Proceedings of the 3rd International Workshop on Distributed Statistical Computing (DSC 2003)
March 20–22, Vienna, Austria *ISSN 1609-395X*
Kurt Hornik, Friedrich Leisch & Achim Zeileis (eds.)
http://www.ci.tuwien.ac.at/Conferences/DSC-2003/

Prediction Analysis of Microarrays in Excel

Balasubramanian Narasimhan

1 Introduction

Prediction Analysis of Microarrays (PAM) is a statistical technique for class prediction using gene expression data using shrunken centroids. It is described in Tibshirani, Hastie, Narasimhan, and Chu (2002). The method of nearest shrunken centroids identifies subsets of genes that best characterize each class. The technique is general and can be used in many other classification problems.

PAM Software for the R (Ihaka and Gentleman, 1996) has been available for some time now from the `http://www-stat.stanford.edu/~tibs/PAM`. Using the software, one can train the classifier, perform cross validation to get an idea of the value to use for thresholding, and do predictions. There has been some demand for a GUI version for PAM.

Our experience with SAM (Significance Analysis of Microarrays, Tusher, Tibshirani, and Chu, 2001) software led us to use Excel as a GUI for the package. SAM was written in Visual Basic, using a Java COM library at the core. However, it was clear that this approach would not be an efficient one in the long run for several reasons. Among them:

- Many statistical tools would have to be rewritten in Java, or Visual Basic wasting valuable development time rather than leveraging already available tools.
- The outcome of the market place battle, Microsoft Java versus Sun Java, is hardly clear. Besides, programming in Microsoft Java is like programming with one hand tied behind your back.
- While Microsoft's commitment to DCOM seems quite firm, its commitment to Java COM seems to have stalled.

At DSC 2001, Neuwirth and Baier (2001) demonstrated an R DCOM server and client. We decided to exploit this technology for PAM to produce an environment much like that of the SAM except with R as the computation engine.

Section 2 describes how the PAM software is organized. Section 3 describes some of the capabilities of the software via an example. Section 4 describes some issues that we faced and future work.

2 The structure of PAM

Since the Excel part of PAM is merely a front-end to the R package, one must install *PAM for R* package first. Then, as is customary for Windows, a point and click `Setup` package is provided to install the Excel addin. Once installed, the user has to activate the addin by using the `Tools` menu and loading the addin.

The Excel GUI is written in Visual Basic using object-oriented VBA throughout. To ensure that a wide audience would be able to use the software, we eschewed newer features and wrote to an Excel 97 target. The software has been tested with R version 1.6.1 on the following platforms: Excel 97 and Windows NT, Excel 2000 and Windows 98, Excel 2000 and Windows ME, Excel 2000 and Windows XP, Excel 2002 and Windows XP.

The software comes with several examples and documentation.

3 A quick tour of PAM

When PAM is installed, two buttons appear in the Excel toolbar titled **PAM** and **PAM Controller**. PAM uses a data format very similar to SAM (see Figure 1). Apart from some leading rows that contain information such as class labels and sample labels, a row corresponds to a gene and a column corresponds to an observation. Like SAM, the data is assumed to be normalized.

One highlights an area of the spreadsheet that represents the data. Then clicking on the **PAM** button brings up the dialog shown in Figure 2. A brief explanation of the fields shown in the dialog.

Class Labels Specifies the row number that contains the class labels.

Sample Labels Specifies the row number that contains the sample labels. Sample labels are optional and so the field can left blank although they are recommended as a means of identifying samples.

Batch Labels Specifies the row containing the batch labels, if any. Batch labels allow one to combine expression data from different experiments.

Imputation Engine PAM handles missing data (denoted by blank cells) by imputing with K-Nearest Neighbors. One can change the number of neighbors required for the imputation. The default is 10.

Web Link Option PAM can hyperlink worksheet cells containing gene ids to the SOURCE database at Stanford so that one can easily search the web database for other information about the gene.

