CHAPTER 6
IDENTIFICATION OF CLUSTERS USING VISUAL VALIDATION
VAT ALGORITHM

Clustering is the process of combining a set of relevant information in the same group. In this process KM algorithm plays a major role to achieve the effective way of grouping information. But in this algorithm number of groups (clusters) selected by user are the main issues. In order to overcome the shortcoming of KM algorithm, Visual Assessment of cluster Tendency (VAT) is used to get the number of clusters in a visual manner (Bezdek & Hathaway 2002). But the estimated result does not match with the true value in many cases. Liang Wang (2009) stated a new Dark Block Extraction (DBE) method for automatically identifying the number of clusters in large amount of datasets. Many methods also available to identify automatically estimate the number of dark blocks in RDI unlabelled datasets (Srinivasulu Asadi et al 2011).

The complex datasets Spectral VAT algorithm (Spec-VAT) is more effective than VAT algorithm. The Spec-VAT algorithms are categories in to A Spec-VAT, P Spec-VAT and E Spec-VAT are also used to find out the number of clusters in a given datasets (Liang Wang et al 2010). But the range of $k$ value is either directly or indirectly given to Spec-VAT algorithms. The algorithm is unsuccessful to highlight the cluster formation in large number of datasets. Havens & Bezdek (2012) implemented a graph-theoretic distance transform method called improved VAT (iVAT) algorithm which extensively reduces the computational complexity. Huband Jacalyn et al (2005) described
Big VAT method for visually estimate the cluster tendency for more datasets than the VAT method.

In this chapter, introduced a new Visual Validation VAT algorithm (V$^2$VAT) is used to enable visual cluster analysis for large datasets. Extensive experimental results are obtained from the proposed algorithm.

### 6.1 INTRODUCTION

Data clustering is a technique in which all the information is divided into desired number of groups. The clustering techniques are also referred as unsupervised technique because there is no particular dependent. The main objective of this chapter is to estimate the number of clusters $k$ in a visual validation manner.

There are several clustering algorithms are available, but it requires the number of clusters $k$ as input parameter of the algorithm, this yields the poor clustering results. The efficiency of clusters is maximum dependent on the initial estimation of number of clusters. Many algorithms are being attempted to find the post clustering process i.e. they attempt to choose the best grouping from a set of alternative group. In contrast, before clustering process starts to finding number of clusters is a difficult task to get the desired number of clusters. The proposed V$^2$VAT algorithm is to attempt the pre-clustering process to find the effective number of clusters.

### 6.2 SYSTEM MODEL

Cluster analysis is the problem of partitioning a set of objects into desired number of similar sets. In the clustering process major issues are identified by the number of clusters in beforehand. Most of the clustering algorithm supports only post clustering process for getting number of clusters.
The proposed method is to solve the issues in getting the effective number of clusters.

### 6.2.1 VAT algorithm

The VAT algorithm is a visually represents the number of clusters in the large amount of datasets. The datasets are represented the dissimilarity matrix, then it is converted into gray scale VAT image. Each pixel represents the scaled dissimilarities value. The pixel values are identified based on the high dissimilarities. ie., viewed as white pixels and low dissimilarities viewed as black pixels to identify the clusters. The diagonal element of the dissimilarity matrix is always zero because the object of each cluster is similar with itself. The elements in the off diagonal is scaled range is (0,1).

#### 6.2.1.1 VAT Algorithm Steps

**Input:** An n × n scaled matrix of pairwise dissimilarities

\[
D = \begin{bmatrix} d_{ij} \end{bmatrix} \text{ with } 1 \geq d_{ij} \geq 0; d_{ji} = d_{ij}; d_{ii} = 0 \text{ for } 1 < i, j \leq n.
\]

1. Initialize \( I = J = 0 \) and \( K = \{1, \ldots, n\} \)

2. Select two least related objects \( o_i \) and \( o_j \) from \( D \)

   Consider \( P(1) = i; I = \{a\}; J = K - \{b\} \)

   Continue up to \( r = 2 \) to \( n \).

3. Select (i, j): most similar objects \( o_i \) and \( o_j \)

   Select \( (i,j) = i \in I, j \in J \)

   Update \( P(r) = j; I = I \cup \{j\}; J = J - \{j\}; \)

**Output:** Obtain reordered dissimilarity matrix \( D^* \) for \( 1 \leq i, j \leq n \)
The VAT algorithm displays a dissimilarity matrix $D$ as a grayscale image, each element is a scaled dissimilarity value $d_{ij}$ between objects $o_i$ and $o_j$. If an object is a member of the cluster, then it should be part of a submatrix with dissimilarity values. This values corresponds to one of the dark blocks along the diagonal of the VAT image, each of which corresponds to one cluster. The RDI is used to represent the row and column of the pixel for corresponding input dataset (Srinivasulu Asadi et al 2010).

There are many possible ways to obtain a RDI, here to use VAT to generate RDI of unlabeled datasets. The Figure 6.1 shows the pictorial representation of VAT algorithm (Bezdek & Hathaway 2002). The algorithm read the dataset and converted into dissimilarity matrix $D$. Reordered dissimilarity matrix $D^*$ are constructed from gray-scale image. Finally to get the number of clusters from a VAT image.

