BIOINFORMATICS

optSelect: using agent-based modeling and binary PSO techniques for ensemble feature selection and stability assessment

Karan Uppal^{1,2,3,4}, 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: XXXXXXXX

ABSTRACT

Motivation: Recent studies have shown that the ensemble feature selection approaches are essential for generating robust classifiers. Existing methods for aggregating feature lists from different methods require use of arbitrary thresholds for selecting the top ranked features and do not account for classification accuracy while selecting the optimal set. Here we present a two-stage ensemble feature selection framework for finding the optimal set of features without compromising on classification accuracy.

Methods and Results: We present herein optSelect, a multi agent-based stochastic optimization approach for nested ensemble feature selection. Stage one involves function perturbation, where ranked list of features are generated using different methods and stage two involves data perturbation, where feature selection is performed within randomly selected subsets of the training data and the optimal set of features is selected within each set using the optSelect. The agents are assigned to different behavior states and move according to a binary PSO algorithm. A multi-objective fitness function is used to evaluate the classification accuracy of the agents. We evaluate the system performance using the random probe method and using five publicly available microarray datasets. The performance of optSelect is compared with single feature selection techniques and existing aggregation methods. The results show that the optSelect algorithm improves the classification accuracy compared to both individual and existing rank aggregation methods. The algorithm is incorporated into an R package, optSelect.

Contact: kuppal2@emory.edu

1 INTRODUCTION

Biomarker discovery is a key component of translational biomedical research (Guyon 2003, Lee 2007, Christin 2010). Most omics technologies measure thousands of variables (genes, metabolites, etc.) and often fall under the category of $n \ll p$ problems that are prone to model over-fitting due to large number of variables (Guyon 2003, Reunanen 2003, Cawley 2010). The large amount of feature space requires application of variable selection techniques to identify most salient variables and generate robust classifiers. This is crucial for targeted validation experiments, designing follow-up studies, and for diagnostic purposes in clinical practice.

*To whom correspondence should be addressed.

Numerous feature selection algorithms have been developed over the last few decades (Saeys 2007, Ma 2008, Christin 2010). The feature selection methods can be classified as: filter, wrapper, and embedded (Saeys 2007). The filter methods use statistical criteria independent of the classifier to select relevant features and the selected features, e.g. $p < 0.05$, are then used to build/train the model. Methods such as t-test, ANOVA, F-test, Chi-sq test, mutual information, etc. can be classified as filter methods. The wrapper methods use a search strategy to evaluate different combinations of subsets of features and select the best model based on the evaluation using a classifier such as Support Vector Machine (SVM, Vapnik 1998). Different search algorithms such as best subset, genetic algorithms, PSO, etc. can be used for finding the optimal set of features; however these methods are prone to over-fitting (Saeys 2007, Christin 2010). The embedded methods include methods such as recursive feature elimination based on SVM, random forests (RF), Lasso, Elasticnet where the variable selection is built-in (Guyon 2002, Breiman 2001, Tibshirani 1996, Zou 2005). 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).

Recent articles have highlighted the importance of aggregating ranking results from multiple methods to achieve a set of stable features that are likely to be reproducible in future studies (Saeys 2008, Boulesteix 2009, Abeel 2010, He 2010). The translation of basic science findings to new interventions has been limited due to irreproducible results (Boulesteix 2009, He 2010, Halsey 2015). The two main reasons of “instable” results: a) data perturbation: inconsistency in selected feature subsets due to sampling variations; b) function perturbation: different rankings of relevance from different methods (Boulesteix 2009, He 2010). Many feature selection approaches use arbitrary rank or significance thresholds to select the number of features without thorough evaluation, which could lead to suboptimal results. Moreover, different algorithms vary differently in performance depending on the distribution of the data and within-class variability (Saeys 2008, Boulesteix 2009). Here we introduce a novel nested ensemble feature selection framework, optSelect, that performs multi-objective optimization using agent-based modeling and binary particle swarm optimization (PSO) techniques to allow aggregation of results from different methods and performs nested feature selection using random subsets of training data to evaluate feature stability (Kennedy and Eberhart 1995, Chuang 2010).

