# A reduced variable neighborhood search approach for feature selection in cancer classification

Angelos Pentelas, Angelo Sifaleras[0000−0002−5696−7021], and Georgia Koloniari

Department of Applied Informatics, School of Information Sciences,
University of Macedonia, 156 Egnatia Str., Thessaloniki 54636, Greece,
apentelas@uom.edu.gr, sifalera@uom.gr, gkoloniari@uom.gr

**Abstract.** In this work we propose a Reduced Variable Neighborhood Search (RVNS) algorithm, to handle the gene selection problem in cancer classification. RVNS is utilized as the search method and gene subsets obtained are evaluated by three learning algorithms, namely support vector machine, k-nearest neighbors, and random forest. Experiments are conducted on five publicly available cancer related datasets, all characterized by a small sample size to dimensionality ratio. Since RVNS seeks gene subsets that yield accurate predictions for all three aforementioned classifiers, the obtained results can be considered more reliable. To the best of our knowledge, the proposed methodology is innovative due to the fact that, it combines the Recursive Feature Elimination (RFE) heuristic with a RVNS algorithm. Despite the large size of the problem instances, the suggested feature selection scheme converges within reasonably short time, when compared to similar methods. Results indicate high performance for RVNS that, is further improved when the RFE method is applied as a pre-processing step.

**Keywords:** Reduced Variable Neighborhood Search · Feature Selection · Cancer Classification.

## 1 Introduction

Compelling technological advances, along with a well-established existent theoretical background, shaped the era of Big Data and Artificial Intelligence. These terms, usually intertwined, imprint the development of tools capable of collecting and storing complex data, as well as methods for mining knowledge from them. Industry and organisations tested and adopted such techniques in a sense that data-driven decisions and operations carry less bias and are, thus, more reliable.

However, the aforementioned trend results in datasets complicated enough that it takes great computational effort for machines to analyze and makes impossible for human experts to interpret, e.g., microarray datasets. In an attempt of achieving a fair trade-off between leveraging all the available information and interpreting an objective's results, Feature Selection (FS) emerged. On a high level, FS can be considered as a technique that, ideally, maintains only relevant

Please cite this paper as:

Pentelas A., Sifaleras A., and Koloniari G., "A reduced variable neighborhood search approach for feature selection in cancer classification", Variable Neighborhood Search (ICVNS 2019), Springer, Cham, LNCS, Vol. 12010, pp. 1-16, 2020.

The final publication is available at Springer via https://doi.org/10.1007/978-3-030-44932-2_1

information, i.e., features, of a dataset about the imminent analysis' scope and discards the rest as irrelevant. Since the FS problem has been proven to be NP-hard [19] and, in addition, in [21] it is implied that the choice of an effective FS method is dataset dependent, various FS techniques have been proposed in the literature. These can be arranged into four groups, namely *Filters*, *Wrappers*, *Embedded*, and *Ensemble*. However, following the recent research studies within the field, an observable shift towards hybridized FS schemes is apparent [1,18,8,3,5]. In the next three paragraphs, all methods are shortly described within a classification task context.

The *Filter* methods rely only on the intrinsic data characteristics, i.e., statistical metrics. Such techniques benefit from a low time complexity and limit the risk of model over-fitting since they do not take the learning algorithm's performance into consideration. The latter can be proven one of their most significant drawbacks, since the predictive ability of a model is a significant concern for domain experts.

*Wrappers* include techniques that continuously search into the feature space, select a feature subset, evaluate its quality by, usually, one classifier and repeat this process until some stopping criteria are met. The selection of a feature subset is typically driven by an intelligent mechanism (e.g., metaheuristics) and is not randomized. Despite being computationally more expensive than the filter methods, these techniques yield more accurate results and manageable sized solutions. Nevertheless, wrappers seem to undergo the risk of model over-fitting.

Trying to balance the pros and cons of the aforementioned FS classes, *Embedded* methods emerged. As stated in [6], such methods use the core of the classifier to establish criteria to rank features. Finally, *Ensemble* techniques, acting like ensemble of classifiers, combine methods described above on the assumption that combining the output of multiple experts is better than the output of any single expert [6]. Nonetheless, both of the aforesaid techniques come with deficiencies. In particular, *Embedded* methods are generally driven by heuristic approaches, thus leading to insufficient exploration of the solution space. *Ensemble* FS schemes, on the other hand, require higher computational time than any single FS technique they incorporate does. Moreover, the contribution of each FS scheme to the final feature subset is not obvious and necessitates examination.

The purpose of this work is to propose an efficient search mechanism for gene selection in cancer classification that limits the drawbacks of wrapper FS techniques, i.e., the risk of model over-fitting and the high computational cost, while it manages to obtain accurate results. To this end, we implement a *Reduced Variable Neighborhood Search* (RVNS) algorithm that searches the solution space in a systematic, yet computationally light, manner. Solutions provided by the RVNS are shared across Support Vector Machine (SVM), $k$-Nearest Neighbor (k-NN) and Random Forest (RF) classifiers for evaluation and their average accuracy, along with the solution's number of selected genes, are taken into consideration by an appropriate evaluation function. As a result, the final gene subsets obtained by our algorithm yield accurate predictions for more than one learning algorithms and findings can be further used with more reliability.

