

Application of Genetic Algorithm/K-Nearest Neighbor Method to the Classification of Renal Cell Carcinoma

Dongqing Liu¹, Ting Shi², Joseph A. DiDonato², John D. Carpten³, Jianping Zhu⁴, Zhong-Hui Duan^{1*}

¹Department of Computer Science, University of Akron, Akron, OH 44325
ldongqi@uakron.edu, duan@uakron.edu

²Department of Cancer Biology, Cleveland Clinic Foundation, Cleveland, OH 44195
shiti@ccf.org, didonaj@ccf.org

³Genetic Basis of Human Disease Research Division, Translational Genomics Research Institute, Tempe, AZ 85281
jcarpten@tgen.org

⁴Department of Theoretical and Applied Mathematics, University of Akron, Akron, OH 44325
jzhu@math.uakron.edu

*To whom correspondence should be addressed

Abstract

In this study, we use a genetic algorithm and k-nearest neighbor method to classify two subtypes of renal cell carcinoma using a set of microarray gene expression profiles of nine samples (three clear cell tumors and six papillary tumors). We show that the genetic algorithm/k-nearest neighbor method can be efficiently used in identifying a panel of discriminator genes. To test the robustness of the algorithm, we perform a bootstrapping analysis that removes one sample from the data set at a time and uses the remaining samples for gene selection. We show that each of the removed samples can be classified correctly. We also analyze the stability of the algorithm and the sensitivity of the algorithm with respect to different samples.

1. Introduction

Renal cell carcinoma (RCC) consists of several different subtypes [1]. Clear cell RCC is the most common one. The genetic alterations are characterized by mutation or hypermethylation of the Von Hippel-Landau gene. Papillary RCC is the second most common subtype. The genetic alterations of the papillary RCC involve the activation of the MET proto-oncogene and trisomy chromosomes 7 and 17. It is expected that the gene expression profiles of the two subtypes are also distinctive and the subtypes can be identified based on the expressions of a panel of genes. The goal of this study is to identify the panel of discriminator genes (PDG) using a genetic algorithm (GA) and the k-nearest neighbor (KNN) method.

2. GA and KNN method

A genetic algorithm improves existing approximate solutions using crossover and mutation as in the natural selection process [2]. This algorithm is used in our study to evolve selected chromosomes, each of which

has 30 genes. The fitness of a chromosome c is measured by its ability to classify samples of known subtypes. For each sample s_i , the expression levels of 30 genes in c constitute a vector v_i . The distance between sample s_i and s_j can be calculated as $d_{ij} = \|v_i - v_j\|$, and the classification of the samples can be accomplished using the KNN method based on the distances calculated [3]. Note that different distance metrics, for example the Euclidean distance and the Pearson distance, can be used in the calculation of d_{ij} .

3. Implementation

The main steps in the algorithm are:

1. Select randomly 100 initial chromosomes to start the evolution process
2. Calculate fitness scores, which is the ratio of the number of samples correctly classified to the total number of samples. A chromosome is considered near optimal (NOC) and kept in a separate pool for later use if it can classify all samples correctly. Note the criterion can be adjusted to less than 100% correct classification based on the level of difficulty of the problem.
3. Evolve selected chromosomes by mutation and crossover if more NOCs are needed.
4. Terminate the process if there are enough NOCs in the pool. In the study reported here, 6000 chromosomes are used in each run.
5. Select PDG from top genes based on the frequency they appear in 6000 NOCs.

The expression data used in this study also has quality scores (signal/noise ratio) that can be used in step 2 and 5 to adjust the fitness scores and selection of PDGs.

4. Analysis of results

All samples can be correctly identified using a PDG of 30 top genes from 6000 NOCs regardless the type of

distance or whether quality scores are used. We also use a bootstrap strategy to test the robustness of our algorithm: We remove one papillary cell RCC sample at a time, use the remaining five papillary cell and three clear cell RCC samples to select the PDG, and then classify the removed sample using the selected PDG. Again, all six samples can be correctly identified regardless the type of distance or whether quality scores are used.