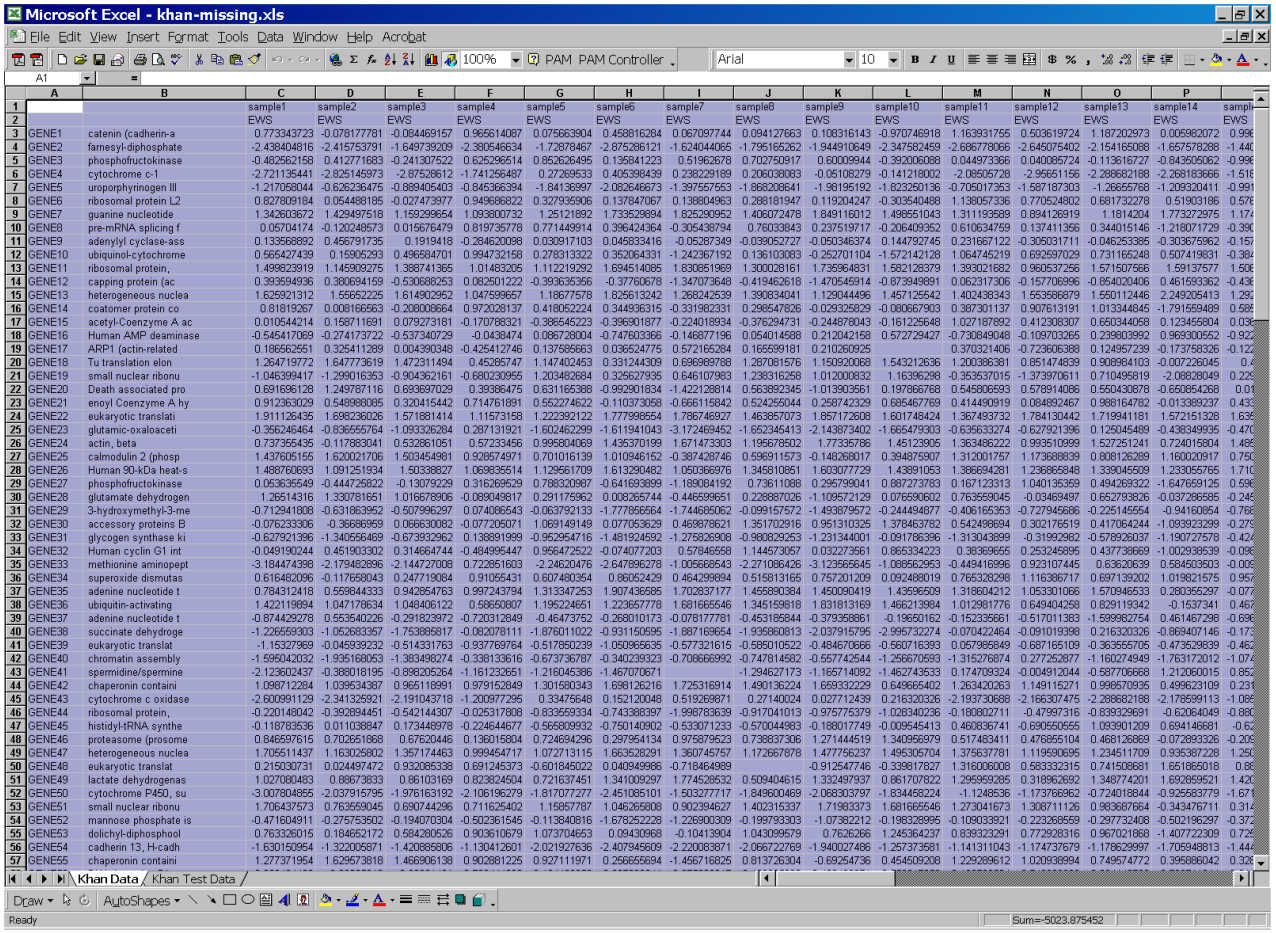

Figure 1: Data format and highlighting data

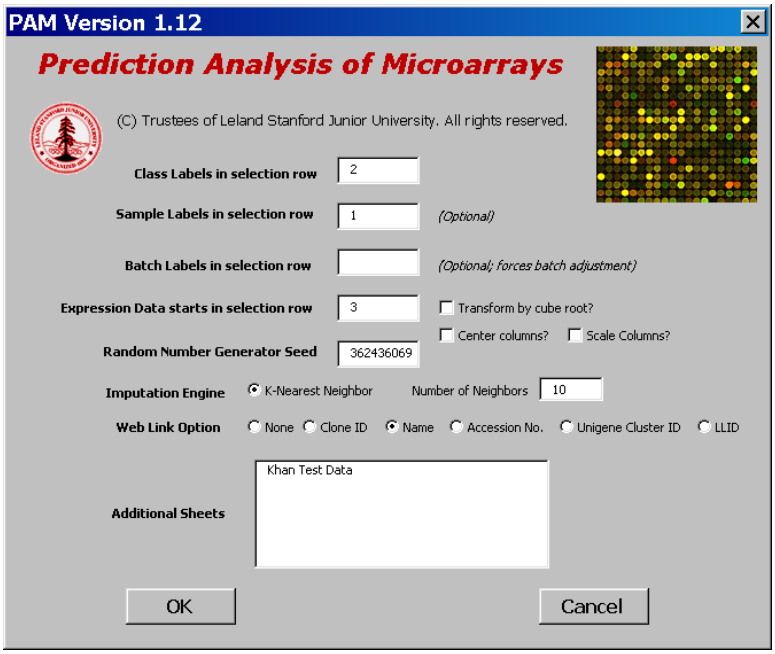

Figure 2: The PAM Dialog Box

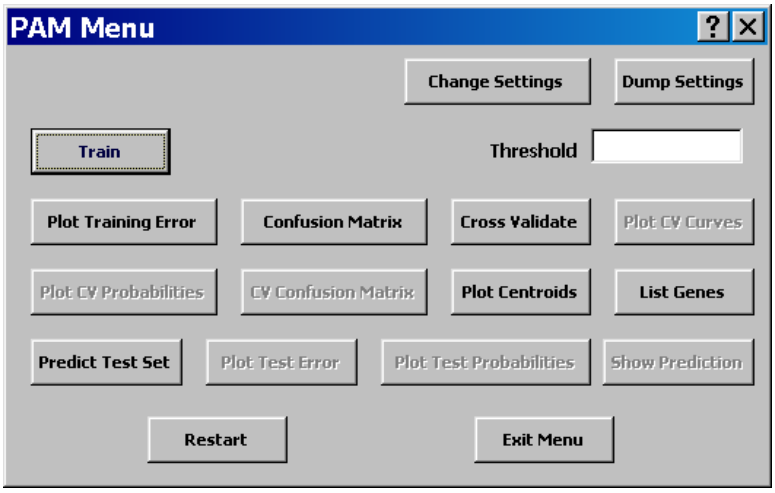

Figure 3: The PAM Controller

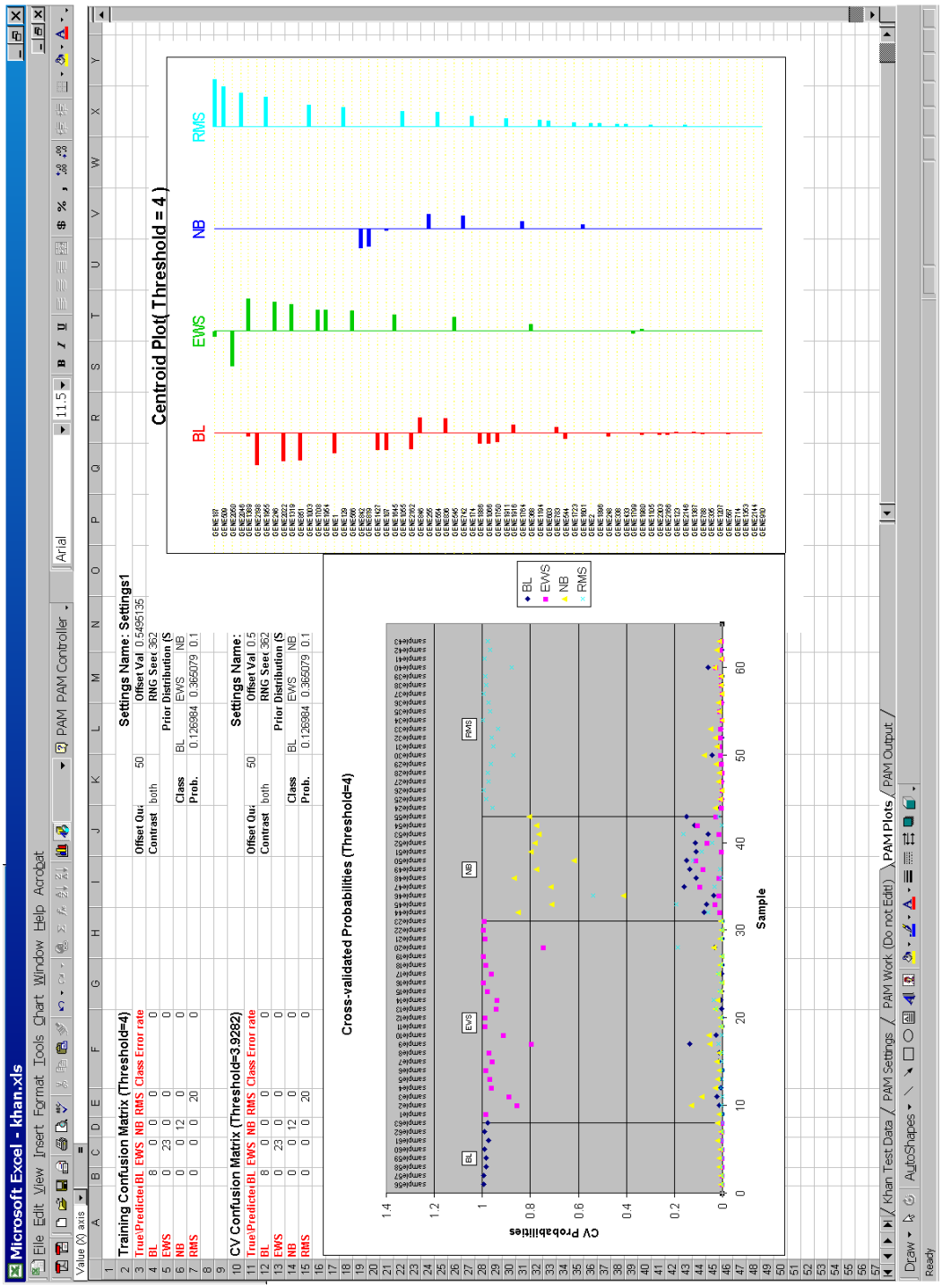

Figure 4: PAM Plots

Additional Sheets A single Excel spreadsheet can have a maximum of 256 columns. When one has more than 256 samples, additional sheets can be specified to overcome this limitation.

Clicking **OK** sends the data over to R. Any missing data are imputed in a new sheet, several other sheets are added and a **PAM Controller** similar to the one shown in Figure 3 pops up.

The following is a brief description of the controls.

Current Threshold is a textbox where one would enter a threshold that specifies the degree of shrinkage used by the classified. This