In the rationale that population-based meta-heuristics have been extensively studied within the FS field, we provide a single-point search meta-heuristic algorithm (i.e., RVNS) that, performs exceptionally in terms of accuracy, final gene subset size, and convergence time. By applying the embedded Support Vector Machine - Recursive Feature Elimination (RFE) technique as a pre-processing step that significantly reduces the feature space, we suggest the RFE-RVNS hybrid method. Both RFE-RVNS and RVNS were tested on five high-dimensional cancer-related datasets, frequently used in cognate research papers.

The structure of this work is as follows. In Section 2, we discuss similar approaches within the gene selection problem, focusing on recent research work and the methods they utilize. Next, we introduce our methodology in Section 3. Section 4 presents the results of our methods on five datasets and a comparison with related well-performing algorithms is quoted. Last comes a short summary of our findings, as well as thoughts for future work and improvements, in Section 5.

## 2 Related work

Focusing on recently conducted studies, in [1] authors implemented two wrapper methods, namely a Genetic Algorithm (GA) and a Geometrical Particle Swarm Optimization (GPSO) to address the gene selection problem. The proposed FS schemes use SVM as their learning algorithm which obtains noteworthy results, after evaluating 4,000 solutions.

Another population-based approach is presented in the work of Alshamlan et al. [3]. A Genetic Bee Colony (GBC) optimization algorithm is applied on a reduced solution space, provided by the Maximum Relevance Minimum Redundancy (MRMR) filter method. SVM's accuracy is again selected as the primary optimization parameter. The overall performance of the hybridized technique is considered acceptable in terms of predictive capability and gene subset size. However, parameter values indicate the requirement of great computational effort, since more than 8,000 evaluations occur.

In a more recent study [5], two hybrid algorithms are presented combining both filter and wrapper FS methods. These two proposed approaches consist of a pre-selection phase, carried out by filter techniques, followed by a search phase that determines a good subset of genes for the classification. A wrapper metaheuristic is responsible for the latter. From an accuracy standpoint, results in eight datasets indicate competitive performance. The computational effort, though, proves underwhelming, with tens of minutes and even hours of runtime. Worth noticing, the classifiers utilized in the two methods are SVM and k-NN, respectively.

Finally, valuable insights come from [18], where authors combine the SVM-RFE embedded method with the MRMR filter one. The novelty of this research work is that, genes are ranked by a convex combination of the relevance given by SVM weights and the MRMR criterion. Results in this case are also acceptable, even though gene subset sizes can not be considered small enough.

With all referenced studies being after 2007, a trend towards hybridized FS schemes becomes apparent. More specifically we note that, filters and embedded methods are in many cases used as a pre-processing step in order to reduce the vast solution space of the gene subset selection problem. Afterwards, wrappers' advantages being exploited, producing small and informative gene subsets. Concerning the learning algorithms used, SVM and k-NN have been the most popular choices.

## 3 Research methodology

In this section, we elaborate on all algorithms used within our research, as well as how they are combined to form the proposed RVNS and RFE-RVNS FS schemes.

### 3.1 Reduced Variable Neighborhood Search

Variable Neighborhood Search (VNS) is a metaheuristic method based on systematic changes in the neighborhood structure within a search, for the solution of various optimization problems. A large number of successful applications of VNS have already been proposed in the literature, [17,23]. In the years following, several variations of VNS emerged, with Reduced VNS (RVNS) being one of them. The essential difference between VNS and RVNS is that, the latter avoids any kind of local search within each neighborhood structure, as shown in Algorithm 1. This fact results in RVNS being computationally lighter than the basic algorithm and, thus; a promising search strategy in large problem instances.

**Algorithm 1:** RVNS pseudocode for a minimization problem

```
initialize solution x
while stopping criteria are not met do
    k = 1
    while k ≤ k_max do
        generate x' a random solution from neighborhood N_k(x)
        if evaluate(x') < evaluate(x) then
            x = x'
            k = 1
        else
            k = k + 1
        end
    end
end
return x;
```

Each candidate solution $s$ is represented as a binary, 1-dimensional array of length $N$, with $N$ denoting the number of genes in each dataset. For instance, a

candidate solution in a dataset with five genes could be: $s = [0, 1, 1, 0, 1]$ which means that, the second, third, and the fifth genes of the dataset are selected; while the first and the fourth are not.

Furthermore, the three following neighborhood structures (i.e., $k_{max} =$ three) are used by both RFE-RVNS and RVNS schemes:

1. *Replace a selected gene of the incumbent solution with an un-selected one.*
2. *Replace two selected genes of the incumbent solution with an un-selected one. If the incumbent solution has only one gene selected, return the incumbent solution.*
3. *Add an un-selected gene to the incumbent solution. If there are no more genes to add, return the incumbent solution.*

The neighborhood order, which is also decisive, is as indicated above. In this manner, RVNS first tries to improve the current solution by keeping the same number of selected genes and, in case that fails, moves to the second neighborhood that reduces the selected genes by one. It is only when both these strategies are unsuccessful that the algorithm will seek a new solution with more selected genes. It should be pointed out that, in all experiments, the initial solution is generated arbitrarily with two randomly selected genes. Therefore, according to the neighborhood definitions above, no exception-handling is required for the case of zero selected genes.

