SANDIA REPORT
SAND2000-3122
Unlimited Release
Printed December 2000

Final Report: Weighted Neighbor Data Mining

Jeffrey J. Carlson, Maritza R. Muguira, J. B. Jordan, G. M. Flachs, and A. K. Peterson

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy under Contract DE-AC04-94AL85000.

Sandia National Laboratories

Issued by Sandia National Laboratories, operated for the United States Department of Energy by Sandia Corporation.

Available to DOE and DOE contractors from
U.S. Department of Energy
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831

Telephone: (865)576-8401
Facsimile: (865)576-5728
E-Mail: reports@adonis.osti.gov
Online ordering: http://www.doe.gov/bridge

Available to the public from
U.S. Department of Commerce
National Technical Information Service
5285 Port Royal Rd
Springfield, VA 22161

Telephone: (800)553-6847
Facsimile: (703)605-6900
E-Mail: orders@ntis.fedworld.gov
Online order: http://www.ntis.gov/ordering.htm

SAND2000-3122
Unlimited Release
Printed December 2000

Final Report: Weighted Neighbor Data Mining

Jeffrey J. Carlson and Maritza R. Muguira
Intelligent System Sensors and Controls
Sandia National Laboratories
P. O. Box 5800
Albuquerque, NM 87185
jjcarls@sandia.gov

J. B. Jordan, G. M. Flachs, and A. K. Peterson
Electronic Vision Research Laboratory
New Mexico State University
Las Cruces, NM 88003

Abstract

Data mining involves the discovery and fusion of features from large databases to establish minimal probability of error (MPE) decision and estimation models. Our approach combines a weighted nearest neighbor (WNN) decision model for classification and estimation with genetic algorithms (GA) for feature discovery and model optimization. The WNN model is used to provide a mathematical framework for adaptively discovering and fusing features into near-MPE decision algorithms. The GA is used to discover weighted features and select decision points for the WNN decision model to achieve near-MPE decisions. The performance of the WNN fusion model is demonstrated on the first of two very different problems to demonstrate its robust and practical application to a wide variety of data-mining problems. The first problem involves the isolation of factors that cause hepatitis C virus (HCV) and requires the evaluation of large databases to establish the critical features that can detect with minimal error whether a person is at risk of having HCV. This requires discovering and extracting relevant information (features) from a questionnaire database and combining (fusing) them to achieve a minimal error decision rule. The primary objective of the research is to develop a practical basis for fusing information from questionnaires administered at hospitals to identify and verify features important to isolate risk factors for HCV. The basic problem involves creating a feature database from the questionnaire information, discovering features that provide sufficient information to reliably identify when a person is at risk under conditions with uncertainties caused by recording errors and evasive tactics of people answering the questionnaire. The results of this study demonstrate the WNN fusion algorithm ability to perform in supervised learning environments. The second phase of the research project is directed at the unsupervised learning environment. In this environment the feature data is presented without any classification. Clustering algorithms are developed to partition the

feature data into clusters based upon similarity measure models. After the feature data is clustered and classified the supervised WNN fusion algorithms are used to classify the data based upon the minimal probability of error decision rule.

Contents

1. INTRODUCTION

The basic problem under investigation is the development of an information fusion system for both the supervised and unsupervised learning environments. In the supervised learning environment the feature data is presented with a classification and in the unsupervised environment the feature data is presented without classification. A WNN fusion algorithm is developed to evaluate and fuse classified feature data into minimal probability of error decision algorithm. The model provides a method to discover the important features in achieving MPE decision algorithms. To generalize the WNN model to handle unclassified feature data, clustering algorithms are developed to classify the feature data based on similarity measures. The models developed are used to discover features for identifying the important factors that cause HCV. The goal is to develop a system with a high probability of detecting when a person is at risk of having HCV. The questionnaire administered has 185 questions (features) related to the general background, medical history, blood exposure, needle exposure, alcohol use, demographics, insect bites, sexually transmitted diseases, and sexual habits of the people entering the hospital. These features are investigated to evaluate their ability to detect whether a person has hepatitis C virus. The thrust of the research is to analyze the feature space of the problem and to establish the features that can be fused to reduce the complexity of the recognition problem, consequently reducing the error probability.

Our approach involved extending our Weighted Nearest Neighbor (WNN) fusion model to improve its ability to fuse large feature spaces to develop minimal error decision monitoring systems. Two types of errors are used to describe the performance of a fusion monitoring system. The first error type is the failure to detect a person with HCV (miss) and the second involves issuing a false alarm when a person does not have HCV. A quantitative feature database was established from the questionnaire database. The WNN fusion model is used to discover and fuse features into near-MPE decision algorithm for detecting when a person is a high risk of having HCV..

The information fusion algorithm selects and weights the features based on estimates of the probability of error. Genetic selection algorithms are used to evolve a weighted feature vector and decision points to minimize the probability of error. Genetic algorithms (GA) are very effective at finding near optimal solutions in complex, high dimensional problems. The properties that make the GA robust to local extrema also make it computational intensive.

This report presents a description of the WNN fusion model with the new features added to improve its performance on a wide variety of data mining problems. Several clustering algorithms are implemented, extending the WNN fusion model to unsupervised learning environments. The WNN model is used to discover and extract relevant information (features) for the HCV data-mining problem and fusing the important features to achieve a minimal error decision rule. A program was developed to read the HCV questionnaire to form a feature database for the fusion algorithm. The feature database was analyzed with a linear correlator to order the features based on their ability to identify a person at risk of HCV. The results of the WNN fusion process are presented to demonstrate its performance on the HCV data-mining problem.

2. WEIGHTED NEAREST-NEIGHBOR FUSION MODEL

A mathematical framework is presented for the WNN fusion model. The problem is formulated in terms of a statistical hypothesis-testing problem. The model insures that the minimal probability of error cannot increase by adding more features [11]. The problem of estimating probability density functions (PDF's) in high dimensional spaces is converted to estimating PDF's in a single nearest neighbor distance space. The WNN fusion model is used to fuse the features into a minimal probability of error decision algorithm. Genetic algorithms are used to select and weight the features to obtain a near minimal probability of error solution.

2.1 Problem Formulation

Many decision and control problems can be formulated as a set of binary statistical hypothesis tests [7,8,10], one for each decision class. For mathematical convenience, the decision and control problem is described here as a binary hypothesis test. In binary hypothesis testing, a statement or claim that something is true, called the null hypothesis H_o, is tested against its alternative H_a to establish with confidence the most probable decision. Consider the features as a vector of random variables $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n)$ used to distinguish the decision class C_j from the other classes and let $x = (x_1, x_2, \ldots, x_n)$ be a given measurement of these random variables. When the features are measured $y = (y_1, y_2, \ldots, y_n)$ from an unknown decision class and compared to measurements from a known class $x = (x_1, x_2, \ldots, x_n)$, the decision is formulated as:

H_o: The features measurement y came from the known class.
H_a: The features measurement y came from a different class.

The feature measurements are perturbed by measurement noise and many random factors, including many environmental and perhaps information warfare factors. The conditional probability density functions (PDF's) under the H_o and H_a hypotheses are represented as $f_{\mathbf{x}}(x_1,x_2,\ldots,x_N|H_o)=f_{\mathbf{x}}(\mathbf{x}|H_o)$ and $f_{\mathbf{x}}(x_1,x_2,\ldots,x_N|H_a)=f_{\mathbf{x}}(\mathbf{x}|H_a)$. If these PDF's are known and the a priori probabilities of occurrence of each hypothesis are also known, an optimal hypothesis test, in terms of minimizing the probability of making an error, can be formulated [7,8,10]. The probability of error for this optimal test can also be computed, giving a bound on the achievable performance associated with the chosen features. The difficulty lies in estimating the PDF's.

Approximating the minimum probability of error using estimates of $f_{\mathbf{x}}(\mathbf{x}|H_o)$ and $f_{\mathbf{x}}(\mathbf{x}|H_a)$ requires a priori knowledge of the probability of occurrence of each hypothesis $P(H_o)$ and $P(H_a)$. These a priori probabilities are often not known and an equally likely assumption is often used to estimate the minimum probability of error (MPE). The MPE is a measure of the overlap of two joint distributions in the feature measurement space that has proven to be an effective measure of the ability of a set of features to distinguish objects from different classes. For continuous random variables where Ω is the set of all $\mathbf{x}$, the MPE parameter is defined in terms of these joint PDF's. For discrete random variables, the MPE statistic is defined in terms of the probability mass functions.

