Available online at: www.basra-science-journal.org

ISSN –1817 –2695

Constructing Support Vector Classifier Depending On The Golden Support Vector

Hesham Saleh Ridha
Basic Sciences Department – College of Dentistry – University of Basrah
Basrah - Iraq
zhesham60@hotmail.com
Received 27-11-2013, Accepted 2-4-2014

Abstract

In order to increase the processing speed of online learning applications represented in its exigent requirement of reducing the amount of Support Vectors, this paper is devoted to present a durable algorithm to construct the well-known Support Vector Classifier, by capturing a unique Vector from each class of training instances. We called that Vector the (Golden Support Vector).

Our algorithm had adopted basic mathematical tools for its constructional phases. The algorithm starts with applying its Hybrid Enclosing Mechanism in order to enclose two sets of mapped instances (Vectors) with the most optimistic non-overlapped curved spaces analogous to the instances distribution. This mechanism is considered as a spring point that leads us directly to separate these spaces with a Strong Separating Hyperplane. From both sides of that Hyperplane, two parallel Supporting Hyperplanes will be released settling down on the first detected Vector, which we called the Golden Support Vector. Each Supporting Hyperplane with its acquired Golden Vector is considered to be the basis to construct the edges of Maximal Margin of Separation Space; which offers best generalization ability not only to the trained instances but also to guarantee high predictive test accuracy for future instances from the same distribution. Finally, the Optimal Separating Hyperplane will intermediate the space of that margin vanishing all other Vectors.

Rare inescapable cases have been discussed, provided with modest solutions as suggestions for future works. Affluent pictorial figures have been spread to emphasize our algorithm credibility.

Keywords: Support Vector Machines, Maximal Margin Classifier, Machine Learning, Statistical Learning, and Data mining.

1. Introduction:

On the basis of Statistical & Machine Learning Theories, the Russian scientist Vladimir N. Vapnic and his colleagues produce the Theory of Support Vector Machines (SVMs), in the nineties of the past century at AT&T Bell Labs in Holmdel, New Jersey [1, 2]. (SVMs) proves to be an excellent statistical tool used for classification; then it was developed to solve regression problems [3].

Since then, Support Vector classifier (SVC) and Support Vector Regressor (SVR) arises as a family under the title of Support Vector Machines (SVMs) [4].

Due to its efficient implementations and many attractive features; (SVC), since it was proposed, has received widespread attention and achieved development that extends to involve many fields of Data Mining and Pattern Recognition applications ranging from (Face Recognition & Image Retrieval, Speaker Verification, Text Categorization, Prediction, Handwriting Recognition, and Bioinformatics) [3,5].

The goal of (SVC), which is our concern here, is to isolate a given training Vectors data set into two distinct classes by an Optimal Separating Hyperplane (OSH) that offers optimum separation space called the Maximal Margin of Separation (MMS) [4, 6, 7, 8]. The mentioned margin should offer best generalization ability not only to the trained data but also to guarantee high predictive accuracy for future data from the same distribution [4]. Noticing that, (SVC) also called as a (Maximal Margin Classifier) [4].

The master key of gaining such goal is to grip small amount of trained Vectors (from both classes) from which the (MMS) can be determined as well as the (OSH) can be obtained. These Vectors are called (Support Vectors), which are considered to be the basis of (SVC) [6, 9, 10]. Without Support Vectors, purposeless separating Hyperplanes will arise [3, 11, 12, 13]. Please see figure (1)

Although (SVC) was considered as an accurate method to solve many classification problems, it is not preferred to use within online learning applications such as (Network Detection, Tumor Recognition, and Constructing Business Intelligence Systems) [14, 15]. In such applications, classification has to be done in great speed. This is because a large set of Support Vectors usually need to form the classifier, making the programming process complex and expensive [10, 16].

Many estimable researches have been introduced for reducing the amount of Support Vectors in order to speed up (SVC) making it more efficient. The goal of this paper is to present simple durable algorithm to construct (SVC), not by reducing the set of support vectors but by capturing a unique Vector from each class of training samples. We called that Vector the (Golden Support Vector).

2. Paper organization:

So far, we have introduced a modest introduction to the grand (SVMs) as a significant method to solve vast area of classification and regression problems; presenting Support Vectors to be the master key to construct (SVC), as well as the major weak point that affects that construction; that is, the set size of these Support Vectors. The remainder of this paper structure will be as follows: "In the next section, we shall touch on the obtainable works related to reducing the amounts of Support Vectors. On the strength of Vapnic definition to the linear classification problem, section (4) will provide insight view to the harmonic environment for the (SVC) constructional stages. Depending on the former sections, section (5) deals with the problem of detecting Support Vectors. Section (6) exhibits our novel algorithmic approach to simulate the classifier task. A brief discussion to our approach will be discussed throughout Section (7), as well as providing considerable suggestions for future work. Finally, the last Section involves a concise summary to our work.

3. Related works:

Several researches were proposed for reducing the amount of Support Vectors. We can list the obtainable papers respectively as follows:

- C.F. Lin et al. [17] proposed a Fuzzy Support Vector Machine (FSVM) based on the idea of adding additional samples into the training samples set in order to give Support Vectors higher degree membership, reducing the influence of other Vectors on the Optimal Hyperplane .
- D. D. Nguyen et al. [18] describes a bottom-up method in order to reduce the complexity of (SVMs) by reducing the number of necessary support vectors, which is included in their solutions. The reduction process iteratively selects two nearest Support Vectors belonging to the same class and replaces them by a newly constructed vector.
- S. S. Keerthi et al. [10] also, suggests an iterative process of adding new Support Vectors that can be stopped when the classifier has reached some limiting level of complexity. The method efficiently forms classifiers, which have smaller number of Support Vectors.
- In their paper, C. Fatichah et al. [19] used an active learning method to select most important training points to re-train a new (SVC). Their method removes training points, which are based on the training accuracy of original training set.
- The work of S. Agarwal et al. [16] proposes a learning procedure to produce a classifier for streaming online data where the maximum size of the Support Vector set is fixed. They used the concept of spanning the set of Support Vectors to decide which Support Vector should be replaced from their set.
- The effort that attracts our attention is what X. Wei et. al. [20] proposed. They produced One Class Support Vector Classifier depending on what they called (Border Support Vectors) that comprises all Vectors relying on a Hyper-Ellipsoid Sphere boundary.

