|  |
| --- |
| BIOINFORMATICS APPLICATION NOTE  **OCFS: an optimal collaborative feature selection framework for and incorporating domain knowledge for high-dimensional data**  Karan Uppal1,2,3,4, Eva K. Lee\*1,2,3  1Center for Operations Research in Medicine and HealthCare, Georgia Institute of Technology, Atlanta, GA, 30332.  2NSF I/UCRC Center for Health Organization Transformation, Georgia Institute of Technology, Atlanta, GA, 30332.  3School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA, 30332.  4School of Biology, Georgia Institute of Technology, Atlanta, GA, 30332.  Received on XXXXX; revised on XXXXX; accepted on XXXXX  Associate Editor: XXXXXXX |

[[1]](#footnote-2)\*abstract

**Motivation:** Improved analytical technologies and better data extraction algorithms in analytic chemistry and related scientific disciplines enable detection of >10,000 chemicals; There are currently limited abilities to automatically identify large numbers of chemicals in these mass spectral analyses. Here we describe an optimization framework to select optimal number of features using single and ensemble feature selection approaches based on a subset generalizability measure. In stage 2, a binary particle swarm optimization algorithm was used to identify minimum number of genes that allowed accurate classification of the samples based on 10-fold cross validation criteria using support vector machine a classifier implemented in the R e1071 package.

Availability: \*Insert URL for sourceforge

# introduction

Biomarker screening and discovery are key components of biomedical research. Most omics technologies measure thousands of variables (genes, metabolites, SNPs, etc.) and often fall under the category of n<<p problems that are prone to model over-fitting. The large amount of information requires application of data mining and technical techniques to identify most salient variables related to the question of interest. In general, the likelihood of erroneous or sub-optimal results is higher as the amount of information increases. Selection of relevant genes for sample classification is a common task in most high-throughput omics studies, where researchers try to identify the smallest possible set of genes that can still achieve good predictive performance. This is crucial for targeted validation experiments, designing follow-up studies, and for diagnostic purposes in clinical practice.

There are three main feature selection schemes: filter, wrapper, embedded, Figure 1. The filter methods use statistical criteria independent of the learning or model evaluation algorithm such as t-test or mutual information that are independent of the model evaluation or learning algorithm to select relevant genes. Variables that meet the filtering criteria, e.g. p<0.05 or false discovery rate less than 5%, are then used for further evaluation. Statistical inference based methods such t-test, Wilcoxon rank sign test, limma, etc. are commonly used in biomedical research due to their simplicity. However, recent studies have reported disadvantages of solely relying upon p-values as that could lead to selection of instable features and results that are not reproducible in validation studies (He 2010, Nature Methods 2015). Furthermore, these methods do not account for relationships between predictors and inter-variable correlation patterns.

Wrapper methods evaluate different feature subsets of features and select the best model based on the evaluation using a learning algorithm such as k-fold cross validation. A variety of methods have been developed for subset selection including exhaustive methods like best subset selection, forward and backward selection, and sequential forward selection; however approaches like best subset selection are not practical for large p problems as the total number of possible subsets is 2p. Stochastic methods such as genetic algorithms and binary particle swarm optimization are

The embedded methods such as recursive feature elimination based on SVM and random forest, Lasso, and ridge regression have built-in variable selection. For instance, Lasso is a coefficient shrinkage method and uses a L1 penalty function to assign a value greater than 0 if a feature is relevant, and 0 otherwise (Saeys 2007).

Feature instability and non-reproducibility of biomarkers have been highlighted as critical issues in several recent review articles and commentaries. Many feature selection approaches use arbitrary rank or significance thresholds to select the number of genes that could lead to irreproducible results (He 2010, Nature Methods 2015). He et al. recognize three main sources of instability in biomarker discovery: a)

Boulestix and Slawski identify two major sources of variability: a) multiplicity of methods for ranking variables by importance; b) variability in results due to perturbations in data.

Moreover, different algorithms vary in performance depending on the distribution of the data and within-class variability (Boulesteix 2009). Advanced variable selection methods such as recursive feature elimination SVM and random forest are well suited for n<<p data and can be used for datasets involving more than two classes (Reference).