The operator $\wedge$ selects the minimum.

$$MPE(x) = \int_{\Omega} P(H_o) f_X(x \mid Ho) \wedge P(H_a) f_X(x \mid Ha) .$$

$$MPE(x) = \sum_{\Omega} P(H_o) f_X(x \mid Ho) \wedge P(H_a) f_X(x \mid Ha)) .$$

2.2 Weighted Nearest-Neighbor Fusion Model

The weighted nearest neighbor fusion model [2] provides a method for analyzing and fusing multiple features to design and optimize a decision process. The fusion process involves discovering the features that can be fused to obtain robust and minimal error decision algorithms. A weighted nearest neighbor (WNN) model is utilized to provide the mathematical framework for fusing features into near minimal probability of error decision algorithms.

Training samples are used to guide the feature selection and fusion process. Each training sample $x = (x_1, x_2, \ldots, x_n)$ represents a point in an n dimensional space. N_j training samples are used to establish a set of NDP_j decision points, dp^j, to characterize the statistical decision surface for each decision class C_j. A weighted distance d_j=WNN(y,dp^j) from an unknown sample y to the nearest

$$WNN(y, dp^j) = \sum_{k=1}^{n} w_k (y_k - dp_k^j)^2$$

neighbor decision point dp^j of class C_j is used to fuse the features and decide class membership. An unknown sample y is given C_j membership if its nearest neighbor decision point is in class C_j and $d_j < T_j$. The thresholds T_j are chosen to achieve the desired false acceptance and rejection rates. The weights $w_k \in [0,1.0]$ and the decision points dp^j are chosen to minimize the probability of error of the decision process using a genetic algorithm search process. A weight of zero effectively eliminates the feature from the decision process and indicates the feature does not contribute to the minimal error solution. The higher the weight the more the feature contributes to the decision process. The key idea in using the WNN distance is to reduce the estimation of n-dimensional PDF's to a single dimensional distance estimation problem.

Training samples are used to estimate the one dimensional conditional probability density functions for the minimal in-class distance $f(\mathbf{d}_j \mid C_j \; \forall j)$ and the minimal out-of-class distance $f(\mathbf{d}_j \mid C_k \; j \neq k)$. The probability of error is estimated by integrating the minimum of the conditional probability density functions over the observation space O(d) of the in-class and out-of-class distances. For discrete probability density functions the probability of error (pe) is given by

$$pe = \frac{1}{2}\sum_{O(d)} f(d_j \mid C_j) \wedge f(d_j \mid C_k : j \neq k)$$

where the symbol $\wedge$ is a minimum select operator and the a priori probabilities are chosen equal $P(C_j)=P(C_k)=½$. The feature weights directly affect the distance measurements that affect the conditional probability functions and the probability of error. The genetic optimization method is used to select the weights and decision points to minimize the pe given a set of potential features. The net result of these operations is to select and fuse the features to achieve a minimal probability of error decision algorithm.

A key variable in the weighted nearest neighbor fusion process is the number of samples required for a minimal probability of error solution. Theoretical methods [7,17,18]can be used to estimate the number of samples required under very stringent parametric estimation conditions. These estimates of the sample size required for reliable parameter estimation are often unrealistically large. Our experience with the WNN decision process, however, indicates that a vastly fewer number of samples can be used to approach the minimal probability of error solution when the decision points are optimized with the GA to minimize the probability of error. Current research is centered on analyzing the training samples to establish on optimal set of decision points for each class to reduce the minimal probability of error. Recent results indicate that the number of decision points can be significantly reduced while at the same time reducing the minimal probability of error.

2.3 Genetic Fusion Process

Genetic algorithms are used to establish the feature weights and the decision points. Several methods have been developed to establish the initial population of decision points for each class. After the initial decision points are establish the GA evaluates the decision points based on the number of correct decisions and the number of errors associated with each decision points. Based on these evaluations new populations of decision points are generated and evaluated. Four methods are available to generate an initial population of decision points. The methods are given below selected by an opcode:

0 => Cluster sample data
1 => Read DP's from a file
2 => Read partial DP's
3 => Use class means of features

The clustering method uses the sample database to find peaks in the feature space. A gradient peak-seeking algorithm is used to find the peaks using an n-dimensional search cube. When a peak is found, the initial starting point for the next search is determined to be the sample point with maximum distance from all previous peaks found. This process attempts to find the peaks that are maximally separated. The process stops when no new peaks are found after repeated trials. This clustering method is relatively fast and provides a good starting population for the GA. The second method allows the user to select random samples from each class as an initial population. Experiments have demonstrated that from five to ten random samples from each class generates an

excellent starting population for problems with forty or more features. The third method allows a user to project a solution from a lower dimensional feature space to a higher dimensional space. Good solutions are easier to find in low dimensional feature spaces. Using this method, additional features can be added and solutions found, insuring the probability of error (pe) will not increase. The feature discovery process uses this approach to locate new features. When a feature is found to decrease the pe the feature is accepted and process of find finding another is repeated. The last method allows the user select the class means for generating an initial population for the GA. The class means often provide a good starting point for the GA search process. Often all these methods lead to relatively similar solutions

The genetic fusion process also allows the user to analyze a large feature space to evaluate the features and find the most significant features to minimize the probability of error. Very large feature spaces require extremely large sample data sets to properly characterize the decision surfaces. In many applications there is not sufficient data to properly characterize the decision surfaces. In these situation one often asks is there a smaller feature set that approaches the MPE decision surfaces. A genetic feature search is provided to find a smaller set of significant features. The genetic feature search starts by analyzing many populations of two features and selects the best two features. The feature search continues by adding randomly selected features to find the best three features. This process continues until no significant improvement in the probability of error can be obtained by adding more features. The feature search algorithms provide a method to evaluate and find the significant features in forming near MPE decision surfaces.

The goal of the GA is to minimize the probability of error (pe) by generating a sequence of feature sets, decision point populations, and feature weights. The genetic algorithm attempts to find the best features and decisions points by continually evaluating different feature sets and decision point populations, replacing the poor performers that are causing errors and updating the feature weights to minimize the probability of error. The process continues until no improvements are made in the pe after repeated trial.

3.0 UNSUPERVISED LEARNING ALGORITHMS

Several clustering methods have been tested to classify sample data sets to find meaningful clusters and isolate outliers for the data-mining problem. The clustering algorithms provide a classified data set directly for the WNN fusion to produce an evaluation of the feature space or near MPE decision surface.

3.1 K-Nearest Neighbor Clustering Model

The K-Nearest Neighbor Clustering approach [19] uses similarity measures based upon mutually shared K-nearest neighbors to combine two feature samples. No assumptions are made of the underling probability density functions and the method works well with smaller data sets. There have been many similarity measures developed to combine feature samples into clusters. Most of the measures are based on some visual concept of a cluster. The similarity measures based on the mutual nearest neighbor concept result in clusters that are visually appealing and work well with the WNN fusion model. Outlier feature points that do not belong to any cluster are generally separated into singleton clusters. The detection of these feature samples is important to many data mining problems, since these samples often represent unusual situations.

Let $\{ \mathbf{x}_0, \mathbf{x}_1, \ldots, \mathbf{x}_N \}$ be the n-dimensional feature sample data set to be clustered into similar classes. Let KNN($\mathbf{x}_i$) be the K nearest neighbors of $\mathbf{x}_i$ and KNN($\mathbf{x}_j$) be the K nearest neighbors of $\mathbf{x}_j$.

KNN Similarity Definition: Two sample $\mathbf{x}_i$ and $\mathbf{x}_j$ are similar iff $\mathbf{x}_i \in$KNN($\mathbf{x}_j$) and $\mathbf{x}_j \in$KNN($\mathbf{x}_i$).

