Supplementary For MGRFE: multilayer recursive feature elimination based on an embedded genetic algorithm for cancer classification

Cheng Peng, Xinyu Wu, Wen Yuan, Xinran Zhang, Yu Zhang, and Ying Li

S1. The datasets used in this study

This study used total 19 benchmark microarrays subdivided into two large Datasets to validate the performance of our proposed MGRFE. In Tables 1 and 2, we provide the brief description of each dataset in Dataset One and Dataset Two.

S2. Pseudocode of the proposed MGRFE

In the main manuscript, we provided the flow chart of the proposed MGRFE in Fig. 2. Here, we supplement the pseudocodes of our methodology. Pseudocode 1 describes the complete procedure of MGRFE. Pseudocodes 2 and 3 explain the two key processes of MGRFE: GaRFE and embedded GA.

S3. Implementation notes and Computation time

We implemented the proposed MGRFE in Python version 3.6.0 environment (https://www.python.org/) on a common laptop computer with Intel(R) Core(TM) i5-4210U CPU and 8G memory. The Python SciPy package version 0.19.0 [3] was involved in the t-test process, and minepy package version 1.2.0 [4] was used to perform the MIC calculation. Some parameter settings about MGRFE: 1) the evolution iteration number of embedded GA was dynamically set to 1 to 3 (smaller iteration number used for larger chromosome length to save time); 2) the reduced feature number between two GA runs, the RFE step, was dynamically set to 1 to 3 (larger reduction step used for larger chromosome length to save time); and 3) the iterative layer number of MGRFE being 3 with

Pseudocode 1: MGRFE: multilayer iterative feature selection using GaRFE

```
Input  : A microarray gene expression data
Output: The optimal gene feature combination
        for phenotype classification
The $t$-test-based gene ranking to generate the candidate gene set $G$;
The MIC-based gene ranking to narrow $G$;
Set $GC$, the list of optimal gene combinations in MGRFE, to empty;
while the maximal iterative layer number not reached do
    Initialize and run a layer of GaRFE (Pseudocode 2) based on $G$;
    for each GaRFE do
        Add the returned optimal gene combinations to $GC$;
    Sort the optimal gene combinations in $GC$ and only preserve the top ranked ones;
    Use the genes in the top ranked gene combinations in $GC$ to form a reduced $G$;
Multiple k-fold CV on the gene combinations in $GC$;
Return the final selected gene combination;
```

- *C. Peng, X. Wu, W. Yuan, X. Zhang, Y. Zhang, and Y. Li are with the College of Computer Science and Technology, Key Laboratory of Symbol Computation and Knowledge Engineering of Ministry of Education, Jilin University, Changchun 130012, China.*
- *Y. Li is the correspondence author. Email: liying@jlu.edu.cn.*

TABLE 1
Summary of the 17 binary classification datasets in Dataset One from ref. [1]

ID	Dataset	Samples	Features	Summary
1	DLBCL[1]	77	7 129	DLBCL patients (58) and follicular lymphoma (19)
2	Pros(Prostate)[1]	102	12 625	prostate (52) and non-prostate (50)
3	Colon[2]	62	2 000	tumour (40) and normal (22)
4	Leuk(Leukaemia)[2]	72	7 129	ALL (47) and AML (25)
5	Mye(Myeloma)[3]	173	12 625	presence (137) and absence (36) of focallesions of bone
6	ALL1[1]	128	12 625	B-cell (95) and T-cell (33)
7	ALL2[1]	100	12 625	patients that did (65) and did not (35) relapse
8	ALL3[1]	125	12 625	with (24) and without (101) multidrug resistance
9	ALL4[1]	93	12 625	with (26) and without (67) the t(9;22) chromosome translocation
10	CNS[1]	60	7 129	medulloblastoma survivors (39) and treatment failures (21)
11	Lym(Lymphoma)[1]	45	4 026	germinalcentre (22) and activated B-like DLBCL (23)
12	Adeno(Adenoma)[1]	36	7 457	colon adenocarcinoma (18) and normal (18)
13	Gas(Gastric)[3]	65	22 645	tumors (29) and non-malignants (36)
14	Gas1(Gastric1)[3]	144	22 283	non-cardia (72) of gastric and normal (72)
15	Gas2(Gastric2)[3]	124	22 283	cardia (62) of gastric and normal (62)
16	T1D[3]	101	54 675	T1D (57) and healthy control (44)
17	Stroke[3]	40	54 675	ischemic stroke (20) and control (20)

In Tables 1 and 2, ”Samples” and ”Features” indicate the total sample number and feature number of each dataset, and ”Summary” column describes the sample classes and the related sample numbers in parenthesis.

