An Enhanced Incremental Leader Ant Clustering With Constraints

K.Sumangala
M.C.A., M.Phil,
Assistant Professor, Research Scholar,
Department of Computer Science, Vellalar
College for Women,Erode-638 012

D. Vasanthi
M.C.A., (M.Phil.,)
Vellalar College for Women, Department of
Computer Science
Erode-638 012,

,

,

ABSTRACT

Clustering task aims at the unsupervised classification of patterns in different groups. Ant-based clustering is a biologically inspired data clustering technique. In the research, three new variants of the Leader Ant Clustering with Constraint algorithm (ILAMC, ILAME and ILACE) are proposed that implements incremental Leader Ant-based clustering and the following constraints: the must-link (ML), cannot-link (CL) constraints and ε – constraints. The main aim of the research is to improve the clustering accuracy, reduce the execution time and providing better convergence, to validate the accuracy using the F-measure and Entropy.

INTRODUCTION

Clustering is a useful technique for the discovery of data and patterns in the underlying data. The goal of clustering is to discover the dense and the sparse regions in a data set. Clustering with constraints has become a topic of significant interest for many researchers because it allows to take into account the knowledge from the domain, expressed as a set of constraints, and thus to improve the efficiency of the analysis [3].

1. ANT CLUSTERING

Ant-based clustering was originally introduced as a computational model of the clustering and sorting behaviors observed in ants. The objective of the ant clustering are to sense the entropy of their locality the pheromone model, optimal solution with limited number of iterations, extract the shortest possible path, allows ant to carry the entire heap of object [7].The two reasons for the specificity of the ant based clustering are they are able to perform the exploratory data analysis and performance can be improved by the key features that would make the algorithm mature tool for diverse applications [1]. Items that were scattered within the environment can be picked up, transported and dropped by the agents(ants).These were biased by the distribution of items within the agents local neighbourhoods.Such that items are isolated or surrounded by dissimilar ones were more likely to be picked up or dropped in the locality of the equivalent ones[1].

2. CONSTRAINTS

Constraint proliferation is the process of using a constraint to reduce the province of a variable based on the domains of the variables. Constraint programming systems have been used in a wide variety of applications and often lead to effective solutions.

The main principles of constraint programming are:
1. Users specify a problem declaratively by providing constraints on variables with domains.
2. Solvers find solutions by constraint proliferation and search**.**

Constraints can be particularly beneficial in data mining, when precise definitions of classes are absent. The aspects of algorithms, theory and applications of the problem of clustering with constraints have been introduced in [1]. The constraints are indeed much easier to provide than class labels even when the user has a very incomplete prior knowledge of the domain.

2.1 Must-Link (Ml) Constraint

Constraints specify that two points S_i and S_j (i≠j) in *S* have to be in the same cluster. As it is well known, ML constraints are transitive; that is to say, ML constraints (S_i,S_j) and (S_j,S_k) imply that there exists a ML constraint(S_i,S_k). Thus, the two constraints can be combined into a single ML constraint, namely (S_i,S_j,S_k). By computing the transitive closure of $Con_=$ a given collection $Con_=$ of ML constraints can be transformed into an equivalent collection $M=\{M_1,M_2,M_3….Mr\}$of constraints[4]. The sets in M are pair wise disjoint and have the following interpretation:

For each set $M_i(1≤i≤r)$,the points in M_i must be all in the same cluster in any feasible solution.

For the feasibility purposes, points which are involved in any ML constraint can be partitioned in to clusters in an arbitrary manner.