The only variable in this similarity measure is the size of K or the size the K nearest neighborhood. If K is small the nearest neighborhoods are small and similar samples are located geometrically close. When K becomes larger, the nearest neighborhoods get larger and similar sample can be located further apart. Some studies require additional members of the KNN($\mathbf{x}_i$) and KNN($\mathbf{x}_j$) sets to match or require additional constraints on the distances to the nearest neighbors. These added complexities allow finer separation of the clusters but make the job of controlling the clustering process much more difficult. Having a single parameter K to control the clustering process greatly simplifies the clustering process.

The KNN clustering algorithm uses the parameter K to refine the resolution of the clusters to obtain a given bound M on the number of clusters representing the data set. Select a K sufficiently large (KT= 15) to meet the bound on the number of clusters.

Step 1: Form the KNN table KNNT(i,j) with rows identified with the data points i = 0, 1, ..., N and the columns representing the j^{th} nearest neighbor j = 0,1, ..., KT . Note that each point is its own 0^{th} nearest neighbor. Consequently, the first element in each row identifies the sample point and the remaining entries in the row represent an ordered list of its nearest neighbors.

Step 2: Select K= 2 to produce the finest clustering of the data set.

Step 3: Copy the first K columns of the KNNT table to a working table RKNNT. Reduced the RKNNT by applying the KNN similarity measure to each pair of rows. Observe that two row i and j are similar if RKNNT(i,0) is in the set { RKNN(j,0), RKNN(j,1), ..., RKNN(j,K) } and RKNNT(j,0) is in the set { RKNN(i,0), RKNN(i,1), ..., RKNN(i,K)}. If two rows are similar replace the largest row label

$$\text{REPLACE} = \max(\text{RKNN}(i,0), \text{RKNN}(j,0))$$

with the smaller row label through out the whole table. Continue applying the similarity measure to row pairs until no pair of remaining rows are similar. The distinct row labels NC left represent the clusters and each sample $\mathbf{x}_i$ belong to the cluster with label RKNN(i,0). Some of the clusters may contain just one or two sample points. These clusters can be removed as outliers with the corresponding reduction in NC.

Step 4: If NC is greater than the expected number of clusters M then increment K and continue with Step 2, else the clustering process is complete and the data is labeled. With the data labeled the supervised WNN fusion process can be used to provide a MPE decision process.

During the process of increasing K to reduce the number of clusters to the desired number M, the number of clusters may suddenly fall to well below M. In fact, if K is raised high enough the number of clusters approaches zero. When this occurs, the process must be stopped before the number of clusters is less than M. If the user still wants to further reduce the number of clusters, the clusters are compared to find the closest pair and merge them into one cluster. This process allows the user to obtain the desired number of clusters.

3.2 Peak Clustering Model

The Peak Cluster Model (PCM) uses a gradient seeking algorithm to find the peaks in the n-dimensional sample data space. After the peaks are found, each sample point is labeled by the peak that it is attracted to by the gradient seeking algorithm. An n-dimensional cube is used in the peak search algorithm to count the number of sample points in the cube. The cube is moved along a trajectory to maximize the number of samples in the cube. When a new peak is found, it is labeled by a new cluster number and all sample points attracted to the peak are given the peak label. This process is continued until all peaks are found.

PCM Similarity Definition: Two samples $\mathbf{x}_i$ and $\mathbf{x}_j$ are similar iff $\mathbf{x}_i$ and $\mathbf{x}_j$ are attracted to the same peak by the gradient search algorithm.

There is only one parameter in the peak-clustering algorithm and it is the size of the n-cube (SC) used in the gradient search for a peak. If the size of the n-cube is small then less smoothing is done and more peaks are found. As the size of the n-cube is increased, more smoothing is done and fewer peaks are found. The size is a function of the number of data samples (N), the number of features (NF) and the number of desired clusters (M). No direct relation has been found for SC so an iterative algorithm is used to adjust SC to obtain the number of desired clusters.

Step 1: Initialize the size of the n-cube

SC(i)= step*(Dmax(i)-Dmin(i)) ,

where step= 0.1 and the quantity (Dmax(i)-Dmin(i)) defines the range of the i^{th} feature. The variable, step, is increased by 0.01 if fewer clusters are desired and decreased by 0.01 if more clusters are desired.

Step 2: Randomly select any sample data point to start the peak-searching algorithm.

Step 3: Use a gradient search algorithm to find and label the peak if it is a new one. Find the sample data point that maximizes the minimal distance to all peaks that have been found. Continue this process until 50 consecutive searches fail to find a new peak.

Step 4: If the number of clusters found is too large then decrement

step= step – 0.01,

and go to Step 1. If on the hand the number of clusters found is too small then increase

step= step +0.01,

and go to Step 1. Finally, when the number of clusters is near your desired number, the process goes to Step 5.

Step 5: Label the data by the peak that each sample data is attracted to by the gradient search algorithm.

This process is relatively fast and produces nice clusters for a large class of problems. Data points located far from the other clusters form singleton clusters, defining the outliers or unusual situations often important to data mining problems. During the process of increasing the n-cube size to reduce the number of clusters to the desired number M, the number of clusters may suddenly fall to well below M. When this occurs, the process must be stopped before the number of clusters is less than M. If the user still wants to further reduce the number of clusters, the clusters are compared to find the closest pair and merge them into one cluster. This process allows the user to obtain the desired number of clusters.

3.3 Furthest Neighbor Clustering Model

The Furthest Neighbor Clustering Model (FNCM) [20] is an agglomerative hierarchical clustering method that merges together a large number of small clusters until it finds the desired number of clusters. The objective is to find the desired number of clusters each with the smallest diameter. The diameter of a cluster is the distance between the furthest two members of the cluster. The method is time consuming since a very large number of clusters are analyzed to find the clusters with the smallest diameter.

Step 1: Assign each data point to its own singleton cluster.

Step 2: Form a test cluster by combining individual clusters together.

Step 3: Find the diameter of the test cluster.

Step 4: Repeat Step 2 and Step 3 until every combination of two separate clusters has been tried.

Step 5: Permanently fuse the two clusters that formed the test cluster with the smallest diameter.

Step 6: Repeat Step 2 through Step 5 until the desired number of clusters is reached.

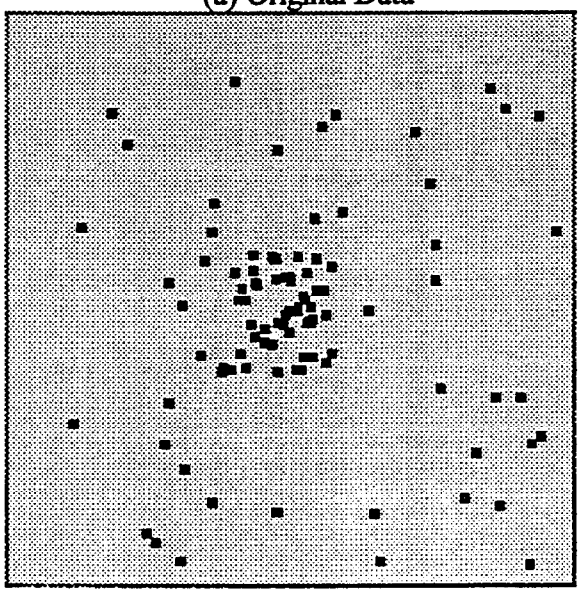
FNCM works well for clusters that are roughly the same size and spherically shaped. The method does not work well for string-like clusters or clusters of different shapes and sizes.

3.4 Clustering Examples

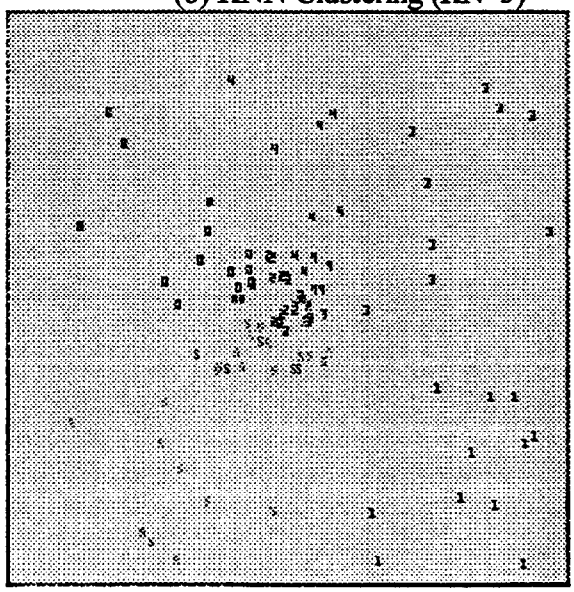
The KNN and PCM are both good clustering techniques that work on a large class of problems; however, all clustering techniques are data set dependent and one technique may work better than another in different situations. The examples given demonstrate the performance of these clustering algorithms on a variety of problems.