**Step 1: Read input dataset**

Scatter plot of a 2D data set  
Unordered image $I(D)$  
Reordered VAT image $I(fg)$

**Step 2: Forming Dissimilarity Matrix (D)**

Let consider $O = \{o_1, o_2, \ldots, o_n\}$

Dissimilarity between objects $o_i$ and $o_j$
Step 3: Reordered Dissimilarity Matrix ($D^*$)

\[
R = \begin{pmatrix}
0 & 0.12 & 0.73 & 0.39 & 0.71 & 0.16 \\
0.12 & 0 & 0.59 & 0.73 & 0.39 & 0.71 \\
0.73 & 0.59 & 0 & 0.12 & 0.74 & 0.16 \\
0.39 & 0.73 & 0.12 & 0 & 0.74 & 0 \\
0.71 & 0.39 & 0.71 & 0.74 & 0 & 0 \\
0.16 & 0.71 & 0.12 & 0.74 & 0 & 0 \\
\end{pmatrix} = I
\]

\[
R = \begin{pmatrix}
0 & 0.12 & 0.73 & 0.39 & 0.71 & 0.16 \\
0.12 & 0 & 0.59 & 0.73 & 0.39 & 0.71 \\
0.73 & 0.59 & 0 & 0.12 & 0.74 & 0.16 \\
0.39 & 0.73 & 0.12 & 0 & 0.74 & 0 \\
0.71 & 0.39 & 0.71 & 0.74 & 0 & 0 \\
0.16 & 0.71 & 0.12 & 0.74 & 0 & 0 \\
\end{pmatrix} = I
\]

Figure 6.1 Pictorial representations of VAT algorithm steps

The Equation (6.1) denotes $n$ objects in the given dataset. In the Equation (6.2) represents the vectorial form of given objects, where $f_i \in R^h$ is each coordinate of the vector $f_i$ provides a feature value of each of $h$ attributes $a_j$ ($j$ value is 1 to $h$) corresponding to an object $o_i$.

In the pairwise similarities or dissimilarities between the objects are represented by $n \times n$ symmetric matrix $D$ this can directly recorded as relational data.

\[
O = (o_1, o_2, \ldots, o_n) \quad (6.1)
\]

\[
F = (f_1, f_2, \ldots, f_n) \quad (6.2)
\]

To convert vectorial data $F$ into dissimilarities $D$ as $d_{ij} = \|f_i - f_j\|$ and $1 < i, j \leq n$ in any vector norm in $R^h$. Generally, the dissimilarity matrix satisfies the condition as $1 \geq d_{ij} \geq 0; d_{ji} = d_{ij}; d_{ii} = 0$ for $1 < i, j \leq n$. The VAT algorithm displays a dissimilarity matrix $D$ as a gray-scale image and element of the dissimilarity matrix are in the range between (0,1). The purpose of highlighting cluster tendency in the datasets that contains the
compact separated clusters is used by RDI. But, highly complex datasets are many in practical applications that are involved. The proposed approach V^2VAT algorithm is used to obtain RDI which combines VAT and spectral analysis to overcome the issues of VAT algorithm.

### 6.2.2 Proposed Algorithm and Problem Formulation

- The new algorithm is proposed to use the divergence matrix gradient with respect to the factorizing matrices which depends heavily on the scales of the matrices instead of dissimilarity matrix. (Kenneth & David 2002)

- The index based validation method provides a solution for the problem where complex datasets exists.

- From the final visual validation image it is easy identify the number of clusters in the given datasets.

- The experiments are conducted in the basis of several real datasets to evaluate the new algorithm in order to check the effectiveness.

The V^2 VAT algorithm applies input in pairwise divergence matrix D as needed. From the matrix D convert to VAT image for getting the number of clusters and reordered matrix produces partitioning the clusters of a given datasets.

### 6.2.2.1 V^2VAT Algorithm

**Input:** D = [d_{ij}] : An \( n \times n \) scaled matrix of pairwise divergence matrix.
1. Find the local scale $\sigma_i$ for each $o_i$ as $\sigma_i = d(o_i, o_k) = d_{ik}$ where $o_k$ is nearest neighbour of $o_i$ without the user defined $k$ value. If $n(o_i) < n(o_k)$ is compared with Equation (6.3) and Equation (6.4)

$$n(o_i) = \sum_{i=1}^{n} \sum_{j=1}^{n} d(o_i, o_j) \quad (6.3)$$

$$n(o_k) = \sum_{j=1}^{n} \sum_{k=1}^{n} d(o_j, o_k) \quad (6.4)$$

2. Select $m$ indices from 1 to $n$ randomly to form the sample index set $I_s$ and set the remaining object indices $I_r$ which are respectively used to get sub-matrices $D_S$ and $D_B$ from matrix $D$.

3. Create the matrices $S \in R^{m \times m}$ from $D_S$ and $B \in R^{m(n-m)}$ from $D_B$ using the weighted Gaussian function $\exp -d_{ij}d_{ji}/\sigma_i\sigma_j$.