  In this paper, we shall discuss their methodology showing its strength points rather than its weak points in order to produce our methodology that is based on their concept but from another point of view.

4. (SVC) constructional phases:

From the source definition provided by the provenance [6], one can announce that the infrastructure of (SVC) includes two phases; they are the Training Phase and the Testing Phase [21, 22]. Depending on the set of trained data, the Training Phase interested in building a Learning Model (i.e. Cognizable Model) to the classifier separation capability. The Learning Model is a mathematical model which was built from a set of sequential processes; they are (Mapping Vectors, Determining Support Vectors, Determining (MMS), and Settling (OSH)) [6]. Relying on that Learned model, the Testing Phase can simply recognize and predict the target values of the unseen data mapping them (in the vector form) to the classifier area as new evidences [3, 9, 23, 24].

Focusing on the Learning model processes, we can declare the following notes:

- The first process concerned with mapping a given [n x n] matrix of instances (i.e. patterns) into two-dimensional graph [11, 25, 26]. These instances should be given as set of (x_n, y_n) spatial Vectors [27], such that each (x) points to a real number that represents the mean value of data instance Vector (array of pattern attributes), while (y) points to the Vector target value for classification (estimation locality) [28]. Figure (2 – a) provides a conceptual example to instances formulation, while figure (2 – b) shows an example of the mapping process to these instances.

- When you reach the second process that is, determining Support Vectors, you will face an intuitively simple problem but appeals as a knotty one. The next section will touch on that problem with elaboration. See figure (2 – c).
- Immediately after determining Support Vectors, two independent sub-processes proceed simultaneously finalizing the classifier shape. They are [6,10] :
  - Two parallel lines will arise and passing through the determined Support Vectors (from both classes) in order to construct the (MMS) as well as the (OSH) comes into being as a perpendicular bisector to the margin width. See figure (2 – d).
  - On acquiring those Support Vectors, which have been considered the goal of building the trustable (SVC) model, all of the remaining Vectors (founded in the model) should be vanished via the second process [9]. See figure (2 – e).
- Hence, any given instance with known (x_i); beyond doubt the Testing Phase can simply recognize and predict its value (y_i) mapping it (in the vector form) to the suitable class as new evidence.

5 .Detecting Support Vectors problem:

Understanding where the Support Vectors are located with respect to the overall vectors dataset and the kind of decision surface they induce; definitely provide a substantial insight into the classifier model [9]. When you pore over again to figure (1 – b), you can realize the following facts:

1st: Support Vectors are those data points that lie ***on*** (or ***near***) the class boundaries [29]; specifically, they lie ***on*** the margin edges [30].

2nd: Since they determine the direction of Separating Hyperplane [3, 29, 31]; then all Support Vectors were found in both classes, should interface each other [17].

At the first sight, the following consideration may come across one's mind; that is to say, those Vectors could be detected by the processes of the determination of class vertices binding them with straight lines to obtain a convex hall that contains the set of all Vectors that delineate the class boundary including Support Vectors. Nevertheless, this process does not grant any guarantee to decide which of these Vectors represents the Support Vectors, because there are many Vectors lying down on the classe's boundaries interfacing each other, that could be nominated to be as Support Vectors. Moreover, a huge amount of purposeless Separating Hyperplanes will arise even if Support Vectors were detected. Figure (3) shows a critical case that declares our assumption. Hence, if we believed in the mentioned facts, that means believing in another enclosing paradigm from which one can exactly determine those Vectors.

In their (Enclosing Machine Learning Concept), X. Wei et. al. [20] they used an Ellipsoid Space to Enclose similar instances (i.e. training samples) enabling the learning machine to “cognize” and “recognize” these instances regardless of the other classes. This point is important to assure that all the existing classes have precisely been learned [20]. Depending on the mentioned concept, they introduce a One Class Support Vector Classifier (OCSVC) using the mentioned Ellipsoid Space that encloses all the target class samples and excludes all outliers. However, their scheme did not depend on detecting Support Vectors; instead, they depend on what they called (Border Support Vectors) that comprises all Vectors relying on the Ellipsoid Space boundary. This border was considered as a standpoint to bring into being an ellipsoid margin of separating space that satisfies a single class [32]. Figure (4) declares the (OCSVC) constructional phases.

The spontaneous question that could be emerged; which is "Why they did not try to use that Ellipsoid Space to enclose both classes detecting all Support Vectors in order to find (OSH)?". The valid answer for that reasonable question has been declared in figure (5) that shows the overlap Ellipsoid Spaces in such cases. The overlap occurs due to the symmetrical property of Ellipsoid Space that leads to comprise all the spaces with or without instances.

For that reason, an exigent necessity appears for providing a solid recognition technique that enables multi-classes to be enclosed at the same time without overlaps, as well as Support Vectors should be detected indisputably, (SVC) should be constructed easily as we will see in the next section.

6 . The Suggested Algorithm:

This section exhibits our novel approach that simulates the classifier task. We shall declare our analytical point of view which show a new generation learning mechanisms used to learn the computer machine how to cognize the well-known facts enveloping them with the most optimistic bow-class analogous to the facts as they distributed avoiding overlapping with the other class. Depending on the deep-seated analytical view, we shall touch on a simple comprehensible algorithm to construct (SVC), not by reducing the set of support vectors but by capturing a unique Vector from each class of training samples. From that Vector the desirable (SVC) will arise.

6 – 1. Analytical view:

The main concern of the suggested algorithm is how to provide a proper environment to the instances (i.e. Vectors) so that the classifier machine can do its tasks easily. In other words, how to enclose all instances, found in the both classes, with two spaces under the following restricted conditions:

1) Each enclosing space should reflect the most optimistic shape analogous to the instances distribution.
2) These enclosed spaces must not overlap whatever their shape cases may be.