2 METHODS

2.1 optSelect: An optimization based approach for nested ensemble feature selection and aggregation

A two-stage procedure is used for finding the optimal set of features. In stage one, 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, recursive feature elimination, random forest, wilcox.test, lasso, elasticnet (Saeys 2007, Christin 2013). The number of features selected impacts the performance of the classifiers and the predictive accuracy (Reunanbam 2003). It is essential to find the optimal set of features. Users have the option to perform sequential backward or forward selection or choose an arbitrary cutoff to select the optimal set of features. Using different ranking criteria allows function perturbation. The different variable selection methods implemented in the CMA package in R are used at this stage (Slawski 2010). A union of the ranked lists from different methods is used as input for stage two. Users have the option to skip stage one and use all features during the optimization procedure; however some selection criteria is recommended prior to stage two to reduce the computational time and perform ensemble selection (Saeys 2007).

In stage two, the newly developed multi-objective stochastic optimization procedure based on agent-based modeling and binary framework is used to perform data perturbation and aggregation using the results from stage one.

Agent-based models involve three key elements (Macal 2010):

- a) Agents and their attributes/behaviors: each agent is assigned to a behavior or rule category, e.g. follows neighbors, moves randomly, etc.
- b) Relationships and interactions between agents: the interactions define how the agents influence and cooperate with each other,
- c) Interaction with the environment: the environment provides feedback to the agent about their movements (e.g. 10-fold cross-validation accuracy)

A binary PSO algorithm is used to simulate the movement of agents according to their behavior. PSO is a 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 determine based on a fitness evaluation function. The movement of each particle, p_i , is determined based on a velocity vector, v_i , and a position vector, x_i . In binary PSO, the position vector, x_i , has d dimensions, where d corresponds to the number of variables (genes, chemicals, etc.). And, $x_{id} \in \{0,1\}$. This allows application of the binary PSO for feature selection, where a feature is selected if $x_{id}=1$. The velocity and position vectors are updated according to equations (1) and (2).

The velocity of each particle, p_i , is updated at iteration $t+1$ according to the equation,

$$v_i^{t+1} = v_i^t + c_1 * r_1 * (pbest_i - s_i^t) + c_2 * r_2 * (gbest_i - x_i^t) \quad (1)$$

where,

i is the current particle

c_1 and c_2 are constant learning factors to control the social influence and global influence,

r_1 and r_2 are random numbers between $[0,1]$ interval,

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

$gbest$ is the best position experienced by any particle in the swarm based on the fitness function, and

x_i is the position in iteration t .

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 based on a sigmoid transformation, S , of the velocity (Figure 1),

for d in 1 to number of variables:

$$S_d = 1 / (1 + \exp(-v_{id}^{t+1})) \quad (2)$$

$$x_{id} = \begin{cases} 1 & \text{if } S_d > r_3 \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

where

x_{id}^t : is the position of the i^{th} agent at time t in dimension d (gene, chemical, etc.),
 v_{id}^{t+1} is the updated velocity of the i^{th} agent at iteration $t+1$ in dimension d ,
 S_{id} is the sigmoid function with values between $[0,1]$ interval for dimension d ,
 r_3 is a random number between $[0,1]$ interval

In the modified binary PSO proposed in this work, users can provide a weight vector to bias the selection process based on expert knowledge or from literature. Studies have shown that incorporating prior knowledge can improve the classification accuracy (He 2010). The random number in (3) is replaced by the weight of the feature $[0 \text{ to } 1]$ to bias the selection process.

Figure 1. Relationship between velocity, sigmoid function, and probability of a feature being selected. The likelihood of a feature being selected increases as the velocity approaches -6 and S approaches 1.

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 (Chuang 2008). Each agent is assigned to one of the four behavioral states $\{C=\text{Confusion}, S=\text{Self-influenced}, N=\text{Influenced by nearest neighbors}, G=\text{Influenced by swarm}\}$ based on the crowd model (Wu 2014). The behavior of the current particle determines which nodes in the swarm network are chosen for interaction and updating velocities as described below:

a) Behavior= N , follows neighbors

$$v_i^{t+1} = v_i^t + c_1 * r_1 * (pbest_i - s_i^t) + c_2 * r_2 * (nbest_i - x_i^t) \quad (4)$$

where

$nbest$ is the centroid (75th percentile) of the k nearest neighbors (default $k=3$),