field becomes visible after training has been done. The choice of the threshold is typically made after a judicious examination of training errors and the cross-validation results. Until a threshold is chosen, many controls remain inactive.

Train Trains the classifier. This is required and is always the first step. Until this is done, several other fields and the buttons remain disabled.

Plot Training Error will plot the training error and place the plot in the **PAM Plots** sheet.

Confusion Matrix button will output a training confusion matrix for a given threshold. If a threshold has not been entered already, then you are asked to enter one.

Cross Validate will do a 10-fold cross validation to help one to choose a threshold that minimizes classification errors.

Plot CV Curves will plot the misclassification errors obtained by cross-validation for various values of the threshold.

Plot CV Probabilities will plot classification probabilities for a specified threshold.

Plot Centroids will plot the shrunken centroids for a specified threshold.

List Genes will list the significant genes with the associated score for each class as shown in Figure 5.

Predict Test Set can be used to predict a test set.

Plot Test Error can be used to plot the prediction errors for a number of values of the threshold.

Plot Test Probabilities will plot the class probabilities for each sample in the test set for a specified threshold.

Show Prediction will create a worksheet with the prediction confusion matrix, if computable, and a list of actual and predicted class labels along with the prediction probabilities for each class.

Figure 4 shows some of the plots produced by PAM.

