AL/EQ-TR-1996-0007

FIP: A PATTERN RECOGNITION PROGRAM FOR FUEL SPILL IDENTIFICATION

A. Faruque, B. K. Lavine, H. T. Mayfield

Armstrong Laboratory (AL/EQL)
139 Barnes Drive, Suite 2
Tyndall AFB FL 32403-5323

ENVIRONICS DIRECTORATE
139 Barnes Drive, Suite 2
Tyndall AFB FL 32403-5323

May 1996

Final Technical Report for Period August 1993 - August 1995

AIR FORCE MATERIEL COMMAND
TYNDALL AIR FORCE BASE, FLORIDA 32403-5323

ARMSTRONG LABORATORY

NOTICES

This report was prepared as an account of work performed by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any employees, nor any of their contractors, subcontractors, or their employees, make any warranty, expressed or implied, or assume any legal liability or responsibility for the accuracy, completeness, or usefulness of any privately owned rights. Reference herein to any specific commercial products, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency, contractor, or subcontractor thereof. The views and opinions of the authors expressed herein do not necessarily state or reflect those of the United States Government or any agency, contractor, or subcontractor thereof.

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely Government-related procurement, the United States Government incurs no responsibility or any obligation whatsoever. The fact that the Government may have formulated or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication, or otherwise in any manner construed, as licensing the holder or any other person or corporation; or as conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

This technical report has been reviewed by the Public Affairs Office (PA) and is releasable to the National Technical Information Service, where it will be available to the general public, including foreign nationals.

This report has been reviewed and is approved for publication.

Howard T. Mayfield

HOWARD T. MAYFIELD, PhD
Project Manager

Michael G. Katona

MICHAEL G. KATONA, PhD
Chief Scientist, Environics Directorate

JIMMY C. CORNETTE, PhD
Chief, Environmental Research Division

Neil J. Lamb

NEIL J. LAMB, Colonel, USAF, BSC
Director, Environics Directorate

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	May 1996	Interim, August 1993 - August 1995

4. TITLE AND SUBTITLE

FIP: A Pattern Recognition Program for Fuel Spill Identification

5. FUNDING NUMBERS

MIPR N93-40

6. AUTHOR(S)

A. Faruque, B. K. Lavine, and H. T. Mayfield

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Armstrong Laboratory, Environics Directorate (AL/EQC)
139 Barnes Drive, Suite 2
Tyndall AFB, FL 32403-5323

8. PERFORMING ORGANIZATION

AL/EQ-TR-1996-0007

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSORING/MONITORING

11. SUPPLEMENTARY NOTES

Supplementary Technical Report for the Project: "Integrated JP-4/JP-8 Database". POC: Dr. Howard T. Mayfield, AL/EQC DSN: 523-6049 Commercial: 904-283-6049

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for Public Release

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

Gas Chromatography and pattern recognition methods (GC-PR) constitute a powerful tool for investigating complex environmental problems, e. g. , realistically analyze large chromatographic data sets and to seek meaningful relationships between chemical constitution and source variables. Recently, out laboratory has investigated the potential of GC-PR as a method for typing fuels in order to directly relate a spill sample to its source. A graphic user interface (GUI) based interactive software called FIP (fuel identification program) has been developed. The development of this software system takes advantage of the high performance computational and visualization routines of the MATLAB programming environment. Both neural networks and statistical pattern recognition techniques are implimented. FIP employs covariance stabalization of the data to ensure correct classification of the gas chromatograms of weathered and unweathered jet fuels.

14. SUBJECT TERMS	15. NUMBER OF PAGES
pattern recognition, neural networks, graphical user interface, multivariate data analysis, gas chromatographic analysis	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	

(The reverse of this page is blank.)

PREFACE

This technical report describes the implementation of a pattern recognition program intended to type classify fuels presented as gas chromatographic profiles. The program, called FIP (fuel identification program) uses a graphic user interface (GUI) for easy and simple interaction with the user. The program offers the user a choice of classification systems using either neural networks or statistical pattern recognition techniques. Various data preprocessing techniques necessary to support the pattern recognition techniques are also provided. Data visualization tools are also provided to permit visual accessment of the data patterns and their relationships. This technical report will provide a manual for the use of the program and an example of its use to classify a collection of jet fuels.

This report was prepared by the Armstrong Laboratory, Environics Directorate, Environmental Research Division (AL/EQC), 139 Barnes Drive, Suite 2, Tyndall AFB, FL 32403-5323. The work was performed by Dr. Abdullah Faruque during a Postdoctoral Associateship, between August 1993 and August 1995.

(The reverse of this page is blank.)

EXECUTIVE SUMMARY

This report constitutes a manual for use of a computer program, FIP, which is a specially built program for the classification and identification of fuels based on data from gas chromatographic profiles and pattern recognition analysis. FIP is an auxiliary program for the MATLAB matrix mathematics package, with an easy-to-use graphical user interface (GUI) providing access to a variety of classification, preprocessing, and display techniques for the analysis of gas chromatographic profiles. Through use of the GUI, the user is provided with pull-down or pop-up menu's to simplify the tasks of entering data, preprocessing the data, displaying results, training or calibrating classifiers, and performing classifications. This report details the techniques selected for inclusion in FIP and describes how to access them through the GUI menus.

The use of the FIP program is illustrated in the manual by the analysis of a typical data set of 228 jet fuel gas chromatograms from a variety of fuels and fuel sources. The representative study describes the entry of the data and the treatments used in the representative study. The classification results from several of the available classifiers are tabulated and discussed.

(The reverse of this page is blank.)

Table of contents

List of Figures

(The reverse of this page is blank.)

List of Tables

(The reverse of this page is blank.)

Introduction

Fuel spill identification is an important problem in the field of environmental chemistry (1-2). Fuel spills must be related to their sources so that models which can predict the results of possible control scenarios can be developed. If a fuel spill is from a leaking underground tank or pipeline, we must be able to determine the type of jet fuel involved, in order to relate the spill sample to its source and ensure proper utilization of repair resources. Interest in establishing the type of fuel responsible for the contamination is also motivated by the clean-up costs, legal fees, and fines which could be incurred by the polluter.

Previous workers (3-4) have shown that a fuel spill sample can be related to its source using gas chromatography (GC). The gas chromatogram of the spilled fuel and a number of suspected fuel sources are examined visually in order to obtain a best match (5). However, this approach to data analysis is often subjective and usually cannot take into account the effects of weathering on the overall GC profile of the fuel. Visual analysis of gas chromatograms also suffers from the drawback that it cannot take into account variations in the operating condition of the gas chromatograph, e.g., aging of the GC column or variations in the temperature programming rate of the column, which can be a serious problem complicating the identification of a leaking fuel (6). Finally, representing a particular class of fuels by a single gas chromatogram cannot always be accomplished since aviation fuel composition is only loosely governed by civilian and military specifications. (Usually, only the flash point and freezing point of a fuel are specified, and it would not be surprising that fuels of quite different composition could satisfy these specifications.) Therefore, evidence based on visual analysis of gas chromatograms is not always persuasive in a court of law, especially in cases involving an unweathered fuel identified as the source of a fuel spill because of marked differences between gas chromatograms of weathered and unweathered jet fuels.

Pattern recognition methods (7-9) offer a better approach to the problem of matching gas chromatograms of spilled fuels to suspected fuel sources. Pattern recognition methods are not subjective and can identify fingerprint patterns in GC data characteristic of fuel type eventhough the fuel samples comprising the training set have been subjected to a variety of conditions. In otherwords, classifiers can be developed from the data that are relatively insensitive to changes in the overall GC profile due to contamination, analytical error, or weathering. In essence, pattern recognition methods can be thought of as providing relations that uncover common properties in the data. For fingerprint data of the type that we are considering, these relations are often expressed in the form of subtle variations in relative peak intensities distributed across several peaks in the gas chromatograms. Pattern recognition methods are especially well suited for extracting this type of information from the large amounts of qualitative and quantitative data present in the gas chromatograms of jet fuels (10-12).

The United States Air Force (USAF) has initiated an in-house project to examine samples of various jet fuels, in order to develop a suitable data base of gas chromatograms representative of the various types of jet fuels. Using conventional pattern recognition techniques, USAF scientists have shown that jet fuels can be identified from GC profile data (13-15). However, gas chromatograms of recovered jet fuels can be a combination of two or more different fuel types complicating their classification. Furthermore, the application of pattern recognition methods to fuel spill data is often complicated by three additional factors:(1) a low object to descriptor ratio, (2) serious colinearities and multicolinearities among the measurement variables in the data, and (3) confounding of the desired group information by experimental artifacts or other systematic variations in the data. Conventional pattern recognition techniques incorporated in commercial software packages cannot properly treat this type of data (16). Hence, there is a need for software that can cope with fuel spill data.

Using MATLAB (17), we have developed a graphical user interface (GUI) based interactive pattern recognition software system for fuel spill identification called FIP (fuel spill identification program). The development of this software system takes advantage of the high performance computational and visualization routines of the MATLAB programming environment. With MATLAB, it becomes a simple matter to develop and implement a modular interactive pattern recognition software system because the graphical user interface can be directly incorporated with the necessary computational

and visualization components. Another advantage of MATLAB is that low level GUI programming, such as X-windows or Microsoft Windows, is not required, so it should be possible for scientists and engineers in the near future to routinely design application specific software packages with MATLAB, without having to write a single line of C or other lower level code. Any application developed under the MATLAB programming environment is portable to a wide variety of hardware platforms, since MATLAB operates in a variety of workstation and personal computing environments. Hence, limitations imposed by the availability of application and hardware specific commercial software for pattern recognition analysis will no longer be a serious problem.

Software Implementation

FIP utilizes MATLAB M files for the computational and GUI components. Each M file performs specific computational and/or GUI tasks which are invoked by the main FIP graphical user interface function module which by itself is a MATLAB M file called FIP.M. FIP's top level GUI is shown in Figure 1. The interface consists of menus for different tasks and graphical objects such as buttons and fields for displaying information from pattern recognition analyses performed on a particular data set. All GUI features are part of the MATLAB Handle Graphics system, which consists of graphics objects, object handles and object properties. The graphics objects include the root screen, figures, text, the various user interface menus and the user interface controls. Each graphics object has a unique identifier called a handle, which is assigned to the object when it is created. These handles are stored in a variable, and the values of the handle are passed when the handle is required by the calling function.