b) Behavior= G : follows global leader

$$v_i^{t+1} = v_i^t + c_1 * r_1 * (pbest_i - s_i^t) + c_2 * r_2 * (gbest_i - x_i^t) \quad (5)$$

where

$gbest$ is the global best position in the swarm,

c) Behavior= C : moves randomly

$$v_i^{t+1} = v_i^t + c_1 * r_1 * (pbest_i - s_i^t) + c_2 * r_2 * (rpos_i - x_i^t) \quad (6)$$

where

rpos is a random position vector,

d) Behavior=S: only self-influenced

$$v_i^{t+1} = v_i^t + c_1 * r_1 * (pbest_i - s_i^t) \quad (7)$$

The behaviors are updated on regular intervals to prevent the population from getting stuck in local optima (user-defined parameter; default: 5 iterations). The fitness of each particle is evaluated using a nested cross-validation procedure and support vector machine (SVM) classifier as shown in Figure 2 (Vapnik 1998). The algorithm uses the internal cross-validation scheme that performs variable selection based on the training set, $train_k$, in the m-fold scheme and uses the left-out subset, $valid_m$, 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 (Cawley 2010). The search process is terminated when the distance between the centroid of the entire population and the global best agent is less than or equal to 2.

2.2. Evaluation experiments

Two experiments were performed to evaluate the performance of the newly developed framework. In the first experiment, a random probe evaluation was performed using the Iris dataset (Fisher 1950). The original dataset has 4 real features and 150 instances belonging to 3 classes of the iris plant. Each real feature was scale normalized ($\mu = 0, \sigma = 1$) and 36 random features with similar distribution were included in the dataset. The instances were divided into 60% train (90; 30 per class) and 40% test (60; 20 per class). The aim was to evaluate the ability of the stage two of the optSelect algorithm to identify real features. In the second experiment, five publicly available microarray datasets: Leukemia (7129 genes, train samples=38, test samples=34, 2 classes; Golub 1999), SRBCT (2308 genes, train samples=63, test samples=25, 4 classes; Khan 2001), Prostate cancer (6033 genes, train samples=61, test samples=41, 2 classes; Singh 2002), MAQCII-ER and MAQCII-PCR (22284 genes, train samples=130, test samples=100, 2 classes) (Popovici 2010). Limma, rfe-SVM, Lasso, Elasticnet, F-test methods implemented in the CMA R package were used for feature selection in stage one. The two rank aggregation methods implemented in the rankAggreg R package, rankAggreg-Monte Carlo and rankAggreg-GA, that optimize the list of ranked features based on the Spearman's footrule were used for comparison in stage two. The evaluation process was repeated using top 5 and top 15 features from stage one. The final performance of different methods is evaluated using an independent blinded test set that was not seen by the optSelect algorithm during the model building stage. The balanced accuracy, average class-wise prediction accuracy, was used for evaluation purpose (Dreyfus 2006).

3 RESULTS

For experiment 1, 3 out of 4 real features selected in 8 out of 10 folds. 2 were selected in all 10 folds. All random features were discarded. Both train 10-fold cross-validation accuracy and balanced accuracy for the test set were 95%.

The results for experiment 2 are summarized in tables 1 and 2. Table 1 shows the test set balanced accuracies for the test sets from the five datasets using top 5 ranked features selected by different methods (limma, rfe-SVM, Lasso, Elasticnet, F.test) in stage 1 followed by different aggregation methods in Stage 2.

1. Split the training set into M learning and validation sets of size S_{train} and S_{valid} (user-defined; default 80% for S_{train})

2. For learning set $train_m$ and validation set $valid_m$:

Initialize the environment with N (default: 20) agents by randomly selecting features for every agent and assigning random behavior according to the behavior distribution (user-defined)

while $distance(population\_centroid, global\_best) > 1$:

Step 1) For every particle, p, evaluate the following multi-objective function in parallel:

$$fitness_{pk} = w_1 * (CV_{train_k} - CV_{train_{perm}}) + w_2 * (CV_{train_k}) + w_3 * (back\_accuracy) - w_4 * (max\_desired\_features - current\_number\_of\_features) \quad (8)$$

where

$w_1 \Rightarrow$ weight for difference between average k-fold CV vs average permuted k-fold CV from three random splits of $train_m$ (default: 0.7),
 $w_2 \Rightarrow$ weight for 10-fold CV (default: 0.25),
 $w_3 \Rightarrow$ weight for back accuracy (default: 0.05),
 $w_4 \Rightarrow$ controlling the penalty for model complexity (default: 0.01),
 $back\_accuracy \Rightarrow$ using a bootstrap sampling of the $valid_m$ at $fold_m$ for training and the current learning set, $train_m$, as test

Step 2) Determine the local best and global best particle that gives highest fitness

If $no\_change > max\_no\_change\_iterations$, then,
 $behavior\_reset()$,
 $position\_reset()$

Step 3) Update the velocity and position of every agent using equations 2-7,

Store the best of features selected for $fold_m$

Evaluate the prediction accuracy using $train_m$ for training the model and $valid_m$ as test set

Return best set of features and validation accuracy for $fold_m$

3. Calculate stability measure for each feature,

$$SM = \frac{\text{Number of learning sets in which a feature is selected}}{\text{Number of learning sets}}$$

4. Aggregate the results for folds 1 to M, by selecting features at different SM thresholds [0 to 1] and selecting the set with highest fitness value

5. Return the average of the accuracy of validation sets [1 to M], outer CV

6. Report results

Figure 2. optSelect algorithm for nested feature selection

The results show that the ensemble approach improves the classification results as compared to individual methods. For both Leukemia and SRBCT test sets, the optSelect algorithm achieved highest accuracy compared to both individual methods in stage 1 and existing aggregation techniques. For the other three datasets, the optSelect algorithm gave similar performance as compared to individual and existing methods.

Table 2 summarizes the balanced accuracy results for the five datasets using the top 15 ranked features selected by different methods (limm, rfe-SVM, Lasso, Elasticnet, F.test) in Stage 1 followed by different aggregation methods in Stage 2. As discussed earlier, the arbitrary thresholds for selecting top ranked features does not guarantee reproducibility of results on a test set. For instance, the performance of rfe-SVM and Elasticnet degrades by 28% for the SRBCT dataset by increasing the number of selected features to 15. On the contrary, almost all methods showed improvements in accuracy for the Leukemia dataset by using greater number of features. Overall, the aggregation stage improved the classification accuracy with the optSelect algorithm performing comparably or better in almost all cases.

4 DISCUSSION

In recent years, various articles have raised the issue of feature instability and irreproducibility that has hindered the translation of results from basic science to clinical domain (Boulesteix 2009, Abeel 2010, He 2010, Halsley 2015). Several methods for rank aggregation have been proposed (Saeys 2008). However, most of these methods use criteria independent of the classification accuracy for aggregating the results and require selection of an arbitrary threshold for top k features to determine the overlap. This could result in degradation of classification performance on an independent or unseen data set.

Here we propose a novel optimization based nested ensemble feature selection framework, optSelect, which addresses this problem by using a two-stage approach for aggregating the results. Stage one involves selection of top ranked features using different methods. The top list of features from different methods is merged and used as input for a multi agent-based optimization procedure to find the most stable set of features with good classification accuracy. The “optimal” set is determined using a multi-objective fitness function as described in Methods. The algorithm incorporates the concepts of function perturbation and data perturbation to select the most optimal and stable set of features (Boluestrix 2009, He 2010). Additionally, the algorithm is designed to prevent agents from getting stuck in local optima. Each agent interacts with other agents based on their behavior state {confusion, follows neighbors, follows global leader, self-influenced}. Search for optimal solution terminates when the global best and the centroid of the population converge, which is determined using the Euclidean distance.

The performance evaluation of the multi agent-based optimization procedure on the random probe experiment (Experiment 1) shows that the newly proposed behavior based search algorithm combined with the multi-objective classification based fitness function allows detection of relevant features even when majority of the features are randomly generate. Evaluation results for the five gene expression datasets (Experiment 2) show that the newly proposed optSelect framework allows aggregation of results from different methods without compromising for the classification accuracy. The results also highlight the dramatic changes in classification accuracies as a result of arbitrary thresholds for selecting top features. Furthermore, the performance of different independent feature selection techniques varies across different datasets. On the contrary, the optSelect algorithm performed consistently across all datasets and improved the classification results on independent test sets in most cases. The output includes outer CV estimates, optimal set of features, and stability measures for each features. Figure 3 shows the stabil-

ity measures of the 6 features from the optimal set selected by optSelect for the MAQCII-ER dataset, where the samples were classified as ER+ve vs ER-ve. The most stable feature that was reproducibly selected in all folds in the nested feature selection was estrogen receptor 1.