2.1.1 Algorithm for ML Constraint

```
Step 1: compute the transitive closure of the constraints in C. Let this computation result  in r sets of points, denoted by $M_1$, $M2 \dots M_R$
Step 2: Let s' =s-$\bigcup_{i=1}^{r} Mi$ ( s' denotes the subset of points that are not involved in any must  link constraint).
Step 3: if $r \geq k_l$ then   (a) $A = (\bigcup_{i=k_i}^{r} Mi) \bigcup s'$.
                               (b) Output  $M1....Mk_l - 1, A$
        Else
            (a)Let t= $k_l$-r .partition s' in to t clusters A1……At arbitrarily.
            (b)Output $M_1,….M_r,A_1….A_t$
```

2.2 Cannot Link (CL) Constraint

Constraint specifies that two point S_i and S_j (i≠j) in *S* must not be placed in the same cluster. Each CL constraints also involve a pair of points S_i and S_j in any feasible clustering points S_i and S_j must not be in same cluster. For any pair of points S_i and S_j in *S*, the distance between them is denoted by $d(S_i , S_j)$ with a symmetric property so that $d(S_i , S_j)=d(S_j, S_i)$.

2.3 ε-Constraint

The constraint specifies a value ε>0 and the feasibility requirement is the following: for any cluster S_i containing two or more points and for any point $S_p \in S_i$ there must exist another point $Sq \in S_i$ such that $d(S_p ,S_q) \leq \varepsilon$. Informally the constraint requires that in any cluster S_j containing two or more points, each point in S_j must have another point within a distance of at most ε. The observation points out that a ε –constraint corresponds to a disjunction of ML constraints.

So, to determine feasibility under a ε-constraint for the set of points S, we first find the subset S_1 containing each point which does not have a ε -neighbor. Let |S | = t, and let C1, C2. Ct denotes the singleton clusters formed from S_1. To cluster the points in S2 = S-S_1 (i.e., the set of points each of which has a ε neighbor it is convenient to use an auxiliary undirected graph.

Definition 4.1. Let a set of points S and a value ε > 0 be given. Let Q $\subseteq$ S is a set of points such that for each point in Q, there is a ε-neighbor in Q.

The auxiliary graph G (V, E) corresponding to Q is constructed as follows

(a) The node set V has one node for each point in Q.

(b) For any two nodes v_p and v_q in V, the edge {V_p and V_q} is in E if the points in Q corresponding to V_p and V_q are ε –neighbors.

2.3.1 Algorithm for ε-Constraint

```
1. Find the set S1 ⊆ S such that no point in S1 has a
ε-neighbor. Let t = |S1| and S2 = S - S1.
2. Construct the auxiliary graph G(V,E) for S2 . Let G
have r connected components (CCs) denoted by $G_1$,
$G_2$. . . $G_r$.
3. Let N* = t + min {1, r}. (Note: To satisfy the ε-
constraint, at least N* clusters must be used.)
4. if N* > $K_u$ then Output “No feasible solution” and
stop.
5. Let C1, C2… Ct denotes the singleton clusters
corresponding to points in $S_1$. Let $X_1$, $X_2$. . . $X_r$
denote the clusters corresponding to the CCs of G.
6. if t + r ≥ $K_u$
then /* We may have too many clusters. */
     (a) Merge clusters $X_{Ku-t}$ , $X_{Ku-t}$ .+1. . ., $X_r$ into a
single new cluster $X_{Ku-t}$ .
(b) Output the $K_u$ clusters $C_1$, $C_2$. . . $C_t$, $X_1$, $X_2$. . $X_{Ku-t}$
else /* We have too few clusters. */
(a)Let N = t + r.
    Construct spanning trees $T_1$, $T_2$. . . $T_r$ corresponding to the
CCs of G.
(b) While (N < $K_l$) do
     (i) Find a tree Ti with at least two nodes.
          If no such tree exists,
               Output “No feasible solution” and stop.
     (ii) Let v be a leaf in tree Ti. Delete v from Ti.
     (iii) Delete the point corresponding to v from cluster Xi
and form a new singleton cluster XN+1 containing that point.
     (iv) N = N + 1.
  Output the K` clusters $C_1$, $C_2$. . . $C_t$, $X_1$, $X_2$, . . ., $X_{K`-t}$.
```

3. COMBINATION OF CONSTRAINTS

The CL-feasibility problem is NP-hard, the feasibility problem for any combination of constraints involving CL constraints are, in general, computationally inflexible The feasibility problem remains efficiently solvable when both a ML constraint and ε-constraint are considered together as well as when ML and CL constraints are considered together as well as CL and ε-constraint. When ML constraints are considered together with a ε-constraint, the feasibility problem is NP-complete. The result points out that when ML and ε-constraints are considered together, the resulting feasibility problem is also NP-complete in general.