All graphics objects have software switches which control how they are displayed. When an object is created, it is initialized with a full set of default property values which can be modified as required. Most of the GUI components of FIP consist of either menus or controls. The menus and submenus are created by using the MATLAB **uimenu** function. The callback property of the menu item created by the **uimenu** function call specifies the action that would be taken when the user selects a particular menu item. The string associated with the callback property is the name of the M file which contains the necessary computational and/or GUI codes for the requested operation. Controls are graphics objects, e.g., push button, check box, radio button, slider, pop-up menu, static text, editable text and frame, which are responsible for performing different actions when manipulated by a mouse. All controls are created using the MATLAB **uicontrol** function. The callback property of the controls perform the specified operation when the user activates the control.

Matlab functions **uimenu** and **uicontrol** are the functions most extensively used in FIP. Input and output file operations are performed using **uigetfile** and **uiputfile** function calls. The whole graphical user interface of FIP has been developed, using only the four above mentioned MATLAB GUI function calls. This is much simpler than what is done in other GUI programming languages, e.g., X Window, Sun View, MS-Window etc., where extensive low level programming is required. FIP was developed on a Sunsparc station operating SunOS 4.1 and MATLAB version 4.2. It has also been ported to PC platforms, running MS-Windows and MATLAB version 4.2. The minimum requirements for a PC platform are a 486DX2-66 Mhz machine (or compatible), 16MB of memory, and VGA color graphics.

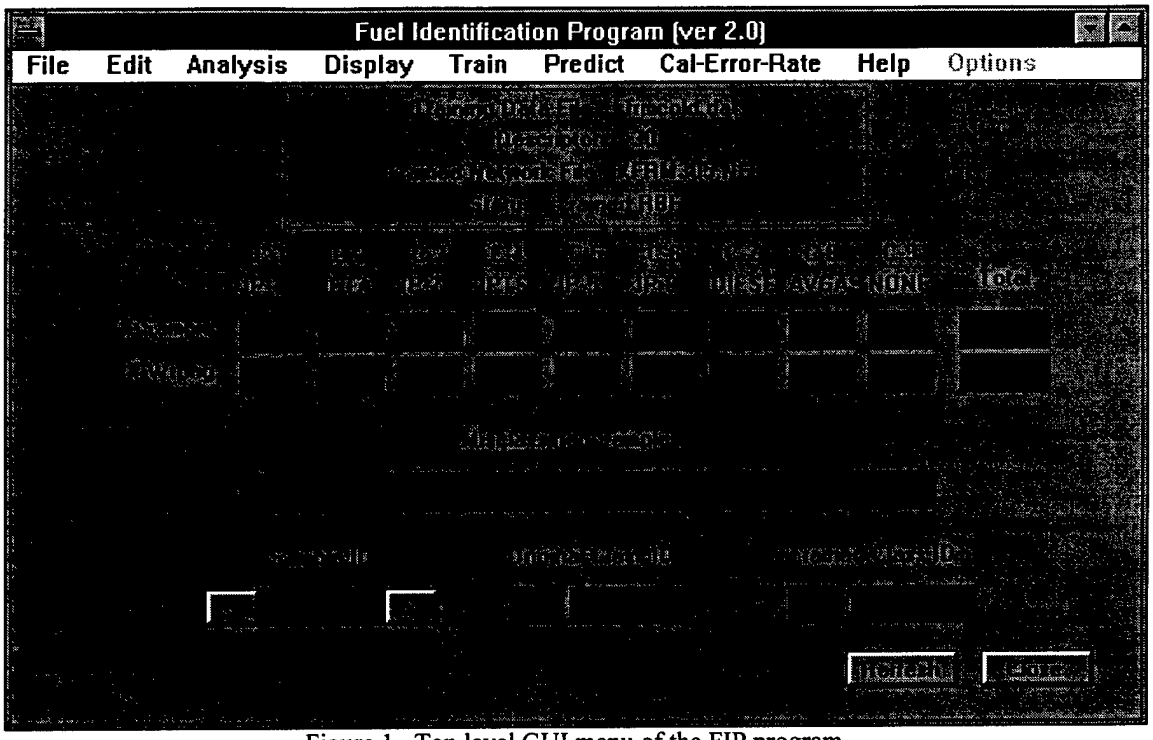

Figure 1. Top level GUI menu of the FIP program.

Main features of FIP

1. Data input and output

Load training data from disk file

Load prediction data from disk file

Load trained network or model from disk file

Save trained network or model into a disk file

Write training results in plain ascii text file

Write prediction results in plain ascii text file

2. Data editing

Remove one or more class from the training data

Remove one or more sample from the training data

Remove one or more descriptor from the training data

3. Data analysis

Feature selection based on variance weights and correlation

Feature selection based on fisher weights and correlation

Fisher and Variance weight calculation

Principal components analysis

Non-linear principal component analysis

4. Classification

Backpropagation networks (BPN)

Radial basis function networks (RBF)

Linear discriminant analysis (LDA)

Quadratic discriminant analysis (QDA)

Regularized discriminant analysis (RDA)

Soft independent modeling of class analogy (SIMCA)

Discriminant analysis with shrunken covariances (DASCO)

K-nearest neighbor (KNN)

5. Prediction

Prediction of unknown fuel profile using a trained network or model designed by any one of the classification methods in FIP.

6. Display results and data:

Display training results

Display prediction result

Display principal components table

Display 2-D plot of any two principal components

Display 3-D plot of the first three principal components

7. Error rate calculation:

Bootstrap error rate calculation from training data

Cross-validated error rate from training data

8. Limited online help is available for some oprerations.

Software and hardware requirements

FIP (ver 2.0) works on the following platforms:

1. Sun Sparkstations running SunOS 4.1 and MATLAB version 4.2a MATLAB Neural Network Toolbox (version 2.0) is required if BPN training method and non-linear principal components analysis are desired.
2. Intel (or compatible) based PC running MS-Windows 3.1 and MATLAB version 4.2 for MS-Windows. MATLAB Neural Network Toolbox (version 2.0) is also required for BPN and non-linear principal components analysis. For reasonable performance, the minimum requirement is a 486DX2-66 Mhz machine with 16 meg sof RAM.

Installations

1. MATLAB for the target platform must be installed properly prior to installation of FIP.
2. Make a directory called FIP on the target hard drive. The program will require about 500KB of disk space.
3. Copy all files from the FIP distribution disk to the newly created directory (in a DOS machine it will be C:\FIP).
4. Add the name of the directory (C:\FIP) to the MATLAB search path by editing matlabrc.m file (see MATLAB user's guide for more details).
5. Start MATLAB
6. Change directory from MATLAB command prompt to the desired working directory where the data files are located. It is a good idea to put data files in a seperate directory (not in C:\FIP).
7. Type fip at the MATLAB command prompt to start FIP.

Data Input

Input of data into the FIP can be accomplished using two data formats: conventional ascii format and FCV ready format. In conventional ascii format, each row is an object and each column is measurement variable. A training set in this format must be arranged in order of the class assignment of the samples. A user will be prompted for the file name, number of classes, and number of samples in each class as shown figure 2 & 3. The FCV ready format is a unique format developed previously for FCV false color data imaging (18). This format requires two files for each data set. One file contains the data matrix (with a .dat extension), and the other file (with an .id extension) has the necessary information about the class assignments. Training set data can also be inputted by loading a previously saved trained network or model file (with a .net extension). A user will be prompted for the name of the trained network file to be loaded as shown in figure 4.

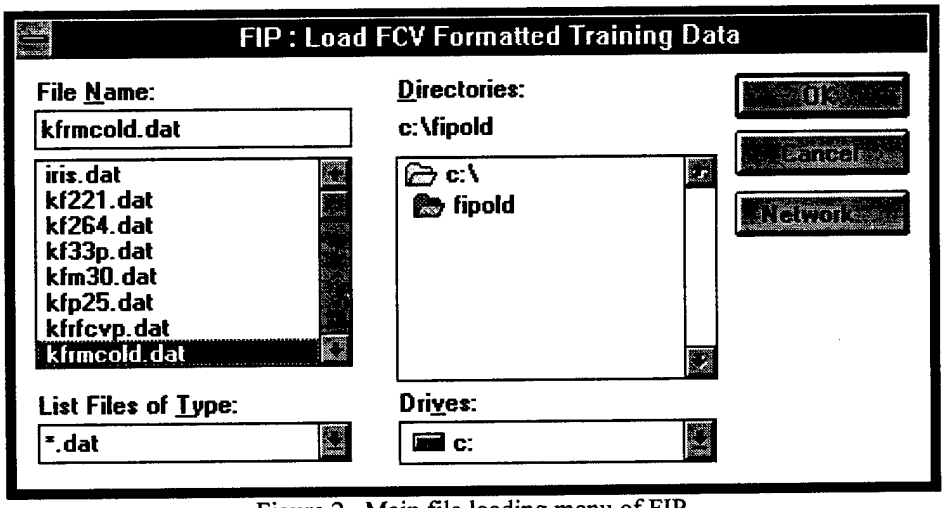

Figure 2. Main file loading menu of FIP.

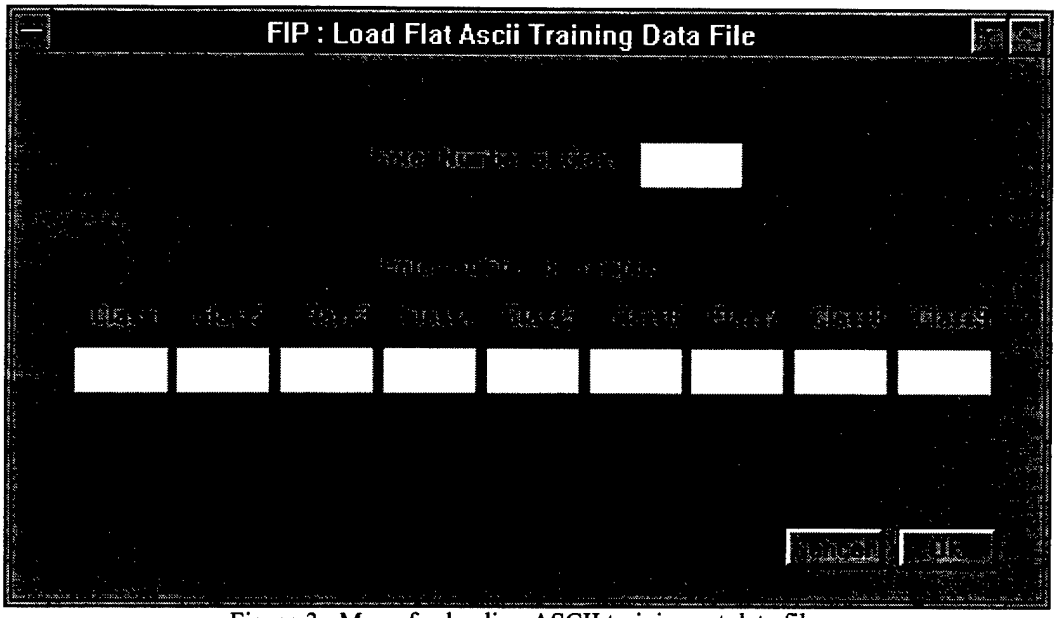

Figure 3. Menu for loading ASCII training set data file.

