PUNE INSTITUTE OF COMPUTER TECHNOLOGY
DHANKAWADI, PUNE-43.

A

SEMINAR REPORT

ON

SUPERVISED CLUSTERING

SUBMITTED BY

Name:Akshay Deshmukh

Roll No: 3417

Class: TE-4

GUIDED BY

Prof. D.T. Mane

COMPUTER ENGINEERING DEPARTMENT

DEPARTMENT OF COMPUTER ENGINEERING

Pune Institute of Computer Technology

Dhankawadi, Pune

Maharashtra 411043

CERTIFICATE

This is to certify that Mr.Akshay Deshmukh, Roll No.3417 a student of T.E. (Computer Engineering Department) Batch 2017-2018, has satisfactorily completed a seminar report on “SUPERVISED CLUSTERING” under the guidance of Prof. D.T. Mane towards the partial fulfillment of the third year Computer Engineering Semester II of Pune University.

Prof. D.T. Mane
Internal Guide
PICT, Pune

Dr. R. B. Ingle
Head
Department of Computer Engineering
PICT, Pune

Place: PICT, Pune
Date:

ACKNOWLEDGEMENT

We sincerely thank our Seminar Coordinator Prof. A.A.Jewalikar and Head of Department Dr. R. B. Ingle for their support.

We also sincerely convey our gratitude to our guide Prof. D.T. Mane, Department of Computer Engineering for his constant support, providing all the help, motivation and encouragement from beginning till end to make this seminar a grand success.

Above all we would like to thank our parents for their wonderful support and blessings, without which we would not have been able to accomplish our goal.

Contents

List of Tables

List of Figures

1 ABSTRACT

Clustering is basically applied in an unsupervised learning domain using particular error functions or module, for ex. an error module that reduces the distances inside a cluster which keeps clusters inbound. Supervised clustering,is a technique which deviates from traditional clustering ,in that it is applied on classified examples with the target of identifying clusters that have high probability function with respect to a single class. In addition, in supervised clustering, we also like to keep the number of clusters minimum, and objects are assigned to clusters using a method of closeness with respect to a given distance function. Supervised clustering evaluates a clustering based on the Number of clusters, k. In general, we like to keep the number of clusters low.My seminar mainly focuses on studying various supervised clustering algorithms using neural networks which can used for Pattern classification/Clustering.

Keywords: Clustering,Pattern Classification,Supervised Learning,Distance Function,Fuzzy Hypershpere,Membership Function.

2 INTRODUCTION

Machine learning is a field of Computer Science. It allows computers to build analytical models of data and find hidden results without any hard coding. It has been applied to a variety of aspects in modern society, ranging from DNA sequences classification, loan defaulter prediction, robot locomotion to natural language processing.

It was initially originated from pattern recognition. Earlier works of the same topic used models including logistic regression, genetic algorithm and inductive learning.

Logistic regression is a statistical model allowing scientists to build predictive prototypes based on a sample. This model is best used for understanding how several independent variables influence a single outcome variable. Though it is useful in some ways it is also limited. Genetic algorithm is based on natural selection and evolution. It can be used to extract rules in propositional and first-order-logic and also to choose the correct sets. Inductive learning's main category is decision tree algorithm.

There are many methods such as logistic regression, linear regression, etc. But in some applications these methods give less accuracy. Hence a new method was evolved called clustering. In clustering, data points are distributed in 'n' dimensional space. According to the similarities between these data points, clusters are formed with respective centroids and raddi. This technique gives good results in various predictive models. This type of clustering is basically under unsupervised. Unsupervised means the data points do not have label associated with them. Thus, they are classified according to the similarities between the data points.

The more efficient type of clustering is supervised clustering. In supervised clustering, the data points are given certain labels. According to the labels, the data is classified into clusters and radii. To see if the accuracy can increase with the supervised clustering, we used it on Iris, Liver, Pima datasets to calculate the accuracy.

3 MOTIVATION

Basically, clustering is associated with Unsupervised. In Unsupervised learning the data points do not have labels. They are clustered according to similarities in between them. Many people use clustering with unsupervised data. The disadvantage of this type of clustering is that the data points are classified only on the basis of their relative position in n-dimensional space. Hence, if a data point of one class is clustered with a data point of another class only on their relative positions in space, then the clustering is partially correct. This can lead to wrong or false cluster formation. Hence, one more technique called supervised clustering can overcome this disadvantage.