***Example 3.1*** Assume a microarray dataset with five genes and an incumbent solution $s = [0, 1, 1, 0, 1]$. Let us denote with $N_i(s)$, $i \in \{1, 2, 3\}$, the sets of neighboring solutions of $s$. According to the three neighborhood structures as defined above, three resulting solutions could be $s_1 = [0, 1, 1, 1, 0] \in N_1(s)$, $s_2 = [1, 0, 1, 0, 0] \in N_2(s)$ and $s_3 = [1, 1, 1, 0, 1] \in N_3(s)$.

### 3.2 Recursive Feature Elimination

Recursive Feature Elimination (RFE) is a heuristic feature ranking approach that determines the importance of each feature based on a learning model's *coefficient* attribute or a *feature importance* metric. RFE is capable of yielding subsets with a specified number of features by repeatedly removing the least significant one(s).

Appertaining to the embedded FS techniques, RFE needs to be associated with a learning algorithm in order to be meaningful. Authors in [14], who introduced the RFE algorithm, combined it with an SVM classifier and successfully tested their SVM-RFE method on two microarray datasets.

In Algorithm 2, the process that SVM-RFE follows to rank all features is given. More specifically, at each iteration, the least significant feature is removed from the *survivable features* vector (i.e., $s$) and is appended to the *ranked list of features* (i.e, $r$) one. The necessity of each feature is quantified by the extent of contribution it occupies in the learning model. In the case of SVM, the importance of each feature is calculated through the $w$ and $c$ vectors, as illustrated in the aforementioned algorithm.

**Algorithm 2:** SVM-RFE pseudocode

**Input**: $X_0 = [x_1, x_2, ..., x_k, ..., x_l]^T$ `// training examples`
**Input**: $y = [y_1, y_2, ..., y_k, ..., y_l]^T$ `// class labels`
initialize subset of surviving features $s = [1, 2, ..., n]$
initialize feature ranked list $r = [\ ]$
**while** $s \neq \emptyset$ **do**
  $X = X_0(:, s)$ `// restrict training examples`
  $\alpha =$ SVM-train($X$,$y$) `// train the classifier`
  $w = \sum_k \alpha_k y_k x_k$ `// compute the weight vector`
  $c_i = (w_i)^2, \forall i$ `// compute the ranking criteria`
  $f = argmin(c)$ `// find the feature with the smallest ranking`
  $r = [s(f), r]$ `// update feature ranked list`
  $s = s(1 : f - 1, f + 1 : length(s))$ `/* eliminate the feature with smallest ranking criterion */`
**end**
**return** $r$;

### 3.3 Learning Algorithms

**Support Vector Machine** In [9], Cortes and Vapnik proposed a remarkably effective learning algorithm called Support Vector Machine (SVM). SVM, conceptually implemented on a very simple idea, seeks for the surface, i.e., hyper-plane, that can optimally segregate two-class training data. Predictions are based on what side of the, already defined, hyper-plane future data are mapped into. Note that SVMs can also be extended for multi-class classification tasks. Its simplicity, flexibility, and satisfactory computational complexity render SVMs superior to many supervised learning algorithms. As a result, several FS methods suggested in the literature have adopted the aforementioned classifier as their primary evaluation metric [1,3,5,11,14,18].

***k*-Nearest Neighbors** $k$-Nearest Neighbors ($k$-NN) is another powerful supervised learning algorithm widely used within the FS process [5,8,22]. It is considered a lazy learning algorithm, i.e., it does not make any assumptions about the underlying data distribution. Given a distance metric and a future data point mapped into the feature space, the class label assigned to the latter depends on the class labels of its k less-distant records. Leveraging mathematical topology's attributes, computation of the k-nearest neighbors can be efficiently achieved.

**Random Forest** A Decision Tree (DT) is a logical structure consisting of parent and children nodes. In a high level approach, a splitting criterion is applied on each parent node in an attempt to yield pure children nodes, i.e., nodes that contain data points of one class, only. The Random Forest (RF) classifier improves the predictive capability of a single DT by incorporating many DTs that are built upon a random subset of data features. The class prediction of

a future instance is justified by the majority of the partial class predictions each DT makes. RF is, thus, an ensemble of classifiers and demonstrates high performance in many machine learning applications, e.g., [4,10].

### 3.4 Hybrid RFE-RVNS method

In an attempt to enhance RVNS's performance, we apply the RFE heuristic approach as a pre-processing step. In that way, a significant number of possibly redundant genes are eliminated and the resulting solution space is handed over RVNS to search into. Therefore, a new search strategy is formed that we refer to as RFE-RVNS. Figure 1 depicts the aforementioned process.