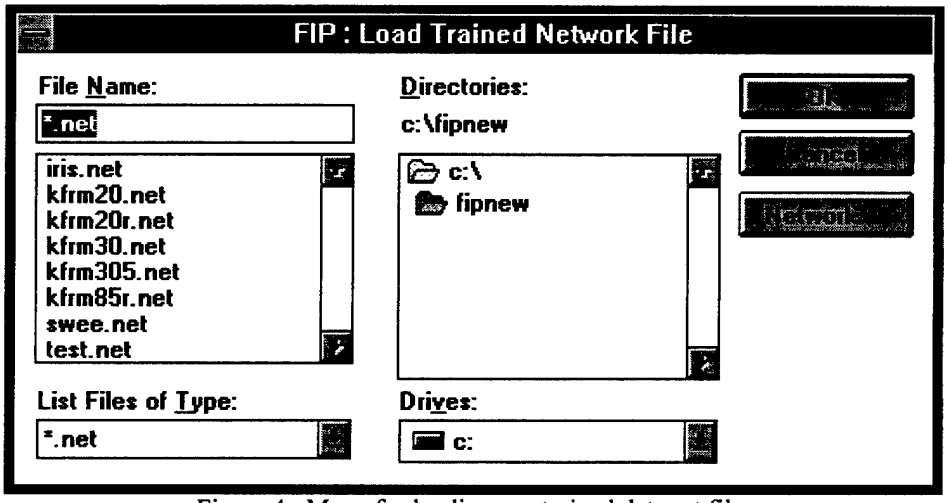

Figure 4. Menu for loading pre-trained data set file.

Data Editing

Once a data set has been loaded into FIP, editing can be performed. The user can delete objects, features, or entire fuel classess which is often necessary due to the presence of discordant observations and measurement variables in the data. A GUI option box will be presented (see figures 5, 6 & 7) when the desired operation is invoked.

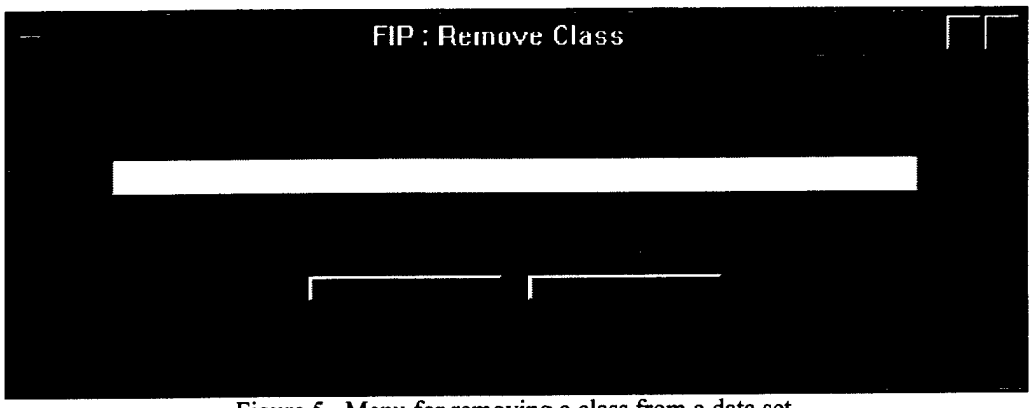

Figure 5. Menu for removing a class from a data set.

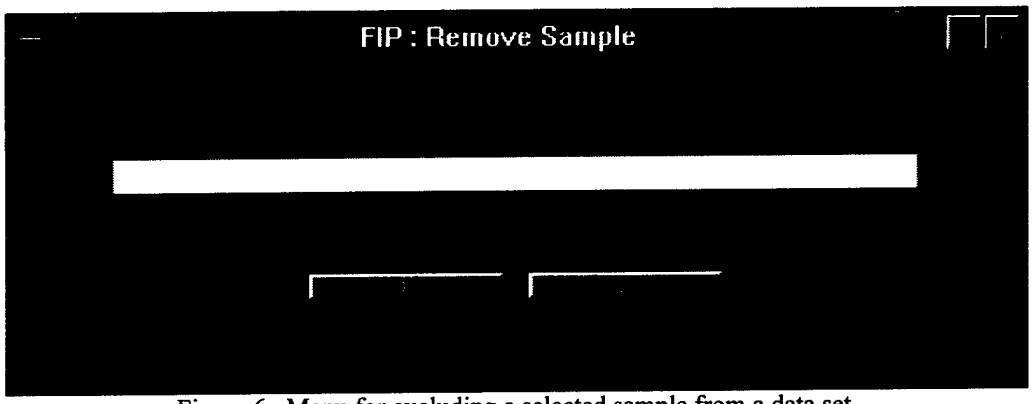

Figure 6. Menu for excluding a selected sample from a data set.

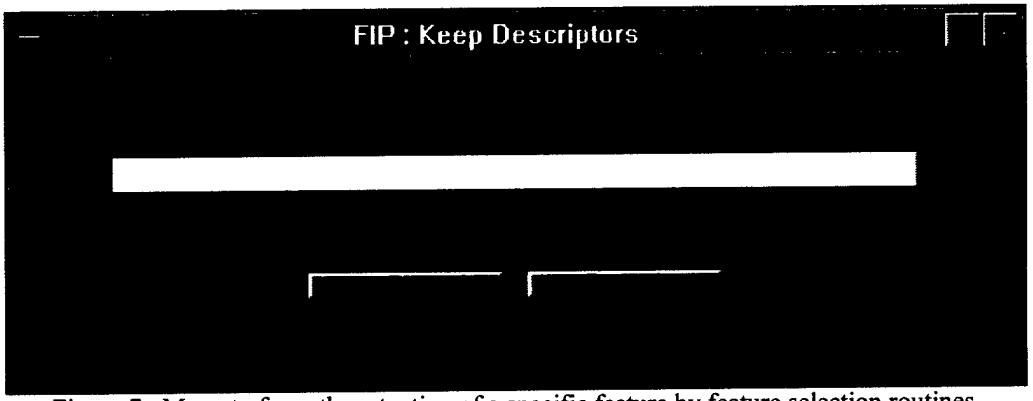

Figure 7. Menu to force the retention of a specific feature by feature selection routines.

Data Output

The data output component of FIP permits the user to write the training and/or prediction set results in an ascii text file, while storing the trained network or model in a binary file. This allows the user to develop a library of trained network or models for later use. The user will be prompted to give a file name (see figures 8,9 &10) for the training results, prediction results, trained network or model to be

stored. The ascii file name for the training or prediction results to be stored should have an extension of .txt, e.g., output.txt. The trained netwok or model file name must have an extension of .net, e.g., test.net

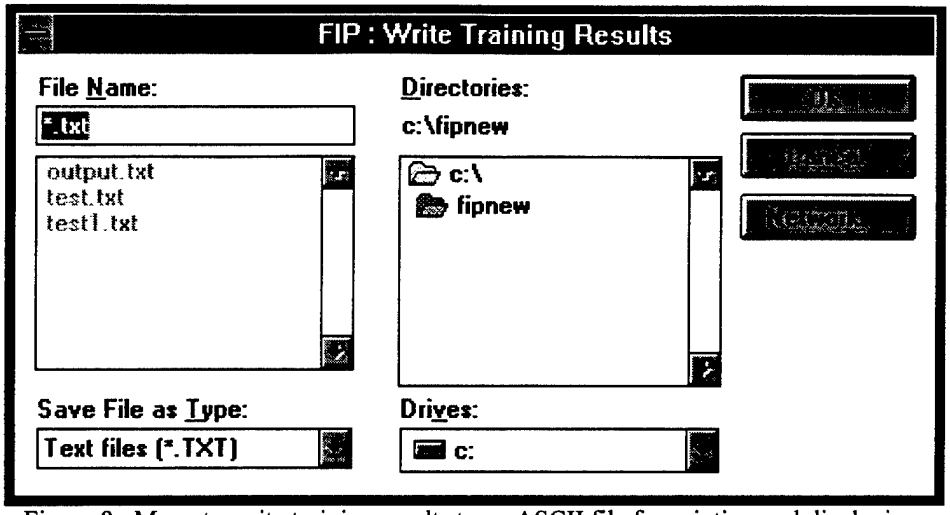

Figure 8. Menu to write training results to an ASCII file for printing and displaying.

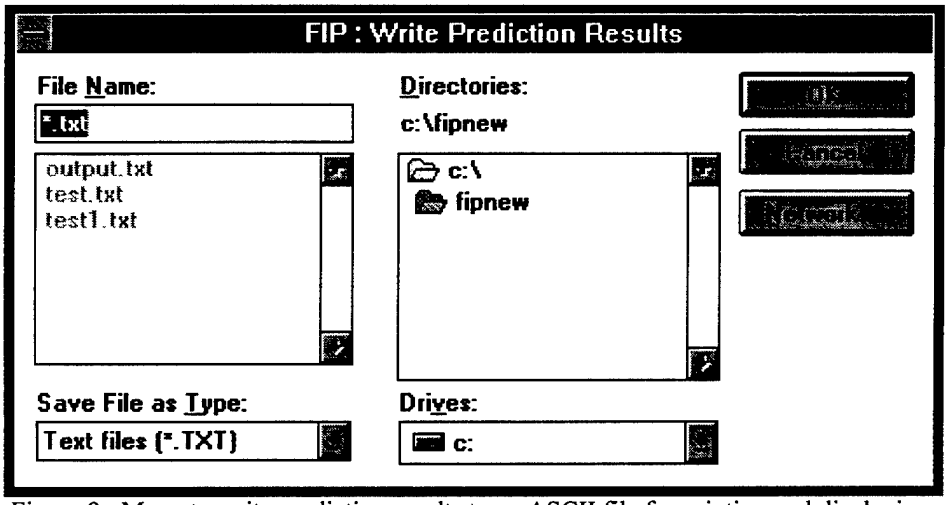

Figure 9. Menu to write prediction results to an ASCII file for printing and displaying.