4. Calculate Eigen decomposition of $S$ and find the approximate Eigen vectors $\bar{U}_F$.

5. Choose the columns of $\bar{U}_F$ that will raise the divergence of matrix and columns form $V_k \in R^{n \times k}$ and normalize the rows of $V_k$ to unit Euclidean norm to generate $V_k'$. Each rows are considered $V_k'$ as a new case to find the new pairwise divergence matrix $D'_S \in R^{m \times m}$ between the sample case to obtain sample SpecVAT images $I(D'_S)$.

6. Find the weighted matrix $W \in R^{n \times n}$ by defining $\omega_{ij} = \exp -d_{ij}d_{ji}/\sigma_i\sigma_j$ for $i \neq j$ and $\omega_{ij} = 0$.

7. Next to find the normalized Laplacian matrix $L' = M^{-1/2}WM^{-1/2}$. 


8. Select the $k$ largest Eigen vectors of $L$ to form the matrix $V = [v_1, ..., v_k] \in \mathbb{R}^{n \times k}$ by stacking the Eigenvectors in columns.

9. Normalize the rows of $V$ with the unit Euclidean norm to generate $V'$.

10. Continue for $i = 1, ..., n$ let $u_i \in \mathbb{R}^k$ be the vector corresponding to the $i^{th}$ row of $V'$ and treat it as a new instance (corresponding to $o_i$). Then construct a new pairwise divergence matrix $D'$ between instances.

11. Select two least related objects $o_i$ and $o_j$ from matrix $D$

   Consider $p(1) = i; I = \{a\}; J = K - \{b\}$

   Continue upto $r = 2$ to $n$.

12. Select $(i, j)$: most similar objects $o_i$ and $o_j$

   Select $(i, j) = i \in I, j \in J$

   Update $p(r) = j; I = I \cup \{j\}; J = J - \{j\}$

13. From the reordered matrix $\tilde{D}' = [d'_{ij}] = [d'_{\pi(i)\pi(j)}]$ for $1 \leq i, j \leq n$.

   A scaled gray-scale image $I(\tilde{D}')$ is obtained.

14. Find the optimal threshold $T^*_k$ that can maximize $\sigma^2_B$ for the image $I(\tilde{D}_k)$, i.e., $T^*_k = \arg \max_{1 \leq T \leq L} \sigma^2_B(T)$.

15. Get the equivalent goodness measure for each SpecVAT image $GM(k) = \sigma^2_B(T^*_k)$.

16. Determine the number of clusters as $c = \arg \max_k GM(k)$
17. For the sample SpecVAT image \( I(D^*_k) \) set the genome of each individual \( x_i \) \((i = 1 \sim b)\) as a binary string of length \( n-1 \), corresponding to the indices of the first \( n-1 \) samples.

18. Randomly set \( c-1 \) element in each \( x_i \) to ‘1’ and other to ‘0’ to create the initial population.

19. Set a fitness function as taking the input \( x_i \) and calculate the candidate partition \( U \) from \( x_i \), and returning the result of objective function.

20. Apply GA for time complexity (Falkenauer 1997) until there is no improvement within \( g = 10 \) generation to find the optimum genome \( x^* \).

21. Change \( x^* \) into cluster partition \( U^* \) (which is equivalent to getting the sizes of each cluster \( \{n_1, ... n_c\} \)). The position \( p_1 \) of the first ‘1’ in \( x^* \) means the first cluster partition is from sample 1 to \( p_1 \). The position \( p_j (j = 2, ... c-1) \) of the \( j^{th} \) ‘1’ means the \( j^{th} \) cluster partition is from sample \( (p_{j-1} + 1) \) to \( p_j \). The \( c^{th} \) cluster partition is from the sample \( (p_{c-1} + 1) \) to \( n \).

22. Get the real objects indices of each cluster \( C_i \) with the permutation index \( \pi(\cdot) \) i.e., \( C_1 = \{o_{\pi(1)}, ..., o_{\pi(n_1)}\} \) and \( C_i = \{o_{\pi(n_{i-1}) + 1}, ..., o_{\pi(n_{i-1})+n_i}\} \) for \( i = 2, ... c \). Get the data partitioning \( \{C_1, ..., C_c\} \).

23. Perform out-of-sample extension using \( V^*_k \) to obtain the cluster labels of the remaining objects indexed by \( I_r \).

Output: The number of clusters \( C \) with data partitioning outcome.
The algorithm mainly used to handle the large amount of dataset into desired number of clusters. In order to get effective clusters, introduces the pairwise divergence matrix (KL divergence) is a non-symmetric measure of the difference between two probability distributions $P$ and $Q$ (Kullback 1959; Kenneth & David 2002). The KL divergence of $Q$ from $P$ is defined to be discrete probability distributions $P$ and $Q$, is defined in Equation (6.5).

$$D_{KL}(P \parallel Q) = \sum_i \left( \frac{P(i)}{Q(i)} \right) P(i) \quad (6.5)$$

where, $P$ and $Q$ is the logarithmic difference between the probabilities

The KL divergence is only defined if $P$ and $Q$ both sum to 1 and if $Q(i)$ implies $P(i)$ for all $i$. The KL measures the expected number of extra bits required to code samples from $P$ when using a code based on $Q$, rather than using a code based on $P$. Typically $P$ represents the "true" distribution of datasets, observations or a precisely calculated theoretical distribution (Kenneth & David 2002). The measure $Q$ typically characterize a theory, model, description and approximation of $P$.