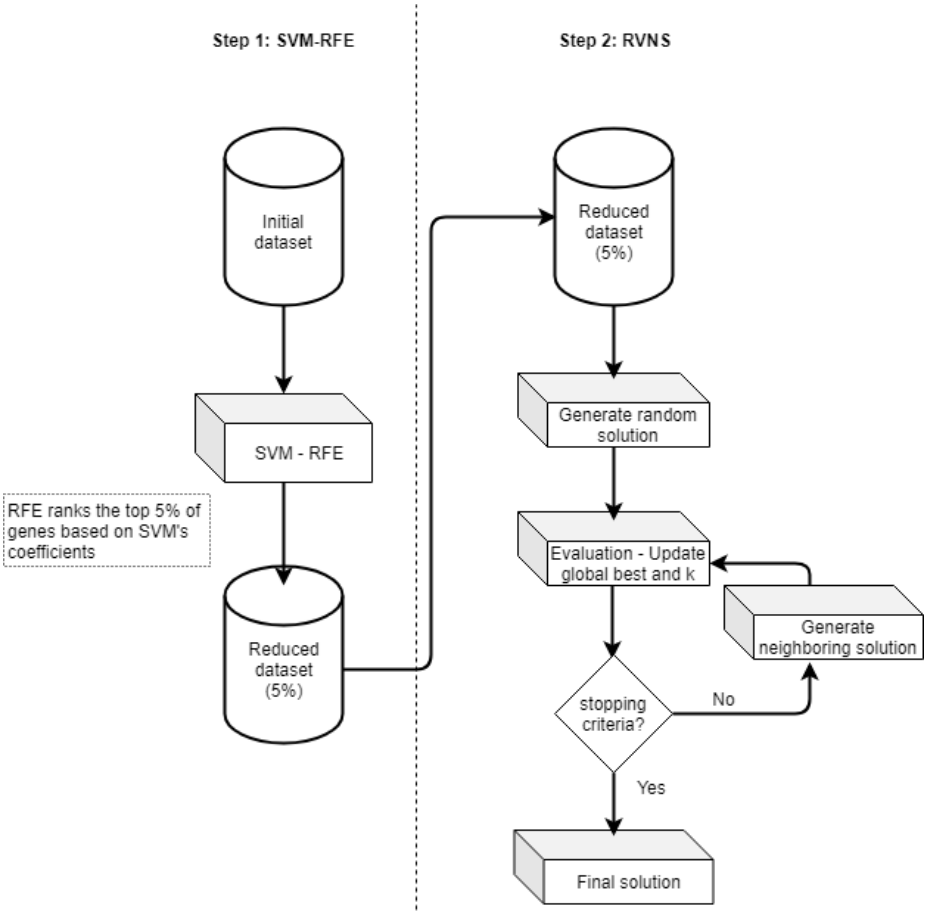


**Fig. 1.** The RFE-RVNS flowchart. The value of $k$ indicates the neighborhood structure the algorithm is searching into.

Each candidate solution is evaluated by four metrics; the accuracy of the three classifiers and the size of the incumbent gene subset. Consequently, we define a fitness function that, is described below:

$$evaluate(s) = \alpha * \underbrace{\frac{3}{a_1(s) + a_2(s) + a_3(s)}}_{a(s)} + (1 - \alpha) * g(s) \quad (1)$$

where $a_1(s)$, $a_2(s)$, and $a_3(s)$ denote the accuracy the SVM, $k$-NN and RF classifiers yield from solution $s$, respectively, and $g(s)$ indicates the number of selected genes. Assuming each predictive model performs at least as good as a random classification, $a_i(s) \in [0.5, 1.0], \forall i \in \{1, 2, 3\}$ (binary classification), thus $\frac{3}{\sum_{i=1,3} a_i(s)} \in [1, 2]$, while $g(s)$ is a positive integer restricted by the number of genes in each dataset. Parameter $\alpha$ acts as weight to the average accuracy of the three classifiers, while $1 - \alpha$ acts similarly to the gene subset size.

The evaluation function in Equation 1 was selected since it can offer a good trade-off between the overall accuracy and the final number of selected genes. Experimentation led us to setting $\alpha$ to 0.99; a value that, is consistent both with our objective of finding informative gene subsets and with the co-domains of $\frac{3}{a(s)}$ and $g(s)$ in Equation 1.

A similar fitness function is used in [1] and manages to balance accurate predictions and small gene subsets, although with different weight values and using the accuracy of a single learning algorithm.

## 4 Experimental results and comparison

All learning algorithms mentioned, the RFE heuristic, as well as the data normalization leverage the Python's scikit-learn library, developed for data science purposes. Experiments are conducted on an Intel i7-7700k 4-core processor, clocked at 4.2Ghz, with 16Gb of RAM. Single runs of both RVNS and RFE-RVNS never exceeded a minute, pre-processing included.

### 4.1 Parameter Settings

**RVNS** Along with the neighborhood structures defined in Section 3, an essential parameter of RVNS that should be specified is the algorithm's termination criteria. In our implementation, we set those to be 300 iterations. The latter indicates that the RVNS algorithm evaluates exactly 300 solutions which is just as 900 classifications, i.e., three classifications per evaluation.

**RFE** The RFE heuristic is applied with an SVM classifier. In each dataset, the SVM-RFE method eliminates 95% of the genes that, are considered irrelevant. The way of achieving this is by removing nine times a 10% (referring to the initial number of genes) of the least important genes and, finally, a 5%. Let us

note that, in a typical RFE execution, such an elimination-step is considered quite large. In order to maintain the computational complexity low, and since RFE is not the primary search method, we apply it with the selected parameters we mentioned above.

**Learning Algorithms** Core parameters of the learning algorithms are empirically selected with two ends in mind; accuracy performance and computational efficiency. Seeking for a balance between these two, we tested $k$-NN with $k$ in $\{1, 3, 5, 7, 9\}$. Additionally, various RF implementations, with number of DT's in $\{10, 20, 30, 40, 50\}$ and pruning depth value in $\{10, 15, 20\}$, helped us proceed to our final choice.