Therefore, if we score that goal; then we will get solid closed spaces that comprise Support Vectors ***on*** their boundaries or ***nearest*** to them. To remove the confusion of the words (***on*** and ***nearest***) we need rigorous tools from which a clear judgment about Support Vectors can be acquired.

The Strong Separation Hyperplane (SSH) is the first comprehensive tool that can provide a Strong Separation Space (SSS) between these Closed Spaces [33]. Please see figure (6 - a). (SSS) can be determined easily throughout preceding the second tool, presented by releasing two Hyperplanes from both sides of (SSH) going ahead to touch these closed spaces. These Hyperplanes will be defined as Supporting Hyperplanes and the space between them will represent the (SSS) [33]. Please see figure (6 - b).

Since each Supporting Hyperplane owns a tangent point with the boundary of each closed space [33], then one can do a simple check to detect the desired Support Vector if it lies ***on*** the boundary or ***nearest*** to it. So, if the tangent point is a Vector belonging to the set of training Vectors rather than verifying the Supporting Hyperplane equation then it will represent the desired Support Vector that lie ***on*** the boundary; else, the ***nearest*** Vector to the Supporting Hyperplane will represent the desired Support Vector. Whatever the case be; we will gain a unique Support Vector from both classes interfacing each other. The captured Support Vectors is called to be the Golden Support Vectors, which is sufficient to determine the edges of (MSS) [34]. Then the edges of (MSS) are those lines that contain these Golden Support Vectors rather they are parallel to the Supporting Hyperplanes (in the case of ***nearest***), or could represent the Supporting Hyperplanes themselves, (in the case of ***on***).

Determining (MMS) allows (OSH) constructional process to proceed easily; that is the (OSH) will represent the perpendicular bisector of the distance between (MMS) edges. Surely, (OSH) will be parallel to (SSH) or could represent (SSH) itself.

Finally, we will get a strong structure that represents the (SVC) that offers best generalization ability not only to the trained data but also to guarantee high predictive accuracy for future data from the same distribution.

6 – 2. Structural View:

Based on the analytical view, we shall set forward a trustable algorithm to build (SVC). In our algorithmic scheme, we shall use a set of the following tools: (Open Disk, Lines, Euclidean distance, Lagrange Interpolation Polynomial, Seed Filling Algorithm, Separating Hyperplane, and two Supporting Hyperplanes).

Our algorithm can be presented as a set which contains the following sequential phases: (Mapping Vectors, Vector Cognition, Class Enclosing, Classes Recognition, Detecting Golden Support Vector, Determining MMS, Locating OSH, and Declaring SVC).

A. The Mapping Phase:

As what was mentioned in section (3–1) the first phase, that initiates our approach, is to map all instances in a Spatial Graph of Vectors that is dependent on instance's indicators [27].

B. The Vectors Cognition Phase:

In this phase, we shall use the mathematical (Closed-Disk) [27] that represents a simple trustable tool to cognize the Vector class type. The Disk starts a scanning process to the mapped graph row by row so that we can capture and record a single type of Vectors according to their indicator. Certainly, this process should be repeated to cognize the other type of Vectors. See figure (7 – a, b, c) please.

C. The Class Enclosing Phase:

Referring to the overlapping weak point on using Ellipsoid Spaces to Enclose Multi-Classes at a time, in this phase we shall enclose both classes using Hybrid Enclosing Mechanism (HEM) so that we can present the most optimistic shapes analogous to the instances distribution without overlap whatever their enclosing cases are.

Looking closely at figure (8), we can show our (HEM) in the following routine:

{Building HEM}

***Input*:** Two sets of spatial identical Vectors

***Output*:** Two sets of Enclosed with the most optimistic shapes analogous to the vectors distribution without overlap.

For each set of identical Vectors ***Do***

Begin

1) Score the two Vectors that own the longest distance. See figure (8 – a).
2) Draw a line between these Vectors. See figure (8 – a).
3) From both sides of this line; detect the most distant vector from that line. See figure (8 - b).
4) From both sides of this line; draw two curves in order to enclose the set of Vectors with two asymmetrical curves. Each curve consists of the end line vectors found from step (1) and the Vector found from step (3). See figure (8 - c). Hence, we will acquire a class that contains two asymmetrical portions analogous to the distribution of these mapped Vectors.
5) Now, fill both portions with a unique color. See figure (8 - d). The class, now, is totally enclosed.

End

1) The formula of scoring the Euclidean distance (D_{pp}) between the two points (a_1,b_1) and (a_2,b_2) is [34]:

$$D_{pp} = \sqrt{(a_2 - a_1)^2 + (b_2 - b_1)^2} \quad \text{……. (1)}$$

2) To draw a line segment between the two points (a_1,b_1) and (a_2,b_2) one can use the (Bresenham's Line Algorithm) [35]. The equation of the line between two points can be presented as follows [34]:

$$y = \left(\left(\frac{b_2 - b_1}{a_2 - a_2}\right) * (x - a_1)\right) + b_1 \quad \text{….. (2)}$$

3) To score the most distant vector from that line can be shown as follows: Let the line end points are A = (a_1,b_1) and B =(a_2,b_2) respectively. Then the distance (D_{PL}) between the point C = (a_3, b_3) and the line ~~AB~~ can be listed by the following equation [34]:

$$D_{PL} = \frac{|((a_2 - a_3) * b_1) + ((a3 - a_1) * b_2) + ((a_1 - a_2) * b_3)|}{\sqrt{(a_1 - a_2)^2 + (b_2 - b_1)^2}} \quad \text{….. (3)}$$

4) The simplest way to draw a curve is by using the Lagrange Interpolation polynomial (LIP) [36]. The (LIP) for three known points ((a_1, b_1), (a_2, b_2), and (a_3, b_3)) can own the following programmable formula :

$LIP_3 = k_1 * x^2 + k_2 * x + k_3$ ...... (4) [36]

Such that:

$$k_1 = \left(\frac{b1}{Q13} + \frac{b2}{Q23} + \frac{b3}{Q33}\right)$$

$$k_2 = \left(\frac{Q11 * b1}{Q13} + \frac{Q21 * b2}{Q23} + \frac{Q31 * b3}{Q33}\right)$$

$$k_3 = \left(\frac{Q12 * b1}{Q13} + \frac{Q22 * b2}{Q23} + \frac{Q32 * b3}{Q33}\right)$$

And

$Q11=(a_3 + a_2)$; $Q12=a_2 * a_3$; $Q13= (a_1^2 - Q11 * a_1 + Q12)$;

$Q21= (a_3 + a_1)$; $Q22= a_1 * a_3$; $Q23= (a_2^2 - Q21 * a_2 + Q22)$;

$Q31= (a_1 + a_2)$; $Q32= a_1 * a_2$; $Q33= (a_3^2 - Q31 * a_3 + Q32)$

5) Finally, for filling the resultant asymmetrical portions of identical classes with a unique color; we used the well-known Seed Filling algorithm (also known as Flood Fill algorithm) [37].