In Supervised Clustering, the data points are given certain labels associated with them. Based on this, a group or cluster is formed. This cluster assures that data points or samples having same labels are grouped together within a cluster. This is an advantage of supervised clustering. It helps to organize the correct clusters based on the labels rather than similarities between the n-dimensional points. The supervised clustering is used in many fields such as marketing, medicals, etc. We aim to study the various supervised learning algorithms[paper reference no]. In addition to this, how supervised clustering can be used with fuzzy hypersphere is implemented on various data sets such as Pima, Liver and Iris dataset. The metrics used is accuracy. The supervised clustering with fuzzy hypersphere gives better results than traditional clustering.

4 SUPERVISED CLUSTERING

4.1 What is Clustering?

Clustering is basically a task in which the data points are divided into set of groups such that data points in same groups belong to the same family or have same characteristics than those in other groups. In an abstract way, the target is to segregate groups with same properties and assign them into clusters or groups. A loose definition of clustering could be the ***"process of organizing objects into groups whose members are similar in some way."*** A cluster is therefore a collection of objects which are "similar" between them and are "dissimilar" to the objects belonging to other classes.

4.2 Types of Clustering

There are three types of clustering methods which are as follows : supervised, semi-supervised and unsupervised clustering.

- **Unsupervised Clustering:** It is a technique using some functions, for example, that minimizes the distances in cluster to keep cluster in-tight.
- **Supervised Clustering:** It is a technique which is applied on classified examples with the objective of identifying cluster's that have high probability density with respect to single class.
- **Semi-Supervised Clustering:** It is applied to enhance an algorithm by using side information in clustering process.

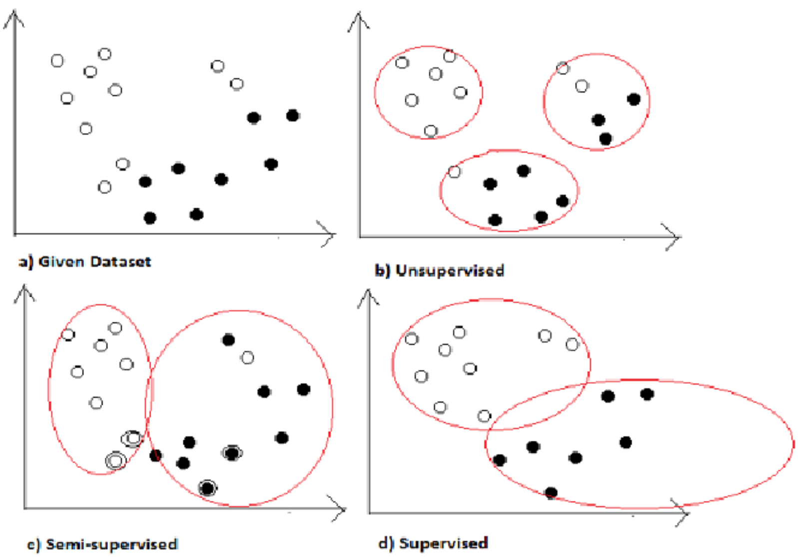

Figure 1: Types of Clustering[1]

4.3 Goals of Clustering

The goal of clustering is to determine the important grouping in a set of unlabelled data. The challenge is to decide what constitutes a good clustering. It is job of user which must supply this criterion in such a way that the result of clustering will suit their needs.

4.4 Supervised Clustering

Supervised Clustering is a type of Clustering. It assumes that the examples are classified and has a goal of determining clusters that have high probability function. Clustering is basically unsupervised to find objects belonging to same group. As we all know, now three types of clustering Traditional, Supervised and Semi-Supervised. Let us understand the Figure given below which aims on finding close or inbound clusters and would produce clustering with four clusters namely A,B,C,D that are produced with traditional clustering. Semi-supervised focuses not only on obtaining clusters, but also it focuses on satisfying constraints with respect to a small set of predefined modules or objects. Patterns belonging to different classes should belong to different clusters while of same class should belong to same cluster.As a result, a semi-supervised clustering algorithm generates clusters E, F, G, and H. Finally, a supervised clustering algorithm which uses a fitness function which maximizes the purity of the clusters while keeping the number of clusters low would produce clusters I, J, K, and L, as depicted in Figure 2.