It is important to determine which algorithm is best suited for selecting the best set of genes from a gene expression dataset. Ensemble feature selection is a new framework for

Several studies have reported the application of using a binary PSO for variable selection (References). Here we introduce a novel ensemble feature selection framework that uses a binary behavior based PSO (B3PSO) to combine the results from different selection methods.

The main goal of our project was to compare the performance of different feature selection algorithms based on the criteria of minimum number of genes selected and maxmum classification accuracy.

# methods

We focused on comparing the performance of F-test, T-test, Kruskal test, Elastic net, Lasso, rfe-SVM, Random Forest, and Binary PSO in this study. The algorithm uses the internal cross-validation scheme that performs variable selection based on the every training set, traink, in the k-fold scheme and uses the left-out subset, testk, for evaluating the performance of the model. The internal cross-validation scheme is shown to out-perform the commonly used external cross-validation scheme where the model evaluation is performed after the selection using all samples. This leads to a biased estimate of the model performance (Reference).

**2.1 Rank based feature selection using sampling evaluation**

A ranked list of features is generated using one or more feature selection algorithms selected by the user. The user can select t.test, f.test, kruskal.test, recursive feature elimination, random forest, Welch.test, Wilcox.test, lasso, elasticnet. The number of features selected impacts the performance of the classifiers and the overall predictive accuracy (Reunanbam 2003, Nijim and Kuhaar 2006). It is essential to find the optimal set of features. A sequential backward or forward selection scheme is used to select the optimal set of features. For every selected algorithm, the number of features is iteratively increased from 1 to X (default 150) in increments of delta (default 5) till there is no improvement in the 10-fold cross validation accuracy for I number of iterations (default 10).

This gives m subsets of top ranked features where m is the number of selected feature selection algorithms. A consensus score of a feature is defined as the number of subsets that include that feature, and its value ranges from 0 to m. Only the top X number of features (default 150; user defined) are retained for further analysis.

**2.2 Binary behavior based particle swarm optimization (B3PSO) for rank aggregation**

Particle Swarm Optimization (PSO) is an agent-based stochastic optimization technique based on the movement and intelligence of swarms developed by James Kennedy and Russell Eberhart in 1995. It comprises of a number of agents/particles that constitute a swarm moving around in the search space looking for the best solution. In most existing versions of the binary PSO algorithm, all particles behave uniformly and generally follow the fully connected topology where each particle is connected to every other particle (Reference). This has been shown to create bias as all particles are following a single best solution (Reference). The B3PSO algorithm is a modified version of the original PSO algorithm that assigns each agent to one of the four behavioral states {C=Confusion, S=Self-influenced, N=Influenced by nearest neighbors, G=Influenced by swarm} based on the crowd model (Su 2014). A time-homogenous Markov chain process is used to update the behavior of particles after n iterations using the nth power of the transition matrix. The use of dynamic topologies reduces the risk of getting stuck in local optima. An internal k-fold cross-validation scheme is used to evaluate the fitness of each subset of variables where the

Every individual moves according to its own experience as well as the experience of others as described below.

The velocity of each particle, pi, is updated at iteration k+1 according to the equation,

vik+1=vik+c1\*rand1\*(pbesti-sik)+c2\*rand2\*(nbesti-xik) (1)

where,

i is the current particle

c1 and c2 are constant learning factors,

rand1 and rand2 are random numbers between [0,1] interaval

pbest is the best position of the particle has experienced based on the fitness function

xi is the position in iteration k, and

In B3PSO, nbest is determined by particle’s behavior as,

* Neighborhood best:

nbest = 75th percentile value for the k nearest neighbors for each dimension, d

* Global best:

nbest= best position experienced by any particle in the entire swarm

* Random position:

nbest = randomly generated vector of size d with values equal to 0 or 1

The velocity of the particle is restricted to be in the interval [-6,6]. The position is updated according to (2) and (3) in binary PSO using a sigmoid function S,

S=1/(1+exp(-vik+1)) (2)

xi =1 if S>rand3 (3)

0 otherwise

where,

xi is the position at iteration k,

vik+1 is the updated velocity at iteration k+1,

Si is the sigmoid function with values between [0,1] interval,

rand3 is a random number between [0,1] interval

The position and velocity vectors of each particle are updated according to equations 1-3.