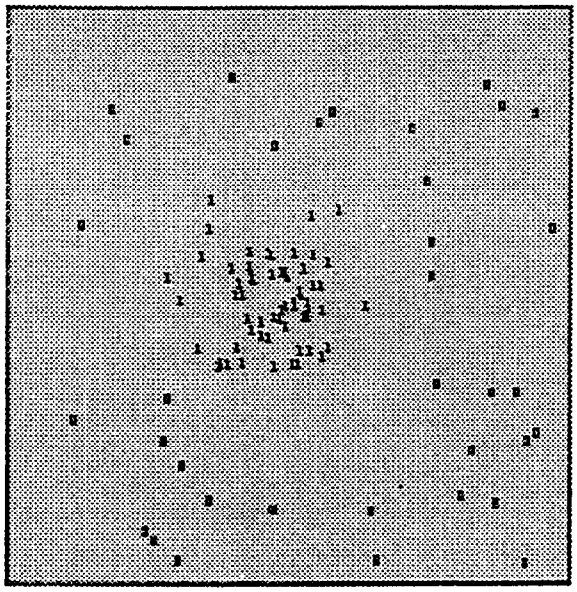
The first example demonstrates the how KN affects the KNN clustering algorithm. With KN =3 the clustering algorithm finds six clusters that are tightly bound. With KN=6 the clustering algorithm produces only two clusters that match the data produced and with KN=10 the data is clustered into only one cluster.

(a) Original Data

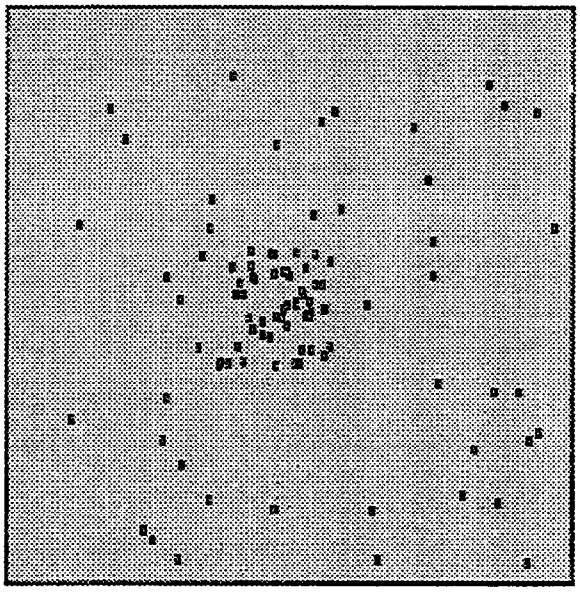


(b) KNN Clustering (KN=3)





(c) KNN Clustering (KN=6)

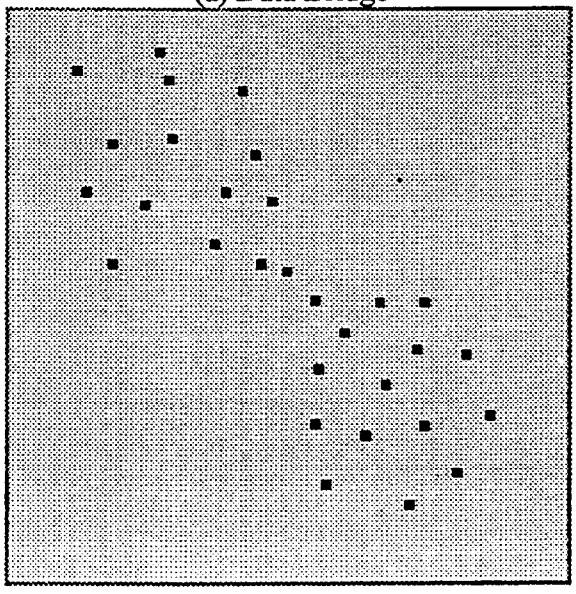


(d) KNN Clustering (KN=10)

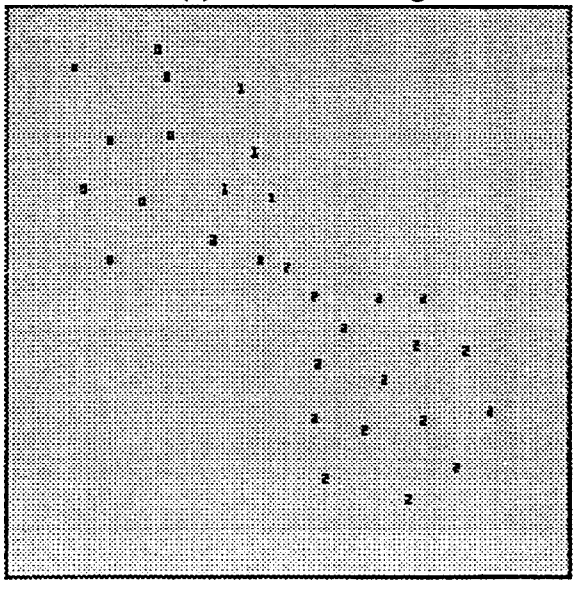
The KNN clustering algorithm presented requires the user to provide an estimate the number (M) of the desired number of clusters. The algorithm adjusts KN to get as near to M clusters as possible.

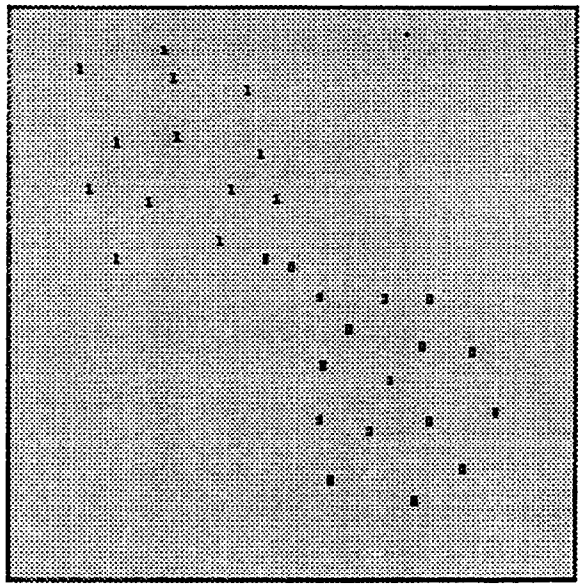
The next example uses the data bridge data set often used to test clustering algorithms to demonstrate the relative performance of the KNN clustering, the peak clustering and the furthest neighbor clustering algorithms. All the clustering algorithms perform well on this problem.