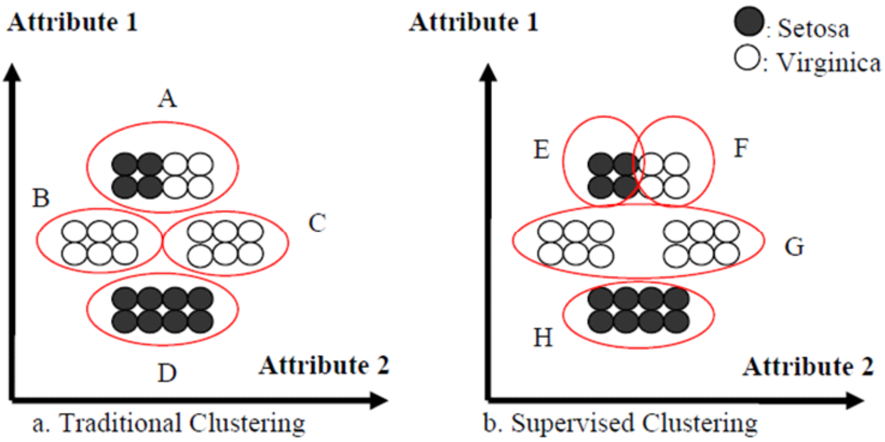

Figure 2: Traditional vs. Supervised Clustering [1]

4.5 Importance of supervised clustering

- Can be used for creating useful background to a dataset.
- Dataset Normalization and compression.
- To form clusters within clusters and to use these clusters to enhance classification.
- To calculate distance functions in distance function learning.

4.5.1 Creating Background of a Dataset

- It shows how patterns of one class distribute in the attribute or feature space, this information which will be visualized will be helpful for finding subclasses of particular classes.
- Statistical summaries can be created for each cluster.
- Meta features such as radii, distances, etc can be found.

4.5.2 Dataset Compression and Editing

The aim of dataset editing is to remove examples from a training set so that the accuracy of a model can be increased. The supervised clustering is useful for editing a dataset to produce a smaller subset. The smaller subset consists of cluster representatives that have been selected by supervised clustering algorithm. There is one algorithm 1-NN classifier for classifying new examples using subset instead of original dataset. This method is called as supervised clustering editing.

Figure 3 illustrates how supervised clustering is used for editing dataset. Figure 3.a shows the dataset that has not been clustered yet. Figure 3.b shows a dataset that was partitioned.

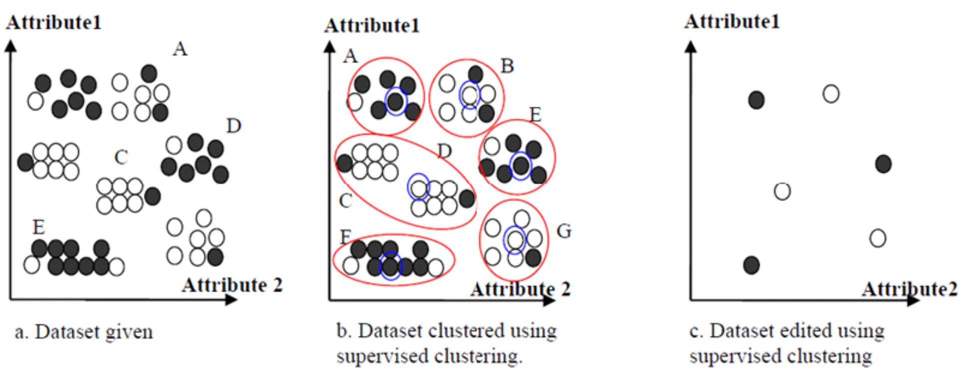

Figure 3: Dataset Compression [1]

4.5.3 Useful in Identifying Subclasses in a Class

The supervised clustering is of great importance. It is useful for identifying the patterns cluster within a cluster .This helps the pattern to go to its correct position rather than going to any arbitrary position. Consider a new comer student comes in class and he wants to identify correct group of friends that can he sit with. The whole class is a cluster but if sits with his liking students then he will be more satisfied. Thus we are creating a cluster within a cluster and supervised clustering helps in doing so.

4.5.4 Meta Features Specifications

Supervised clustering helps in creating meta features such as radii, distances between clusters.This meta parameters are also good to visualize to clear or see the difference of clustering in a distinct way. These are some of the advantages of supervised clustering. Hence supervised clustering is a powerful clustering technique compared to traditional clustering.