Fitness evaluation: The fitness of each particle is evaluated using an internal cross-validation,

fitnesspk = w1\*(CVk-foldtrain – CVperm) + w2\*(CVk-foldtest

The search process is terminated when the distance between the global best position and the centroid of the individual best position is less than x.

The B3PSO algorithm is summarized in Figure 1.

**2.3. OCFS: Optimal collaborative feature selection**

A new feature selection algorithm based on a consensus scoring methodology was implemented. The algorithm is divided into three steps. The first two steps use the CMA package implemented in R, and the last step uses a particle swarm optimization based search strategy to find the optimal set of features that would allow maximum classification accuracy.

The user uploaded dataset is first randomly divided into distinct training and test sets based on the splitting percentage provided by the user (default: 60% train, 40%test).

In the **first step** of the feature selection process, a ranked list of features is generated using one or more feature selection algorithms selected by the user. The user can select t.test, f.test, kruskal.test, recursive feature elimination, random forest, Welch.test, Wilcox.test, lasso, elasticnet. The number of features selected impacts the performance of the classifiers and the predictive accuracy (Reunanbam 2003, Nijim and Kuhaar 2006). It is essential to find the optimal set of features.

A sequential backward or forward selection scheme is used to select the optimal set of features. For every selected algorithm, the number of features is iteratively increased from 1 to X (default 150) in increments of delta (default 5) till there is no improvement in the 10fold cross validation accuracy for I number of iterations (default 10).

This gives m subsets of top ranked features where m is the number of selected feature selection algorithms. A consensus score of a feature is defined as the number of subsets that include that feature, and its value ranges from 0 to m. Only the top X number of features (default 150; user defined) are retained for further analysis. . A gene was identified as being differentially expressed if it was selected by at least two methods. The consensus approach increases the confidence in the gene selection process as some methods are too stringent while others too lenient.

In the **second** stage, the consensus subset of features selected by at least M\_min (default=100%) algorithms is then used as an input for a novel behavior based binary PSO algorithm, B3PSO. The PSO algorithm is designed to avoid particles from getting stuck in local optima. The search criteria (local or global search) depends on the rank of a particle. In other words, the best particle will have a narrow or local search space while the worst particle will have a much broader or a global search space controlled by the inertia variable in the velocity update equation of the PSO algorithm.

B3PSO Algorithm:

Basic idea: Particles x[i][j] are 2D arrays where index i is the particle and index j is the feature. If x[i][j]=1, then the feature j is selected, otherwise it is not selected.

Initializaiton

1)Initially every particle xi has p randomly selected features where p<d

2) After time step 1, the features of a particle are set to 0, or 1 depending on its velocity. If the S measure <rand() then feature is not selected, otherwise it is selected.

3)

Each particle chooses a topology based on their behavior.

The PSO algorithm is designed to avoid particles from getting stuck in local optima. The search criteria (local or global search) depends on the rank of a particle. In other words, the best particle will have a narrow or local search space while the worst particle will have a much broader or a global search space controlled by the inertia variable in the velocity update equation of the PSO algorithm. Get the consensus features that are selected in each set.

We used the Classification for Microarrays (CMA) package - a comprehensive Bioconductor package for supervised classification with high dimensional data [2]. The package has built-in functions for performing feature selection using F-test, T-test, Elasticnet, Lasso, rfeSVM, Kruskal test, and Random Forest. We performed feature selection and cross-validation using the knn classifier from CMA package. The function has in-built feature to tune the value of k to be used.

# results

The performance of the scoring algorithm was evaluated using three publicly available gene-expression datasets for our study. The three datasets varied in their level of complexity. The NCI60 dataset had 60 training samples belonging to 9 classes, but there was no test data, and we could not split the dataset due to very few samples. The SRBCT dataset and Leukemia dataset had both training and test data.

All datasets had number of genes >> number of samples. Table 1 summarizes the details of the three datasets. The results show that the ensemble approach improves the classification results as compared to individual methods.