4.1 Stability of the algorithm

Given the random nature of any GA-based method, we are concerned about the stability of the algorithm, i.e. how the final result changes from different runs of the same code. For example, in six runs of the same code using all 9 samples with Euclidean distance and no quality scores, there are about 80% - 90% common genes for each pair of different runs. Across all six runs, only 67% of the top 30 genes are in common. This inconsistency makes it difficult to select a PDG that will provide correct classification in different runs.

To improve the consistency of the results and stability of the algorithm, we add an intersection step in the selection of top genes for the PDG: Run the code m times, select the set $S_i, i=1, \dots, m$, of 50 top genes based on the frequency they appeared in the near optimal chromosomes, and then take the intersection of $S_i, C=\cap S_i$, as the PDG. From 6 runs of the same code with the same parameters, we are able to select a PDG of 30 discriminatory genes, common to all six runs, which can classify all samples correctly. This approach significantly improves the consistency of the results from different runs of the same code. Note that the size of S_i and the parameter m can be adjusted and the intersection C can be taken for fewer than m sets.

4.2 Sensitivity of PDG to different samples

Another issue that needs to be considered is the sensitivity of the PDG to changes of samples. Ideally, the PDG should remain approximately the same for the same subtypes of tumors. We select six PDGs by removing one of the six papillary cell RCC samples at a time using the Euclidean distance. Each PDG has 30 genes. Although all six PDGs can be used to identify all 9 samples correctly, there are only 53% - 80% common genes between each pair of PDGs, and 53% common genes across all 6 PDGs.

To reduce this sensitivity of the PDG to different samples, we use the similar approach discussed in 4.1: Remove one sample $P_i, i=1, \dots, 6$, at a time from the 6 papillary cell RCC samples, select the set $S_i, i=1, \dots, 6$, of 50 top genes using the five papillary cell and three clear cell RCC samples based on the frequency they

appear in the near optimal chromosomes, and then take the intersection of $S_i, C=\cap S_i$, as the PDG. Note that we can augment the PDG by taking intersections of five or four of all $S_i, i=1, \dots, 6$. For the 6 runs described early, the final PDG of 30 genes contains 22 genes that are common to all six S_i 's and 8 genes that are common to at least five S_i 's. The final PDG of 30 discriminatory genes can classify all samples correctly.

4.3 Effects of quality scores and different distance metrics on the PDG

The quality score is an indication of the reliability of a particular gene expression level. We use it to give more reliable data higher weight in the calculation. The selected PDGs with and without quality scores for the cases we tested demonstrate about 10% differences.

We have observed significant differences in the top genes selected using Euclidean and Pearson distances. In various test cases, only 45% - 50% of the top 50 genes from 6000 NOCs using the two distances are in common. Further, the final PDGs generated by the Euclidean and Pearson distances are not equivalent. For example, the PDG generated using the Euclidean distance can classify all 9 samples correctly using the Pearson distance, while the PDG generated by the Pearson distance can only identify 7 of the 9 samples correctly using the Euclidean distance.

5. Conclusions

We have shown that the GA/KNN method can be an effective tool for classifying different subtypes of RCC. By using intersections of multiple gene sets from different runs or different samples, we can improve the consistency of the results and reduce the sensitivity of the PDGs to different samples. Future studies include testing of the method using more complex data set, in depth analysis of the effect of different distances on the selection of PDGs, and possible development of more suitable distance metrics for microarray data analysis.

6. References

1. P.C. Walsh, A. B. Retik, E. D. Vaughan, A. J. Wein, L. R. Kavoussi, A. C. Novick, A. W. Partin, C. A. Peters, *Campbell's Urology*, 8th ed, Saunders, New York, 2003.
2. D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley, Boston, Massachusetts, 1989.
3. L. Li, C. R. Weinberg, T. A. Darden, and L. G. Pedersen, Gene selection for sample classification based on gene expression data: study of sensitivity to choice of parameters of the GA/KNN method, *Bioinformatics*, 17:1131-1142, 2001.