4.6 Applications of supervised clustering

Supervised Clustering is a type of clustering of automatically fitting a clustering algorithm with the help of a training set consisting of patterns and labels of these patterns. It is used in many tasks such as :

- **Image segmentation:-** Image segmentation is a process of partitioning a digital image into multiple segments. Generally the unsupervised based clustering is applied. But it was observed that with the help of supervised clustering the efficiency increased. Only a part of database need to be searched. For this, supervised clustering came useful.
- **News article clustering:-** Supervised clustering is used in news article clustering. News are given certain labels such as crime, sports, politics. Hence, it becomes easy to form clusters of news on the basis of labels. When a new news comes it goes into correct cluster formed.
- **Earthquake Studies:-** In this specific regions are formed in cluster which are given the label as danger zone or safe zone. When a new region comes, so if this is close to the proximity of danger it will be identified as danger or safe.
- **Biology:-** Classification of plants and animals with their features and labels.
- **Email batch clustering:-** There are mails such as spam and not span. Clusters of spam messages and non-spam messages can be formed according to the labels. So with the help of this when a new message comes it will be compared to both classes. If it falls in the cluster of spam it is given label as spam otherwise non-spam.

These were the applications of supervised clustering. It can be used in many fields which can improve efficiency over other models such as Linear Regression, Logistic Regression, SVM.

4.7 Algorithms in Supervised Clustering

In traditional clustering that is unsupervised clustering the different fitness functions are used to classify the patters in a cluster. For example in k-means clustering centroid is used as fitness function. The fitness functions used by supervised clustering are different from traditional clustering. It evaluates based on following two criteria :

Class Impurity: It is measured by percentage of minority examples in different clusters of a clustering X. A minority example is an example that belongs to a class different from most frequent class in its cluster.

Number of clusters k: This is the number of clusters for objects which should be low.

TABLE OF NOTATIONS

Notation	Description
$O = o_1, ...o_n$	Objects in a dataset
n	No. of objects in the data set
$d(o_i, o_j)$	Distance between objects oi and oj
c	The no. of classes in the dataset
Ci	Cluster associated with the i^{th} representative
$X = C_1, ...C_k$	A clustering solution consisting of Clusters C_1 to C_k
$k = \mid X \mid$	The no. of clusters in a solution X
$q(X)$	A fitness function that evaluates a clustering X

Table 1: Table of Notations[1]

The fitness function which is used in the supervised clustering is as follows:

$$q(X) = Impurity(X) + \beta * Penalty(k) \tag{1}$$

where,

$$Impurity(X) = percentage\, of\, minority\, example\, of\, other\, classes \tag{2}$$

$$Penalty(k) = \begin{cases} sqrt((k-c)/n) & \text{if } k \geq c \\ 0, & \text{if } k < c \end{cases}$$

With n being total number of examples and c being number of classes in dataset. The parameter β,

$$(0 < \beta <= 2.0) \tag{3}$$

determines the penalty that is associated with number of clusters $\boldsymbol{k}$. Higher values of β indicates larger penalties.

Two special cases are there for above fitness function: The first case is ***X1*** that uses only clusters. The second case is a clustering ***X2*** uses ***n*** clusters and assigns a single object to each cluster, therefore each is pure.

We obtain,

$$q(X1) = Impurity(X1)\ and\ q(X2)\ =\ \beta \tag{4}$$

4.7.1 Representative-Based Algorithms

There are many algorithms for supervised clustering. The target of RBA is to find ***k*** representatives that fit the dataset properly. We will find distance of testing sample from these ***k*** representatives and assign to nearest representative. ***The goal is to find subset Pr of P such that the clustering obtained through it should minimize our q(x).*** The advantage of this is it is useful for data summarization and no new distances have to be computed as that of ***k*** means.

4.7.2 Partitioning Around Medoids (PAM)

It is a clustering algorithm called as k-medoid which aims to find ***k*** represents objects among objects in the data set minimizing the fitness function as stated above. Here a new function similar to fitness function called ***Tightness(X)*** is used. It denotes the chosen representative is how much close or bound to that object in dataset. Initially a random is chosen but as move further more representatives are added.

$$Tightness(X) = 1 \mid objects \mid \Sigma\ distance(object, medoid(object)\ ) \tag{5}$$

Where medoid(object) is the medoid representative of cluster that object belongs to. The number of clusters is input to this algorithm. PAM is again divided into two algorithms. First one is called as BUILD algorithm which begins with a set of representatives that initially contains the medoid of complete dataset and as we move further new representatives are added which minimize the fitness function. The second algorithm PAM called SWAP which tries to improve the clustering obtained by BUILD by exploring all possible replacements of medoids by non-medoids picking the replacement that enhances the fitness function the most. If algorithm can't proceed further then algorithm terminates.