D. Classes Recognition Phase:

As we have seen, the former phase grants us with two solid non-overlapped spaces to be under our control. Therefore, the next step is to separate these spaces with a Strong Separating Hyperplane (SSH) that provides a strict separation space between two enclosed spaces [33]. See figure (9 – a) please. Although (SSH) does not offer the best generalization separation ability for these instances (i.e.Vectors); but it gives an initial sight view to recognize these enclosed spaces to be as enclosed classes. Moreover, (SSH) can guide us to score the goal of this research; that is detecting the Golden Support Vector from each of these classes.

(SSH) can be defined to be the perpendicular bisector of the line segment, which joins the closest points (i.e. Vectors) from both sides of classes [33]. The mathematical model of (SSH) can be shown as follows [38]:

Let *C* and *D* be two-closed set in the plane.

Let *c* and *d* represent the closest points between these two sets, such that:

$c = (x_i, y_i) \in C$ and $d = (x_j, y_j) \in D$

Then the (SSH) represents the perpendicular bisector of the line segment that joins the points (*c*) and (*d*) will own the following equation :

$$ax + by + k = 0 \quad \ldots\ldots (5)$$

such that :

$a = 2(x_i - x_j)$; $b = 2(y_i - y_j)$; $k = (x_j^2 + y_j^2) - (x_i^2 + y_i^2)$

Noticing that, the process of finding the closest points (i.e. Vectors) from both sides of classes can be accomplished by a simple iterative process as follows:

{Finding Closest Vectors}

Input: Two sets of Enclosed with the most optimistic shapes analogous to the vectors distribution without overlap.

Output : Recording the closest pair of Vectors in each identical class.

Begin

1) ***Scan*** the spatial graph to ***Detect*** and ***Record*** all points that rely on the enclosed spaces according to its indicators.
2) ***Isolate*** these points in a two sets (*A* and *B*).
3) ***For*** every point (a_i) found in (*A*) **Do**
4) ***For*** every point (b_j) found in (*B*) **Do**

Begin

4.1) *PointsDistance* = Euclidean distance $D(a_i, b_j)$

4.2) *CheckPoint* = Euclidean distance $D(a_i, b_{j+1})$

4.3) ***If*** *(PointsDistance* $\leq$ *Checkpoint)*

4.3.1) ***Then***

Begin

4.3.1.1) *MinimumDistance = PointsDistance*

4.3.1.2) *Nominated pairs are* (a_i and b_j)

End

4.3.2) ***Else***

Begin

4.3.2.1) *MinimumDistance = CheckPoint*

4.3.2.2) *Nominated pairs are* (a_i and b_{i+1})

End

End If

End

End For
End For
Closest pairs = Nominated pairs
End

E. Detecting Golden Support Vectors Phase:

Now, we can use (SSH) to detect the Golden Support Vectors by introducing the following steps:

1) From both sides of (SSH), two parallel Supporting Hyperplanes will be released in order to scan the separation space between the enclosed classes settling down on touching the classe's perimeter. See figure (9 – b). Each Supporting hyperplane will hold the following equation [34] :

   $mx - y - (mx_h - y_{h)} = 0$ …. (6)

   Such that:

   $m = (-\frac{a}{b})$ the slope of (SSH) from equation (5)

   (a and b) are equation (5) parameters

   (x_h, y_h) represents the tangent point between Supporting Hyperplane and the Enclosed Class; rather it represent the closest points to the (SSH). See the former phase.

2) We can use these Supporting Hyperplanes to detect a unique Golden Support Vector which is sufficient enough to plot the edges of (MMS). The desired goal can be scored by doing the following simple test.

{Golden Support Vector Check Point}

Input: Two sets of Enclosed with the most optimistic shapes analogous to the vectors distribution without overlap.

Output: Two Support Vectors, each one represents the Golden Support Vector of a single enclosed class.

For both classes ***Do*** the following Checkpoint

Begin

If the (Tangent point between Supporting Hyperplane and Enclosed Class) is a Vector that belong to the set of instances.

Then

Record that Vector as a ***Golden Support Vector*** (because this Vector lies ***on*** the class boundary).

Else

Move the Supporting Hyperplane inside the enclosed class carefully, recoding the first touchable Vector as a ***Golden Support Vector.*** In other words, use a loop to look for the ***nearest*** Vector to the Supporting Hyperplane then record that Vector as a ***Support Vector***. Please look at figure (9 - c).

End if

End

- We should mentioned that, the formula of detecting the distance (D) between a Vector and a Supporting Hyperplane holds the following equation [34]

  $D = \frac{|ax_1 + by_1 + c|}{\sqrt{a^2 + b^2}}$ ….. (7)

  Such that:

  (a , b , and c) are the Supporting Hyperplane parameters from equation (6)

  (x_1, y_1) represents a Vector that belong to a class.

- Also, we should notice that, there are many cases in which, there could exist more than one Vector that owns the nearest distance to the Supporting Hyperplane. In such cases, all of these Vectors will be organized in the list of ***Golden Support Vectors***. Please look at figure (9 - c):

F. Determining the space of Maximal Margin of Separation (MMS) Phase:

Detecting the set of Golden Support Vectors leads to the dawn of (MMS); that is, the new locations of Supporting Hyperplanes will represent (MMS) edges. Gaining the distance between these edges means knowing the amount of separation space.