The number of neighboring classes that $k$-NN takes into consideration before classifying an unknown patient is set to five and the RF classifier predicts class labels by consulting with 20 10-depth pruned decision trees. Moreover, the SVM classifier is implemented with a linear kernel meaning that, it searches for the best linear hyper-plane that is able to discriminate the data. The accuracy obtained from each learning algorithm is averaged after a 10-fold cross-validation.

### 4.2 Data Description and Preprocessing

The proposed methodology is tested on five publicly available cancer-related datasets; the Leukemia , Lung, Ovarian, Colon, and Breast cancer datasets. The first four were originally taken from the public Kent Ridge Bio-medical Data Repository, which is now hosted in the ELVIRA Biomedical Data Repository (`http://leo.ugr.es/elvira/DBCRepository`). The Breast Cancer Dataset was available under `https://data.mendeley.com/datasets/v3cc2p38hb/1`. Sample size, dimensionality, and the number of classes of each dataset are depicted in Table 1.

**Table 1.** Dataset characteristics

| Dataset | Sample size | Number of genes | Number of classes | Reference |
|---|---|---|---|---|
| Leukemia | 72 | 7,129 | 2 | [12] |
| Lung | 181 | 12,533 | 2 | [13] |
| Ovarian | 253 | 15,154 | 2 | [20] |
| Colon | 62 | 2,000 | 2 | [2] |
| Breast | 590 | 17,814 | 2 | [7] |

All data values are normalized and missing ones are replaced by zero's, i.e., their mean. It should be noted that, only in the Breast cancer dataset, a few missing values are found.

### 4.3 Performance of RFE-RVNS and RVNS

The performance of the proposed algorithms on the selected datasets is depicted in Tables 2-6. The metrics measured are the *Best*, *Mean*, and *Worst* values of each of the classifiers' accuracy, along with the respective number of genes values (*#Genes*). The *Average accuracy* metric, which is measured as the average accuracy value of SVM, $k$-NN and RF in a single run, should not be interpreted as a typical learning algorithm's accuracy, but rather as the ability of the proposed algorithms to obtain informative genes for all classifiers.

**Table 2.** Performance of RFE - RVNS and RVNS algorithms when applied with the SVM, k-NN and RF classifiers on the Leukemia dataset after ten independent runs.

| | **RFE-RVNS** | | | **RVNS** | | |
|---|---|---|---|---|---|---|
| **Metric** | **Best** | **Mean** | **Worst** | **Best** | **Mean** | **Worst** |
| Average accuracy | 99.58 | 98.88 | 97.64 | 97.44 | 94.26 | 89.35 |
| SVM accuracy | 100 | 98.75 | 94.58 | 100 | 95.18 | 86.67 |
| k-NN accuracy | 100 | 99.58 | 97.08 | 97.5 | 93.84 | 87.56 |
| RF accuracy | 100 | 98.32 | 97.08 | 97.5 | 93.76 | 87.2 |
| #Genes | 3 | 3.8 | 5 | 2 | 4.7 | 8 |

**Table 3.** Performance of RFE - RVNS and RVNS algorithms when applied with the SVM, k-NN and RF classifiers on the Lung cancer dataset after ten independent runs.

| | **RFE-RVNS** | | | **RVNS** | | |
|---|---|---|---|---|---|---|
| **Metric** | **Best** | **Mean** | **Worst** | **Best** | **Mean** | **Worst** |
| Average accuracy | 99.44 | 98.65 | 97.22 | 98.55 | 95.67 | 91.88 |
| SVM accuracy | 99.44 | 98.73 | 97.22 | 98.36 | 95.75 | 91.17 |
| k-NN accuracy | 100 | 98.67 | 97.22 | 98.36 | 95.19 | 91.14 |
| RF accuracy | 100 | 98.56 | 97.22 | 98.92 | 96.07 | 93.33 |
| #Genes | 2 | 2.2 | 3 | 2 | 2.5 | 3 |

Commenting upon figures in Tables 2, 3, 4, and 6, RFE-RVNS managed to obtain a maximum, i.e., the maximum of bests, of 100% accuracy and a minimum, i.e., the minimum of worst, of 97.22%, while the corresponding values for RVNS are 99.10% and 89.35%, respectively. In the case of the Colon dataset, both RFE-RVNS and RVNS faced some adversities in finding small and informative gene subsets with a mean accuracy of 91.23% and 87.22%, respectively. Notable is the fact that, in the Ovarian dataset, RFE-RVNS managed to simultaneously yield 100% accuracy for all three classifiers.

Concerning the gene subset size, the mean number of selected genes is impressively small under both approaches, with 3.8-gene and 4.7-gene subsets being

**Table 4.** Performance of RFE - RVNS and RVNS algorithms when applied with the SVM, k-NN and RF classifiers on the Ovarian cancer dataset after ten independent runs.

| | **RFE-RVNS** | | | **RVNS** | | |
|---|---|---|---|---|---|---|
| **Metric** | **Best** | **Mean** | **Worst** | **Best** | **Mean** | **Worst** |
| Average accuracy | 100 | 99.18 | 97.87 | 98.67 | 96.94 | 94.77 |
| SVM accuracy | 100 | 99.49 | 97.62 | 99.6 | 98.06 | 95.63 |
| k-NN accuracy | 100 | 99.21 | 98 | 98.4 | 97.04 | 94.52 |
| RF accuracy | 100 | 98.85 | 97.6 | 95.74 | 95.74 | 91.39 |
| #Genes | 2 | 2.4 | 3 | 3 | 4.1 | 7 |

