A comparison among accuracy of KNN, PNN, KNCN, DANN and NFL

Jamshid Tamouk[1], Farshid Allahakbari[2]

[1]Computer Engineering Department, E.M.U University

[2]Electrical and Electronic Department, E.M.U University

***Abstract*— Accuracy of classifications methods from Satellite and iris imagery dataset are very important for eco-environment monitoring and iris type classification sequentially. It is tried to implement classifying methods on some given data files and in this article the results and accuracies among KNN, PNN, KNCN, DANN and NFL methods are compared.The results respectively show that the overall accuracies of each data are approximately %87.70 , %84.80 , %93.88, %80.70 and %80.95 .It is indicating that the KNCN and KNN classifiers have greatly better accuracies than the other mentioned methods which leads to conclusion that KNCN is the best among these five classification methods with 96.67% for iris data set and 91.09% for satellite image dataset.**

Keywords— *Classification; KNN; PNN; KNCN; DANN; NFL*

I. INTRODUCTION

In this project, it is tried to implement classifying methods on some given data files and compare the results together in different cases with different parameters. First of all, we have a pre-processing step, data normalizing and then the chosen methods are K-Nearest Neighbor (KNN), Probabilistic Neural Network (PNN), K-Nearest Centroid Neighbor (KNCN), Discriminant Adaptive Nearest Neighbor Classification (DANN) and Nearest Feature Line (NFL) known as methods for classifying data.

KNN definition: In pattern recognition, the k-nearest neighbor algorithm (kNN) is a method for classifying objects based on closest training samples in the feature space. k-nearest neighbor method is one of the simplest machine learning algorithms which an object is classified by a majority vote of its neighbors then the object will assign to the class with most frequent nearest sample among k nearest neighbors (k is a positive integer, typically small). If k = 1, then the object is simply assigned to the class of its nearest neighbor which is called nearest neighbor (NN).

PNN definition: Probabilistic neural network (PNN) can compute nonlinear decision boundaries by replacing the sigmoid activation function used in neural networks with an exponential function and an activation functions .

KNCN definition: K-Nearest Centroid Neighbor algorithm or more simply KNCN concept is based on the idea that the neighborhood of a test sample is subject to two constraints. First, by the distance criterion, the K neighbors of a test sample must be as near as possible. Second, by the symmetry criterion, their centroid must be also as close to test sample as possible.

DANN definition: Discriminant Adaptive Nearest Neighbor starts to classify based on Linear Discriminant Analysis (LDA) to estimate an effective metric for computing neighborhoods. K-Nearest Neighbor also will be used for completion.

NFL definition: The nearest feature line (NFL) method extends the classification capability of the nearest neighbor (NN) method by taking advantages of multiple (more than one) templates per class. It effectively improves the classification performance especially when the number of templates per class is small. This problem is frequently encountered in many applications such as face recognition problems.

II. DATA AND METHODS

K-Nearest neighbors (KNN) is an approach that classifies data with respect to number of nearest distance to system training data KNN. The steps of KNN method are shown below:

Step 1: Training phase storing vectors with keeping their class labels.

Step 2: Determining K parameter (by user).

Step 3: Calculating given point (testing point) distance to all training points and storing the result. Distance can be calculated from different metrics like Euclidean, Minkowski, Manhattan and etc. Here, we have used Euclidean metric: Dist($\bar{x}$, $\bar{y}$)=($\sum_{i=1}^{d}|x_i - y_i|^2$)$^{1/2}$4.

Step 4: Tested point will assign to the class with most frequent nearest sample among k nearest neighbors. Testing point will be belonged to a class which have the most frequency of nearest distance among K lowest distances. For the simplest case (K=1), testing point belongs to class that have lowest distance to its points which is called nearest neighbor. i.e. in figure 1, k=5 and the class which has more nearest neighbors is class 2. Then given point belongs to class 2.

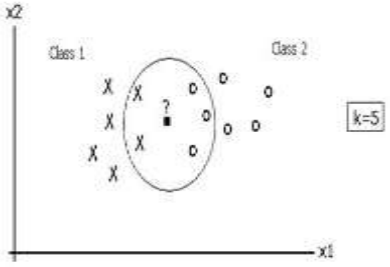

Fig. 1 KNN Classification (K=5)

PNN Method: Probabilistic Neural Network is a mapping from a pattern vector to a class. It is made up of 3 layers. Layer 1, input layer contains feature values and numbers of its elements are equal to

number of the vector dimensions. Layer 2, hidden layer contains weights which are function of training data. That's why the number of training data and these layer elements are the same. Each node in hidden layer corresponds to a Gaussian function ($g_c=e^{(\overline{W}_n^T \overline{X}-1)/\delta^2}$) that is associated with feature vector in the same class in layer 3, output layer. Greatest summary of values input to layer 3 is the result. δ is design parameter and will be tuned to give the best result. The steps of PNN method are shown below:

Step 1: Training phase, calculating weight of each node in hidden layer with respect to weight function and training vector placed in Input layer (In this case, $\overline{W}_i = \overline{X}_i$)
Step 2: Placing each testing vector in input layer to calculate $g_c= e^{(\overline{W}_n^T \overline{X}-1)/\delta^2}$ with respect to transposed weight vector and given testing vector.
Step 3: Summing all output function value (g_c) from hidden layer nodes for each class.
Step 4: For our application result will be the one with maximum value (probability).

