

# CAPIES: DNA Microarray-Based Class Prediction System for Computational Diagnostics

Sung Geun Lee<sup>1</sup>  
sglee@istech21.com

Bonghee Seo<sup>1</sup>  
bhseo@istech21.com

Yang Seok Kim<sup>1,2</sup>  
yskim@istech21.com

<sup>1</sup> Bioinformatics Unit, ISTECH Inc. #704, Hyundai Town Vill, 848-1 Janghang-dong, Ilsan-gu, Goyang city, Gyeonggi-do, 411-380, Korea

<sup>2</sup> Cancer Metastasis Research Center, Yonsei University College of Medicine, 134 Shinchon-dong, Seodaemun-gu, Seoul, 120-752, Korea

**Keywords:** gene selection, classification, generalization error

## 1 Introduction

DNA microarray technologies have given new opportunities and insights in medical applications [2]: clinical diagnosis, prognosis and tumor subtype classification. The advantage of DNA chip data analysis has been successfully demonstrated in classification of complex diseases like cancer that is involved with complicated genetic mechanisms. CAPIES is a DNA microarray-based class prediction program. As an integrated class prediction tool, it includes such modules as preprocessing, gene selection, classification, permutation test and generalization error estimation. The pipeline, starting from gene selection via sample classification to error estimation, provides a one-stop solution for DNA microarray-based class prediction problems esp. in disease diagnostics.

## 2 Method and Results

The individual modules of CAPIES work both independently and interactively. Each module contains its own graphic view and can be connected as a linkage for various combinations among algorithms. Preprocessing deals with file management or processing such as editing, filtering, and imputation. Gene selection is to single out the genes that are differentially expressed in some specific classes: i.e. finding out a class-characterizing set of genes. It can help increase the accuracy of classification. In microarray design, gene selection can be beneficial in cost reduction since smaller number of genes would cost less, computationally or financially. CAPIES provides five effective options for gene selection: Null, BSS/WSS, Regularized t-test, Modified Wilcoxon test, and User-defined selection. The supervised classification section in CAPIES contains three algorithms: KNN (K Nearest Neighbor), Nearest Centroid, and an adaptive machine learning method. The generalization error estimation of CAPIES will help users to decide the degree of belief for the classification results. The widely used LOOCV is not enough to guarantee the reliability of generalization error of given classifiers [1, 3]. CAPIES provides three approaches: LOOCV, stratified k-fold, and Bootstrap coupled with permutation test. In statistical aspects, these approaches are complementary so that accuracy bias may be avoided.

CAPIES is an easy-to-use GUI tool (Fig. 1). It is implemented using C++ and Microsoft's IDE (Integrated Development Environment) VC++ 6.0 for computational speed maximization. Consequently, the target platforms of CAPIES are oriented to Windows. To make the system as flexible and extensible as possible, we spent much time in the analysis and design phase of the software engineering process. We stringently followed the OOAD (Object-Oriented Analysis and Design) principles and used UML (Unified Modeling Language) as the notational tool. As the entry point and communication channel to the internal system, our tool has a simple and easy-to-understand interface. The menu system consists of representative items for each functional subsystem. Routine operations could

be performed using the toolbar which can float anywhere on the user's desktop or can be docked anywhere in the main window. The project bar that shows current project's status is also a floating window and so can be either docked or floated according to the user's preference. Every result view is displayed in a consistent manner with the algorithms and their parameters heading first. Classification results are displayed with detailed information for why each sample was assigned to the specific class. Individual projects could be saved as separate project files for later usage and further investigation. So you need not start a new project if the data sets that are training data or test data are the same. The time-saving effects of this functionality would be tremendous in a long-time scale. The system uses a CSV (Comma Separated Value)-like text file format for the input microarray expression data. The format can use tab character in place of comma as the column delimiter.

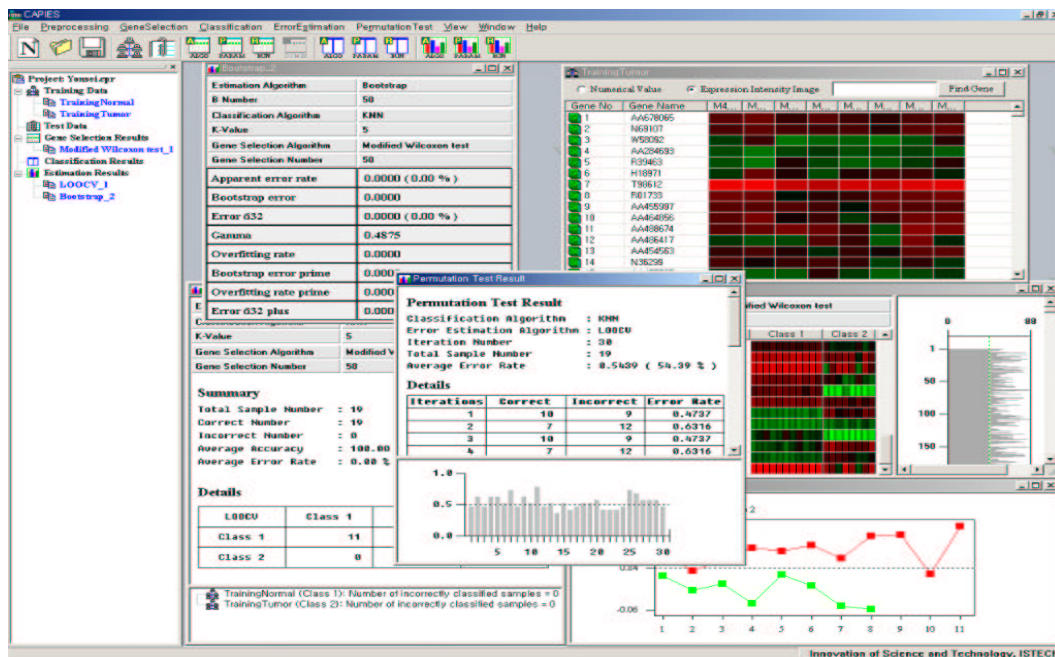


Figure 1: User interface of CAPIES. Toolbar and project-bar are all dockable windows, which means that they can be docked or floated according to the user's preference. The toolbar contains shortcut buttons for routine operations. The project-bar shows current configuration of a project. Double-clicking an item in the project-bar brings out a view specific to the item type. Due to spatial limitation, this figure shows only a few of the various graphic views CAPIES provides. Three types of error estimation result view are shown: Bootstrap, LOOCV, and Permutation test. As can be seen in the figure, gene selection result views and training data views show gene-expression image. The view named 'TrainingTumor' is an example of training data view. A gene selection result view that used 'Modified Wilcoxon Test' as a feature selection method is also shown in this figure. The view in the right-bottom corner of the figure, which can be shown by double-clicking an item in the gene selection result list, is a gene expression profile graph view. With this graph view, how well a gene is differentially expressed in training classes can be checked intuitively and easily.

## Acknowledgments

This work was supported by the Ministry of Health and Welfare of Korea.

## References

- [1] Ambrose, C. and McLachlan, G.J., Selection bias in gene extraction on the basis of microarray gene expression data, *Proc. Natl. Acad. Sci. USA*, 99:6562–6566, 2002.
- [2] Berns, A., Gene expression in diagnosis, *Nature*, 403:491–492, 2000.
- [3] Radmacher, M.D., McShane L.M., and Simon R., A paradigm for class prediction using gene expression profiles, *J. Comput. Biol.*, 9:505–511, 2002.