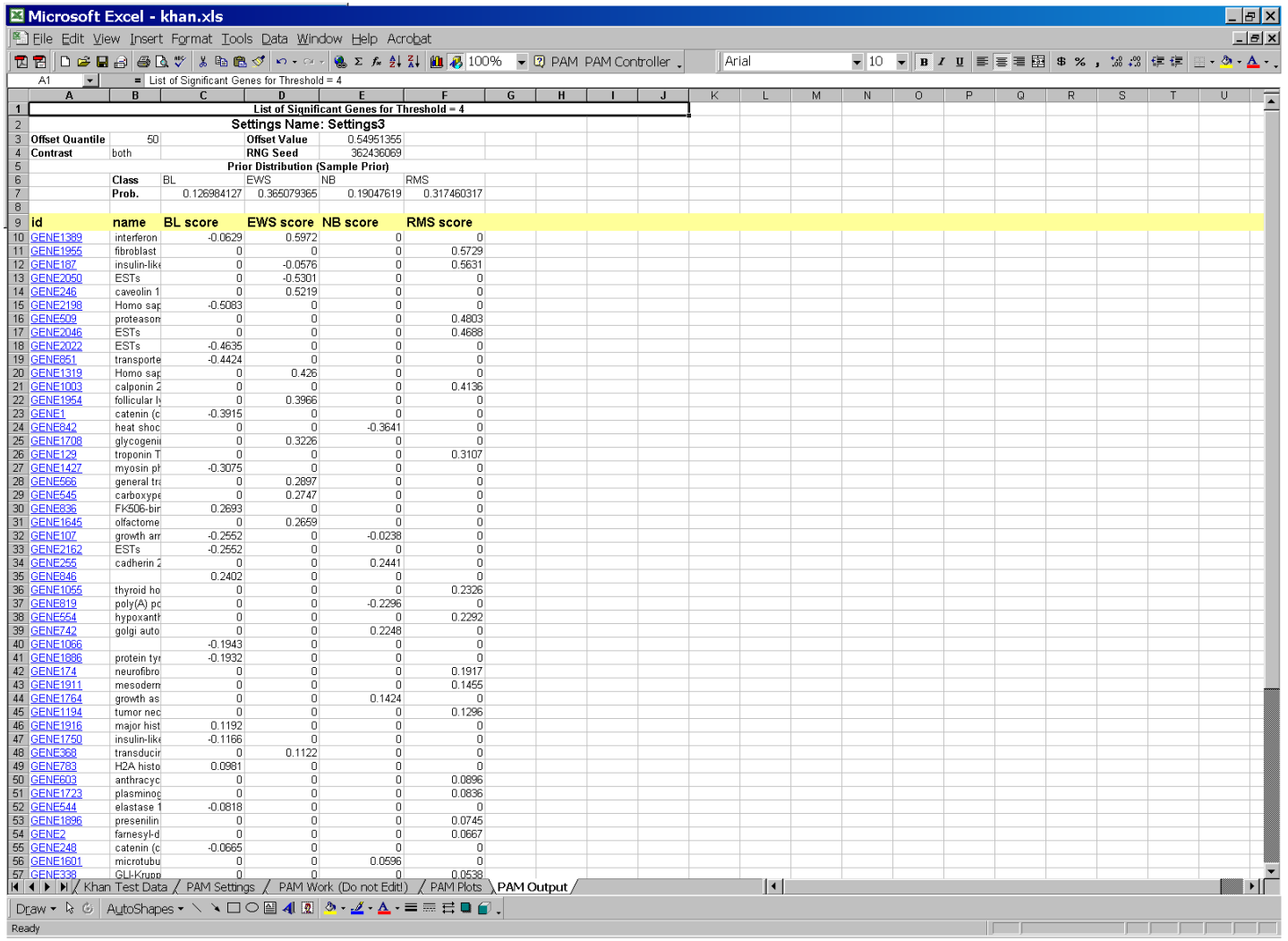

Figure 5: PAM listing of significant genes

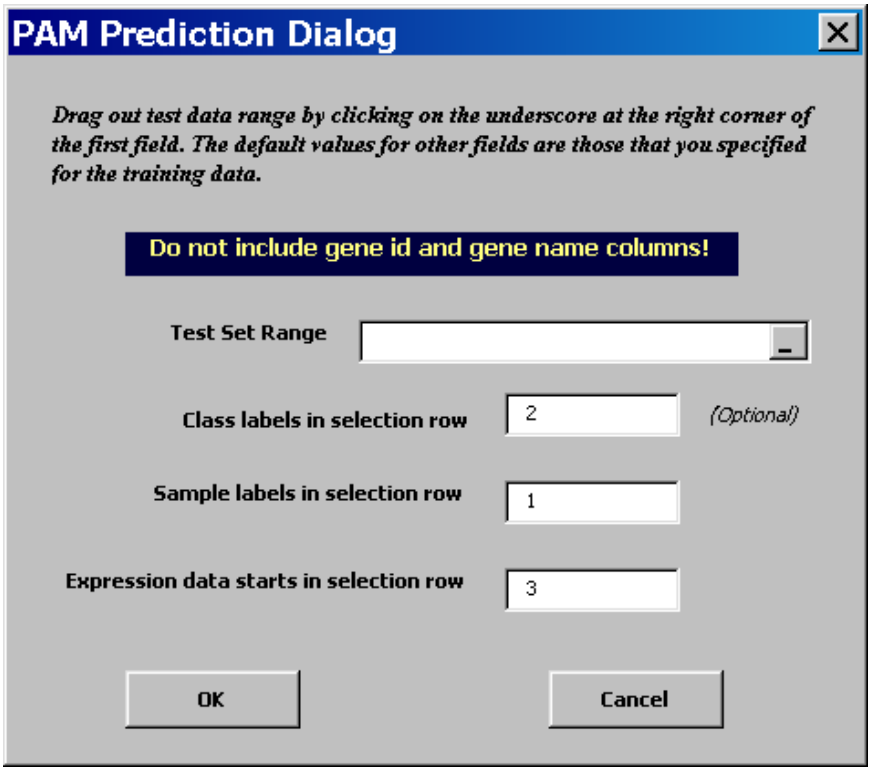

Figure 6: The PAM Prediction Dialog

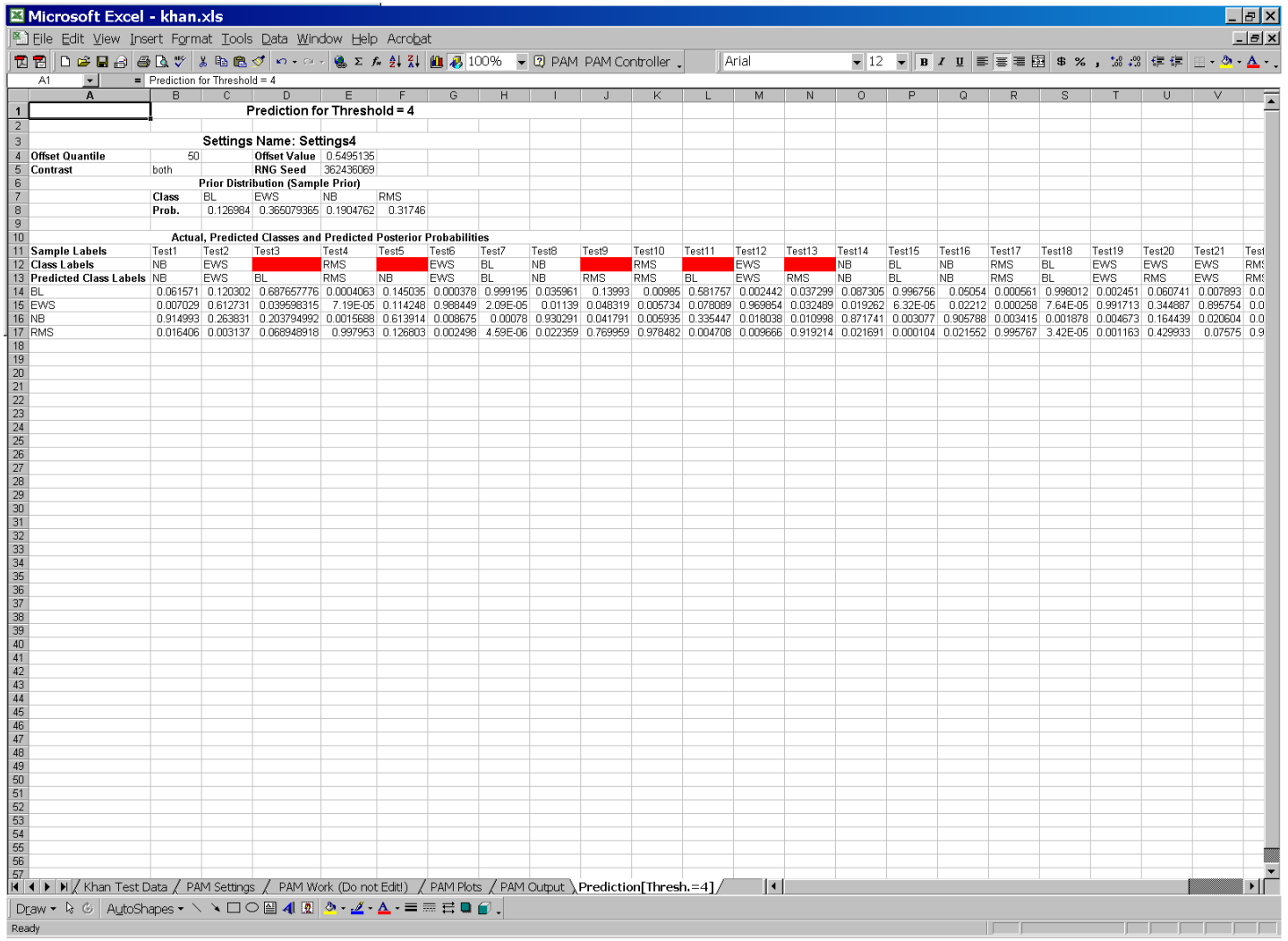

Figure 7: The PAM prediction output for Khan data

3.1 Prediction

Class labels for test data can be predicted by once again highlighting an area and clicking on the **Predict Test Set** button. A prediction dialog (see Figure 6) allows the user to specify the characteristics of the test set. A new worksheet with the predicted class labels along with the posterior probabilities is output. If the test data has all class labels specified, then a confusion matrix is also provided. See Figure 7 where some class labels are missing and highlighted in red.

4 Discussion

The current version of PAM uses Thomas Baier's DCOM server version 0.99. This version has some limitations in handling mixed data types and therefore such data have to be parsed and handled in Visual Basic before sending them over to R. Baier has since released a newer version of the DCOM server and Duncan Temple Lang also has an R DCOM server. Newer versions of PAM will be modified to use these servers as they become available.

Excel's facilities for graphics use a very different model compared to R. Reproducing some R plots in Excel is troublesome and tedious, if not impossible, although many tricks are available. For example, we were