The algorithm need not mention the entire object, each object is verified using discrete probability distributions $P$ and $Q$ as converted into pairwise divergence matrix $D$ to be loaded into memory at once, because may not be sufficient for large datasets. The given input matrix $D$ is very large incrementally loads it to perform step (1) without user defined $k$ value.

To handle the large datasets, propose a feasible approximate sampling plus extension (sample index set $I_k$) to enable both visual validation estimation and dividing the datasets of a sub matrix $D_S$ and $D_B$ from matrix $D$ are done in step (2).
In the random sampling, other active sampling techniques of weighted gaussian function are used to extracting the required number of objects from complete grouping solution defined by Fowlkes et al (2004) in step (3).

Consider $n \times n$ symmetric positive semidefinite matrix ($F$), can be decomposed in the Equation (6.6).

$$F = U_F \Sigma_F U_F^T$$  \hspace{1cm} (6.6)

where, $\Sigma_F$ is Eigen values of $F$ and $U_F$ is associated Eigenvectors.

Let consider, with no alternate $m$ columns of $F$ which are randomly sampled. Let $A$ be the $n \times m$ matrix of these sampled columns, and $S$ be the $m \times m$ matrix consisting of $m$ columns with equivalent $m$ rows. To obtain symmetric positive semidefinite matrix ($F$) columns and rows of $F$ without changing as mention in Equation (6.7)

$$F = \begin{pmatrix} S & B \\ B^T & C \end{pmatrix} \text{ simplified as } A = \begin{pmatrix} S \\ B^T \end{pmatrix}$$  \hspace{1cm} (6.7)

where, $B \in R^{m \times (n-m)}$ contains the elements from the sampled objects to the remaining ones, $C \in R^{(n-m) \times (n-m)}$ contains the elements between all the remaining objects.

$S$ is very small but $C$ is usually large and let consider $m \ll n$ objects.

The purpose of finding the numerical selection to Eigen function problems, the Nystrom method has been used for fast Gaussian process for image segmentation as mentioned in the Equation (6.8).

$$F \approx \tilde{F} = A S^+ A^T$$  \hspace{1cm} (6.8)

where, “$+$” is the pseudo inverse.
The Nystrom method indirectly finds the value of \( C \) by using \( BS^+B^T \) and the resulting approximate Eigen values and Eigen vectors of \( F \) are denoted in Equation (6.9)

\[
\mathbf{\Sigma}_F = \left( \frac{n}{m} \right) \Sigma_F \text{ and } \mathbf{\bar{U}}_F = \sqrt{\frac{m}{n}} \mathbf{A}\mathbf{U}_S \Sigma_S^+
\]  

(6.9)

The Eigen vectors generated from the Nystrom approximation function are not exactly orthogonal because they are extracted from the Eigen vectors of matrix \( S \). Finding the approximated orthogonalized Eigen vectors \( \mathbf{\bar{U}}_F \) is adopted using this method which is referred in step (4). In a more explicit form the approximated Eigen vectors are mentioned in the Equation (6.10)

\[
\mathbf{\bar{U}}_F = \begin{pmatrix} \mathbf{U}_S \\ \mathbf{U}_S \Sigma_S^+ \end{pmatrix}
\]  

(6.10)

The Eigen decomposition estimation process is done by Nystrom method because of the time complexity and sample expansion from \( \mathbf{\bar{U}}_F \) as mentioned in step (5).

In order to find approximated Eigen vectors of \( F \) take only the matrix \( A \) or \( S \) or \( B \) as needed. In this process, Eigen decomposition of small sample matrix \( S \in \mathbb{R}^{m \times m} \) is practical, and multiplication with the matrix \( B \) (i.e., \( \mathbf{B}^T \mathbf{U}_S \Sigma_S^+ \)) is also an effective solution.

From the sample images \( I(D'_i) \) specific local scaling parameter are used for self tuning of the specified object \( i \) and \( j \). The objects are assigned to each clusters based on high affinities within clusters and low affinities across clusters. Next to form a normalized Laplacian matrix mentioned in the Equation (6.11).

\[
\hat{\mathbf{L}} = \mathbf{M}^{-\frac{1}{2}}(\mathbf{M} - \mathbf{W})\mathbf{M}^{-\frac{1}{2}} = \mathbf{I} - \mathbf{L'}
\]  

(6.11)
where, \( I \) is the identity matrix.

Changing \( I - L' \) with \( L' \) in the algorithm step (6 and 7) only changes the Eigen values from \( 1 - \lambda_i \) to \( \lambda_i \) and not the Eigen vectors.

Next step (8 and 9) select \( k \) minimum Eigen vectors of matrix \( L' \) to matrix \( V \) of matrix columns and next to normalize the rows to get \( V' \) matrix.

The new representations matrix \( D' \) step (10) are restricted by maximum value of \( k \) Eigen vectors of \( L' \) is the rank \( k \) subspace that best approximates \( W \), in which the original objects are indirectly transferred.