4.7.3 Supervised Partitioning Around Medoids (SPAM)

This algorithm is a slight variation of the algorithm PAM that uses the fitness function ***q(X)*** instead of ***Tightness(X)***. The number of clusters k is an input parameter to the algorithm. It consists of two sub-algorithms. Sub-algorithm SBUILD starts selecting the medoid of the members of the most frequent class in the data set as the first representative. After that, it repeatedly and greedily adds to the current set of representatives a non-representative object that, if added to the set of representatives, would generate a clustering that produces the minimum value for the fitness function ***q(X)***. The second sub-algorithm, SSWAP, tries to improve the clustering produced by SBUILD by exploring all possible replacements of a single representative by a single non-representative. SPAM terminates if no replacement can be found that leads to a clustering with a better (lower) fitness value with respect to $\boldsymbol{q(X)}$.

4.7.4 Top Down Splitting Algorithm (TDS)

This is a very simple algorithm that aims at creating clusters quickly. It starts by assigning all data objects to a root cluster. After that the algorithm recursively splits clusters by replacing the medoid of the cluster with two medoids: the medoid of the most frequent class in the cluster and the medoid of the second most frequent class in the cluster. Clusters are only split if $\boldsymbol{q(X)}$ does not increase as a result of the split. This splitting procedure is recursively applied to newly generated clusters.

5 FUZZY HYPERSPHERE

5.1 Introduction

Fuzzy Logic is a technique which is basically based of degrees of truth rather than general true of false. It includes 0 and 1 as extreme cases of truth but also includes the various states in between so that result could not be tall or short but 38 of tallness or 64 of shortness. It can be used to describe how information is processed inside human brains. Fuzzy gives probabilistic results rather than binary results which can be useful.

Fuzzy hypersphere is basically the n-dimensional space which maps to lowest abstraction level. In simple manner all the n dimensional features combine together to form a hypersphere. This hypersphere is created by using fuzzy logic which we know that it gives continuous values rather than discrete results.

5.2 Fuzzy Hypersphere Neural Networks

A neural network is based on a collection of nodes called neurons just like brain. Each connection between artificial neurons can transmit a signal from one to another. In common neural networks artificial neurons are real number and the output of each artificial neurons is calculated by non-linear function of the sum of its inputs. Artificial neurons and connections typically have a weight that adjusts as learning continues. Artificial neurons have a threshold such that only if signal crosses that threshold then only signal is sent.

A fuzzy set is a set whose elements which have a degree of membership. The membership can be any value from the closed interval [0,1]. i.e. an element may belong to a particular set with degree of membership = 0.7. In other words, it can be said that the membership of an element is described with the help of a membership function valued in the real unit interval.

The fuzzy hypersphere neural network uses fuzzy set hyperspheres as pattern clusters and classes are represented by union of fuzzy set hyperspheres.

Membership function plays an important role in FHNN. This neural network uses fuzzy membership function to give 100% training accuracy. Membership function is actually used for testing sample. When any pattern comes we use membership function to decide that to which class this pattern belongs.

5.3 Modified Membership Function

We discussed that membership function plays an important role while classifying any sample. In FHNN the membership function is defined as follows :

$$d(P_h, O_j, r_j) = f(dist, r_j) \tag{6}$$

where, P_h is h^{th} input pattern P = P_1, P_2, P_3,.....P_h,....P_n
O_j is j^{th} centroid of a cluster O = O_1, O_2, O_3...O_h...O_n,
dist is distance between O_j and P_h,
r_j is radius of O_j.

$$f(dist, r_j) = \begin{cases} 1 & \text{, if } dist \leq r_j \\ r_j/dist & \text{, otherwise} \end{cases}$$

Limitations:
The membership function discussed above has some limitations. This Membership function uses the max membership value as 1.But the maximum value can be more than one also. Consider one case where a point lies in intersection of two circles. Now both this circles will give membership value as 1.Now it becomes difficult to assign label for input pattern for a cluster as both returns 1. This is the big disadvantage of this function. The scenario is illustrated below.

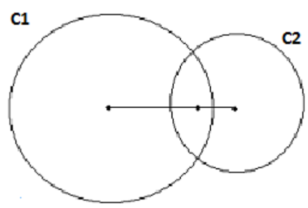

Figure 4: Initial Membership Function [3]

Here for C1 and C2 the P1 lies within both circles. The above membership function will return 1 as value. But it will become too difficult to classify the pattern.