(a) Data Bridge

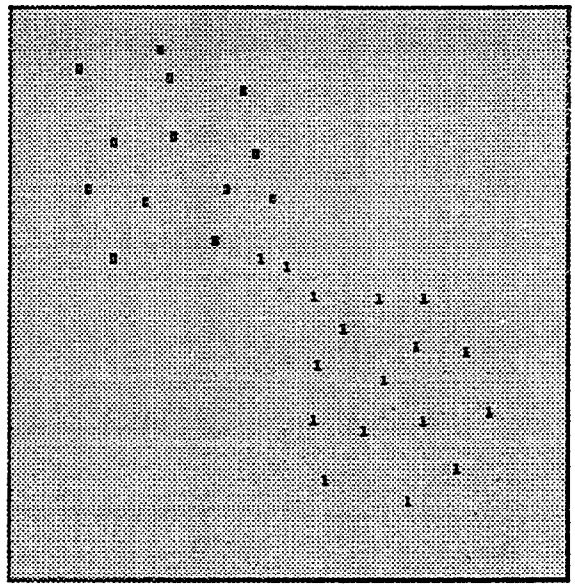


(b) KNN Clustering





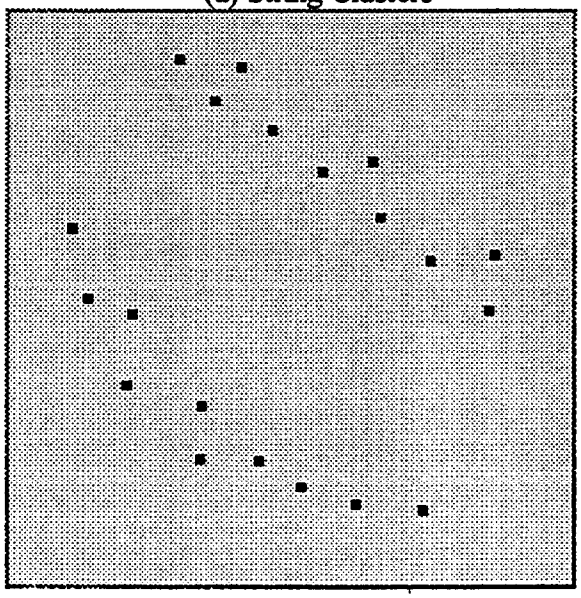
(c) Peak Clustering



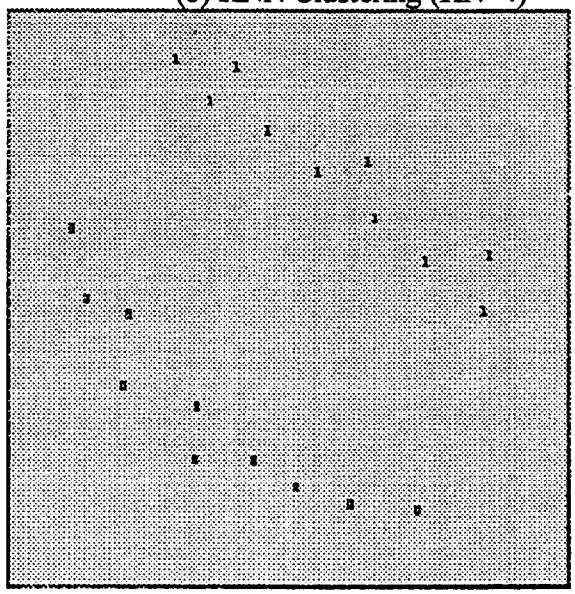
(d) Furthest Neighbor

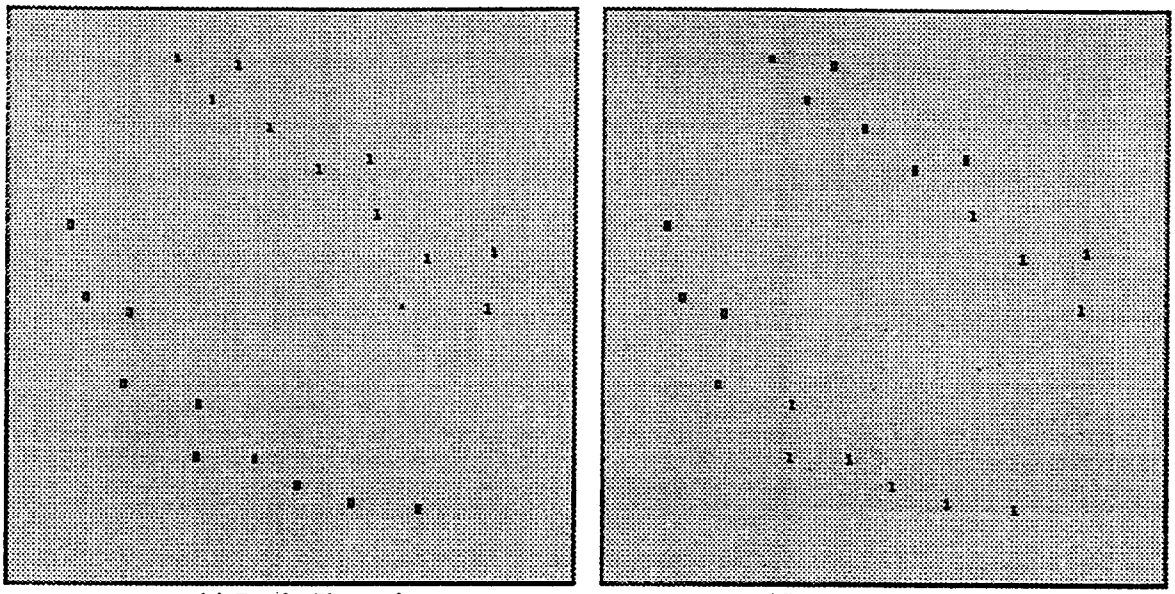
The next problem is the string clusters. Both the KNN and Peak clustering algorithms easily separate the strings where as the furthest neighbor algorithm tends to break up the strings.

(a) String Clusters



(b) KNN Clustering (KN=4)