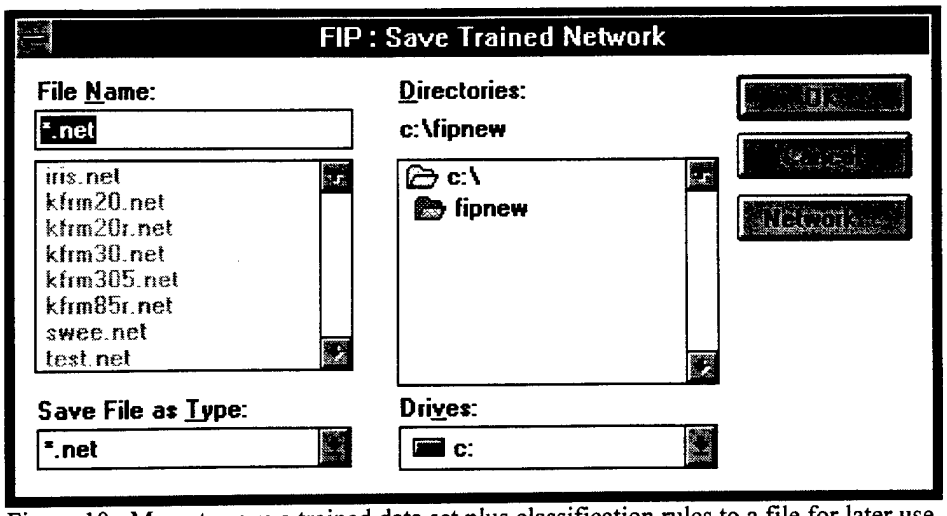

Figure 10. Menu to save a trained data set plus classification rules to a file for later use.

Data Analysis

Data analysis is divided into three main conponents: (1) principal components analysis, (2) fisher and variance weight calculations and (3) feature selection. The principal component analysis (PCA) module of FIP uses the loaded data set to perform either statdard PCA (via singular value decomposition or direct eigenanalysis) or non-linear PCA (via autoassociative feed forward neural networks trained by the backpropagation learning rule). Using MATLAB's graphics routines, 2-dimensional or 3-dimentional plots of the principal components can be displayed for direct viewing. A principal component table (Figure 11) will be presented after a principal component analysis run is done. This table displays the values of upto 10 eigenvalues and cumulative variances. This table also includes GUI push buttons to invoke 2-d principal components plot, 3-d principal components plot, eigenvalue plot and cumulative variance plot. Each sample in the principal components plot is labeled, using either sample number or the class assignment of the sample. Each fuel class is assigned a unique color in the principal components plot for better visualization. Samples misclassified by network or discriminant models are marked in red color in the principal components plot. This feature can increase the information content of a principal components plot, allowing the user to better understand the structure of the data.

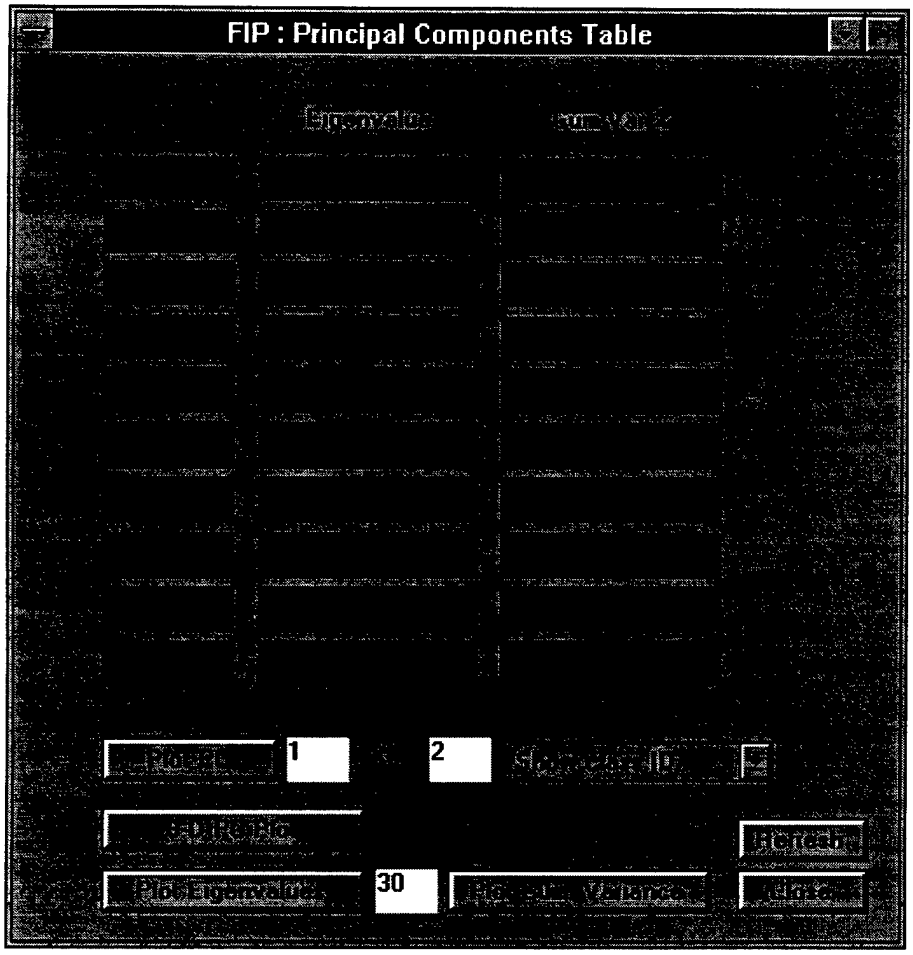

Figure 11. Principal components analysis result display and plot control menu.

Feature selection component of FIP will select the desired number of features as requested (see figure 12) using the variance or fisher weight and correlation. The highest weighted feature is selected as the first feature. The remaining features are then decorrelated from the chosen feature. Of the decorrelated features, the one with the highest variance or fisher weight is chosen as the second selected feature. The process repeats until either a specified number of features have been selected or the variance or fisher weight falls below a specified tolerance (the default variance weight tolerance is 1.05). The result of the feature selection will be written to feature.dat file in the working directory. The training and prediction set data will be adjusted automatically according to the selected features so that a user can run different classification methods with the selected features.

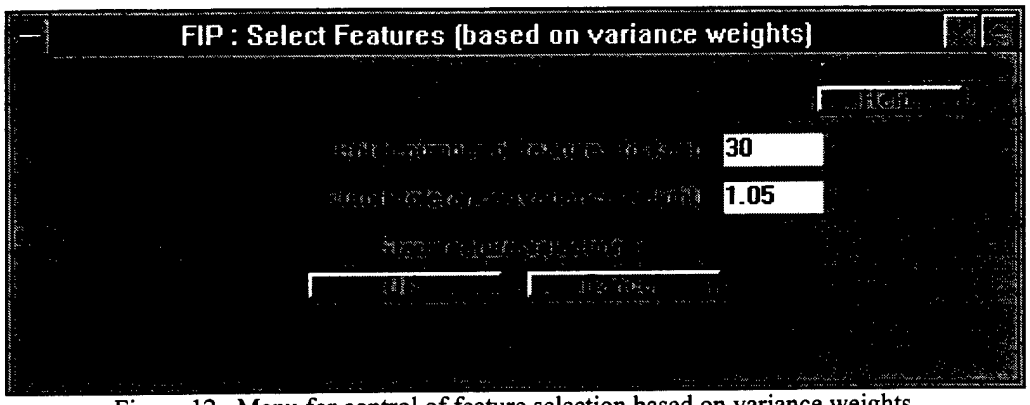

Figure 12. Menu for control of feature selection based on variance weights.

Classification and Prediction

Classification algorithms in FIP include both neural networks and statistical pattern recognition techniques. Neural network methods include the back propagation neural network (BPN) and radial basis function (RBF) neural network. MATLAB Neural Network Toolbox is required to run back propagation training method. The training parameters for both methods can be changed through the GUI trainig option menu box, which will be presented after invoking the desired training method. Statistical pattern recognition component sof FIP includes linear discriminant analysis (LDA), quadratic discriminant analysis (QDA), regularized discriminant analysis (RDA), soft independent modeling of class anlogy (SIMCA) and discriminat analysis with shurken covariances (DASCO). K-nearest neighbor (KNN) classification is also part of FIP. All of the classification methods in FIP present the user with a GUI option box for parameter selection when that method is invoked. Once a data set has been trained using any of the classification methods described above, the prediction on unknown data can be performed easily by invoking the "predict using trained model" item from the main menu option of FIP.

Backpropagation Neural Network:

A GUI training option box will be presented (see figure 13) when this method is invoked. The user can select either backpropagation-1 or backpropagation-2 method of training. Backpropagation-1 uses the gradient descent algorithm. It also uses momentum and adaptive learning rate to speed the learning process. Backpropagation-2 uses the Levenberg-Marquardt approximation technique. This optimization technique is more powerful, but it requires more memory. This method should only be implemented on a sparcstation in the case of GC-fuel data. If the dimenisionality of the data is low, than it should work on PC platform as well. All the training parameters should be selected as desired by invoking "Set Training Parameters" push button. Figure 14 shows the GUI option box for parameter selection. The default values are a reasonable starting point in most cases.

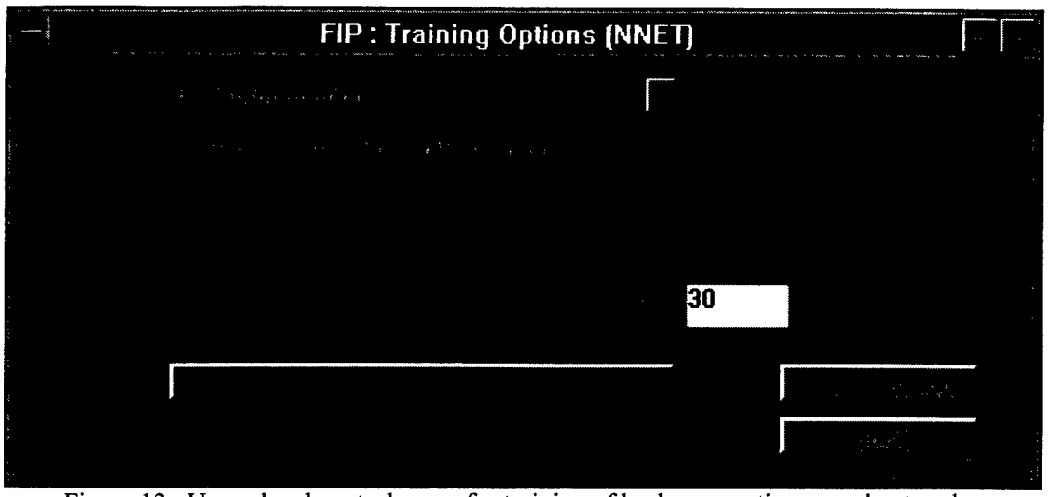

Figure 13. Upper level control menu for training of backpropagation neural networks.