# DISCUSSION

# CONCLUSION

It can be concluded that the embedded feature selection algorithms perform better than the filter and wrapper methods. Rfe-SVM performed best on the Leukemia and NCI60 dataset, but Lasso gave better performance on the SRBCT dataset. Thus, the performance of a feature selection algorithm depends on the dataset being used, the number of samples in the train and test dataset as well as the number of classes.

Most feature selection algorithms (mainly the filter and wrapper methods) generate models that over-fit to the training data, and thus generalize poorly. Recursive Feature Elimination- SVM method and Lasso achieves better performance because they minimize over-fitting by penalizing the irrelevant features, or removing them at every iteration.

More samples and better classification algorithms are required for building robust models for large-class datasets like NCI60. A comparison of different classification algorithms should be considered for building the trained model.

b) Binary PSO

**For n iterations:**

Initialize the swarm with N particles by randomly selecting p features for every particle

For every particle, evaluate the fitness function:

Build SVM model using the train data (140 samples)

Evaluate the model using the validation data (70 samples) and determine the classification accuracy

Determine the local best and global best particle that gives highest classification accuracy

Update the velocity of every particle,



Update the position of every particle,

S=1/(1+exp(-vi))

xi =1 if S>random number

0 otherwise

**Fig. 1.**Scoring algorithm workflow

**Table 1.**Description of datasets

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Dataset** | **Description** | **Number of Samples** | **Number of genes** | **Classes** | **Source of dataset** |
| NCI60 –RNA HU6800 | Gene expression data of 9 cell lines obtained from 60 cancer patients. **Already processed and filtered.** | Train: 60 | 6112 | 9 | Staunton et al. **Chemosensitivity Prediction by Transcriptional Profiling** Proc Natl Acad Sci USA 2001 Sep 11;98(19):10787-92 <http://discover.nci.nih.gov/cellminer/loadDownload.do> |
| Leukemia | Acute myeloid leukemia (AML) and acute lymphoblastic leukemia (ALL) gene expression data. **Processed, but not filtered.** | Train: 38  Test: 34 | 7130 | 2 | Golub et al. **Molecular Classification of Cancer:Class Discovery and Class Prediction by gene expression profiling** *Science* 15 October 1999: Vol. 286. no. 5439, pp. 531 - 537 <http://www.broad.mit.edu/cgi-bin/cancer/publications/pub_paper.cgi?mode=view&paper_id=43> |
| Small round blue cell tumors (SRBCT) | Gene expression data of 4 tumors (neuroblastoma, rhabdomyosarcoma, Hodgkin lymphoma, and the Ewing family tumor | Train: 63  Test: 20 | 2098 | 4 | Khan et al. **Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks**  <http://stat.ethz.ch/%7Edettling/bagboost.html> |

**Table 2.**Benchmark results

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Feature Selection Algorithm** | **Accuracy (%) using the top n=3 genes** | **Accuracy (%) using the top n=5 genes** | **Accuracy (%) using the top n=10 genes** | **Accuracy (%) using the top n=15 genes** | **Accuracy (%) using the top n=25 genes** | **Accuracy (%) using the top n=50 genes** |
| Elasticnet | 50 | 60 | 55 | 60 | 60 | 65 |
| Ttest | 60 | 50 | 50 | 65 | 60 | 60 |
| rfe-svm | 40 | 35 | 55 | 70 | 50 | 50 |
| Random Forest | 50 | 45 | 60 | 75 | 55 | 70 |
| F-test | 50 | 50 | 70 | 68 | 55 | 55 |
| Kruskal | 50 | 50 | 40 | 60 | 60 | 65 |
| Lasso | 70 | 75 | 50 | 60 | 60 | 65 |

acknowledgements

*Funding*:

References

Eg: Alexandrescu,A. (2001) Modern C++ Design: Generic Programming and Design

Song Wu, Sun Quanbin (2014) Computer Simulation of Leadership, Consensus Decision Making and Collective Behavior in Humans. PLOS ONE.

1. \*To whom correspondence should be addressed. [↑](#footnote-ref-2)