**Table 5.** Performance of RFE - RVNS and RVNS algorithms when applied with the SVM, k-NN and RF classifiers on the Colon cancer dataset after ten independent runs.

| | **RFE-RVNS** | | | **RVNS** | | |
|---|---|---|---|---|---|---|
| **Metric** | **Best** | **Mean** | **Worst** | **Best** | **Mean** | **Worst** |
| Average accuracy | 93.73 | 91.23 | 88.57 | 89.68 | 87.72 | 85.24 |
| SVM accuracy | 96.90 | 93.36 | 88.57 | 93.33 | 89.50 | 84.05 |
| k-NN accuracy | 96.90 | 92.69 | 89.05 | 90.00 | 87.79 | 84.05 |
| RF accuracy | 90.24 | 87.64 | 84.05 | 88.57 | 85.88 | 80.48 |
| #Genes | 4 | 5.5 | 8 | 3 | 5.5 | 8 |

**Table 6.** Performance of RFE - RVNS and RVNS algorithms when applied with the SVM, k-NN and RF classifiers on the Breast cancer dataset after ten independent runs.

| | **RFE-RVNS** | | | **RVNS** | | |
|---|---|---|---|---|---|---|
| **Metric** | **Best** | **Mean** | **Worst** | **Best** | **Mean** | **Worst** |
| Average accuracy | 99.27 | 98.83 | 98.37 | 99.1 | 98.35 | 97.06 |
| SVM accuracy | 99.32 | 98.83 | 98.47 | 99.32 | 98.39 | 96.95 |
| k-NN accuracy | 99.32 | 98.97 | 98.31 | 99.32 | 98.39 | 97.12 |
| RF accuracy | 99.32 | 98.7 | 98.14 | 99.16 | 98.29 | 97.12 |
| #Genes | 1 | 1.7 | 2 | 2 | 2.5 | 3 |

the largest average ones for RFE-RVNS and RVNS respectively. Again, in the Colon dataset, the behavior differs a little with slightly larger gene subsets.

While both methods perform worthy, not only in terms of yielding informative, to all classifiers, gene subsets, but also small sized ones, the dominance of RFE-RVNS over RVNS cannot be overlooked.

Questioning whether one learning algorithm is favored over the others, or whether their predictive ability significantly varies, Figure 2 shows that only in the case of the Colon cancer dataset, the RF model performs somewhat worst.

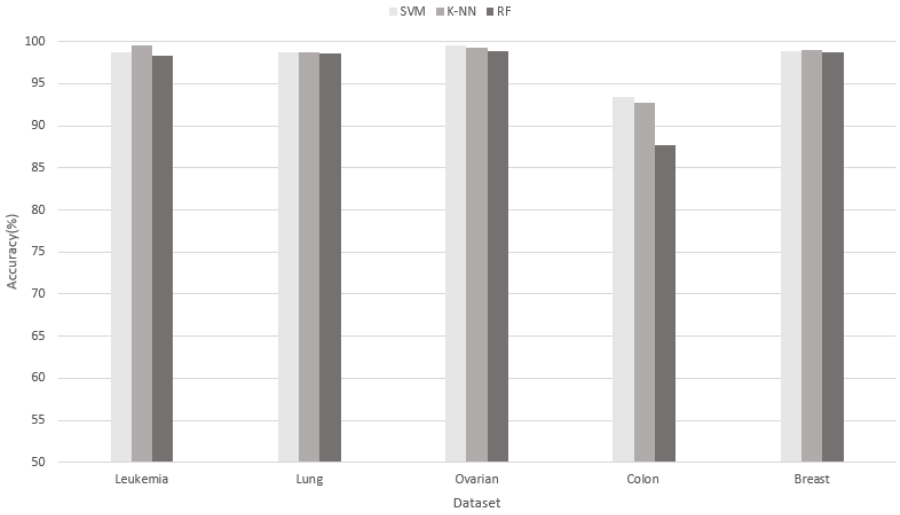


**Fig. 2.** SVM, $k$-NN and RF mean accuracy in each dataset obtained by RFE-RVNS.

Trying to decipher our method's behavior on the Colon dataset, we depict the classifiers' accuracy on current best solutions found every 15 iterations of a typical RVNS execution. The graphs illustrated in Figure 3 indicate that, gene subsets obtained often improve one learning algorithm's performance but worsen another's, e.g., iterations 40 and 145. This phenomenon adds to our intention of implementing a gene selection strategy that returns informative gene subsets for more than one classifier, in the sense of quality and reliability.