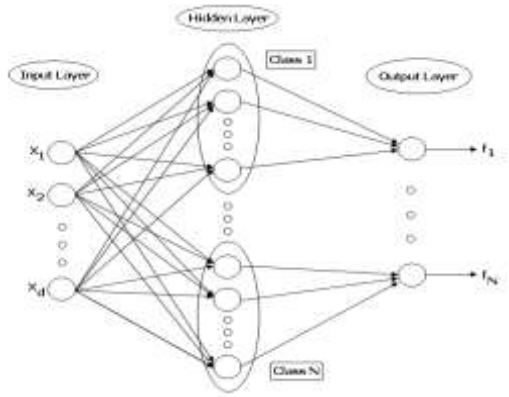

Fig. 2 PNN Classification

KNCN Method: The steps of K-Nearest Centroid Neighbor algorithm are shown below:
Step 1: Find the k nearest centroid neighbours of q.
Step 2: Assign to q the class with a majority of votes among its k nearest centroid neighbours.

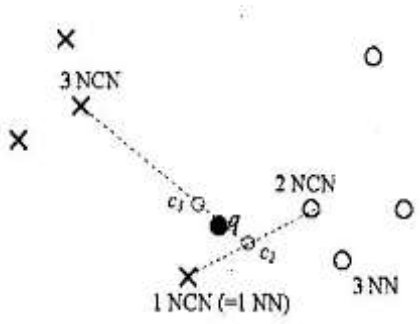

Fig. 3 3-NCN classification (Grabowski,2004)

DANN Method: Discriminant Adaptive Nearest Neighbor Classification is a method that is using LDA and KNN for classification. The steps of DANN method are shown below:

Step 1: Initialize the metric $\sum$ = I, the identity matrix.

Step 2: Spread out a nearest neighborhood of KM points around the test point xo, in the metric $\sum$.

Step 3: Calculate the weighted within and between sum of squares matrices W and B using the points in the neighborhood.

Step 4: Define a new metric $\sum$ = W-1/2[W-1/2BW-1/2 + εI]W-1/2

Step 5: Iterate steps 1, 2, and 3.

Step 6: At completion, use the metric $\sum$ for k-nearest neighbor classification at the test point xo. (Hastie el.al.,1996).

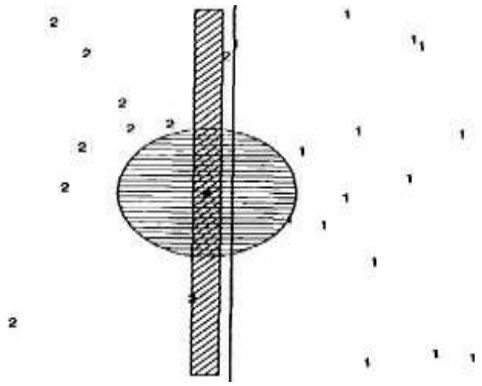

Fig. 4 The vertical strip denotes the NN region using(Hastie el.al.,1996)

NFL Method: The nearest feature line (NFL) finds the distance between given point (testing point) and the line passes from each 2 training points of the same class is the basis of classifying in this method. The steps of NFL method are:

Step 1: Training phase storing vectors with keeping their class labels.

Step 2: Calculate the distance (perpendicular) between the given point and the line passes from each 2 training points of the same class. Euclidean distance between point P(xp,yp) and line l passes from points (x0,y0) , (x1,y1) in 2-dimentional space obtains as:

$$\text{Dist(P,l)}=\sqrt{\left(x_p - x_0 - \lambda(x_1 - x_0)\right)^2 + \left(y_p - y_0 - \lambda(y_1 - y_0)\right)^2}$$

$$\lambda = \frac{(x_1 - x_0)(x_p - x_0) + (y_1 - y_0)(y_p - y_0)}{(x_1 - x_0)^2 + (y_1 - y_0)^2}$$

And it can be developed for higher dimensionality.

Step 3: The class which has the minimal line distance to the point is the result.

All processes of this method are shown in figure 5.

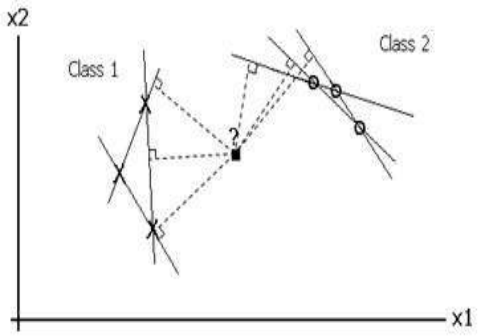

Fig. 5 NFL Classification

III. PROBLEM DEFINITION

Before starting the process, researchers need to know more about the feature of methods which depends on their performance that they can choose appropriate methods for classıfying, in fact they should be able to choose the best method for their usage, so in this paper, it's attempted to deal with five methods and consider differences among them, indeed this paper can be effective for finding the most accurate method among these classification methods.