Proposed Membership Function The modified membership function is as follows:-

$$d(P_h, O, r) = f(D, r) \tag{7}$$

where, P_h is h^{th} input pattern P = P_1, P_2, P_3,.....P_h,....P_n ,
O_j is centroid array of a cluster O = O_1, O_2, O_3...O_h...O_n,
D is distance array between

$$(P_h, O_j) \ \forall \ O_j \epsilon \ O \tag{8}$$

r is radius array for every centroid.

$$f(D, r) = max(r_j/D_j) \ \forall \ (r_j, D_j) \ \epsilon \ (r, D) \tag{9}$$

Now the above problem can be solved using modified membership function.

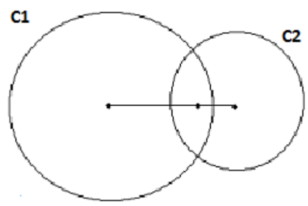

Figure 5: Modified Membership Function [3]

The distance of testing sample from ***C1*** is more than the distance of ***C2*** from sample. Hence the membership function value ***(r1/D1)*** will be less than as it is farther from centroid of ***C1*** where as the membership function value of ***C2*** will be more than that of ***C1***. Hence the label for the testing sample will be label associated with ***C2*** . In short the modified membership function is useful when multiple clusters return 1 as membership function value.

5.4 Architecture of FHNN

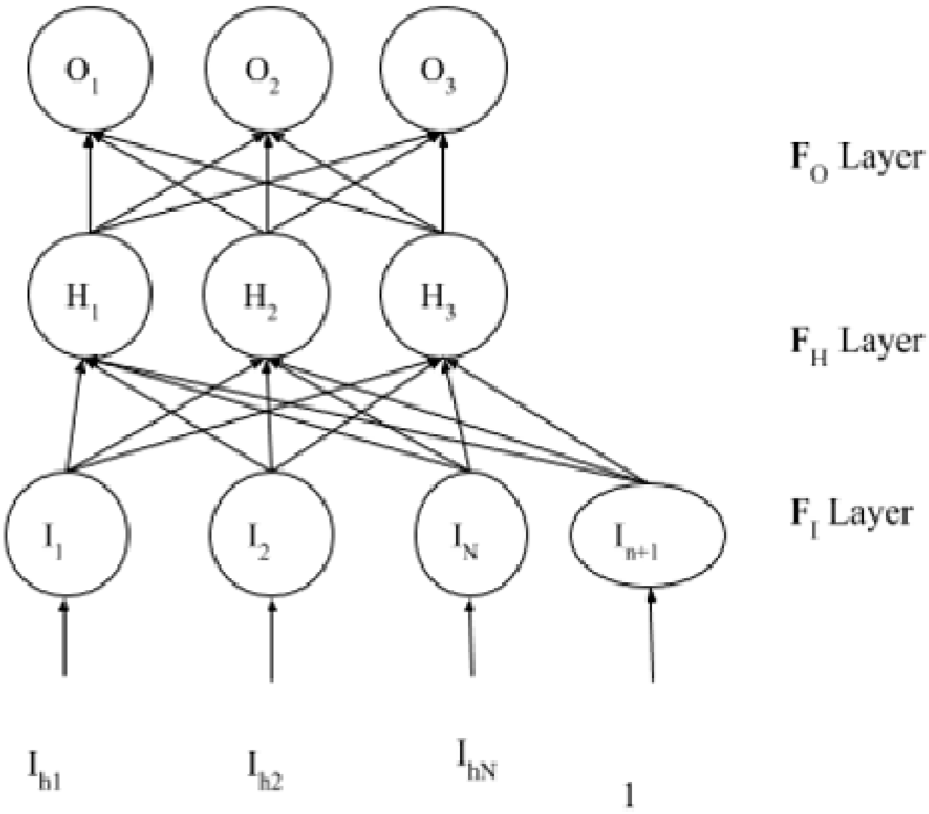

Figure 6: Architecture of FHNN [3]

As show above the ***N*** nodes in ***F1*** layer accept ***N*** dimensional pattern and the input $(N+1)^{th}$ is fixed to 1. The nodes in ***F1*** layer don't do any processing and simply pass on the information to the hidden layer.

The ***J*** nodes in F_h layer are hidden layer nodes whose processing is characterized by a fuzzy proposed membership defined as above.

The last layer is centroid layer called as F_0 layer. It indicates the formation of clusters with a centroid and having a certain label associated to it.

The role of activation function as such of neural networks is performed by membership function. It triggers a node to pass on the signal to other node.