[1] These datasets were retrieved from http://www.broadinstitute.org/cgi-bin/cancer/datasets.cgi.

[2] Colon and Leuk datasets were downloaded from the R/Bioconductor packages *colonCA* and *golubEsets*, respectively.

[3] These datasets were downloaded from https://www.ncbi.nlm.nih.gov/geo/.

TABLE 2
Summary of the 3 classification datasets in Dataset Two from ref. [2]

ID	Dataset	Classes	Samples	Features	Summary
1	SRBCT[1]	4	88	2 308	EWS (29), NHL (11), NB (18) and RMS (25)
2	ALL_AML[2]	2	72	7 129	ALL (47) and AML (25)
3	MLL[3]	3	72	12 582	ALL (24), MLL (20) and AML (28)

[1] SRBCT dataset was downloaded from http://research.nhgri.nih.gov/microarray/Supplement/. This dataset includes 88 samples totally, but five of them are irrelevant and thus only 83 samples were used.

[2] ALL_AML in Dataset Two and Leuk in Dataset One are the same dataset in actual.

[3] MLL dataset was retrieved from http://portals.broadinstitute.org/cgi-bin/cancer/publications/pub_paper.cgi?paper_id=63.

Pseudocode 2: GaRFE: recursive feature elimination with embedded GA

```
Input  : Candidate gene set $G$, Maximal chromosome length $L$
Output: The optimal gene feature combinations in GaRFE
Randomly generate the first GA population $P$ from $G$ with chromosome length equal to $L$;
Set $GC$, the list of optimal gene combinations in GaRFE, to empty;
do
    Execute embedded GA (Pseudocode 3) using the population $P$;
    Add the returned gene combinations by GA to $GC$;
    if the current chromosome length $> 1$ then
        Chromosome length drop: each individual in $P$ randomly discard several genes with the number equal to the RFE step;
while the current chromosome length $\geqslant 1$;
Sort the optimal gene combinations in $GC$ and only preserve the top ranked ones;
Return the optimal gene combinations in $GC$;
```

Pseudocode 3: Embedded GA

```
Input  : GA population $P$, Maximal evolution times $T$
Output: Updated $P$, The optimal gene feature combinations in GA
Set $GC$, the list of optimal gene combinations in GA, to empty;
while the maximal evolution times $T$ not reached do
    Perform mutation operator;
    Perform crossover operator;
    Fitness calculation of each GA individual by k-fold CV;
    Truncation selection to form the updated $P$;
Sort the GA individuals in $P$ and select the top ones to form $GC$;
Return the updated population $P$ and the optimal gene combinations in $GC$;
```

three, two and one GaRFE processes at each layer is usually enough. In the experiments, we limited the size of the final selected gene combination in each dataset to below 10 genes. According to the experiment records, in each of the 19 microarray datasets, the running time of MGRFE is commonly between 500 seconds (8.33 minutes) and 900 seconds (15 minutes). Additionally, it is well worthy to mention that the final chosen gene subset in each dataset might be already found by the first GaRFE process in the first layer of MGRFE, which just costs 2∼3 minutes. More implementation details and experiment results of MGRFE in the 19 datasets are available at https://github.com/Pengeace/MGRFE-GaRFE.

Because Kar *et al.* also employed an evolutionary-computation method PSO, which is similar to GA, to select minimal informative genes in microarray and provided their program running time records on three datasets SRBCT, ALL_AML, and MLL [2], here, we offer a simple running time comparison between their method and MGRFE. Their PSO-based method cost 2.7956, 2.7906 and 7.1488 hours on SRBCT, ALL_AML and MLL respectively to find their optimal gene subsets. In contrast, MGRFE merely used 10.8230, 9.0108 and 8.8739 minutes respectively in the same three datasets and thus showed much higher converge speed. Moreover, according to Tables 4, 5 and 6 in the main manuscript, the gene subsets selected by MGRFE had smaller sizes but higher classification accuracies compared with Kar *et al.*'s method. We noted that Kar *et al.* didn't employ the filter techniques to cut down the feature search space and their binary-coded PSO demanded high time cost to converge when dealing with the thousands of genes in each microarray.

S4. Performance of MGRFE in 10-time 10-fold CV

In the main manuscript, the performance of MGRFE on the two large Datasets in 10-time 10-fold cross validation (CV) are shown in the box-plot form. Here, we supplement the detailed mean accuracies ($Mean\ Acc$s) and standard deviations ($S.D.$s) of MGRFE on all the 19 datasets. In each dataset, 10-fold CV is repeated 10 times based on different random seeds. In each 10-fold CV, the mean accuracy value in 10-fold is calculated and recorded. Then after 10 repetitions of the 10-fold CV, the $Mean\ Acc$ and $S.D.$ of MGRFE in a dataset is calculated from the recorded total 10 mean accuracy values.