### 4.4 Comparison

As stated earlier, the proposed fitness function tries to achieve high performance on three learning algorithms while maintaining a small gene subset size. However, most related work was conducted by targeting one or two ends. Thereby, within the context of a *search strategy* comparison, the objective function of RFE-RVNS and RVNS is modified in order to take only the SVM's accuracy into consideration, meaning that only a third of the classifications originally

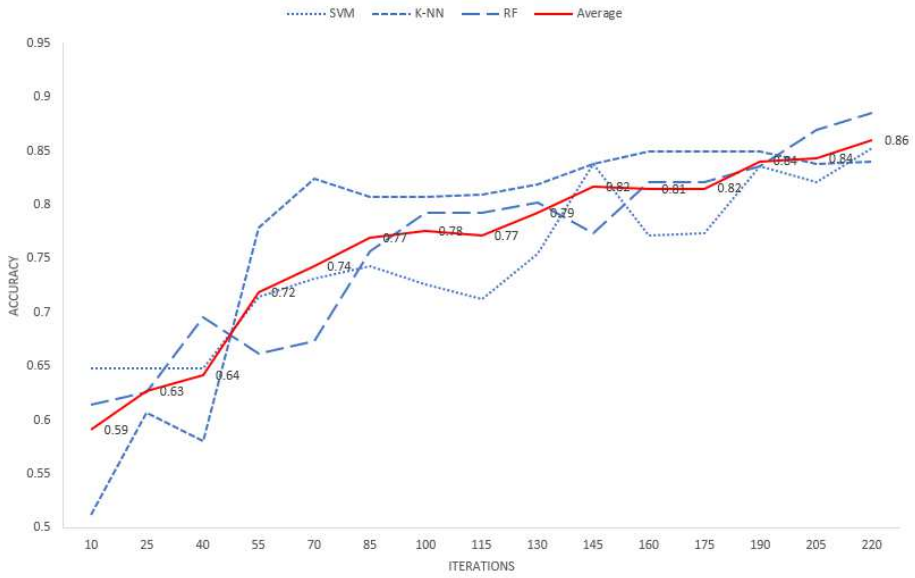


**Fig. 3.** SVM, $k$-NN, RF and average accuracy values from a typical execution of RVNS on the Colon dataset.

made will occur. That allows us to intensify the search capability of RFE-RVNS and RVNS by increasing the iterations from 300 to 500 and 1,000 respectively. Comparatively, in most related studies, wrapper methods tend to evaluate a few thousands candidate solutions as mentioned in Section 2.

It is also in the same section that we refer to the Genetic Algorithm (GA) [1], the Geometrical Particle Swarm Optimization (GPSO) [1], the Genetic Bee Colony (GBC) [3] and the Maximum Relevance Minimum Redundancy - SVM-RFE (MRMR+SVM) [18] algorithms. Furthermore, in Table 7, performance of FS schemes like the Multiple Filter Multiple Wrapper (MFMW) [15] and the Ensemble Neural Network (ENN) [16] is presented. Lastly, it must be pointed out that, the Feature Selection - Random Projection (FS+RP) [24] method does not appertain to typical FS techniques presented in this paper; instead, it is associated with the feature extraction ones. However, we proceed to a comparison with it since, to the best of our knowledge, no other FS methods tested on the exact Breast cancer dataset can be found in the literature.

In Table 7, results indicate that RFE-RVNS outperforms well-known gene selection methods in all datasets except for the Colon one, while RVNS also obtains notable results. Thus, a small, yet informative, gene subset can be successfully obtained under a Variable Neighborhood Search strategy. Compared to similar methods, our algorithms require less amount of computational time since they evaluate significantly less candidate solutions.

**Table 7.** The performance of RFE-RVNS and RVNS algorithms, when applied only with the SVM classifier, compared to similar methods.

| Reference | Leukemia | Lung | Ovarian | Breast | Colon |
|---|---|---|---|---|---|
| RFE-RVNS | 99.86[4] | 99.51[3] | 99.80[3] | 99.12[2] | 96.69[5] |
| RVNS | 98.84[5] | 98.67[3] | 98.55[4] | 98.66[2] | 93.74[6] |
| GA [1] | 95.86[4] | 99.49[4] | 98.83[4] | - | 100[3] |
| GPSO [1] | 97.38[3] | 99.00[4] | 99.44[4] | - | 100[2] |
| GBC [3] | 96.43[5] | - | - | - | 91.51[5] |
| MFMW [15] | - | 98.34[6] | - | - | 95.16[6] |
| MRMR+SVM [18] | 98.35[37] | - | - | - | 91.68[78] |
| ENN [16] | - | - | 99.21[75] | - | 81.48[-] |
| FS+RP [24] | - | - | - | 98.97[>100] | - |

## 5 Conclusions and Future Work

In this paper, our aim was to suggest an efficient wrapper feature selection method capable of yielding informative gene subsets for cancer classification. Therefore, we proposed a Reduced Variable Neighborhood Search algorithm as the primary search strategy. In many cases though, performance of different learning algorithms may significantly vary, despite learning from the same data (i.e., gene subsets). Consequently, we evaluated each gene subset by three classifiers, i.e., support vector machine, k-nearest neighbors and random forest, and balanced the extra computational effort by enforcing considerably less, compared to the literature, classification attempts. In addition to that, we applied the Recursive Feature Elimination heuristic method to reduce the feature space which was then given to RVNS to search into.

Both RFE-RVNS and RVNS performed well despite the large size of problem instances and the computationally intensive 3-model building. Results on five well-known publicly available microarray datasets indicate high performance of RVNS that manages to obtain high accuracy for all three classifiers while still keeping the gene subset size relatively small. By applying RFE and executing the RVNS algorithm on a significantly reduced feature space (5% of the initial size), the total performance is considerably improved. As a result, small-sized gene subsets obtained can be suggested to experts with higher reliability.

We conclude by acknowledging that, an algorithm's robustness constitutes an important performance criterion. The development of an appropriate initialization (construction) method might add to that direction. Further study on the latter, along with testing our method on more datasets and different domains (e.g., text classification) will concern us in future work.