5.5 Learning Algorithms

Notations and Symbols:-

k = 1, 2........k denotes the classes of samples
O = $O_1, O_2....O_n$ where n is no of centroid clusters formed
P = Total number of patterns of all classes
a_k = Total number of patterns belonging to class k
t_k = P - a_k denoting total number of patterns belonging to other than class k
I is the input pattern matrix
r = $r_1, r_2, ...r_n$ radii associated with centroid clusters

Algorithm FHNN()

Step 1: Compute the inter-class distance for every pattern in the class k A new matrix A^k is created which is of shape $a_k * t_k$. It will store interclass distance of every pattern in every class to that of other classes. Different types of distances are used to compute the distance. Mainly Manhanntan, Eucledian distances are used.

$$A^k = [dist(I_i - I_j)] \tag{10}$$

where, i = 1,2,......,a_k
j = 1,2,......,t_k

Step 2: Calculate intra-class distance for each class $\boldsymbol{k}$ A new matrix is used named B^k of shape $a_k * a_k$

$$B^k = [dist(I_i - I_j)]\ of\ size\ a_k * a_k \tag{11}$$

where, i = 1,2,.......,a_k
j = 1,2,......,a_k

Step 3: Calculate min-interclass distance matrix for each class k

$$W^k = \min_{j=1}^{t_k}(A_{ij})^k \tag{12}$$

where, i = 1,2,.......,a_k
j = 1,2,.......,t_k

Step 4: Find the centroid that covers maximum patterns

Step 4.1: Initialize a count array of size $a_k * (a_k + 1)$

$$pcount^k[a_k * (a_k + 1)]$$

Consider each pattern in class k as initial centroid and its radius as minimum of interclass distance which we computed from W^K, that is,

$$tempr^k{}_i = w^k{}_i \tag{13}$$

Step 4.2: Check whether this centroid covers its intra-class neighboring points

for j <- 1 to a_k do
    if B_j^k <= $tempr_i^k$ do
        increment $pcount^k$[i,a_k]
        mark as visited $pcount^k$[i,j]<-1
    end
end

Step 4.3: Choose that centroid which covers maximum patterns of its class

index<-maximum(pcount)
Add that centroid in resultant O matrix
increment pointer of

Step 4.4: Update that centroid radius to its max-intraclass distance within its cluster

max1 <- 0
for j <- 1 to a_k do
    if $pcount^k$[index,j] is 1 and max1 <= B^k[index,j] do
        max1 = B^k[index,j]
    end
end

Step 4.4.1 Check wheather given centroid covers only himself

if maximum(pcount) = 0 do
    update its radius as half of minimum interclass distance
    that is, r[pointer] <- W^k[index] / 2
else
    r[pointer] <- max1
end

Step 4.5: Associate label to that cluster the same as that of its centroid

Step 4.6: Mark all these points as visited which are covered in the cluster

```
for j <- 1 to a_k do
    if pcount^k [index,j] = 1 do
        mark as visited I^k[j] <- 1
    end
end
```

Step 5: Decrement a_k by visited number of patterns in cluster

$$a_k < -a_k maximum(pcount) \tag{14}$$

Step 6: Repeat the steps from 1 to 5 until a_k becomes zero and for each class k

Step 7: Halt

6 IMPLEMENTATION AND RESULTS

The Fuzzy Hypersphere learning algorithm was implemented which is defined in above section. The Supervised Clustering along with Fuzzy combination came to be fruitful as good classification technique. It was implemented on following data sets:-

- **Iris**

  Iris data set is widely popular data set.Total 150 labelled samples of plants are given in this data set. In all there are four features namely Sepal-length, Sepal-Width, Petal-Length, Petal-Width in cm. By using these features we have to predict class of plant that Iris-setosa, Iris-versicolor, Iris-virginca.

  I used 120 training samples which lead to **100%** training accuracy. Remaining 30 samples were used for testing. Moreover K-fold validation technique was used which led to on an average **96%** testing accuracy.

K-fold	Training Samples No.	Testing Samples No.	Accuracy
1	120	30	100%
2	120	30	96.66%
3	120	30	96.66%
4	120	30	93.33%
5	120	30	93.33%
Average = 96%			

Table 2: Iris Data set results

- **Liver**

  It is another popular data set. In this data set we have to predict whether the given input is liver patient or not. The data set consists of 10 feature columns. The 11^{th} column is output label associated to pattern. Some of the features are gender,tot-proteins etc.