IV. RESULTS AND DISCUSION

The accuracies obtained for each datasets with KNN method and their average are as following table:

TABLE I
KNN BEST ACCURACY RESULTS

Data-file name	Best accuracy
Iris	%98.000
Satimage	%77.404

The accuracies obtained for each data file with PNN method and their average are as following table:

TABLE III
PNN BEST ACCURACY RESULTS

Data-file name	Accuracy of first-fold	Accuracy of second fold	Accuracy of third-fold	Accuracy of forth-fold	Accuracy of fifth-fold	Total accuracy (Average)
Iris	%100	%100	%93.3	%93.3	%100	%97.3
Satimage	%69.3	%71.4	%71.4	%67.7	%81.4	%72.2

The best accuracies obtained for each data file with KNCN method are shown in following table:

TABLE III
KNCN BEST ACCURACY RESULTS

Data-file name	Best accuracy
Iris	%96.673
Satimage	%91.091

The best accuracies obtained for each data file with DANN method are shown in following table:

TABLE IIIV
DANN BEST ACCURACY RESULTS

Data-file name	Best accuracy
Iris	%91.064
Satimage	%91.091

The accuracies obtained for each data file with NFL method and their average are as following table:

TABLE VIV
NFL BEST ACCURACY RESULTS

Data-file name	Accuracy of first-fold	Accuracy of second-fold	Accuracy of third-fold	Accuracy of forth-fold	Accuracy of fifth-fold	Total accuracy (Average)
Iris	%90.0	%90.0	%96.6	%86.6	%96.6	%92.0
Satimage	%75.6	%73.5	%70.8	%61.3	%67.9	%69.8

The following, summarized results of different methods for different data files are shown in table VII.

TABLE VV I
TOTAL ACCURACY RESULTS

Data-file name	Total accuracy of KNN	Total accuracy of PNN	Total accuracy of KNCN	Total accuracy of DANN	Total accuracy of NFL	Best method
Iris	%98.000	%97.333	%96.673	%91.064	%92.000	KNN
Satimage	%77.400	%72.275	%91.091	%70.346	%69.892	KNCN
Total Average	%87.700	%84.804	%93.882	%80.705	%80,946	

As can be seen, the method KNCN gives the best accuracy among the other methods for different data files. The next best method with respect to accuracies is KNN. All data samples are used interchangeably in KNN, PNN, KNCN, DANN and NFL approaches as both training and testing data. In each execution of algorithms (with different training and testing data) an accuracy value is obtained and gives the percentage of truly classified data.

V. CONCLUSIONS

In conclusion the majority of researchers have worked on satellite and identification images in these days, so searching about eco-environment monitoring and identifying application can be a significant issue. However in this project it is tried to implement classifying methods for some given data files and this article compares the results and accuracies among KNN, PNN, KNCN, DANN and NFL methods. Eventually the results of classification method on datasets (i.e."iris" and "satimage") show that KNCN method had the best efficiency among other mentioned methods.

REFERENCES

[1] S. M. Metev and V. P. Veiko, *Laser Assisted Microtechnology*, 2nd ed., R. M. Osgood, Jr., Ed. Berlin, Germany: Springer-Verlag, 1998.
[2] J. Breckling, Ed., *The Analysis of Directional Time Series: Applications to Wind Speed and Direction*, ser. Lecture Notes in Statistics. Berlin, Germany: Springer, 1989, vol. 61.
[3] S. Zhang, C. Zhu, J. K. O. Sin, and P. K. T. Mok, "A novel ultrathin elevated channel low-temperature poly-Si TFT," *IEEE Electron Device Lett.*, vol. 20, pp. 569–571, Nov. 1999.
[4] M. Wegmuller, J. P. von der Weid, P. Oberson, and N. Gisin, "High resolution fiber distributed measurements with coherent OFDR," in *Proc. ECOC'00*, 2000, paper 11.3.4, p. 109.
[5] R. E. Sorace, V. S. Reinhardt, and S. A. Vaughn, "High-speed digital-to-RF converter," U.S. Patent 5 668 842, Sept. 16, 1997.
[6] (2002) The IEEE website. [Online]. Available: http://www.ieee.org/
[7] M. Shell. (2002) IEEEtran homepage on CTAN. [Online]. Available: http://www.ctan.org/tex-archive/macros/latex/contrib/supported/IEEEtran/
[8] *FLEXChip Signal Processor (MC68175/D)*, Motorola, 1996.
[9] "PDCA12-70 data sheet," Opto Speed SA, Mezzovico, Switzerland.
[10] A. Karnik, "Performance of TCP congestion control with rate feedback: TCP/ABR and rate adaptive TCP/IP," M. Eng. thesis, Indian Institute of Science, Bangalore, India, Jan. 1999.
[11] J. Padhye, V. Firoiu, and D. Towsley, "A stochastic model of TCP Reno congestion avoidance and control," Univ. of Massachusetts, Amherst, MA, CMPSCI Tech. Rep. 99-02, 1999.
[12] *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification*, IEEE Std. 802.11, 1997.