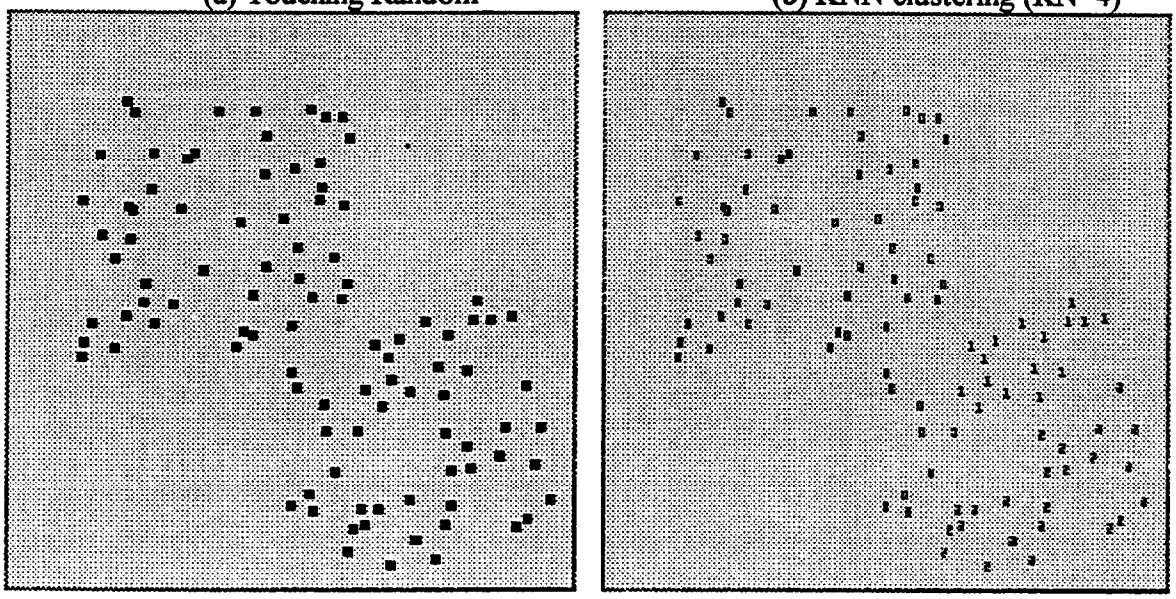


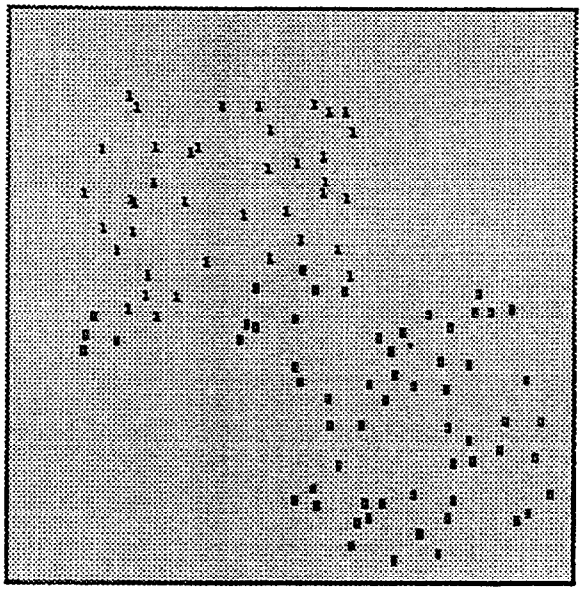


(c) Peak Clustering (d) Furthest Neighbor

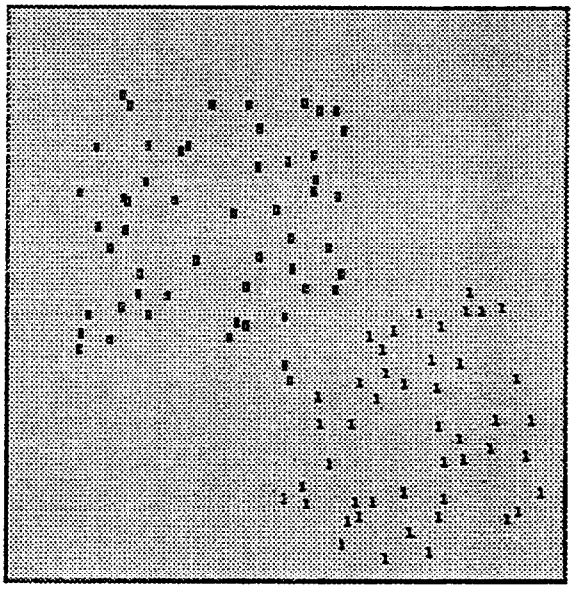
The next data set was generated by randomly generating two clusters with a small overlap. The performance of the algorithms is again compared. All algorithms perform well. Observe that the KNN clustering separates one cluster into two visible clusters

(a) Touching Random (b) KNN clustering (KN=4)





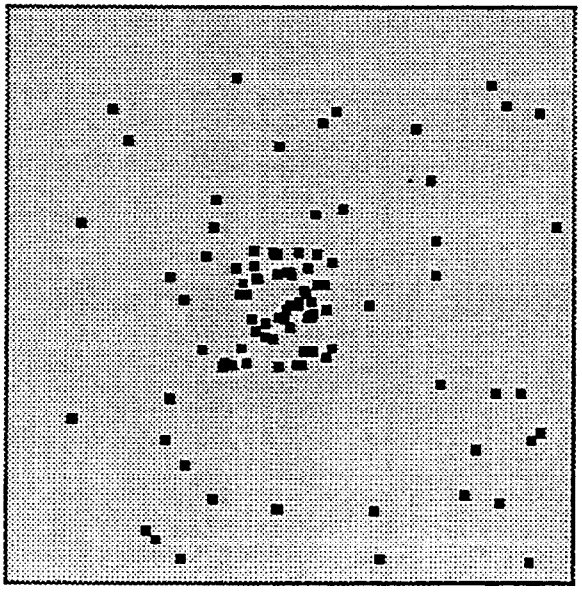
(c) Peak Clustering



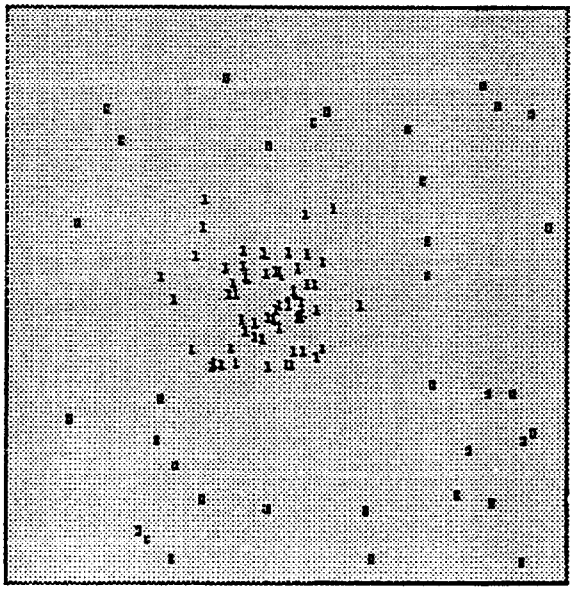
(d) Furthest Neighbor

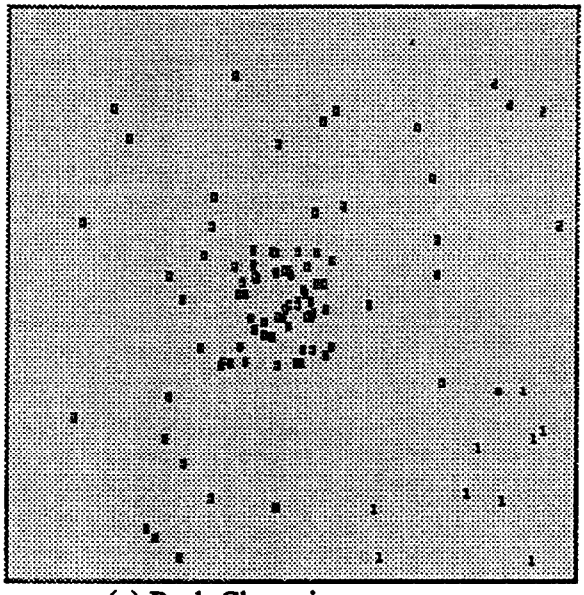
In the next example, the two random clusters are completely overlapped. The KNN algorithm still separates the clusters. The peak clustering and the furthest neighbor algorithms do not separate the overlapped clusters.

(a) Overlapped Random Clusters

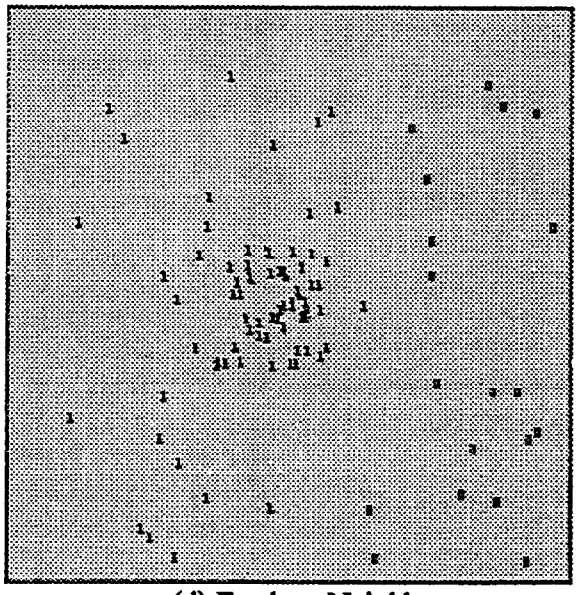


(b) KNN Algorithm (KN=6)





(c) Peak Clustering



(d) Furthest Neighbor

In summary, the KNN algorithm performs best in finding visible clusters and also isolates outlier points that are important to many data mining problems. The peak clustering algorithm works well with large data sets with Gaussian-like clusters. The furthest neighbor algorithm performs much like the peak clustering algorithm but is more computationally expensive.

4.0 HCV FEATURE EVALUATION

The HCV questionnaire has 185 questions related to the general background, medical history, blood exposure, needle exposure, alcohol use, demographics, insect bites, sexually transmitted diseases, and sexual habits of the people entering several hospitals in New Mexico. Many of the questions related to alcohol consumption and other questions allow free form answers that are very difficult to read and quantify. For these responses, algorithms were developed to quantify the answers into the number of beers consumed per week or the number of ounces of whisky consumed per week. In the future, these questions should be developed to require the response to be mapped into a set of quantitative options.

With a great amount of effort a program was developed to read the raw questionnaire database and form the quantitative feature database. Since many questions were related to the same behavior, the responses were mapped into a single composite feature to reduce the dimensionality of the problem. A program was developed to evaluate the features based on their linear correlation to whether or not a person has HCV. The linear correlation varies from +1 (directly correlated) to a -1 (inversely correlated). The program computes the false-alarm and miss-probabilities for each feature. The results are given where CORR column in the linear correlation, PFA is the false alarm probability, PMISS is the miss probability, PERR is the total error probability and the last column defines the feature. The list is ordered based on the linear correlation. The best features in terms of the linear correlation are on top of the list. Observe the correlation with the probability of error.