3.1 Combination of ML & ε- Constraint

The feasibility problem for the combination of ML and ε-constraints is NP-complete. We use a reduction method from a problem to prove the result which is known to be NP-complete

3.2 Combination of CL & ε-Constraint

The CL feasibility is NP-Complete. The fact in conjunction with the proof in the appendix implies that the CL feasibility problem is computationally inflexible even when the number of constraints is linear in the number of points. For each point s_i, create a graph node v_i, and for each CL constraint {s_i, sj}, create the undirected edge {v_i, vj}.

4. LEADER ANT CLUSTERING WITH CONSTRAINTS

The Leader Ant clustering algorithm (LA) is inspired from the chemical recognition system of ants [3].

4.1 Chemical Recognition System of Ants

Each ant has its own aroma called label that is spread over its “skin”. The label acts as an identity card and is partially determined by the genome of the ant and by the substances extracted from its environment (mainly the nest materials and the food)[3][6].The template can be either fixed or computed as the mean value of the distance values *d (i,j)* predicted between Nb_{Learn} couple of ants *i* and *j* [3].The ants are selected randomly and the template is continually updated and is used at each convergence between two ants, to decide if they should accept each other and exchange chemical cues The continuous chemical exchanges between the nest mates lead to the establishment of a colonial odor between the nest mates and recognized by every nest mates, according to the “Gestalt theory” [6][3].

4.2 The Leader Ant Model

The LA is inspired by the real ants and the clustering problems. In LA, an ant is described as an artificial agent. The artificial agent is described by three parameters [3].

1. The ***genome*** is associated with an inimitable objective of the data set.
2. The ***template*** **i**s the identical for all artificial agents and is either fixed or computed experimentally as the mean value of the Distance values *d(i,j)* estimated between Nb_{learn} couples of ants i and j randomly selected with *d(i,j)*,the distance value between object associated to the agent *i* and *j* .

$$Template = \frac{\sum_{Nb_{Learn}} d(i,j)}{Nb_{Learn}}$$

4.3 Label

The ***label*** reflects the nest membership of each artificial agent. At the beginning, the value is set to zero as no suggestion is made concerning the initial membership of agents.LA is a one-pass agglomerative algorithm that iteratively selects at random a new agent *a.* The agent a is selected based on the criteria that it has not been assigned in any of the nest. Agent determines its label or nest membership by simulating $Nb_{meetings}$ meetings with randomly selected agents from each existing nest *k* in [0,NbMaxNests]. During these meetings, the agent *a* estimates the similarity of its genome with those of ants from the evaluated nest *k*. At the end, the distance D(a,k) between the agent *a* and the nest *k* is computed as the mean distance over the $Nb_{meetings}$ meetings.

$$D(a,k) = \frac{1}{Nb_{meeting}} * \sum_{j=1}^{Nb_{meetings}} d(a, agent_j^k)$$

Where $agent_j^k$ is the j_{th}, jε[1, Nbmeetings] randomly selected ant from nest k. If no nest exists or if the indicated distance value is under the template value, the agent creates its own new nest [2].

NbMaxNests=NbMaxNests+1(create new nest)

$Label_a$=NbMaxNests

In the contradictory, the agent joins the nest with the lowest indicated distance value by setting its label as follows

$$\mathbf{Label_a = argmin_{k\varepsilon[1,\ NbMaxNests]}\ D(a,k)}$$

Finally, when all agents are assigned to a nest, the smallest nests whose size is under a fixed percentage of the total number of objects n can optionally be deleted and their agents reassigned to the other clusters. Table 3.1 presents the algorithm of the Leader Ant method[2].