Since all of Support Vectors lies ***on*** the sides of (MMS) edges, then one can get the needed distance by knowing the distance between one of these Vectors and the facing edge. First, we must know the equation of each edge in order to capture that distance.

- As we knew, each edge of (MMS) represents a line, which is parallel to (SSH) and determined by one (or more than one) Support Vector. Depending on these facts , the equation of the each edge will be as follows:

  $\boldsymbol{mx - y - (mx_v - y_v) = 0}$ …. (8)

  Such that:

  (m) is the slope of (SSH) from equation (5)

  (x_v, y_v) represents the location of Support vector.

- The relevant distance ($\boldsymbol{D_{MMS}}$) between a Support Vector and the facing edge will be as follows ;

  $\boldsymbol{D_{MMS}} = \frac{|ax_v + by_v + c|}{\sqrt{a^2+b^2}}$ ….. (9)

  Such that:

  (x_v, y_v) represents a Vector that belong to a class.

  (a , b , and c) are the faced edge parameters from equation (8)

G. Locating (OSH) phase:

After knowing the edges of (MMS), one can determine the (OSH) that divides the two classes into two spaces providing the best generalization ability to test the existed Vectors rather than the future Vector that needs to be tested. (OSH) represents the Hyperplane that intermediates the space of (MMS). Although we had acquired the mentioned space but we must know its equation to get the wanted equation of (OSH). The equation of (OSH) can be driven from the equation of the line that is; the perpendicular bisector of the line that represents the space of (MMS). That line can be determined using the following routine:

<u>{Finding OSH }</u>

<u>Input:</u>

- Two Golden Support Vectors each one represents the Support Vector of a single enclosed class
- Margin of Separation (MMS) edges.

<u>Output:</u> Optimum Separating Hyperplane.

<u>Begin</u>

1) Choose any of Support Vectors
2) Throughout that Vector, detect the equation of line that is orthogonal to the faced (MMS) edge.
3) Record the intersection point between these lines.
4) Get the equation of the perpendicular bisector to that orthogonal line.
5) Record that line to be the desirable (OSH).

<u>End</u>

Mathematically [34], we can illustrate the above routine as follows:

- Let (x_a, y_a) represents a Vector belong to the (MMS) edge of class A.
  Let $m_{SSH}\ x - y - (m_{SSH}\ x_b - y_b) = 0$ …. (10)
  represents the equation of the faced edge;
  such that (x_b, y_b) is a Support Vector that belongs to the faced Class B.

  (m_{SSH}) is the slope of (SSH) from equation (5).

  Then the equation of line passing through (x_a, y_a) orthogonal to the faced edge is:
  $m_{orth}\ x - y - (m_{orth}\ x_a - y_a) = 0$ …. (11)
  Such that:
  $m_{orth} = -\frac{1}{m_{SSH}}$ …. (12)
- The intersection point between these orthogonal lines can be derived to hold the following form $(x_{intersect}, y_{intersect})$:Such that

$$x_{intersect} = \left(\frac{y_a - y_b - m_{orth}\ x_a}{m_{SSH} - m_{orth}}\right);$$
$$y_{intersect} = (m_{SHH} * x_{intersect}) - (m_{orth} * x_a) + y_a \quad \text{…. (13)}$$

- ∴ (OSH) which is perpendicular bisector of the orthogonal line segment that joins the Support Vector points ($S_v = (x_a, y_a)$) and the intersection point ($P_{intersect} = (x_{intersect}, y_{intersect})$). See figure (9 - d) please.
  The programmable mode that exhibits (OSH) directly, can be presented as follow [38]:
  $Ax + By + K = 0$ …… (14)
  such that :
  $A = 2(x_a - x_{intersect})$; $B = 2(y_a - y_{intersect})$; $K = (x_{intersect}^2 + y_{intersect}^2) - (x_a^2 + y_a^2)$

H. Declaring SVC Phase:

The final phase announces (SVC) as it should be, that is; showing the (Instance Vectors including Support Vectors, Optimal Separating Hyperplane, and Maximal Margin of separation), vanishing all other details. Look at figure (10) please.

7. Discussion and Suggestions For Future Work:

Although the main purpose of this paper focused on a novel constructive technique for reducing the amount of Support Vectors to the singular Golden Support Vector; but it obviously exhibits behind the scenes a new generation of enclosing machine learning that we should stand upon; that is, the (HEM).

Like all learning mechanisms [20], (HEM) used to learn the computer machine how to cognize the well-known facts in order to enclose them in a single pattern class. Dislike all of these mechanisms [20]; the (HEM) enclosing process based on a rigorous approach to envelope all identical facts with the most optimistic bow-class analogous to the facts as they distributed avoiding overlapping with the other class. From that mechanism, one can learn the computer machine how to echo the most complex instinct learning ability that human perception can do; that is, the online eye cognition process.

Using the (HEM) mechanism enhanced with the Golden Support Vector approach allows the machine to recognize things of the same kind more easily without affecting the existing knowledge; as well as providing the ability of comparing things found in related classes. We think that our approach, that holds twofold criteria, could be applied to enhance many fields of Image Processing and Pattern Recognition such as (Cancers Cells Detection, Weather

Forecasting, Driver Fatigue observation, Targets Identification, etc.).

As we all knew, the term (SVC) represents an algorithm that exhibits classifier machines based on special parameters called Support Vectors, which steers the (OSH) [39]. As in any other algorithmic schemes, there are always pessimistic cases that lead to some weak points. In our approach, the classifier cannot accomplish its work properly (or could be halted) in the presence of the following inescapable rare cases, they are:

- The first case we called the Contiguous Case. In such case, the mentioned (HEM) produces two enclosed classes contiguous by a single border point. See figure (11 - a) please. In this situation, (SSH) cannot be determined as well as all of the classifier process will halt due to the absence of the closest enclosing points from both classes. To avoid this situation, we suggest splitting these classes into two independent classes by adapting the axes of the tangent point with a minimal micron so that to create two closest points instead of that tangent point.
- The second case we called the Conjoined Case. In such case, the (HEM) produces two overlapped-enclosed classes conjoined by many points. Look at figure (11 - b) please. This situation will reflect two Siamese Classes that cannot be classified. To avoid this situation, we suggest considering the conjoined region to be as a new independent enclosed class that owns a new instances having a new behavior. That means we will hold three different classes that needs to be separated by two (OSH) using the suggested algorithm in section (6).