Next step (11 to 12) apply a new pairwise divergence matrix \( D \) to VAT algorithm to validate the matrix \( D \) into reordered dissimilarity matrix \( D' \).

Otsu (1979) stated each image pixel denoted as L gray scale. The total number of pixel level \( l \) is represented by \( m_l \) and finds the sum of pixels \( N \) by the Equation (6.12).

\[
N = \sum_{l=1}^{L} m_l \quad (6.12)
\]

Then the probability distribution mentioned in the Equation (6.13)

\[
p_l = \frac{m_l}{N}, p_l > 0, \sum_{l=1}^{L} p_l = 1 \quad (6.13)
\]

The image pixels are splitted into \( C_1 \) and \( C_2 \) as mentioned in cluster blocks, between cluster blocks in VAT image the threshold at level \( T \). Consider \( C_1 \) and \( C_2 \) pixel level are \( T=1 \) to \( L \). The probabilities of class represented by the Equation (6.14 and 6.15).

\[
\omega_1 = P(C_1) = \sum_{l=1}^{L} p_l \quad (6.14)
\]
In order to find the class mean levels using Equation (6.16 and 6.17).

\[
\mu_1 = \sum_{l=1}^{T} lP(l|C_1) = \sum_{l=1}^{T} \frac{lp_l}{\omega_2} = \frac{\mu(T)}{\omega(T)} \tag{6.16}
\]

\[
\mu_2 = \sum_{l=T+1}^{L} lP(l|C_2) = \sum_{l=T+1}^{L} \frac{lp_l}{\omega_2} = \frac{\mu_T - \mu(T)}{1-\omega(T)} \tag{6.17}
\]

where, \(\omega(T) = \sum_{l=1}^{T} p_l\) and \(\mu(T) = \sum_{l=1}^{T} lp_l\) are zeroth and first-order sum flow of the histogram up to the T th level.

\(\mu_L\) is the total mean level of actual image.

Note, \(\omega_1\mu_1 + \omega_2\mu_2 = \mu_L\) and \(\omega_1 + \omega_2 = 1\).

The class variances are in the Equation (6.18 and 6.19)

\[
\sigma_1^2 = \sum_{l=1}^{T} (l - \mu_1)^2 P(l|C_1) = \sum_{l=1}^{T} (l - \mu_1)^2 \frac{p_l}{\omega_1} \tag{6.18}
\]

\[
\sigma_2^2 = \sum_{l=T+1}^{L} (l - \mu_2)^2 P(l|C_2) = \sum_{l=T+1}^{L} (l - \mu_2)^2 \frac{p_l}{\omega_2} \tag{6.19}
\]

McLachlan (2005) stated discriminant criteria and Otsu (1979) stated measurement of evaluating the class by the Equation (6.20 and 6.23)

\[
\alpha = \frac{\sigma_B^2}{\sigma_W^2}, \beta = \frac{\sigma_T^2}{\sigma_W^2}, \gamma = \frac{\sigma_B^2}{\sigma_T^2} \tag{6.20}
\]

where,

\[
\sigma_W^2 = \omega_1 \sigma_1^2 + \omega_2 \sigma_2^2 \tag{6.21}
\]

\[
\sigma_B^2 = \omega_1 \omega_2 (\mu_2 - \mu_1)^2 \tag{6.22}
\]

\[
\sigma_T^2 = \sum_{l=1}^{T} (l - \mu_l)^2 p_l = \sigma_W^2 + \sigma_B^2 \tag{6.23}
\]
The above equations are between class variance and total variance respectively. In order to maximize $\alpha$, $\beta$ and $\gamma$ to achieve effective threshold to overcome the optimization issues.

Step (13 to 16) represent reordered matrix $\tilde{D}'$ to $I(\tilde{D}')$ gray-scale image are obtained. Next to find optimal threshold and to maximize $\sigma_{\tilde{B}}^2$. Finally get the number of clusters ($c$) from $\sigma_{\tilde{B}}^2(T_{K}^*)$ as in Equation (6.24).

$$c = \arg\{\max\} \sigma_{\tilde{B}}^2(T_{K}^*) \quad (6.24)$$

Step (17) in order to partitioning the clusters ($C_1, \ldots, C_c$) the SpecVAT image $I(\tilde{D}')$ is generated from VAT image to set $n$ objects, permutation index and size of population ($P$) and number of clusters ($c$).

Consider the genome of each object $x_i$ ($i = 1\sim b$). To make an initial population randomly assign $c-1$ element in each $x_i$ to 1 else 0 in step (18).

Step (19) consider the fitness function as taking the input $x_i$ and calculate the candidate partition $U$ from $x_i$ and returns the result based on the objective function mentioned in the Equation (6.25).

$$E(U, \tilde{D}') = E_{b} - E_{w} \quad (6.25)$$

where, $E_{b}$ and $E_{w}$ are the mean of dissimilarity within the dark regions of image $I(\tilde{D}')$

Step (20) use GA algorithm to find the time complexity Equation (6.26).

$$\text{Time complexity} = O(bg_{f_{0}}) \quad (6.26)$$

where, $b$ is a population
g is the number of generations  

\( f_0 \) is a computational complexity of a objective function.