Figure 3. Stability measures for features in the optimal set selected by optSelect.

Limitations and future work: The current study did not assess the effect of using data normalization methods and classifiers on classification accuracy. Escalante et al. have recently showed application of PSO for full model selection (pre-processing methods, feature selection, and classification algorithms). Future work will focus on extending the current framework to full model selection.

5 CONCLUSION

We have developed a novel multi-stage nested ensemble feature selection algorithm, optSelect, which accounts for function perturbation and data perturbation during the feature selection process. The results show that the ensemble framework improves the classification accuracy over independent feature selection methods and existing rank aggregation methods.

ACKNOWLEDGEMENTS

Funding:

REFERENCES

1. Guyon et al. An introduction to variable and feature selection Journal of Machine Learning Research 2003 (3): 1157-1182
2. Lee EK. Large-scale optimization-based classification models in medicine and biology. Ann Biomed Eng 35: 1095–1109, 2007
3. Christin C, Hoefsloot HC, Smilde AK, Hoekman B, Suits F, Bischoff R, Horvatovich P. A critical assessment of feature selection methods for biomarker discovery in clinical proteomics. Mol Cell Proteomics. 2013 Jan;12(1):263-76. doi: 10.1074/mcp.M112.022566. Epub 2012 Oct 31.
4. Reunanen J. Overfitting in Making Comparisons Between Variable Selection Methods. Journal of Machine Learning Research 2003 (3): 1371-1382

5. Cawley GC and Talbot NLC, Over-fitting in model selection and subsequent selection bias in performance evaluation, Journal of Machine Learning Research, 2010. Research, vol. 11, pp. 2079-2107, July 2010.
6. Ma S, Huang J. Penalized feature selection and classification in bioinformatics. Brief Bioinform. 2008 Sep;9(5):392-403. doi: 10.1093/bib/bbn027. Epub 2008 Jun 18.
7. Saeys Y, Inza I, Larrañaga P, A review of feature selection techniques in bioinformatics, Bioinformatics 23 (2007) 2507.
8. Vapnik, V. (1998). The Support Vector Method of Function Estimation. In Nonlinear Modeling (pp. 55-85). Springer US.. Retrieved January 2, 2014, from http://link.springer.com/chapter/10.1007/978-1-4615-5703-6_3
9. Guyon I, Weston J, Barnhill S, and Vapnik V. (2002). Gene Selection for Cancer Classification using Support Vector Machines. Mach Learn, 46(1-3), 389-422. <http://dx.doi.org/10.1023/A:1012487302797>.
10. Zou, H., & Hastie, T. (2005). Regularization and variable selection via the Elastic Net. Journal of the Royal Statistical Society Series B, 67(2), 301-320.
11. Breiman, L. (2001). Random Forests. Mach Learn, 45(1), 5-32. <http://dx.doi.org/10.1023/A:1010933404324>
12. Tibshirani R. Regression shrinkage and selection via the Lasso. J. Roy Stat Soc B 1996;58:267–88.
13. Saeys Y, Abeel T, Peer YVD (2008). Robust Feature Selection Using Ensemble Feature Selection Techniques. In: Machine Learning and Knowledge Discovery in Databases (pp. 313-325). Springer Berlin Heidelberg. Retrieved August 23, 2013, from http://link.springer.com/chapter/10.1007/978-3-540-87481-2_21
14. Boulesteix AL, Slawski M. Stability and aggregation of ranked gene lists. Brief Bioinform. 2009 Sep;10(5):556-68. doi: 10.1093/bib/bbp034. Review.
15. He Z, Yu W. Stable feature selection for biomarker discovery. Comput Biol Chem. 2010 Aug;34(4):215-25. doi: 10.1016/j.compbiolchem.2010.07.002. Epub 2010 Aug 10. Review.
16. Abeel T, Helleputte T, Peer YVD, Dupont P, Saeys Y. (2010). Robust biomarker identification for cancer diagnosis with ensemble feature selection methods. Bioinformatics, 26(3), 392-398. <http://dx.doi.org/10.1093/bioinformatics/btp630>
17. Halsey LG, Curran-Everett D, Vowler SL, Drummond GB. The fickle P value generates irreproducible results. Nat Methods. 2015 Mar;12(3):179-85. doi: 10.1038/nmeth.3288.
18. Kennedy J. and Eberhart R. (1995). "Particle Swarm Optimization". Proceedings of IEEE International Conference on Neural Networks IV. pp. 1942–1948. doi:10.1109/ICNN.1995.488968
19. Chuang LY, Chang HW, Tu CJ, Yang CH. Improved binary PSO for feature selection using gene expression data. Comput Biol Chem. 2008 Feb;32(1):29-37. Epub 2007 Sep 25.
20. Slawski et al. Classification for Microarrays – a comprehensive Bioconductor package for supervised classification with high dimensional data BMC Bioinformatics 2008 Vol. 9 pp439