8. Conclusions:

Statistical learning is a vast growing area and Vapnic's Support Vector Classifier (SVC), provides means to learn two sets of non-identical instances mapped in the form of Vectors. (SVC) presents itself as an Optimal Separating Hyperplane (OSH) that naturally partitioning the space between these Vectors into two independent classes disjoined by Maximal Margin of Separation (MMS). The superior elements of this significant classifier are those special Vectors that rest on the margin boundaries steering (OSH). These Vectors are called the Support Vectors.

Many researches were developed to solve the exigent requirement of all online learning applications represented in reducing the amount of Support Vectors in order to increase the processing speed. Among all of these researches, the Enclosing Machine Learning was aroused to facilitate the training phase throughout enclosing all identical Vectors in a single class using regular shapes such as Ellipsoid Space.

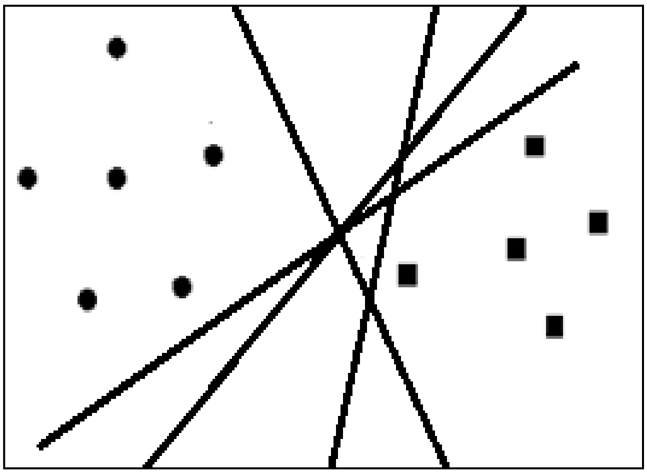
Although the Ellipsoid Space provides a substantial tool to recognize a single class but it is not suitable to Enclose Multi-Classes at a time due to its weak point that is; the overlapping cases. The overlap occurs due to the symmetrical property of Ellipsoid Space that leads to comprise spaces with or without instances.

In this research, a Hybrid Enclosing Mechanism (HEM) was exhibited to enclose both sets of Vectors in order to present the most optimistic curved shapes analogous to the instances distribution that does not overlap whatever their enclosing cases be. This mechanism is considered to be as a spring point that leads us directly to score the goal of this paper that is; presenting a simple comprehensible algorithm to construct (SVC), not by reducing the set of support vectors but by capturing a unique Vector from each class of training samples. From that Vector the desirable (SVC) will arise. We called that Vector the (Golden Support Vector).

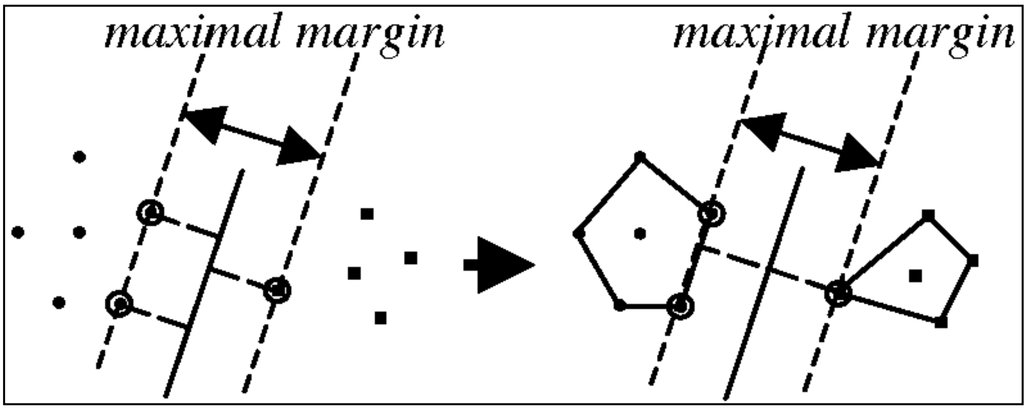
Basic mathematical knowledge is interjected in the algorithm phases. Rare inescapable cases have been discussed, provided with modest solutions as suggestions for future works. Affluent pictorial evidences have spreads to emphasize our algorithm credibility.

Figure captions

Figure No.	Caption
1	(SVC) conceptual view.
2	(SVC) Constructional Phases.
3	Difficulties of detecting Support Vectors using classe's convex hall.
4	One-Class Hyper-Ellipsoid Classifier constructional phases.
5	Enclosed Ellipsoid Spaces fail to accomplish SVC tasks.
6	Strong Separating Hyperplane and Supporting Hyperplane .
7	Cognition Phase using Closed Disk.
8	Enclosing Phase.
9	Detecting Support Vectors, providing (MMS), and Locating (OSH).
10	(SVC) in its final Phase.
11	Rare inescapable cases.



(a)



(b)

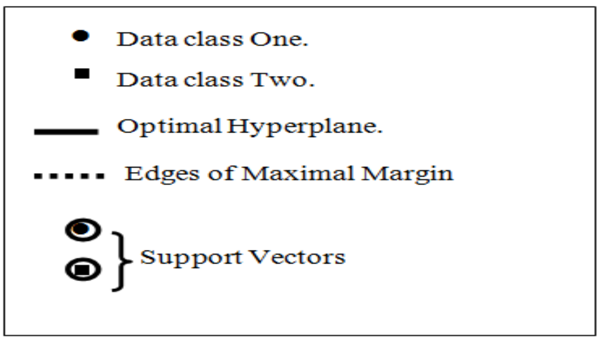


Figure (1): (SVC) conceptual view [14].
(a) : Numerous amount of Hyperplanes could separate the two classes.
(b) : The optimal Hyperplane that provide the maximal margin of separation space related with Support Vectors

Data Instances Format						
Vector no.	Feature 1	Feature 2	----	Feature m	Mean Value μ	Classification Value (Good, Bad) (1,0)
1						

N						

(a): Vectors (instances) formulation [26] (with adaptation).