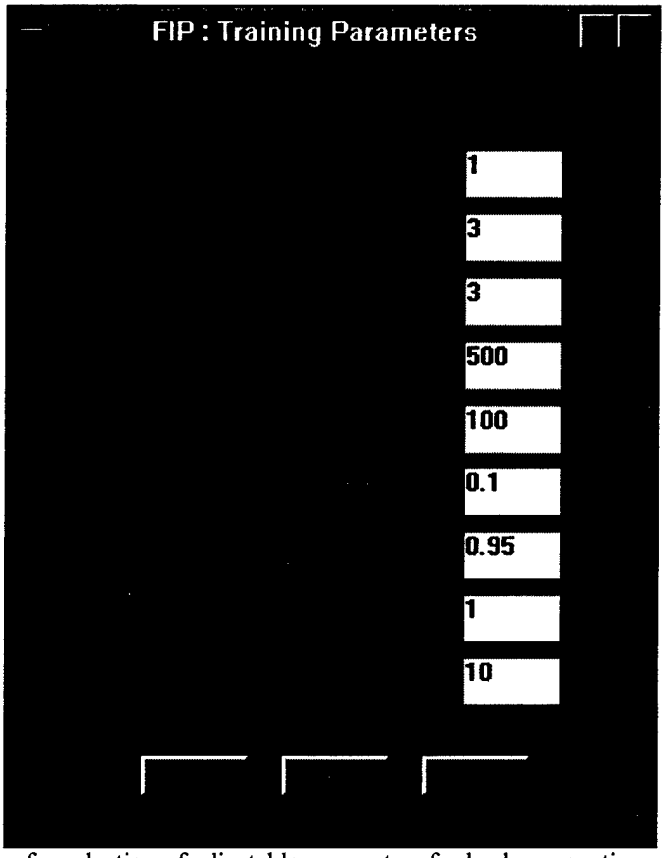

Figure 14. Menu for selection of adjustable parameters for backpropagation neural networks.

Radial Basis Function (RBF) Neural Network:

A GUI training option box is presented (see figure 15) when this method of training is invoked. The user must provide values of covariance mixing and spread constant parameters. With the correct model

parameters RBF performs very well in the case of GC-fuel data. It can be viewed as neural network implementation of regularized discriminant analysis.

Covariance mixing parameter:

The default value of covariance mixing parameter is 0. It can have any value between 0 and 1. This parameter determines the degree of mixing of the covariance matrix towards the pooled covariance matrix (which is used in LDA). A value of 1 (which is the LDA case) involves using an equal norm weighting matrix for each hidden node kernel function , whereas a value of 0 (which is the QDA case) involves using a different norm weighting matrix for each hidden node kernel function. For GC fuel data, the default value of 0 is good for cases where the features selection method used is from FIP. If the dimensionality of the data is high, a little mixing (a value of 0.1) can provide a lower error rate.

Spread constant:

This parameter determines the spread of the gaussian radial-basis kernel function for each hidden node. The value must the greater than zero. The default value is 0.1 and the usual values are between 0.1 and 0.5 for GC fuel data. A lower value (e.g., 0.01) will provide better prediction results when the number of features is very high (e.g., 85). The optimum value is the one that gives the lowest bootstrap or cross-validated error rate and the best prediction set results.

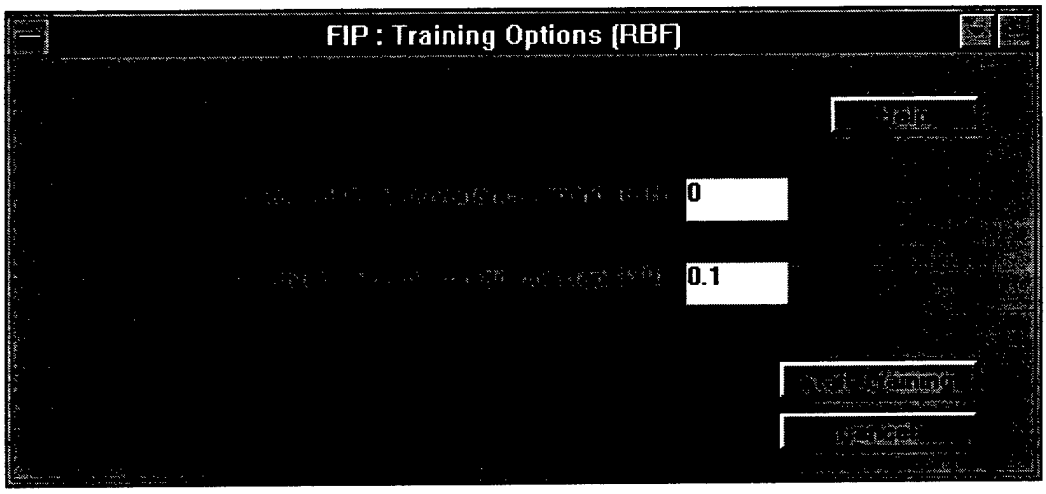

Figure 15. Menu for selection of adjustable parameters for training radial basis neural networks.

K-nearest neighbor (KNN):

A GUI training option box will be presented (see figure 16) when this method of training is invoked. The user must provide the number of nearest neighbor to be used. The default value is 5. A check box is provided to use autoscaled data.

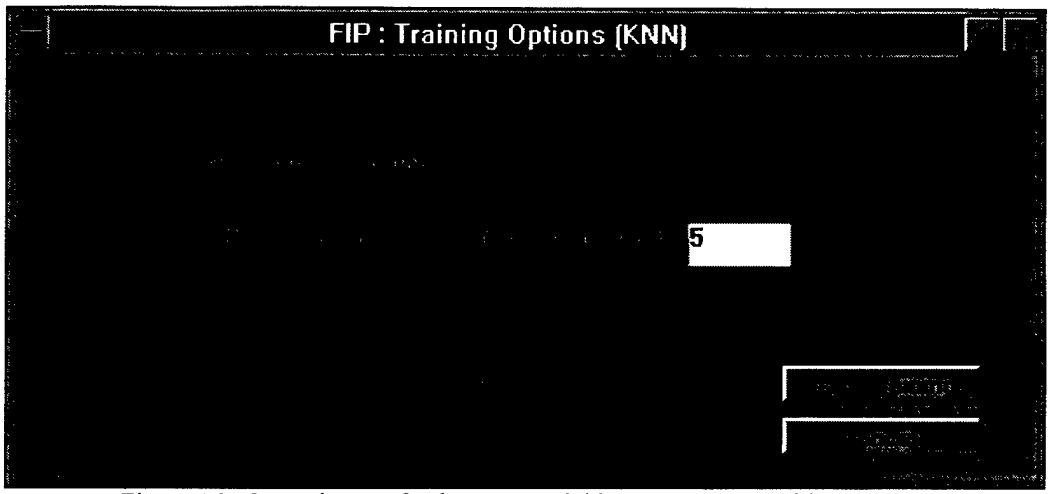

Figure 16. Control menu for *k*-nearest neighbor pattern recognition system.

Linear Discriminat Analysis (LDA) :

A GUI training option box will be presented (see figure 17) when this method of training is invoked. The user may select either raw data or a selected number of principal components as the training data set. The default is the raw data. A check box is provided for autoscaled data. LDA assumes equal class covariance matrices, which is the reason for using the pooled covariance matrix which is calculated from the training set samples. LDA does not require any user selected model parameters. It is computationally efficient and is expected to perform well in homoscedastic cases when sample size is large compared to the dimensionality of the measurement space. In a well-determined homoscedastic case, LDA should give better perfomance because it has fewer estimated parameters.

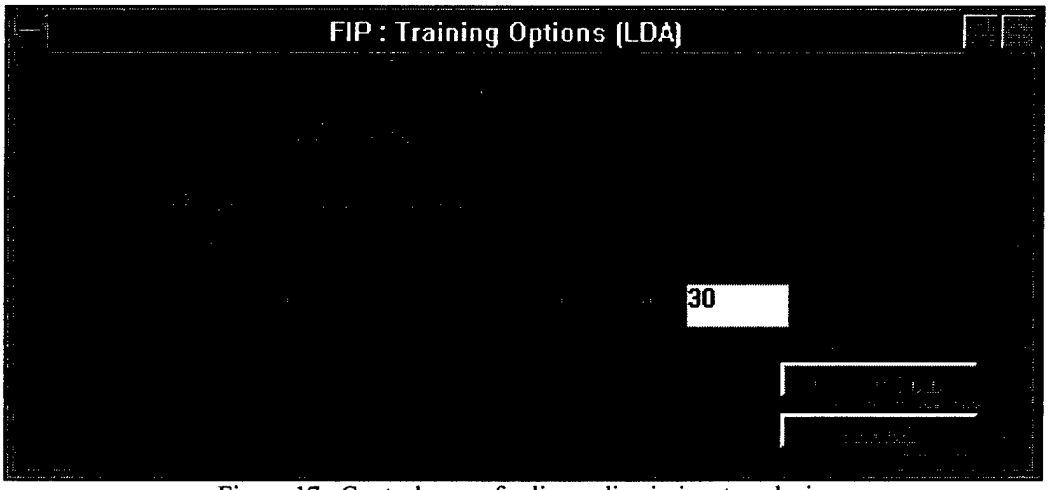

Figure 17. Control menu for linear discriminant analysis.

Quadratic Discriminant Analysis (QDA):

A GUI training option box is presented (see figure 18) when this method of training is invoked. The user may select either raw data or principal components of the data . The default is the raw data. A check box is provided for autoscaled data if desired. QDA uses a different class covariance matrix which is calculated from the training set samples for each class. QDA does not require any user specified model parameters. It is computationally efficient and is expected to perform well in heteroscedastic cases, when sample size is large compared to the dimensionality of the measurement space. In a well-determined heteroscedastic case, QDA should perfom better than regularized methods because it estimates fewer parameters.