CORR	PFA	PMISS	PERR	Features
0.040	0.000	0.061	0.061	Whiskey Drinker
0.026	0.003	0.058	0.061	Drugs for sex
0.023	0.005	0.056	0.061	HCV partners
0.022	0.018	0.041	0.058	Heroin user
0.022	0.008	0,053	0.061	Methadone user
0.022	0.008	0.053	0.061	Tattoos done in prison
0.019	0.036	0.030	0.066	Drugs by injection
0.019	0.010	0.053	0.063	Shared needles
0.019	0.008	0.056	0.063	Had Syphilis
0.016	0.025	0.046	0.071	Smoked crack
0.015	0.008	0.058	0.066	Sexually transmitted diseases
0.014	0.066	0.028	0.094	Cocaine user
0.014	0.013	0.056	0.069	Paid for sex
0.012	0.010	0.058	0.069	Money for sex
0.012	0.038	0.046	0.084	Amphetamines user
0.010	0.030	0.051	0.081	Lived in prison
0.009	0.020	0.056	0.076	Pierced with shared needle
0.009	0.043	0.048	0.091	Hepatitis
0.008	0.030	0.053	0.084	Shared other drug equipment
0.008	0.048	0.048	0.096	Sexual relation with drug addict
0.008	0.048	0.048	0.096	Relative with HCV
0.008	0.041	0.051	0.091	Homeless

Feature Evaluation Continued

0.008	0.013	0.058	0.071	Sexual behavior
0.008	.183	0.018	0.201	Tattoos
0.007	0.015	0.058	0.074	Needle User
0.007	0.099	0.036	0.135	Tattoos done commercial
0.006	0.010	0.061	0.071	Bedbugs bites 100-1000
0.005	0.099	0.043	0.142	Accident with loss of blood
0.005	0.063	0.051	0.114	Medical care outside US
0.005	0.025	0.058	0.084	Had Gonorrhea
0.004	0.102	0.046	0.147	Blood brothers
0.004	0.058	0.053	0.112	Mosquito bites 100-1000
0.004	0.124	0.043	0.168	Blood before 1992
0.004	0.292	0.023	0.315	Hispanic
0.003	0.015	0.061	0.076	Live with someone with HCV
0.003	0.015	0.061	0.076	Had Venereal warts
0.003	0.124	0.043	0.168	Blood transfusion
0.003	0.086	0.051	0.137	Rheumatoid Arthritis
0.003	0.556	0.008	0.563	Income <20k
0.003	0.168	0.041	0.208	Anal sex
0.003	0.401	0.020	0.421	Ibuprofen use
0.002	0.119	0.048	0.168	Day care worker
0.002	0.213	0.038	0.251	Jet gun used
0.002	0.170	0.043	0.213	Blood transfusion
0.002	0.363	0.025	0.388	Sex during menstruation
0.002	0.398	0.023	0.421	Gender
0.002	0.041	0.058	0.099	Had Chlamydia
0.002	0.589	0.013	0.602	Oral sex
0.002	0.003	0.061	0.063	Number of tattoos
0.002	0.373	0.028	0.401	Sex with women
0.002	0.023	0.061	0.084	Sexual relation with HCV partner
0.002	0.175	0.046	0.221	Military service
0.002	0.203	0.043	0.246	Single
0.001	0.236	0.041	0.277	Live in country
0.001	0.099	0.053	0.152	Shaved outside US
0.001	0.025	0.061	0.086	Mosquito bites >1000
0.001	0.081	0.056	0.137	Diabetes
0.001	0.173	0.048	0.221	Drink well water
0.001	0.058	0.058	0.117	Sexual contact outside US
0.001	0.216	0.046	0.261	Lived outside US
0.001	0.317	0.038	0.355	Mosquito bites 10-100
0.001	0.061	0.058	0.119	Drink bottled water
0.000	0.718	0.013	0.731	Operation/surgery
0.000	0.561	0.023	0.584	Bedbugs bites <10

Feature Evaluation Continued

0.000	0.099	0.056	0.155	Traveled to Mediterranean
0.000	.033	0.061	0.094	Bedbugs bites 10-100
0.000	0.071	0.058	0.129	Shot to protect next pregnancy
0.000	0.036	0.061	0.096	Other race
0.000	0.000	0.063	0.063	Bedbugs bites >1000
0.000	0.000	0.063	0.063	Previously married
-0.000	0.264	0.046	0.310	Health care worker
-0.000	0.751	0.013	0.764	Teeth cleaned
-0.000	0.678	0.018	0.695	Currently in relationship
-0.000	0.005	0.063	0.069	Beer Drinker
-0.000	0.089	0.058	0.147	Cancer
-0.000	0.416	0.038	0.454	Dental surgery
-0.001	0.675	0.020	0.695	Married
-0.001	0.678	0.020	0.698	Drink city water
-0.001	0.543	0.030	0.574	Around animals
-0.001	0.096	0.058	0.155	Received Gamma Globulin
-0.001	0.472	0.036	0.508	Traveled outside US
-0.001	0.320	0.046	0.365	Aspirin use
-0.001	0.477	0.036	0.513	Tylenol use
-0.001	0.058	0.061	0.119	Black
-0.001	0.548	0.033	0.581	Pierced body part
-0.001	0.119	0.058	0.178	Acupuncture in US
-0.001	0.124	0.058	0.183	Acupuncture
-0.001	0.140	0.058	0.198	Anemia
-0.001	0.695	0.023	0.718	Live in town
-0.001	0.434	0.043	0.477	Have children
-0.002	0.513	0.038	0.551	Sex with men
-0.002	0.264	0.053	0.317	Pierced at commercial
-0.002	0.274	0.053	0.327	Traveled to Latin America
-0.002	0.102	0.061	0.162	Traveled to Asia
-0.002	0.107	0.061	0.168	Medications by injections
-0.002	0.254	0.056	0.310	Income 20-40k
-0.002	0.485	0.046	0.530	Mosquito bites <10
-0.003	0.003	0.063	0.066	Number/type of sexual partners
-0.003	0.008	0.063	0.071	Wine Drinker
-0.003	0.003	0.063	0.066	Had Non-gonococcal urethritis
-0.003	0.003	0.063	0.066	Other relationship
-0.003	0.003	0.063	0.066	Drink rain water
-0.003	0.005	0.063	0.069	Live other
-0.003	0.005	0.063	0.069	Drink other water
-0.003	0.008	0.063	0.071	Acupuncture outside US
-0.003	0.015	0.063	0.079	Drugs this week

Feature Evaluation Continued

-0.003	0.018	0.063	0.081	Drink river water
-0.003	0.018	0.063	0.081	Had Herpes
-0.003	0.033	0.063	0.096	Traveled to Africa
-0.003	0.048	0.063	0.112	Income >70k
-0.003	0.056	0.063	0.119	Long term relationship
-0.003	0.069	0.063	0.132	Steroids use
-0.003	0.079	0.063	0.142	Income 40-70k
-0.003	0.094	0.063	0.157	Acupuncture in NM
-0.003	0.551	0.046	0.596	White

Using these features and their ranking, the WNN fusion process selected the following features to minimize the pe. Many attempts were made to establish new features to further decrease the probability of error. When the new features are added, the new feature is given zero weight by the GA optimization algorithm, which means the new feature does not lower the pe. The variable ans[i] represents the response to the i^{th} feature in the database. These features are taken directly out of the C program that generates them.

Feature 0 <=> Gender

```
f[0]=ans[5];                    // 1=> male and 0 =>female
```

Feature 1 <=> Needle and drug users

```
f[1]  =ans[74]*105;             //drugs by injection
f[1] +=ans[77]*95;              //heroin user
f[1] +=ans[81]*75;              //shared needles
f[1] +=ans[82]*50;              //share drug equip
f[1] +=ans[83]*35;              //smoked crack
f[1] +=ans[80]*20;              //methadone
f[1] +=ans[78]*10;              //amphetamines
f[1] +=ans[79]*10;              //cocaine
```

Feature 2 <=> Blood transfusion

```
f[2]  =ans[53]*100;             //before 1992
f[2] +=ans[52]*20;              //after 1992
```

Feature 3 <=> Sexual behavior

```
f[3]  = ans[172]*105;      //sex with HCV
f[3] += ans[173]*95;       //sex with drug user
f[3] += ans[108]*75;       //drugs for sex
f[3] += ans[107]*50;       //money for sex
f[3] += ans[129]*35;       //sex outside US
f[3] += ans[177]*20;       //anal sex
f[3] += ans[175]*10;       //sex during menstruation
```

Feature 4 <=> Social status and HCV friends

```
f[4]  = ans[147]*105;      //lived with HCV person
f[4] += ans[145]*95;       //relative with HCV
f[4] += ans[150]*75;       //lived in prison
f[4] += ans[149]*50;       //been homeless
```

The weights associated with the composite features are chosen to help separate the clusters and place emphasis on the more important questions. During the feature selection process many other features were selected but were given weight of zero by the GA weight optimizing process, which effectively removes them from the decision process. During the early stages of the feature selection process, the alcohol consumption features performed well but as the needle and drug user features were added the weight of the alcohol consumption features went to zero, implying their HCV detection ability is contained in the other features.