In step (21) convert \( x^* \) into cluster partition \( U^* \) to get the sizes of each cluster \( \{n_1, \ldots, n_c\} \). In order to get effective objective function use the Equation (6.27)

\[
U^* = \arg [\max] E(U, D')
\]  

(6.27)

The computer model in which the population of conceptual representations are called as genome. The candidate solutions are called individuals to an optimization issues which evolves the effective solutions.

Step (22) using permutation index \( \pi() \) to retrieve real objects indices in each cluster \( C_i \) number of clusters. Step (23) to obtain the cluster labels of each cluster using out-of-samples extension using \( V'_{k,r} \) remaining objects indexed by \( I_r \).

In addition to this algorithm, this visual partitioning procedure can also operate on the VAT image or other reordered divergence images. Finally to get the number of clusters (c) and data partitioning.

### 6.3 RESULTS AND DISCUSSION

In this chapter the enhanced visual validation approach \( V^2 \) VAT is implemented by finding the number of clusters and partitioning the given datasets automatically. The real-world datasets are used to evaluate the performance of proposed method, it is taken from the UCI machine learning repository which is described in Table 1.1 of chapter 1. The following sections are described by the simulation results of the entire datasets.
6.3.1 Simulations Results

The proposed algorithm is stimulated by using Wine, Iris, Vehicle, Glass, Liver and Wisconsin datasets. The datasets characteristics and the results of $V^2$VAT method are summarized in Table 6.1.

Table 6.1 C value for different data set using VAT and $V^2$VAT

<table>
<thead>
<tr>
<th>Datasets</th>
<th>Number of instances</th>
<th>Number of attributes</th>
<th>Attribute Characteristics</th>
<th>Actual Number of clusters</th>
<th>$C$ value in VAT</th>
<th>$C$ value in $V^2$VAT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wine</td>
<td>178</td>
<td>13</td>
<td>Integer, Real</td>
<td>3</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>Iris</td>
<td>150</td>
<td>4</td>
<td>Real</td>
<td>3</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>Vehicle</td>
<td>946</td>
<td>18</td>
<td>Integer</td>
<td>4</td>
<td>5</td>
<td>4</td>
</tr>
<tr>
<td>Glass</td>
<td>214</td>
<td>10</td>
<td>Real</td>
<td>7</td>
<td>6</td>
<td>7</td>
</tr>
<tr>
<td>Liver</td>
<td>345</td>
<td>7</td>
<td>Categorical, Integer, Real</td>
<td>3</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>Wisconsin</td>
<td>699</td>
<td>10</td>
<td>Integer</td>
<td>2</td>
<td>3</td>
<td>2</td>
</tr>
</tbody>
</table>

In this section the performance of the proposed algorithm are investigated based on the input dataset. The results on the Wine dataset are shown in Figure 6.2. The given input wine dataset are converted into histogram of the original dissimilarity matrix $D$ as shown in Figure 6.2 a. The histogram is generated based on the average value of inter-cluster and intra-cluster distances, it is a multi model representation of dissimilarity matrix $D$. The objects are assigned to each clusters in the basis of high affinities within clusters and low affinities across clusters.
Figure 6.2 Results on the Wine dataset. (a) Histogram of the original dissimilarity matrix $D$. (b) Histogram of Laplacian Matrix $D'$. (c) Morphological Dilation filtered image $I$. (d) Distance Transform image $I$. (e) Diagonal projection signal $H$ from $I$. (f) VAT image. (g) $V^2$VAT image.

Figure 6.2 b displayed in the normalized Laplacian matrix $D'$ is generated based on the cluster formation. Next step to form the morphological dilation filtered image (I) are shown in Figure 6.2 c. In order to get the segmented image, clear morphological operations are used to perform binary image filtering. Morphological filtering is one of the processing techniques
used to modify the structure of objects. The Dilation and erosion are the two fundamental morphological operations which are used to form the effective cluster structure.

In order to convert the morphologically filtered image into an enlightening the dark block structure sequence, the values of pixels that are in the main diagonal axis of the image are shown in Figure 6.2 d. To find the clear result of count C use visual information to construct the diagonal projection signal (H) from the distance transform image (I), it shows the clear separation between major peaks in the signal.

Towards the VAT algorithm the VAT image is generated which is shown in Figure 6.2 f, it is not clearly shown the diagonal part of the image. The diagonal part of image is only to identify the number of clusters. The VAT image $I(D^*_k)$ is a set the genome of each individual as a binary string of length, corresponding to the indices of the first $n-1$ samples. The $V^2$VAT image are uptained based on the probability distribution to image pixels that are splitted into number of clusters within the cluster blocks as shown in Figure 6.2 g. The $V^2$VAT algorithms gives correct value of C is 3 but the VAT algorithms gives only 2.

The Iris dataset simulation results are illustrated in Figure 6.3. It is observed that the C value in VAT algorithm gives 4, but the proposed $V^2$VAT gives as 3 are shown in Table 6.1.
Figure 6.3  Results on the Iris dataset. (a) Histogram of the original dissimilarity matrix D. (b) Histogram of Laplacian Matrix $D'$. (c) Morphological Dilation filtered image I. (d) Distance Transform image I (e) Diagonal projection signal H from I. (f) VAT image. (g) $V^2$VAT image.