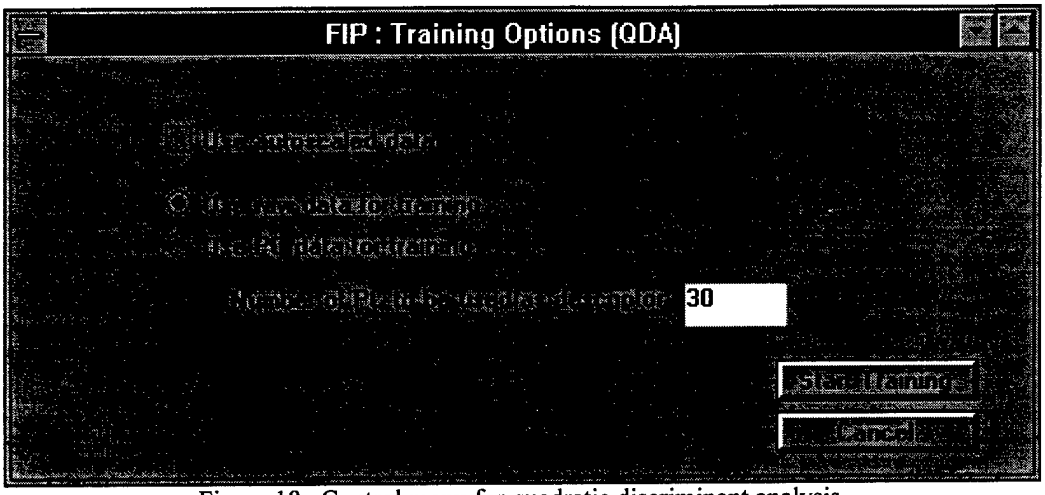

Figure 18. Control menu for quadratic discriminant analysis.

Regularized Discriminant Analysis (RDA):

A GUI training option box will be presented (see figure 19) when this method of training is invoked. A check box is provided to use autoscaled data . The user must select the values of lamda and gamma. RDA uses a complex biasing scheme to get better class covariance estimates. RDA is expected to perform well when the sample size is small compared to the dimensions of the measurement space. For high dimensional GC-fuel data , RDA is usually a good choice.

The default value of lamda is 0.1, and it can have any value between 0 and 1. Lamda is the regularization parameter that controls the mixing of the covariance matrix towards the pooled estimate. The default value of gamma is 0 and it can also assume any value between 0 and 1. Gamma is the regularization parameter that controls the shrinkage towards a multiple of the identity matrix.

For lamda=1 and gamma=0, RDA becomes LDA. For lamda=0 and and gamma=0, RDA becomes QDA. For GC fuel data the default value of lamda (0.1) and gamma (0) is good for most cases. The optimum value of lamda and gamma is the one which gives the lowest bootstrap or cross-validated error rate

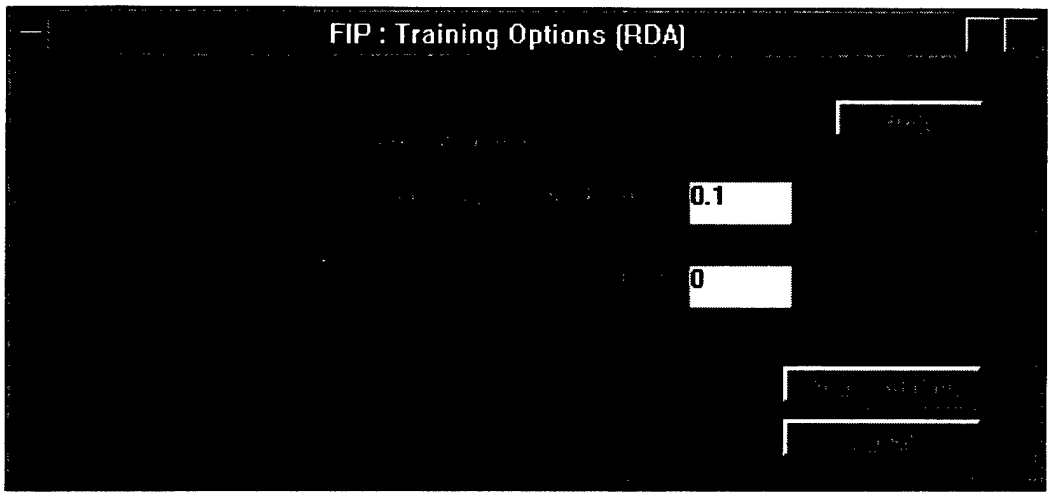

Figure 19. Control Menu for Regularized Discriminant Analysis.

Soft Independent modeling of class analogy (SIMCA):

A GUI training option box is presented (Figure 20) when this method of training is invoked. A check box is provided to use autoscaled data. The user must select the number of principal components necessary to describe each class. The optimum number of principal components is the one which gives the lowest bootstrap or cross-validated error rate. It often performs well when the observation to descriptor ratio is low. SIMCA is a variation of QDA and is a favorite classification method in chemistry.

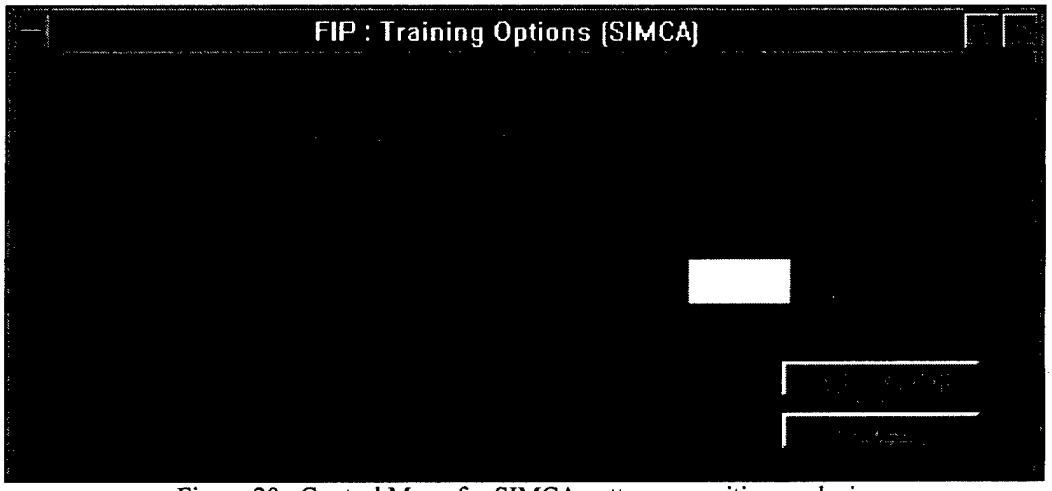

Figure 20. Control Menu for SIMCA pattern reognition analysis.

Discriminant Analysis with shrunken covariances (DASCO):

A GUI training option box is presented (see figure 21) when this method of training is invoked. A check box is provided to use autoscaled data . The user must select the percent of cumulative variances associated with the primary sub-space. The optimum value is the one that gives the lowest bootstrap or cross-validated error rate. It is expected to perform well with data that has a low object to descriptor ratio.

DASCO is a variation of QDA. It has been claimed that DASCO proviides a more reliable estimate of class covariance than SIMCA.

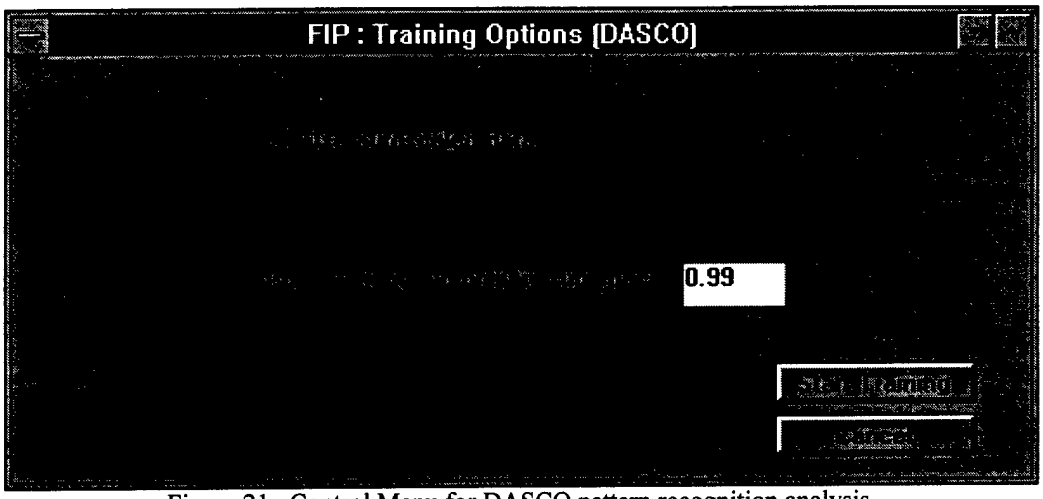

Figure 21. Control Menu for DASCO pattern recognition analysis.

Calculation of error rate

FIP provides two ways to calculate classification error rate: (1) bootstrap error rate and (2) cross-validated error rate. Since the apparent error rate is biased too high, it is recommended to use either bootstrap or cross validated error rates to evaluate the relative performance of different classification methods, as well as in selecting the optimum model parameters. Cross-validated error rate is calculated using the leave one out algorithm. It is computationally very intensive for a larger data set. The bootstrap method uses 10 random training set and test set generated from the available training set data. This method is computationally efficient. For GC-fuel data ,10 bootstrap cycles are enough to get a stable error rate. Before invoking boostrap or cross-validated error rate calculation for a particular classification method, data must be trained with that method using the necessary model parameters. The error rate is displayed after the cycle has been completed. It is possible to abort the operation if necessary by pressing Control-C simultaneously in the MATLAB command window. After such an abort operation, training set data must be reloaded from the disk file. It is recommended to save a trained model, as well as training or prediction set results in a disk file before starting the calculation of bootstrap or cross-validated error rate.

Representative Study

228 fuel samples representing five different types of jet fuels (JP-4, Jet-A, JP-7, JPTS, and JP-5) were obtained from Wright Patterson Air Force Base (OH) and Mukilteo Energy Management Laboratory (WA). The fuel samples were splits from regular quality control standards used by these two laboratories to verify the authenticity of manufacturers' claims that purchased fuels meet designated specifications. The quality control standards were collected over a three year period and constituted a representative sampling of the fuels.

The fuel samples, after they had arrived for the study, were immediately stored in sealed containers at -20^{0}C prior to analysis by gas chromatography. The gas chromatograms of these neat jet fuel samples were used as the training set (see Table 1). The prediction set consisted of 21 gas chromatograms

of weathered jet fuel (see Table 2). 11 of the 21 weathered fuel samples were collected from sampling wells as a neat oily phase which was found floating on top of the well water. 7 of the 21 fuel samples were recovered fuels extracted from the soil near various fuel spills. (Methylene chloride was used to extract the fuel from the soil via a quick swirl extraction.) The other 3 fuel samples had been subjected to weathering in the laboratory.