TABLE 3.1: The LA Algorithm

```
Input: data set S with n points
Output: partition of the data set

1. Initialization of Artificial Ants
2. Template computation or template fixed experimentally
3. Iterative nests Building
        3.1 Selection of an artificial ant a
        3.2 For each existing nest k
                - Random meetings with ants from k
                - Estimation of distance between a and nest k
                - Assignment to nest k or building of a new nest
4. Deletion of smallest Nests (option)
```

5. INCREMENTAL ANT BASED CLUSTERING

The incremental Leader ant clustering algorithm incorporates the following modules [2]

1. Initialize clusters
2. Amend clusters dynamically
3. Sustain cluster model with a varying crisscross

5.1 Initialize clusters

In the initial state(t=t_0), select N_0 objects from the current dataset, and dispense the N_0 objects consistently at random on the *Z X Z* crisscross. Initialize all ant clusters to be unlade (unload).

A subspace with a cluster has lower entropy than a subspace without cluster [2] .Information entropy is introduced in the clustering algorithm. Denote *S X S* as the region in which a leader agent lies. Assuming independence of attributes, the entropy of the *S X S* area including a set of objects is defined by equation (1), where *p(X)* is defined by equation (2), obj_num is the total number of objects in *S X S*; x_num is the number of objects whose attribute X_i has value x.

$$E(s2) = -\sum_{i=1}^{n}\sum_{x\varepsilon X_i} p(x)\log p(x) \quad (1)$$

$$p(x) = \frac{x\_num}{obj\_num} \quad \mathbf{(2)}$$

Take entropy as the criterion for a leader ant to pick up or drop items, we propose the following algorithm for initial clustering [2].

5.1.1 SWARM_CLUSTER ALGORITHM

```
    Initialize parameters: $Z, s, t_{max}, N_a$
  For every object $o_i$ do
  Place $o_i$ randomly on the plane of $Z \times Z$
                // Z is grid
    End For
    For $n = 1$ to $N_a$ do //$N_a$ number of agents
                Place leader ant at randomly selected
                site in $Z \times Z$
    End For
    For $t = 1$ to $t_{max}$ do//$t_{max}$ maximal times that a
  Leader agent changes.
   For $n = 1$ to $N_a$ do
      If ((leader ant unlades) and (site occupied
  then by object $o_i$))
  Compute entropy $E1, E2$
       If ($E1 > E2$) then drop $o_i$ (dropping rule
       End if
  End If
      Move to randomly selected neighbor site not
  occupied by other leader ant
    End For
  End For
   For each site $(x, y)$ in $Z \times Z$ do
  Compute entropy of the surrounding area
  $s \times s$ area
    Compute pheromone $\tau(x, y)$
    End For
```

In the algorithm, when all agent clusters stop moving, the initial clusters have been formed. For the next incremental clustering, to each pixel (x,y) in the *Z X Z* plane, compute the entropy of the surrounding *S X S* area according to equation (1), and if *S X S* area is empty set entropy of the area with a maximum. Then compute the pheromone concentration τ(x, y) of the surrounding *S X S* area according to equation (3).

$$\tau(\text{x, y}) = \frac{obj\_num}{1+S\ X\ S} \qquad (3)$$

Where obj_num is the number of objects in the surrounding *S X S* area.

5.2 Modify clusters dynamically

After the initial clusters form, the cluster model will be updated from time to time as the database changes, through insertions and deletions. In incremental clustering a leader agent moves according to the following rule instead of moving randomly [2].

```
A. Rule

 1. If (leader ant carrying a new object)
    If τ(x,y)>Φ
          Move to the pixel$(x', y')$, where$(x', y')$ is an
      empty site within the $s \times s$area

Else
          Move to a randomly selected site not
      occupied by another leader ant
  End If
 End If

 2. If (leader ant unlades)
     If τ(x,y)>Φ
        Move to the site of the nearest new object
    Else
        Move to a randomly selected site not occupied
 by another leader ant
    End if
 End if
```

In the above rule, ***Φ*** is a random number. For a leader agent carrying a new object, the greater τ(x, y) is, the higher the probability that ***τ(x,y)>Φ***, or in other words, the more likely it is to move to one of the clusters. Conversely, for an unlading leader agent, the smaller pheromone concentration is, the higher the probability it is to move to the nearest new object, pick up a new object, and move to one of the existing clusters.