Correspondingly the Vehicle dataset simulation results are shown in Figure 6.4. It shows C value in VAT algorithm gives 5, but the $V^2$VAT gives as 4 are shown in Table 6.1.
Figure 6.4 Results on the Vehicle dataset. (a) Histogram of the original dissimilarity matrix $D$. (b) Histogram of Laplacian Matrix $D'$. (c) Morphological Dilation filtered image $I$. (d) Distance Transform image $I$ (e) Diagonal projection signal $H$ from $I$. (f) VAT image. (g) $V^2$VAT image.

The Glass dataset simulation results are shown in Figure 6.5. Here the C value VAT algorithm gives 6, but the $V^2$VAT gives as 7 are shown in Table 6.1.
Figure 6.5  Results on the Glass dataset. (a) Histogram of the original dissimilarity matrix $D$. (b) Histogram of Laplacian Matrix $D'$. (c) Morphological Dilation filtered image $I$. (d) Distance Transform image $I$ (e) Diagonal projection signal $H$ from $I$. (f) VAT image. (g) $V^2$VAT image.

The Liver dataset simulation results are shown in Figure 6.6. Here the C value VAT algorithm gives 2, but the $V^2$VAT gives as 3 are shown in Table 6.1.
Figure 6.6  Results on the Liver dataset. (a) Histogram of the original dissimilarity matrix $D$. (b) Histogram of Laplacian Matrix $D'$. (c) Morphological Dilation filtered image $I$. (d) Distance Transform image $I$ (e) Diagonal projection signal $H$ from $I$. (f) VAT image. (g) $V^2$VAT image.

The Wisconsin dataset simulation results are shown in Figure 6.7. Where the $C$ value VAT algorithm gives 3, but the $V^2$VAT gives as 2 are shown in Table 6.1.
Figure 6.7 Results on the Wisconsin dataset. (a) Histogram of the original dissimilarity matrix $D$. (b) Histogram of Laplacian Matrix $D'$. (c) Morphological Dilation filtered image $I$. (d) Distance Transform image $I$ (e) Diagonal projection signal $H$ from $I$. (f) VAT image. (g) $V^2$VAT image.

From the above simulations result shows the proposed $V^2$VAT algorithm gives the better C value than the existing VAT algorithm for each dataset.
6.3.2 Performance Measures

The proposed algorithm is evaluated using performance measures such as accuracy, precision, recall and F-measure with benchmark datasets are described in Table 1.1 of chapter 1.

Table 6.2 describes the $V^2$VAT proposed investigation method that are much efficient with a maximum performance than the existing methods KM and VAT algorithms.

**Table 6.2 Comparision table of KM,VAT and $V^2$VAT algorithm**

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Data sets</th>
<th>KM</th>
<th>VAT</th>
<th>$V^2$VAT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy</td>
<td>Wine</td>
<td>65.16</td>
<td>95.50</td>
<td>96.06</td>
</tr>
<tr>
<td></td>
<td>Iris</td>
<td>87.33</td>
<td>98</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td>Vehicle</td>
<td>67.02</td>
<td>86.17</td>
<td>98.08</td>
</tr>
<tr>
<td></td>
<td>Glass</td>
<td>69.15</td>
<td>97.19</td>
<td>97.19</td>
</tr>
<tr>
<td></td>
<td>Liver</td>
<td>68.11</td>
<td>97.10</td>
<td>97.10</td>
</tr>
<tr>
<td></td>
<td>Wisconsin</td>
<td>72.1</td>
<td>96.13</td>
<td>98.14</td>
</tr>
<tr>
<td>Precision</td>
<td>Wine</td>
<td>0.65</td>
<td>0.955</td>
<td>0.960</td>
</tr>
<tr>
<td></td>
<td>Iris</td>
<td>0.87</td>
<td>0.98</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Vehicle</td>
<td>0.67</td>
<td>0.98</td>
<td>0.980</td>
</tr>
<tr>
<td></td>
<td>Glass</td>
<td>0.69</td>
<td>0.97</td>
<td>0.971</td>
</tr>
<tr>
<td></td>
<td>Liver</td>
<td>0.67</td>
<td>0.97</td>
<td>0.971</td>
</tr>
<tr>
<td></td>
<td>Wisconsin</td>
<td>0.72</td>
<td>0.96</td>
<td>0.981</td>
</tr>
<tr>
<td>Recall</td>
<td>Wine</td>
<td>0.65</td>
<td>0.954</td>
<td>0.960</td>
</tr>
<tr>
<td></td>
<td>Iris</td>
<td>0.87</td>
<td>0.98</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Vehicle</td>
<td>0.67</td>
<td>0.861</td>
<td>0.981</td>
</tr>
<tr>
<td></td>
<td>Glass</td>
<td>0.69</td>
<td>0.971</td>
<td>0.972</td>
</tr>
<tr>
<td></td>
<td>Liver</td>
<td>0.68</td>
<td>0.969</td>
<td>0.970</td>
</tr>
<tr>
<td></td>
<td>Wisconsin</td>
<td>0.72</td>
<td>0.961</td>
<td>0.981</td>
</tr>
<tr>
<td>F-measure</td>
<td>Wine</td>
<td>0.65</td>
<td>0.955</td>
<td>0.96</td>
</tr>
<tr>
<td></td>
<td>Iris</td>
<td>0.87</td>
<td>0.98</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Vehicle</td>
<td>0.67</td>
<td>0.86</td>
<td>0.981</td>
</tr>
<tr>
<td></td>
<td>Glass</td>
<td>0.69</td>
<td>0.97</td>
<td>0.98</td>
</tr>
<tr>
<td></td>
<td>Liver</td>
<td>0.67</td>
<td>0.97</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Wisconsin</td>
<td>0.72</td>
<td>0.985</td>
<td>0.99</td>
</tr>
</tbody>
</table>
Accuracy comparison