  The data set consists of 583 samples out of which I choose 504 for training and 79 for testing.The training algorithm gave **100%** training accuracy. I used k-fold validation which gave on an average **71%** testing accuracy

K-fold	Training Samples No.	Testing Samples No.	Accuracy
1	504	70	73.611%
2	504	70	69.44%
3	504	70	68.04%
4	504	70	70.83%
5	504	70	69.44%
6	504	70	72.11%
Average = 70.65% ≈ 71%			

Table 3: Liver Data set results

- **Pima**

  It is another data set which gives information of Indian Diabetes Patient. In this data set we have to predict whether the given input pattern suffers from diabetes or not by seeing to features of input pattern. Eight feature columns are given and 9^{th} column is considered as output label.

  The data set comprises of 768 training samples. Out of which 576 are chosen for training which leads to 100% training accuracy. Remaining 192 are used for testing. K-fold validation technique is used.

K-fold	Training Samples No.	Testing Samples No.	Accuracy
1	572	192	75.52%
2	572	192	69.791%
3	572	192	68.75%
4	572	192	79.685%
Average = 73.43%			

Table 4: Pima Data set results

- **Glass dataset**

  This is another popular data set which is available on UCI repository. The task is to classify the given input pattern into type of glass. Six classes of glass are present in this data set. In all total nine feature columns are present. The 10^{th} column is a label.

  In all there are 210 input samples. Out of which 150 are chosen for training which led to 100% training accuracy. Remaining 50 are used for testing. K-fold validation technique is used which led to accuracy on an average 76.67%.

K-fold	Training Samples No.	Testing Samples No.	Accuracy
1	160	50	82%
2	160	50	74.3%
3	160	50	76.6%
4	160	50	73.8%
Average = 76.67%			

Table 5: Glass Data set results

7 LITERATURE SURVEY

Year	Conference/ Journal Paper	Author Name	Algorithms used	Salient features
2004	Supervised Clustering - Algorithms and benefits [1]	Christoph F. Nidal Zeidat, and Zhenghong Zhao	PAM, SPAM, TDS	Supervised clustering and its features, Proposed algorithms for supervised clustering, Applications of Supervised Clustering
2017	Multi-label Classification Systems by the Use of Supervised Clustering	N.Rastin, M.Z.Jahromi ,M.Taheri	Supervised Clustering	Proposed Multilabel classifiation using Supervised Clustering, Intially performed using unsupervised clustering, Datasets used were From Mulan library
2001	Fuzzy Hypersphere Neural Network Classifier	U.V. Kulkarni ,T.R. Sontakke	Learning algorithm for Fuzzy Hypershpere	Proposed Fuzzy Hypersphere using Supervised Clustering, Concept of membership function for a cluster was introduced, Evaluation was done on Iris dataset with 93% accuracy, Gives 100% training accuracy
2017	Pruned Fuzzy Hypersphere Neural Network (PFHSNN) for Lung Cancer Classification	D. N. Sonar ,U. V. Kulkarni	Learning algorithm for fuzzy hypersphere with Pruning process.	Proposed Fuzzy with Pruning Process, Gives 100% training accuracy, Dataset was from JSRT database Japan

Table 6: Literature Survey

8 CONCLUSION

Successfully studied Supervised clustering, it's features, algorithms, advantages. Supervised clustering plays an important role as a pattern classification technique. As clustering is associated with Unsupervised algorithms such as K-means, K-NN but it can be concluded that clustering can not only be performed by Unsupervised techniques but also by Supervised techniques. Then we studied how Supervised clustering can be integrated with Fuzzy Hypersphere.

By using labels of patterns, clusters are formed with respective centroids and raddi. The cluster contains only same class patterns and not different class patterns. When a new sample comes for testing membership function it is used to place that testing pattern into appropriate cluster.

We performed Fuzzy Learning algorithm on various data sets such as Iris, Liver, Pima and Glass. The outcomes were certainly good. The metrics accuracy led to **100%** training accuracy on every data set. K-fold validation was used to calculate the testing accuracy. Hence, it can be said that supervised clustering techniques can be applied to most of the pattern classification techniques which would yield fruitful results.

References

[1] Christoph F. Eick, Nidal Zeidat, and Zhenghong Zhao(2004) Supervised Clustering – Algorithms and Benefits, *16th IEEE International Conference on Tools with Artificial Intelligence*, 774-776.

[2] Niloofar Rastin,Mansoor Zolghadri Jahromi,Mohammad Taheri(2017) Multi-label Classification Systems by the Use of Supervised Clustering, *Artificial Intelligence and Signal Processing Conference (AISP)*, 246-249.

[3] U.V. Kulkarni, T.R. Sontakke(2001) Fuzzy Hypersphere Neural Network Classifier, *10th IEEE International Conference on Fuzzy Systems.* (Cat. No.01CH37297),3,1221-1226.