After all leader agents stop moving, to each pixel (x,y) in the *Z X Z* plane, amend the entropy of the surrounding *S X S* area according to equations (1), and modify pheromone of the surrounding *S X S* area according to equation (4).

$$\tau(\text{x, y}) = \frac{(1-\lambda)obj\_num + \Delta obj\_num}{1+S\ X\ S} \quad (4)$$

Where

Obj_num is the number of previous objects in *S X S*;

Δobj_num is the number of new objects in *S X S*;

λ Is the pheromone evaporation rate.

The pheromone evaporation rate can be assumed to be set constant.

5.3 Maintaining the clustering model

The clustering model changes with the arrival of dynamically inserted or deleted data. A database is used to save the position of each pixel on the Z X Z plane. In the initial state, each pixel on the Z X Z plane is set to a null value. The schema of the database is (x,y,τ,entropy,pheromone), where(x,y) is the coordinate of each pixel on the *Z X Z* plane, t is the arrival time of the occupying object, entropy is the value E(s^2) of the pixel, and the pheromone is the τ (x,y) value[2].

B. The features of the proposed clustering algorithm

Compared to other methods, to pre-specify the number of clusters is not necessary. It can hit upon clusters of random shape without predefined preconception In addition, it has further four main features [2]:

(1) The factor influencing a leader agent's picking up or dropping action is entropy; each action of picking up or dropping can reduce the entropy of the previous patch, and thus speed up clustering.

(2) The number of parameters which need to be specified for constructing a clustering model is small. In our method there are only 5 parameters (Z, s, tmax, Na, λ).

(3) After the initial or incremental clustering, we compute and save the entropy and pheromone

for each pixel in the *Z X Z* plane for sequent clustering. The agents movements are guided by the pheromone for locating new objects

(4) The temporal complexity of locating objects in initial clustering is 0(t_{max} x N_a), and the temporal complexity of computing the entropy and pheromone is *0(Z X Z).* They are all independent of the number of objects. The one reason why swarm intelligence is especially applicable to dynamic environments.

5.4 Incremental Leader Ant clustering with ML & CL constraints (ILAMC)

The ILAMC algorithm which integrates the must-link and cannot-link constraints to ILA algorithm is presented. ML constraints are transitive; that is to say, ML constraints(si,sj) and (sj,sk) imply that there exists a ML constraint(si,sk) [3].The two constraints can be combined into a single ML constraint, namely(si,sj,sk) [4]. By computing the transitive closure of con$_=$ the ML constraints can be converted in to equivalent collection of constraints. The transitive closure computation in the ILAMC can be carried out as follows [4]. An undirected graph G with n node is constructed, one node for each point in the data set, and an edge between two nodes if the corresponding points appear together in the ML constraint [4]. The set of transitive closure is used to reduce the number of points which has to be considered during the clustering [4]. The set of CL can be used at each iteration directly to determine the clusters when estimating the distance between a nest and an artificial ant (equation 2).

5.5 Incremental Leader Ant clustering with ML and ε-constraints (ILAME)

ILAME algorithm that implements the ML constraints and ε- constraint in ILA algorithm. Similarly to ILAMC algorithm, the transitive closure is constructed for the ML constraints and the transitive closure has to satisfy the ε-constraint [4]: for any transitive closure M_k and for any point $S_p \varepsilon M_k$, there must be another point $S_q \varepsilon M_k$ such that d $(s_p,s_q) \leq \varepsilon$.

5.6 Incremental Leader Ant clustering with Cannot-link and ε-constraints (ILACE)

A constraint version of ILA algorithm is derived by using CL constraints and ε-constraint [4]. The CL constraints and ε-constraints are used directly in the clustering process. A set of CL constraints can be applied to the iteration to find the clusters when estimating the distance between the nest and an agent [4].