In this section the performance is evaluated in terms of accuracy. Accuracy is calculated by the Equation (1.6) mentioned in chapter 1.

![Accuracy comparison chart of V^2VAT algorithm](image)

**Figure 6.8 Accuracy comparison chart of V^2VAT algorithm**

Figure 6.8 shows simulation and analytical results of accuracy rate. It is observed that proposed V^2VAT algorithm gives better accuracy in terms of effective number of clusters in a visual representation but existing KM and VAT gives less accuracy for all the datasets. The number of datasets and the accuracy parameter are considered as x and y axis respectively. The proposed method accuracy is smartly increased from 90% to 95.15% of efficiency.

Precision Comparison

Precision is described as the fraction of a cluster that consists of objects of a specified class. Precision can be calculated by the Equation (1.7) mentioned in chapter 1.
Figure 6.9 shows the precision and datasets comparison. It is analyzed with the existing KM and VAT algorithm with proposed V$^2$VAT algorithm, which finds the effective number of clusters in all the datasets. The number of datasets and the precision parameter are considered as $x$ and $y$ axis respectively. The proposed V$^2$VAT algorithm gives much better precision rate than the exiting methods as represented in the Figure 6.9.

Recall comparison

Recall is defined as probability of related objects is selected from the specified class. i.e., Recall is described as combination of all objects are grouped in to a specific class. Recall can be calculated by the Equation (1.8) mentioned in chapter 1.
Figure 6.10 Recall comparison chart of V^2VAT algorithm

Figure 6.10 shows the recall and datasets comparison. It is to analyze the existing KM and VAT algorithm with proposed V^2VAT algorithm, it gives the better number of clusters in all datasets. The number of datasets and the recall parameter are considered as x and y axis respectively. Hence the proposed V^2VAT algorithm gives much better recall rate than the exiting methods.

F-Measure Comparison

F-measure is a combination of precision and recall that measures the extent to which a cluster contains only objects of a particular class and all objects of that class. The performance is evaluated in terms of F-measure, it can be calculated by the Equations (1.9) and (1.10) given in chapter 1.
Figure 6.11 F-measure comparison chart of V²VAT algorithm

Figure 6.11 shows the F-measure and datasets comparison. The proposed method V²VAT gives better results in terms of selecting number of clusters in a visual manner. But existing KM and VAT algorithms gives less F-measure rate. The number of datasets and the F-measure parameter are considered as x and y axis respectively. Hence the proposed V²VAT algorithm gives much better F-measure rate than the exiting methods.

ROC Curve

Figure 6.12 displays the ROC graphs for all the datasets. It is very useful technique to represent the performance with FPR and TPR of all the datasets, it can select the certain conditions which are described in chapter 1. Figure 6.12 a presents the Wine dataset representation, less FPR and perfect TPR on both the algorithms. Figure 6.12 b shows Iris dataset represent TPR is higher in V²VAT than VAT algorithm. Similarly Vehicle, Glass, Liver and Wisconsin dataset gives higher TPR in V²VAT than the existing methods as represented in Figures 6.12 c – 6.12 f datasets respectively.
Figure 6.12 a Wine dataset  
Figure 6.12 b Iris dataset  
Figure 6.12 c Vehicle dataset  
Figure 6.12 d Glass dataset  
Figure 6.12 e Liver dataset  
Figure 6.12 f Wisconsin dataset  

Figure 6.12 ROC comparison graph of V^2VAT algorithm
Form the above experimental results it is observed that $V^2$VAT gives better results. At the outset, conclude that $V^2$VAT outperforms than KM and VAT algorithms to identify the best number of clusters in a visual manner for entire datasets.

6.4 SUMMARY

This chapter investigates $V^2$VAT algorithm to estimate the number of clusters in a given datasets automatically. The $V^2$VAT algorithm input gets only pairwise divergence matrix as user defined parameter. The $V^2$VAT algorithm is more effective in terms of visually represent the number of clusters for all the datasets. The direct visual validation method is an option for number of clusters in a simple representation and easy to read all the datasets. A series of experiments are conducted using different dataset have to performs well in terms of both number of clusters and data partitioning. The proposed method is not used the MO validation, hence further studies is needed to validate the clusters in a MO environment.