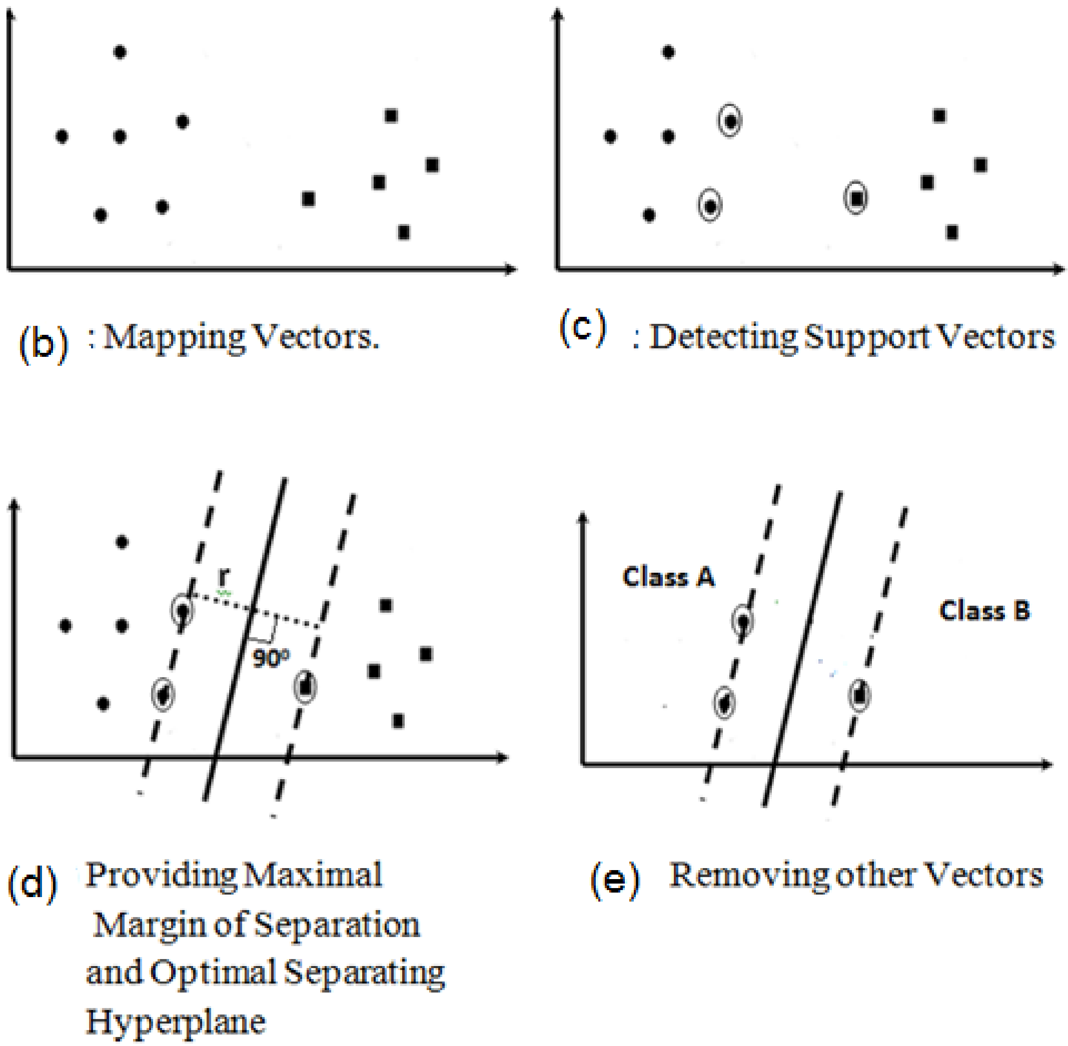
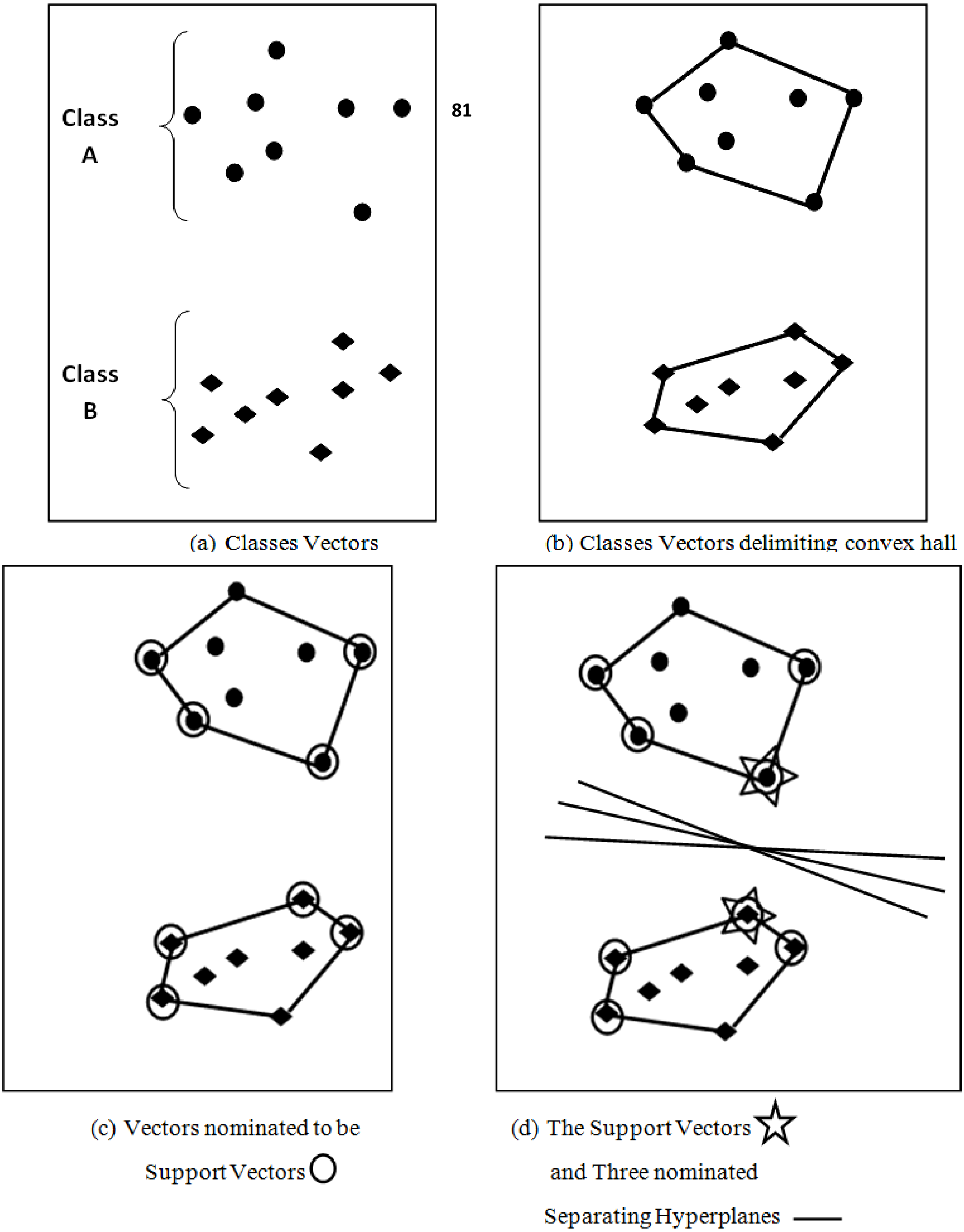