TABLE 1: TRAINING SET

Number of samples	Fuel-Type
54	JP-4 (fuel used by USAF fighters)
70	Jet-A (fuel used by civilian airliners)
32	JP-7 (fuel used by SR-71 Reconnaissance plane)
29	JPTS (fuel used by TR-1 and U-2 aircraft)
43	JP-5 (fuel used by Navy jets)

TABLE 2: PREDICTION SET

SAMPLE #	IDENTITY	SOURCE
PF007	JP-4	SAMPLING WELL AT TYNDALL AFB[a]
PF008	JP-4	SAMPLING WELL AT TYNDALL AFB[a]
PF009	JP-4	SAMPLING WELL AT TYNDALL AFB[a]
PF010	JP-4	SAMPLING WELL AT TYNDALL AFB[a]
PF011	JP-4	SAMPLING WELL AT TYNDALL AFB[a]
PF012	JP-4	SAMPLING WELL AT TYNDALL AFB[a]
PF013	JP-4	SAMPLING WELL AT TYNDALL AFB[a]
KSE1M2	JP-4	SOIL EXTRACT NEAR A SAMPLING WELL AT TYNDALL[b]
KSE2M2	JP-4	SOIL EXTRACT NEAR A SAMPLING WELL AT TYNDALL[b]
KSE3M2	JP-4	SOIL EXTRACT NEAR A SAMPLING WELL AT TYNDALL[b]
KSE4M2	JP-4	SOIL EXTRACT NEAR A SAMPLING WELL AT TYNDALL[b]
KSE5M2	JP-4	SOIL EXTRACT NEAR A SAMPLING WELL AT TYNDALL[b]
KSE6M2	JP-4	SOIL EXTRACT NEAR A SAMPLING WELL AT TYNDALL[b]
KSE7M2	JP-4	SOIL EXTRACT NEAR A SAMPLING WELL AT TYNDALL[b]
STALE-1	JP-4	WEATHERED IN LABORATORY[c]
STALE-2	JP-4	WEATHERED IN LABORATORY[c]
STALE-3	JP-4	WEATHERED IN LABORATORY[c]
PIT1UNK	JP-5	SAMPLING PITT AT KEYWEST NAVAL AIRSTATION[d]
PIT1UNK	JP-5	SAMPLING PITT AT KEYWEST NAVAL AIRSTATION[d]
PIT2UNK	JP-5	SAMPLING PITT AT KEYWEST NAVAL AIRSTATION[d]
PIT2UNK	JP-5	SAMPLING PITT AT KEYWEST NAVAL AIRSTATION[d]

[a]The sampling well was near a previously functioning storage depot. Each well sample was collected on a different day.

[b]Dug with a hand auger at various depths. Distance between sampling well and soil extract was approximately 80 yards.

[c]Old JP-4 fuel samples which had undergone weathering in a laboratory refrigerator.

[d]Two pits were dug near a seawall to investigate a suspected JP-5 fuel leak.

Prior to gas chromatographic (GC) analysis, each fuel sample was diluted with methylene chloride, and the diluted fuel sample was then injected onto a GC capillary column using a split injection technique. High speed GC profiles were obtained using a high efficiency fused silica capillary column (Hewlett Packard, Analytical Products Group, P.O. Box 9000, San Fernando, CA 91341-9981) 10 m long with an internal diameter of 0.10 mm and coated with 0.34 μm of a bonded and cross-linked 5% phenyl-substituted polymethylsiloxane stationary phase. The column was temperature programmed from 60 to 270^0 degrees Centigrade at 18^0 per minute using an HP-5890 gas chromatograph equipped with a flame ionization detector, a split/splitless injection port, and an HP-7673A autosampler. Gas chromatograms representative of the five fuel types in this study are shown in Figure 1.

The GC data were digitized and stored using an HP-3357 laboratory automation system implemented on an HP-1000-F minicomputer. A FORTRAN program was used to translate the integration reports into ASCII files formatted for entry into SETUP (18), a computer program for peak matching. SETUP correctly assigned the peaks by first computing the Kovat's retention index for the compounds eluting off the GC column. Since the n-alkane peaks are the most prominent features present in the gas chromatograms of these fuels (19), it was a simple matter to compute the Kovat's retention index for each GC peak. The peak matching program then analyzed the GC data in three distinct steps. First, a template of peaks was developed by examining integration reports and adding features to the template which did not match the retention indices of previously observed features. Second, a preliminary data vector was produced for each gas chromatogram by matching the retention indices of GC peaks with the retention indices of the features in the template. A feature would be assigned a value corresponding to the normalized area of the GC peak in the chromatogram. Unmatched peaks were zeroed, whereas poorly resolved and tailing peaks were excluded from the analysis. Third, the frequency of each feature was computed, i.e., the number of times a particular feature is found to have a nonzero value was calculated, and features below a user specified number of nonzero occurrences (which was set equal to 10% of the total number of fuel samples in the training set) were deleted from the data set, whereas features that passed the non-zero frequency criterion were retained. The peak matching software yielded a final cumulative reference file containing 85 identities though not all peaks were present in all chromatograms. Hence, for pattern recognition analysis, each gas chromatogram was initially represented as a 85 dimensional data vector, $x = (x_1, x_2, x_3, ... x_j, ..., x_{85})$, where x_j is the area of the jth peak. The data vectors were normalized to constant sum, i.e., each x_j was divided by the total integrated peak area.

Because outliers have the potential to adversely affect the performance of statistical and pattern recognition methods, outlier analysis was performed on each fuel class in the training set prior to pattern recognition analysis using the generalized distance test (20) which was implemented via SCOUT (21). 3 Jet-A and 4 JP-7 fuel samples were found to be outliers by both tests at the 0.01 level; therefore, these 7 fuel samples were removed from the data base. Hence, the set of data -- 221 gas chromatograms of 85 peaks each -- was transferred via floppy diskette to FIP (21) which was installed on a SUN SPARC II workstation. The data were standardized and autoscaled so that each variable (peak) had a mean of zero and a standard deviation of one within the entire set of 221 gas chromatograms. Thus, autoscaling ensured that each feature had equal weight in the analysis.

The pattern recognition analyses were directed toward three specific goals: (1) finding discriminants that can correctly classify neat jet fuels on the basis of legitimate chemical differences between the different types of fuels, (2) studying the structure of the GC data to seek obscure relationships with mapping and display methods, and (3) developing the ability to predict the class membership of weathered fuels. Both principal component and statistical discriminant analysis were used to analyze the fuel data.

The first step was to apply PCA to the analysis of the training set data, in order to obtain information about the overall trends present in the data. Each fuel sample or gas chromatogram was represented as a point in a 2-dimensional principal component map. The JP-4, JP-7, and JPTS fuel samples were well separated from one another and from the gas chromatograms of Jet-A and JP-5 fuel samples in

the map, suggesting that information characteristic of fuel type is present in the high speed gas chromatograms of the neat jet fuels. Because this projection is made without the use of information about the class assignment of the fuel samples, the resulting separation is, therefore, a strong indication of real differences in the hydrocarbon composition of these fuels as reflected in their gas chromatographic profiles.

The overlap of Jet-A and JP-5 fuel samples in the principal component map suggests that gas chromatograms of these two fuel materials share a common set of attributes which is not surprising because of the similarity in their physical and chemical properties, e.g., flash point, freezing point, vapor pressure, and distillation curve. Mayfield and Henley in a previous study (19) observed that gas chromatograms of Jet-A and JP-5 fuels were more difficult to classify than gas chromatograms of other types of processed fuels because of the similarity in the overall hydrocarbon composition of these two fuel materials. Nevertheless, Mayfield and Henley also concluded that fingerprint patterns exist within the high speed gas chromatograms of Jet-A and JP-5 fuels characteristic of fuel type, which is consistent with score plot of the second and third largest principal components of the 85 GC peaks, suggesting that differences do indeed exist between the hydrocarbon profiles of Jet-A and JP-5 fuels. Since the second and third largest principal components do not represent the directions of maximum variance in the data, we must conclude that most of the information contained within the 85 GC peaks is not about the differences between GC profiles of Jet-A and JP-5 fuels.

To better understand the problems associated with classifying gas chromatograms of Jet-A and JP-5 fuels, we found it necessary to re-examine this particular classification problem using PCA. An examination of a principal component map developed from 85 GC peaks of 110 Jet-A and JP-5 fuel samples revealed a very interesting result. Although the Jet-A and JP-5 fuel samples lie in different regions of the principal component map, the data points representing the JP-5 fuels form two distinct subgroups in the map, which could be a serious problem since an important requirement in any successful pattern recognition study is that each class in the data set is represented by a homogeneous collection of objects. In other words, it will be difficult to adequately represent the gas chromatograms of the JP-5 fuels by a single prototypical class vector which is necessary in order to successfully implement SDA or variations of it. Therefore, it is important that we identify and delete from the data set the GC peaks responsible for the sub-clustering of the JP-5 fuel samples in the 85-dimensional pattern space.

Hence, the following procedure was used to identify GC peaks strongly correlated with the sub-clustering. First, the JP-5 fuel samples were divided into two categories on the basis of the observed sub-clustering. Next, the variance weights were computed for the GC peaks so that peaks strongly correlated with this sub-clustering could be identified. Variance weights were also computed for the following category pairs: JP-4 vs JP-5, Jet-A vs JP-5, JP-7 vs JP-5, and JPTS vs JP-5. A GC peak was retained for further analysis, only if its variance weight for the sub-clustering dichotomy was lower than for any of the other category pairs. 27 GC peaks were retained for further study. A plot of the scores of the two largest principal components of the 27 GC peaks obtained from the 221 neat jet fuel samples shows separation for all the fuel classes. Since PCA does not directly utilize class information about the fuel samples in the development of a map for the data, the eigenvector projection should be viewed in the context of this study as a conservative estimate of differences in hydrocarbon composition of the fuels as reflected by their GC profiles. In otherwords, the fact that fuel samples in the principal component map cluster according to fuel type suggests that information is contained within the gas chromatograms of the fuels characteristic of fuel type.

Table 3 shows the results of K-nearest neighbor, i.e., K-NN, which was also used to analyze the data. (The K-NN method categorizes the data vectors in the training set according to their proximity to other objects of pre-assigned categories.) It is evident on the basis of K-NN and the PCA map that in the 27-dimensional measurement space the five fuel classes are well separated, and each fuel class is represented by a homogeneous collection of objects.