S.No	Data Set	Accuracy(F-Measure)	Entropy Value		
			ILAME	ILACE	ILAMC
1	Glass	0.0892	0.4749	0.4766	0.4756
2	Iris	0.0769	0.5958	0.3792	0.3792
3	Lymph	0.0769	0.5930	0.6005	0.6136
4	Thyroid	0.1366	0.5930	0.5848	0.5848
5	Wine	0.1250	0.5336	0.5236	0.5332
6	Wbcd	0.1250	0.5942	0.6154	0.6227

5.7 Algorithm of Incremental Leader Ant Clustering

```
Input: data set S with n points
Output: partition of the data set

1. Initialization of Artificial Ants
2. Template computation or template fixed experimentally
3. Iterative nests Building
        a) Selection of an artificial ant a
        b) For each existing nest k
        c)Random meetings with ants from k
        d) Estimation of distance between ant a and nest k
        e) Assignment to nest k or building of a new nest based on the SWARM_CLUSTER algorithm.
4.Initialize the clusters based on the entropy value
5. Modify the cluster value based on the 2 Rule.
6. Maintain the Cluster Model by saving the changes in a Database.
7.Based on the Constraints Selection
    a) Select the ILAME for ML and ε-Constraint to be satisfied.
    b) Select the ILACE for CL and ε-Constraint to be satisfied.
    c) Select the ILAMC for ML and CL constraint to be satisfied.
8. Deletion of smallest Nests (option).
```

6. EVALUATION METHODS

A.F-Measure

F-measure combines the precision and recall concepts from information reclamation. We then calculate the recall and precision of that cluster for each class as:

$$precision(i,j) = \frac{n_{ij}}{n_j} \quad (1) \qquad recall(i,j) = \frac{n_{ij}}{n_i} \quad (2)$$

where n_{ij} is the number of objects of class i that are in cluster j, n_j is the number of objects in cluster j, and n_i is the number of objects in class i. The F-Measure of cluster j and class i is given by the following equation

$$F(i,j) = \frac{2 \cdot precision(i,j) \cdot recall(i,j)}{precision(i,j) + recall(i,j)}$$

The F-Measure values are within the interval [0, 1] and larger values indicate higher clustering quality.

B. Entropy

Entropy measures the purity of the clusters class labels. Thus, if all clusters consist of objects with only a single class label, the entropy is 0. However, as the class labels of objects in a cluster become more varied, the entropy increases.

Table 6.2. Accuracy value for F-Measure and Entropy.

To compute the entropy of a dataset, we need to calculate the class distribution of the objects in each cluster as follows:

$$E_j = \sum_i p_{ij} \log(p_{ij})$$

where the sum is taken over all classes. The total entropy for a set of clusters is calculated as the weighted sum of the entropies of all clusters, as shown in the next equation

$$E = \sum_{j=1}^{m} \frac{n_j}{n} \cdot E_j$$

where n_j is the size of cluster j , m is the number of clusters, and n is the total number of data points.

7. EXPERIMENTAL RESULTS

The performance evaluation of the proposed approaches is evaluated using Iris, Glass, Thyroid, Lymphoma, and Wine, Wisconsin Breast Cancer Database (WBCD) based on the clustering accuracy and convergence behavior.

7.1 Clustering Accuracy

The results of LA and MELA are compared with the proposed Incremental Leader Ant clustering with Must-link and ε-constraint (ILAME) to evaluate the clustering efficiency. The proposed ILAME is compared with LA and MELA on three aspects: average minimum accuracy, average mean accuracy and average maximum accuracy over 100 runs.

S.No	Data set	Average Min Accuracy	Average Mean Accuracy	Average Max Accuracy
1.	Glass	60	80	85
2.	Iris	62	81	85
3.	Thyroid	65	83	83
4.	Lymph	63	85	82
5.	Wine	63	82	84
6.	Wbcd	64	82	81

Table 7.1 Accuracy comparison with LA, MELA, ILAME.

The results are shown in figures. ILAME performs better than MELA and LA. LA aims at satisfying all the constraints whereas MELA uses the constraints to refine the search space and to indirectly help to define the number of clusters as such information is not initially available as a parameter of the method.