S5. The gene probes selected by MGRFE

The gene probes finally selected by MGRFE on all the 19 datasets are listed in Table 4. These differentially

TABLE 3
$Mean\ Acc$ and $S.D.$ of MGRFE on 19 benchmark datasets in 10-time 10-fold CV

Dataset	DLBCL	Pros	Colon	Leuk	Mye	ALL1	ALL2	ALL3	ALL4	CNS
$Mean\ Acc$	0.987	0.979	0.971	0.982	0.933	0.998	0.880	0.920	0.963	0.980
$S.D.$	0.007	0.003	0.012	0.007	0.011	0.004	0.013	0.007	0.010	0.007
Dataset	Lym	Adeno	Gas	Gas1	Gas2	T1D	Stroke	SRBCT	MLL	
$Mean\ Acc$	1.000	1.000	1.000	0.974	1.000	0.897	1.000	1.000	0.997	
$S.D.$	0.000	0.000	0.000	0.004	0.000	0.014	0.000	0.000	0.006	

Note that the $Mean\ Acc$ and $S.D.$ of MGRFE in dataset ALL_AML are same as the records in Leuk for these two are the same dataset in actual.

TABLE 4
The gene probes finally selected by MGRFE on the 19 microarray datasets

Dataset	Probe number	Gene probes
DLBCL	3	[*X69433_at*, *Z84497_s_at*, *M15205_at*]
Pros	4	[*37639_at*, *38634_at*, *1909_at*, *37537_at*]
Colon	6	[*Hsa.36952*, *Hsa.36696*, *Hsa.94*, *Hsa.442*, *Hsa.5226*, *Hsa.5756*]
Leuk	2	[*M23197_at*, *M31523_at*]
Mye	7	[*35977_at*, *33130_at*, *31366_at*, *34571_at*, *38013_at*, *1368_at*, *41150_r_at*]
ALL1	1	[*38319_at*]
ALL2	8	[*37502_at*, *39885_at*, *1291_s_at*, *39408_at*, *1838_g_at*, *819_at*, *31331_at*, *39336_at*]
ALL3	8	[*38907_at*, *38478_at*, *34284_at*, *37693_at*, *201_s_at*, *34497_at*, *37809_at*, *41259_at*]
ALL4	6	[*39631_at*, *38119_at*, *36795_at*, *36873_at*, *39905_i_at*, *1265_g_at*]
CNS	7	[*S76475_at*, *M96739_at*, *X64624_s_at*, *X93511_s_at*, *K01911_at*, *S78693_f_at*, *X78565_at*]
Lym	3	[*GENE3332X*, *GENE3261X*, *GENE1191X*]
Adeno	1	[*D43636*]
Gas	3	[*225571_at*, *236118_at*, *237466_s_at*]
Gas1	3	[*213125_at*, *41037_at*, *208897_s_at*]
Gas2	2	[*212344_at*, *210766_s_at*]
T1D	7	[*1566232_at*, *215728_s_at*, *215612_at*, *226585_at*, *239474_at*, *219870_at*, *244223_at*]
Stroke	4	[*1567009_at*, *240084_at*, *239389_at*, *233835_at*]
SRBCT	5	[*245330.0*, *784257.0*, *43733.0*, *784224.0*, *295985.0*]
MLL	3	[*38242_at*, *37710_at*, *1389_at*]

expressed genes could be potential biomarker candidates that are useful to related phenotype researches.

REFERENCES

[1] R. Ge, M. Zhou, Y. Luo, Q. Meng, G. Mai, D. Ma, G. Wang, and F. Zhou, "Mctwo: a two-step feature selection algorithm based on maximal information coefficient," *BMC bioinformatics*, vol. 17, no. 1, p. 142, 2016.

[2] S. Kar, K. D. Sharma, and M. Maitra, "Gene selection from microarray gene expression data for classification of cancer subgroups employing pso and adaptive k-nearest neighborhood technique," *Expert Systems with Applications*, vol. 42, no. 1, pp. 612–627, 2015.

[3] E. Jones, T. Oliphant, and P. Peterson, "Scipy: Open source scientific tools for python," 2014.

[4] D. Albanese, M. Filosi, R. Visintainer, S. Riccadonna, G. Jurman, and C. Furlanello, "minerva and minepy: a c engine for the mine suite and its r, python and matlab wrappers," *Bioinformatics*, vol. 29, no. 3, pp. 407–408, 2013. [Online]. Available: ⟨GotoISI⟩://WOS:000314892000022