TABLE 3: K-NN CLASSIFICATION RESULTS

CLASS	NIC	1-NN	3-NN	5-NN	7-NN
JP-4	54	54	54	54	54
JET-A	67	67	67	67	67
JP-7	28	28	28	28	28
JPTS	29	29	29	29	29
[a]JP-5	43	43	41	36	37
TOTAL	221	221	219	214	215

[a]Misclassified JP-5 fuel samples were categorized as JET-A. This result is not surprising because of the similarity in the hydrocarbon composition of these two fuel materials (see reference 19).

A five-way classification study involving the JP-4, Jet-A, JP-7, JPTS, and JP-5 fuel samples in the truncated pattern space was also undertaken using QDA, LDA, SIMCA, back propagation neural networks (BPN), discriminant analysis with shrunken covariance (DASCO), and regularized discriminant analysis (RDA). DASCO and RDA, like SIMCA, also utilize nonsample based methods to stabilize the inverse of the class covariance matrix which is then substituted into the quadratic discriminant analysis rule. DASCO, like SIMCA, partitions the pattern space into a primary and secondary subspace. The contribution of the primary subspace to the inverse of the covariance matrix is estimated directly from the primary eigenvalues, whereas the eigenvalues associated with the secondary or complementary subspace are averaged like in SIMCA. (In SIMCA the primary eigenvalues are ignored.) RDA employs a more complex scheme to obtain a biased estimate of the class covariance matrix. RDA shrinks the class covariance matrix towards the pooled covariance matrix, while simultaneously shrinking the eigenvalues of the class covariance matrix towards equality (by shrinking the resulting estimates towards multiples of the identity matrix). Optimum values of these shrinkage parameters are computed for a given data set by cross validating on the total number of misclassifications. (In otherwords, a vector of misclassifications as a function of the shrinkage parameter is generated, with the value of the evaluated parameter corresponding to the lowest error rate selected.)

Results from the 5-way classification study involving the 221 neat jet fuel samples are shown in Table 4. The recognition rates for the discriminants developed from the 27 GC peaks using LDA, QDA, SIMCA, DASCO, RDA, or BPN are very high. Evidently, the gas chromatograms of the neat jet fuels contain information characteristic of fuel type.

TABLE 4: TRAINING SET RESULTS

METHOD	APPARENT[1]	BOOTSTRAP[2]	CROSS VALIDATION[3]
LDA	96.8%	96.0%	93.7%
QDA	100%	97.8%	97.3%
SIMCA[4]	99.5%	98.3%	96.4%
DASCO[5]	100%	99.2%	97.7%
RDA[6]	100%	99.2%	98.2%
BPN	100%	99.2%	99.5%

[1]The ability of the discriminant to correctly classify those samples with which it was developed.

[2]60% of the samples were chosen at random from the training set, and a classification rule is developed from these fuel samples. The classification rule is validated using all of the fuel samples in the original training set, and the fraction of samples correctly classified in the validation set is computed. This procedure is repeated ten times, and the recognition rate for the classifier is equal to the average classification success rate obtained for the validation set.

[3]The classification rule is developed on one part of the training set, with the other part functioning as a mock test set. This process is repeated until all training set samples have been used as test set samples. The recognition rate is equal to the fraction of mock test set samples correctly classified.

[4]An 8 principal component model was developed for each fuel class. The number of principal components for each class model was determined by cross validating on the total number of misclassifications.

[5]The primary subspace was defined by the 8 largest principal components.

[6]Gamma was set equal to 0.0 and lambda was set equal to 0.2.

To test the predictive ability of these GC peaks and the discriminants associated with them, a prediction set of 21 gas chromatograms was employed (see Table 2). The gas chromatograms in the prediction set were run a few months before the neat jet fuel gas chromatograms were run and thus constituted a true prediction set. Table 5 summarizes the results of this experiment. RDA, DASCO, and SIMCA correctly classified all of the weathered fuel samples in the prediction set, whereas LDA and BPN misclassified 14 of the 21 weathered fuel samples. QDA misclassified 4 of the 21 weathered fuels. The disparity between the recognition and classification success rates for the discriminants developed using LDA or BPN would suggest that both cross validated and bootstrapped estimates of the error rate can be overly optimistic figures of merit, despite claims made to the contrary. (The apparent recognition rate is considered to be too optimistic by all workers in the field, because the samples used in the design of the classifier are the same ones used for testing, so differences between this figure of merit and the classification success rate obtained for samples in the prediction set is not unexpected.) Evidently, a reliable estimate of the error rate for a classifier requires the use an independent sample test set, i.e., samples that have not been used in the design of the classifier.

TABLE 5: PREDICTION SET RESULTS

METHOD	ERROR RATE
LDA	14 MISCLASSIFIED (ALL JP-4)
QDA	4 MISCLASSIFIED (ALL JP-5)
SIMCA[1]	0 MISCLASSIFIED
DASCO[1]	0 MISCLASSIFIED
RDA[1]	0 MISCLASSIFIED
BPN[2]	14 MISCLASSIFIED

[1]Posterior probability for the samples correctly classified was greater than 75%.

[2]Best results.

The fact that QDA outperformed LDA (see Table 5) comes as no surprise because the assumption of equality between class covariance matrices is not justified in this problem, as evidenced by the unequal dispersion of the points representing the different fuels in the plot of the two largest principal components obtained from the 27 GC peaks. With regards to BPN, we attribute its poor performance to overfitting of the training set data which is a serious problem with certain types of artificial neural networks. The fact that SIMCA, DASCO, and RDA outperformed QDA is also not surprising since these methods were developed specifically for small sample/high dimensional settings. However, the fact that SIMCA, DASCO and RDA performed equally well in this study raises questions about the designation of either RDA or DASCO as a so-called best method for pattern recognition problems involving data sets with a low object to descriptor ratio. In all likelihood, these three methods perform equally well with real chemical data, so observed differences in performance between SIMCA, DASCO, and RDA for a given problem are probably application specific.

Finally, the high classification success rate obtained for the weathered fuels suggests that information about fuel type is present in the gas chromatograms of weathered fuels. This is a significant result since the changes in composition that occur after a processed fuel is released into the environment constitute a major problem in fuel spill identification. These changes arise from evaporation of lower molecular weight alkanes, microbial degradation, and the loss of water soluble compounds due to

dissolution. However, the weathered fuel samples used in this study were recovered from a subsurface environment. Loss of lower alkanes due to evaporation is severely retarded in a subsurface environment, and only a comparatively small number of jet fuel components are soluble in water. (If the selective evaporation of lower alkanes was not retarded in the subsurface environment, the weathered JP-4 fuel samples which are high in volatiles could not have been identified using discriminants developed from the gas chromatograms of the neat jet fuels.) Hence, the predominant weathering factor in subsurface fuel spills is probably biodegradation due to the action of microbial organisms which does not appear to have a pronounced effect on the overall GC profile of the fuels. Therefore, the weathering process for aviation turbine fuels in subsurface environments is greatly retarded in comparison to surface spills, thereby preserving the fuel's identity for a longer period of time.

Acknowledgement

This research was supported in part by an appointment of Abdullah Faruque to the Postgraduate Research Program at the Tyndall Air Force Base administrated by the Oak Ridge Institute for Science and Education through an interagency agreement between the U.S. Department of Energy and the U.S. Air Force Armstrong Laboratory.

References

1. Cohen, S. Z.; Creeger, S. M.; Carsel, R. F.; Enfield, C. G. In "Treatment and Disposal of Pesticide Wastes"; Kruager, R. F.; Seiber, J. N., Eds.; ACS Symposium Series No. 259, American Chemical Society: Washington, D.C., 1984, pp. 297-325.

2. Zemo, D.A.; Bruuya, J.E.; Graf, T.E. Groundwater, 1995, 147-156.

3. Kawahara, F.K.; **J. Chromatogr. Sci.**, 1972, 10, 629-635.

4. Kawahara, F.K. and Yang, Y.Y. **Anal. Chem.**, 1976, 48, 651-656.

5. Yang, W.C.; Wang, H., Water Res., 1977, 11, 879-885.

6. Jurs, P.C.; Lavine, B.K.; Stouch, T.R., NBS Journal of Research, 1985, 543-549.

7. Dunn, W.J.; Stalling, D.L.; Schwartz, T.R.; Hogan, J.W.; Petty, J., Analytical Chemistry, 1984, 56, 1308-1315.

8. Duda, R.O.; Hart, P.E., "Pattern Classification and Scene Analysis," John Wiley & Sons, NY, 1973.

9. Lavine, B.K.; Carlson, D., Anal.Chem., 1987, 59, 468A-472A.

10. McLachlan, G.J. Discriminant Analysis and Statistical Pattern Recognition, John Wiley & Sons, NY., 1992, pp. 129-167.

11. A.M.Harper, D.L.Duewer, B.R.Kowalski, and J.L.Fasching, "ARTHUR and Experimental Data Analysis: the Heuristic Use of Polyalgorithms," in Chemometrics: Theory & Application, B.R.Kowalski, (Ed.), ACS Symposium Series 52, Washington, D.C. 1977.

12. Sharaf, M.; Illman, D.; Kowalski, B.R. Chemometrics, John Wiley & Sons, NY. 1986, pp 195.

13. Lavine, B.K.; Qin, X.; Stine, A.; and Mayfield, H.T.; **Process Control and Quality**, 1992, 2, 347-355.

14. Lavine, B.K.; Stine, A.; Mayfield, H.T. **Anal. Chim. Acta.**, 1993, 227, 357-367.

15. Lavine, B.K. **Chemolab**. 1992, 15, 219-230.

16. B.K.Lavine, A.B.Stine, H.Mayfield, and R.Gunderson, JCICS, 1993, 33, 826-834.

17. MATLAB User's Guide, The Math Works Inc., Natic, Mass, 1993

18. Mayfield, H.T.; Bertsch, W. **Comput. Appl. Lab**. 1983, 1, 130-137.

19. Mayfield, H.T.; Henley M. Monitoring Water in the 1990s: Meeting New Challenges, American Chemical Society for Testing and Materials, Hall, J.R.; Glayson, G.D. (Eds.). Philadelphia, PA 1991, pp. 578-597.

20. Schwager, S.J.; Margolin, B.H. **Ann. Stat.**, 1982, 10, 943-953.

21. Stapanian, M.A.; Garner, F.C.; Fitsgerald, K.E.; Flatman, G.T.; Nocerino, J.M. **J. Chemom.**, 1993, 7, 165-176.

22. Lavine, B.K.; Faruque, A.; Mayfield, H. **J.Comp.Inf.Sci.**, submitted.