4.0 HCV FUSION RESULTS

The WNN fusion process was performed on the HCV feature database, resulting in a total probability of error PE = 0.023. The false alarm probability is PFA = 0.011 and the probability of miss is PMISS = 0.012. Five weighted-features and seventeen decision points were generated by the GA. The five weighted features are given below with their associated weights.

Feature	Weights
f[0] <=> Gender	1.00
f[1] <=> Needle and drug users	2.00
f[2] <=> Blood Transfusions	1.00
f[3] <=> Sexual behavior	0.50
f[4] <=> Social status and HCV friends	5.00

The higher weights imply the features play a more important role in the decision process. Hence, the social status and whether a person has HCV friends play a dominant role in the decision process. The second most important feature is needle and drug users. Followed by blood transfusion before 1992 and gender. The gender feature is surprising until you consider the fact that the features helps select the decision points and often different factors place male and female samples in risk of HCV.

Seventeen decision points are selected to implement the decision logic. Ten of the decision points are used to classify the samples not a risk of HCV (class=0) and seven are used to classify samples at risk of HCV. If an unknown sample is closest to a decision point of class=0, then the sample is considered not to be at risk of HCV. However, if an unknown sample is closer to a class=1 decision point, then the person is at risk of HCV. The decision points are given below.

f[0]	f[1]	f[2]	f[3]	f[4]	Class
0.18	0.00	0.00	171.6	128.0	0.00
0.49	-60.80	96.00	109.40	0.00	0.00
0.30	-60.80	151.20	-72.60	122.00	0.00
0.33	345.80	-3.60	-6.60	62.00	0.00
0.77	247.00	36.00	296.40	64.00	0.00
0.52	243.20	138.00	148.60	-32.00	1.00
1.60	0.00	0.00	132.00	0.00	0.00
0.33	-64.60	36.00	-66.00	120.00	0.00
-0.32	185.80	-13.20	235.20	48.00	1.00
0.32	-60.80	115.20	150.80	128.00	0.00
0.66	178.60	106.80	-130.80	63.00	1.00
1.30	285.80	18.00	54.60	-64.00	1.00
0.56	140.60	2.40	-71.60	187.00	1.00
1.45	0.00	0.00	0.00	50.00	0.00
0.15	-114.00	0.00	-99.00	0.00	0.00
1.30	573.00	18.00	272.00	125.00	1.00
1.00	402.00	0.00	140.00	215.00	1.00

Under the best no risk condition of each feature, their value would be zero and their value increases as their probable risk increases. The decision points with class=0 classify the people not at risk of HCV and the decision points with the class=1 classifies people at risk of HCV. Since the first feature is gender (0=>female and 1=>male), observe the different feature values of the other features of the decision are quite different for male and females. The estimated total probability of error is pe = 0.023 for the WNN fusion logic which is likely to be near the MPE given the uncertainties caused by recording errors and evasive tactics of people answering the questionnaire. The GA process of selecting and weighting the features and establishing the decision points is time consuming but the time required to perform the decision logic is very fast. An unknown sample is simple compared to each decision point and given the class of the closest.

5.0 CONCLUSIONS

The results support the conclusion that the WNN fusion model can be used to discover and fuse features from large databases to establish near-MPE decision and estimation models. The performance of the individual features is established in terms of their detection and false alarm probabilities. Individually, the features for the HCV database have relatively low detection probabilities. However, when fused together in a higher dimensional space their combined performance is much improved. The basic idea is that a few features can be easily confused but it is difficult to simultaneously confuse a large number of features. This concept is illustrated by fusing the HCV database into six features using the WNN fusion model to establish a near-MPE decision logic to decide whether a person has high risk of having HCV. Several clustering algorithms are developed to find clusters and isolate outlier samples. The KNN clustering and the peak clustering algorithms perform well on many sample problems.

6. REFERENCES

1. Scott, D. R., *The k-nearest neighbor statistic with applications to electronic vision systems*, Ph.D. Dissertation, New Mexico State University, Dec., 1990.
2. Cover T. and P. Hart, *Nearest Neighbor Pattern Classification*, IEEE Trans. on Information Theory, 1963.
3. McClave and Scheaffer, *Probability and Statistics for Engineers*, Duxbury Press, 1995.
4. Beer, Cynthia, *The Tie Statistic and Texture Recognition*, Ph.D. Dissertation, New Mexico State University, Dec., 1989.
5. Flachs, G. M., Qiang Meng, Xiaohua Niu, Wei Wang and Zhonghao Bao, Final Report for Entry Control Project for the Contract with Sandia National Laboratories, New Mexico State University, 1994.
6. Carlson, J, J. Jordan, G. Flachs, Q. Meng, X. Niu, W. Wang, Z. Bao, D. Marten,"High Confidence Identity Verification Using Multiple, Coarse Features Acquired from the Face, Hand and Voice," Proceedings of the 11th Annual Security Technology Symposium, Virginia Beach, VA, June 19-22, pp. 249-256, 1995.
7. Nguyen H. T. and G. S. Rogers, *Fundamentals of Mathematical Statistics*, Vol. II Statistical Inference, New York: Springer-Verlag, 1989.
8. Mood, A. M., Graybill, F. A., and D. C. Boes, *Introduction to the Theory of Statistics*, New york: McGraw-Hill, 1974.
9. Papoulis, A., *Probability, Random Variables and Stochastic Processes*, New York: McGraw-Hill, 1984.
10. Hogg, R, V. and A. T. Craig, *Introduction to Mathematical Statistics*, New York: Macmillan, 1978.
11. Carlson, Jeffrey J., *Decision-Making Complexity with Applications in Electronic Vision*, Ph.D. Dissertation, New Mexico State University, Sept. 1988.
12. Beer, Cynthia, *The Tie Statistic and Texture Recognition*, Ph.D. Dissertation, New Mexico State University, Dec., 1989.
13. Daubechies, I., "Orthonormal Bases of Compactly Supported Wavelets," *J. Math. Phys.*, **31**:1898-1900, 1990.
14. Flachs, G. M., J. B. Jordan, C. L. Beer, and D. R. Scott, "Feature Space Mapping for Sensor Fusion," *Journal of Robotic Systems*, Vol. 7, No. 3, pp. 373-393, June, 1990.
15. Choe, H., *A comparative analysis of statistical, fuzzy, and artificial neural pattern recognition techniques*, Ph.D. dissertation, New Mexico State University, Dec., 1992.
16. Jordan, J. B. and H. Choe, "A comparative analysis of statistical, fuzzy, and artificial neural pattern recognition techniques," *Proceedings of SPIE Signal Processing, Sensor Fusion, and Target Recognition*, vol. 1699, pp. 166-176.
17. Johnson, N. I. and S. Kotz, *Continuous Univariate Distributions-1*, Boston: Houghton Mifflin, 1970.
18. Jordan J. B., G. M. Flachs, and Z. Bao. "Feature Based Methodology for Sensor Fusion," Final Report to Sandia National Laboratory ,November 26,1996.
19. Belur V. Dasarathy, *NN Pattern Classification Techniques*, IEEE Computer Society Press, Chapter 8, 1991.
20. Duda R. and Peter Hart, *Pattern Classification and Scene Analysis*, Wiley & Sons, 1973.

7.0 DISTRIBUTION LIST

10	MS	1003	Jeffrey J. Carlson, 15211
1		1003	Rush D. Robinett III, 15211
1		0188	LDRD Office, 4001
5		1010	Sharon L. Blauwkamp, 15222
1		9018	Central Technical Files, 8945-1
2		0899	Technical Library, 9616
1		0612	Review and Approval Desk, 9612 For DOE/OSTI
1		0161	Patent and Licensing Office, 11500