## Acknowledgements

The second author has been funded by the University of Macedonia Research Committee as part of the "Principal Research 2019" funding scheme (ID 81307).

## References


1. Alba, E., Garcia-Nieto, J., Jourdan, L., Talbi, E.G.: Gene selection in cancer classification using PSO/SVM and GA/SVM hybrid algorithms. In: IEEE Congress on Evolutionary Computation (CEC 2007). pp. 284–290 (2007)
2. Alon, U., Barkai, N., Notterman, D.A., Gish, K., Ybarra, S., Mack, D., Levine, A.J.: Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. Proceedings of the National Academy of Sciences **96**(12), 6745–6750 (1999)
3. Alshamlan, H.M., Badr, G.H., Alohali, Y.A.: Genetic bee colony (GBC) algorithm: A new gene selection method for microarray cancer classification. Computational biology and chemistry **56**, 49–60 (2015)
4. Belgiu, M., Drăguţ, L.: Random forest in remote sensing: A review of applications and future directions. ISPRS Journal of Photogrammetry and Remote Sensing **114**, 24–31 (2016)
5. Bir-Jmel, A., Douiri, S.M., Elbernoussi, S.: Gene selection via BPSO and backward generation for cancer classification. RAIRO-Operations Research **53**(1), 269–288 (2019)
6. Bolón-Canedo, V., Sánchez-Marono, N., Alonso-Betanzos, A., Benítez, J.M., Herrera, F.: A review of microarray datasets and applied feature selection methods. Information Sciences **282**, 111–135 (2014)
7. Cancer Genome Atlas Network: Comprehensive molecular portraits of human breast tumours. Nature **490**(7418), 61 (2012)
8. Chuang, L.Y., Yang, C.H., Wu, K.C., Yang, C.H.: A hybrid feature selection method for DNA microarray data. Computers in biology and medicine **41**(4), 228–237 (2011)
9. Cortes, C., Vapnik, V.: Support-vector networks. Machine learning **20**(3), 273–297 (1995)
10. Díaz-Uriarte, R., De Andres, S.A.: Gene selection and classification of microarray data using random forest. BMC Bioinformatics **7**(1), 3 (2006)
11. Duan, K.B., Rajapakse, J.C., Wang, H., Azuaje, F.: Multiple SVM-RFE for gene selection in cancer classification with expression data. IEEE Transactions on Nanobioscience **4**(3), 228–234 (2005)
12. Golub, T.R., Slonim, D.K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J.P., Coller, H., Loh, M.L., Downing, J.R., Caligiuri, M.A., et al.: Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. science **286**(5439), 531–537 (1999)
13. Gordon, G.J., Jensen, R.V., Hsiao, L.L., Gullans, S.R., Blumenstock, J.E., Ramaswamy, S., Richards, W.G., Sugarbaker, D.J., Bueno, R.: Translation of microarray data into clinically relevant cancer diagnostic tests using gene expression ratios in lung cancer and mesothelioma. Cancer research **62**(17), 4963–4967 (2002)
14. Guyon, I., Weston, J., Barnhill, S., Vapnik, V.: Gene selection for cancer classification using support vector machines. Machine learning **46**(1-3), 389–422 (2002)
15. Leung, Y., Hung, Y.: A multiple-filter-multiple-wrapper approach to gene selection and microarray data classification. IEEE/ACM Transactions on Computational Biology and Bioinformatics **7**(1), 108–117 (2010)
16. Liu, B., Cui, Q., Jiang, T., Ma, S.: A combinational feature selection and ensemble neural network method for classification of gene expression data. BMC Bbioinformatics **5**(1), 136 (2004)

17. Mladenović, N., Sifaleras, A., Sörensen, K.: Editorial to the special cluster on variable neighborhood search, variants and recent applications. International Transactions in Operational Research **24**(3), 507–508 (2017)
18. Mundra, P.A., Rajapakse, J.C.: SVM-RFE with MRMR filter for gene selection. IEEE Transactions on Nanobioscience **9**(1), 31–37 (2010)
19. Narendra, P.M., Fukunaga, K.: A branch and bound algorithm for feature subset selection. IEEE Transactions on Computers (9), 917–922 (1977)
20. Petricoin III, E.F., Ardekani, A.M., Hitt, B.A., Levine, P.J., Fusaro, V.A., Steinberg, S.M., Mills, G.B., Simone, C., Fishman, D.A., Kohn, E.C., Liotta, L.: Use of proteomic patterns in serum to identify ovarian cancer. Lancet **359**(9306), 572–577 (2002)
21. Reunanen, J.: Search strategies. In: Feature Extraction, pp. 119–136. Springer (2006)
22. Rogati, M., Yang, Y.: High-performing feature selection for text classification. In: Proc. of the 11th ACM International conference on Information and Knowledge Management. pp. 659–661 (2002)
23. Sifaleras, A., Salhi, S., Brimberg, J. (eds.): Variable Neighborhood Search - 6th International Conference, ICVNS 2018, Sithonia, Greece, October 4-7, 2018, Revised Selected Papers, Lecture Notes in Computer Science, vol. 11328. Springer, Cham (2019)
24. Xie, H., Li, J., Zhang, Q., Wang, Y.: Comparison among dimensionality reduction techniques based on random projection for cancer classification. Computational Biology and Chemistry **65**, 165–172 (2016)