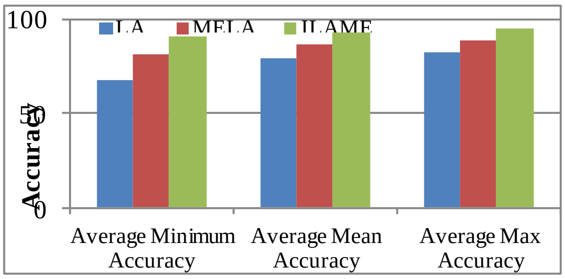

Figure 7.1: Accuracy Comparison with LA, MELA and ILAME.

7.2 Convergence Behavior

A convergence behavior is the fixed iteration number or cluster result which does not change after a certain number of iterations.

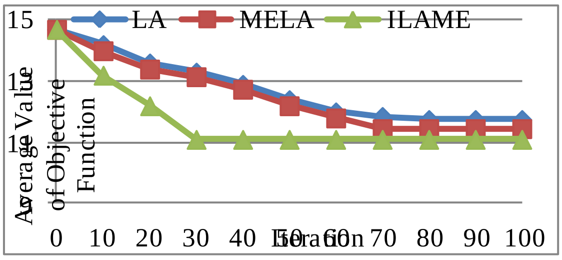

Figure 7.2: Convergence behavior Comparison with LA, MELA and ILAME.

8. DATA SET RESULTS

The dataset results shows that the clustering accuracy is increased using the Incremental constraint with the other the constraint like ML, CL, ε-constraint.the result for the Iris data set is shown in the following figures.

8.1 Iris Data Set

The iris data set was created by R.A. Fisher. The donor of the iris data set is Michael Marshall. The data set has four attributes like sepal length in cm, sepal width in cm, petal length in cm, petal width in cm. The data set has classes like iris setosa, iris versicolour; iris virginica. The data
set has 150 instances.

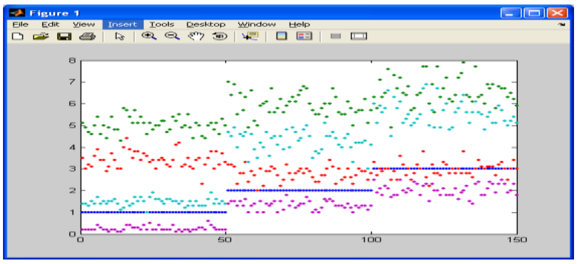

Figure 8.1 Iris Data Set Plotted Initially

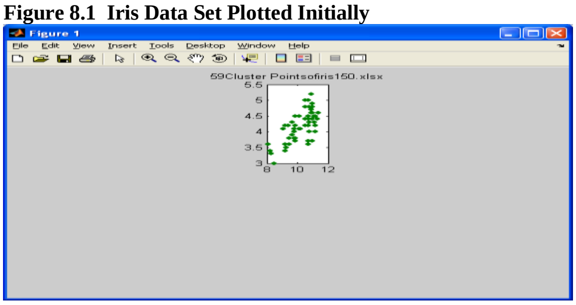

Figure 8.1 Accurate Clustered 59 Points Plotted Using ILAME Constraint

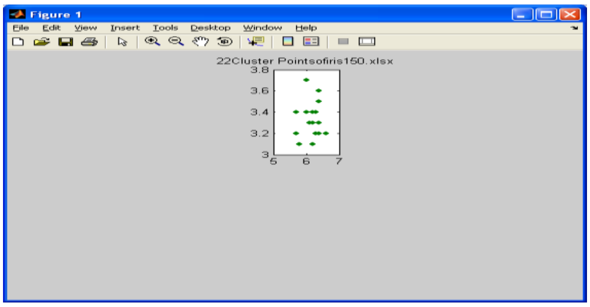

Figure 8.1 Accurate Clutsered 22 Points Plotted Using ILACE Constraint

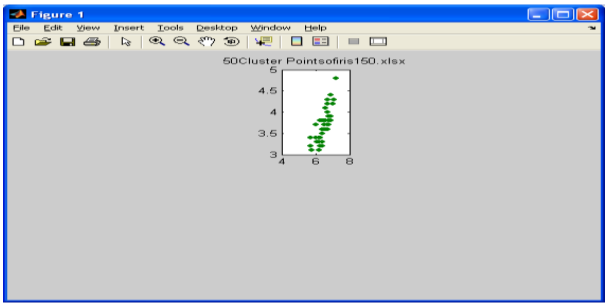

Figure 8.1 Accurate Clustered 50 Points Plotted Using ILAMC Constraint.