unable to reproduce the centroid plot and resorted to embedding the plot in a worksheet. On the other hand, Excel plots are interactive and many users seem comfortable in dealing with them, so that yields an advantage.

With the sizeable number of plots that PAM produces, there are presentation issues that come to the fore. Plots tend to overlap one another and the user has to physically relocate them. Some generated plots usually are more comprehensible when resized. Embedding the plots in worksheets, as PAM does, alleviates the problem to some extent.

Slapping a GUI in front of a package can sometimes obscure the details for more sophisticated users. Such users might wish to see the actual results of the computation rather than just the final results. We have addressed this by creating the special worksheet named **PAM Worksheet (Do Not Edit!)**. This sheet contains intermediate results of computations done by PAM. Every column in this worksheet is given a heading and an elaborate comment is provided so that the user can discern what it represents. If desired, these results can be used for further computations.

In the two months the software has been available, we have found that the installation process needs improvement. The current version demands that the user install an R package and then follow it up with an installation of the Excel Addin. Despite elaborate instructions, this is one step too many for the common Windows user and it would be ideal to make this a one-step process. In addition, error handling needs to be improved.

The ideas used in developing PAM should enable us to build a similar interface for Bioconductor.

PAM is freely available from the PAM Website at `http://www-stat.stanford.edu/~tibs/PAM`.

References

R. Ihaka and R. Gentleman. R: A language for data analysis and graphics. *Journal of Computational and Graphical Statistics*, 5(3):299–314, 1996.

E. Neuwirth and T. Baier. Embedding R in standard software, and the other way round. In K. Hornik and F. Leisch, editors, *Proceedings of the 2nd International Workshop on Distributed Statistical Computing, March 15-17, 2001, Technische Universität Wien, Vienna, Austria*, 2001. URL http://www.ci.tuwien.ac.at/Conferences/DSC-2001/Proceedings/. ISSN 1609-395X.

R. J. Tibshirani, T. J. Hastie, B. Narasimhan, and G. Chu. Diagnosis of multiple cancer types by shrunken centroids of gene expression. *Proceedings of the National Academy of Sciences*, 99(10):6567–6572, May 2002.

V. Tusher, R. Tibshirani, and C. Chu. Significance analysis of microarrays applied to ionizing radiation response. *Proceedings of the National Academy of Sciences*, 98(9):5116–5121, April 2001.

Affiliation

Balasubramanian Narasimhan
Dept. of Statistics and Dept. of Health Research & Policy
Stanford University
Stanford CA 94305
E-mail: naras@stat.stanford.edu