21. Macal CM and North MJ. Tutorial on agent-based modeling and simulation. Journal of Simulation. 2010. (4). 151-162.
22. Wu S, Sun Q. Computer simulation of leadership, consensus decision making and collective behaviour in humans. PLoS One. 2014 Jan 17;9(1):e80680. doi: 10.1371/journal.pone.0080680. eCollection 2014.
23. 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
24. Khan et al. Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks Nature Medicine June 2001: Volume 7 no. 6, pp 673-679
25. Singh D, Febbo PG, Ross K, Jackson DG, Manola J, Ladd C, Tamayo P, Renshaw AA, D'Amico AV, Richie JP, Lander ES, Loda M, Kantoff PW, Golub TR, Sellers WR. Gene expression correlates of clinical prostate cancer behavior. Cancer Cell. 2002;1(2):203–209.
26. Popovici V, Chen W, Gallas BG, Hatzis C, Shi W, Samuelson FW, Nikolsky Y, Tsyganova M, Ishkin A, Nikolskaya T, Hess KR, Valero V, Booser D, Delorenzi M, Hortobagyi GN, Shi L, Symmans WF, Pusztai L. Effect of training-sample size and classification difficulty on the accuracy of genomic predictors. Breast Cancer Res. 2010;12(1):R5. doi: 10.1186/bcr2468. Epub 2010 Jan 11.
27. Dreyfus G and Guyon I. Chapter 2: Assessment Methods. Feature Extraction. Studies in Fuzziness and Soft Computing. Vol 207. 2006

Table 1. Using top 5 ranked features in Stage 1. The balanced accuracies for the independent test sets are reported here.

	Method\Dataset	Leukemia	SRBCT	MaqcII_ER	MaqcII_PCR	Prostate
Stage 1	Limma	0.89	0.94	0.87	0.7	0.77
	rfe-SVM	0.94	0.92	0.88	0.66	0.77
	Lasso	0.94	0.9	0.87	0.62	0.75
	Elasticnet	0.89	0.94	0.87	0.72	0.77
	F.test	0.93	0.94	0.87	0.66	0.77
Stage2 (Aggregation)	rankAggreg-monte carlo	0.89	0.92	0.87	0.74	0.77
	rankAggreg-GA	0.89	0.94	0.87	0.64	0.77
	optSelect	0.96	1	0.87	0.72	0.77

Table 2. Using top 15 ranked features in Stage 1. The balanced accuracies for the independent test sets are reported here.

	Method\Dataset	Leukemia	SRBCT	MaqcII_ER	MaqcII_PCR	Prostate
Stage 1	limma	0.96	0.99	0.88	0.71	0.77
	rfe-SVM	0.96	0.661	0.86	0.68	0.77

Error! No text of specified style in document.

	Lasso	0.96	1	0.87	0.64	0.77
	Elasticnet	0.96	0.66	0.88	0.68	0.77
	F.test	0.96	0.986	0.88	0.66	0.77
Stage2						
(Aggregation)	rankAggreg-monte carlo	0.94	1	0.88	0.75	0.77
	rankAggreg-GA	0.96	0.96	0.87	0.73	0.77
	optSelect	0.96	1	0.9	0.73	0.83