9. SUMMARY

In the research the problem of Leader ant based clustering is discussed and proposed the Incremental Leader ant based clustering with constraints for effective clustering performance. However, the complexity analysis shows that both adding and removing constraints is typically intractable. In the research, the constraint addition problem is also focused and show that just adding a single constraint (be it ML or CL) is in the worst case intractable.

The proposed approach aims at constructing the knowledge model incrementally for a dynamically changing database. It makes use of a swarm of special leader ants, i.e. an ant colony, and imitates their natural behaviors to form clusters of arbitrary shape gradually, unnecessary to pre-specify the number of clusters. The algorithm applies information entropy to model behaviors of leader ants, such as picking up and dropping objects, and guides leader ant movement by pheromone in incremental stages.

10. SCOPE FOR FUTURE WORK

There are many directions for future research. An Enhanced incremental leader ant based version of the ILA algorithm can be proposed by combining all the constraints for better clustering performance. Execution Time can be reduced using the ILAME with some enhancements. A new constraint named δ constraint can be used in clustering. The δ constraint can be used with other constraints to improve the clustering accuracy.

11. REFERENCES

[1] Andre L.Vizine,Leandro N.de Castro, Eduardo R.Hrusehka,Ricardo R. Gudwin,Towards Improving Clustering Ants: An Adaptive Ant Clustering Algorithm

[2] Bob McKay, Bo Liu,Jiuhui Pan, Incremental Clustering Based on Swarm Intelligence.

[3] Bernadette-Meunier, Leader Ant Clustering With Constraints,

[4] I. Davidson, M. Ester and S.S. Ravi, “Clustering with constraints: Feasibility issues and the K-means algorithm”, in proc. SIAM SDM 2005, Newport Beach, USA.

[5] I. Davidson, M. Ester and S.S. Ravi, “Agglomerative hierarchical clustering with constraints: Theoretical and empirical results”, in Proc. of Principles of Knowledge Discovery from Databases, PKDD 2005.

[6] B. Hölldobler and E. Wilson (1990), The Ants, Chapter colony odor and kin recognition. p. 197-208. Spinger Verlag, Berlin, Germany.

[7] Daniel Barbará, Julia Couto, Yi Li, COOLCAT: An Entropy-based Algorithm for Categorical

Clustering, Proceedings of the Eleventh International Conference on Information and KnowledgeManagement, 582-589, 2002.

[8] D. Klein, S.D. Kamvar and C.D. Manning, “From Instance-Level constraintes to space-level constraints: Making the most of Prior Knowledge in Data Clustering”, in proc. 19th Intl. on Machine Learning (ICML 2002), Sydney, Australia, Jyly 2002, pp. 307-314.

[9] N. Monmarche, M. Slimane and G. Venturini (1999), “On improving clustering in numerical databases with artificial ants”, in D. Florence, J. Nicoud and F. Mondala, LNAI, Swiss Federal Institute of Technology, Lausanne, Switzerland, pp. 626-635.

[10] Shi Yong; Zhang Ge; “Research on an improved algorithm for cluster analysis”,International Conference on Consumer Electronics, Communications and Networks (CECNet), Pp. 598 – 601, 2011

[11] K. Wagstaff, C. Cardie, S. Rogers and S. Schroedl, “Constrained Kmeans clustering with background knowledge”, in: Proc. Of 18th Int. Conf. on Machine Learning ICML’01, pp. 577 - 584.

[12] K. Wagstaff, Intelligent clustering with instance-level constraints, PhD Thesis of Computer Science, 2002, Cornell University, USA