Figure (2): (SVC) Constructional Phases



(a) Classes Vectors

(b) Classes Vectors delimiting convex hall

(c) Vectors nominated to be Support Vectors ○

(d) The Support Vectors ☆ and Three nominated Separating Hyperplanes ——

Figure (3): Difficulties of detecting Support Vectors using classe's convex hall

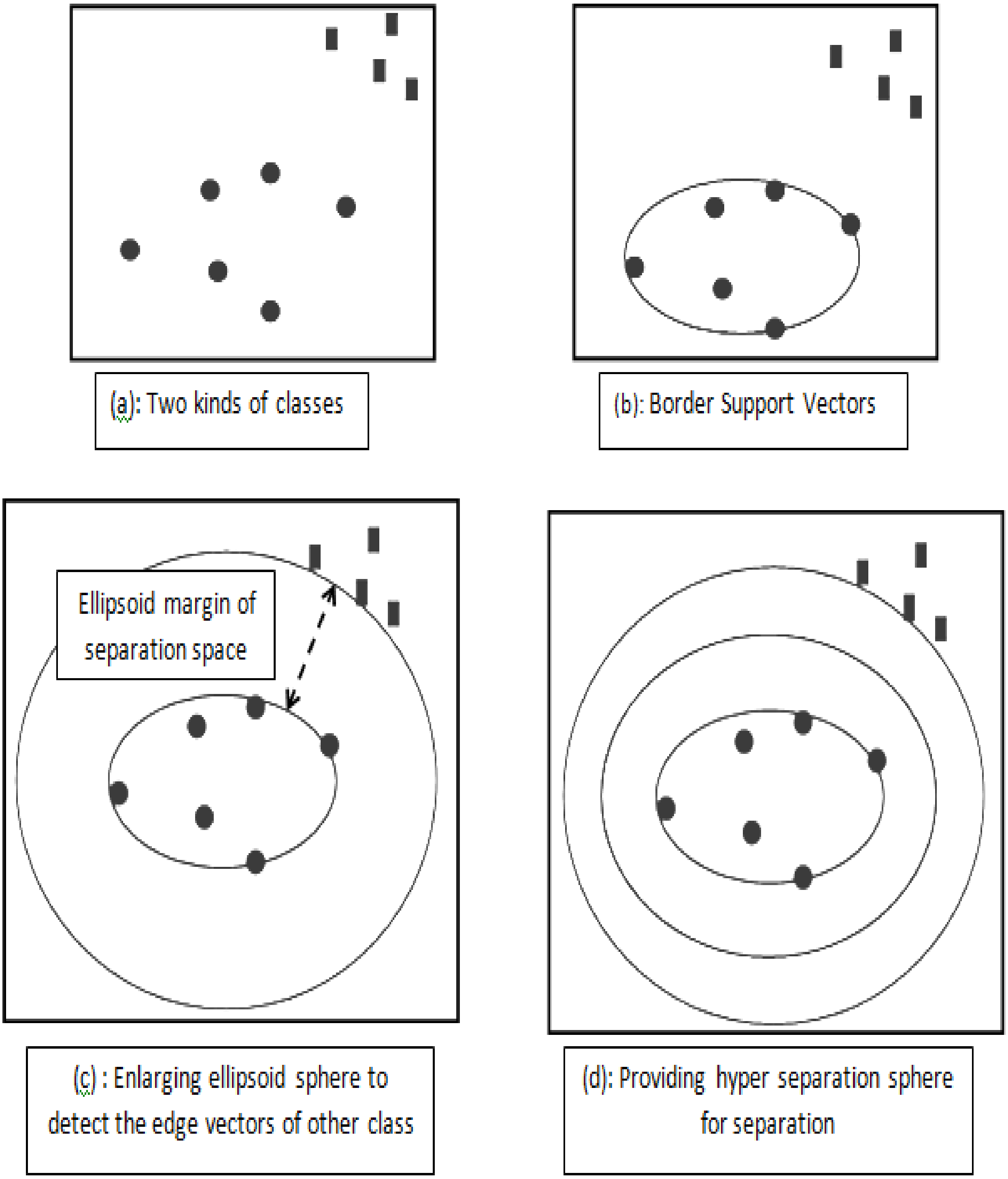
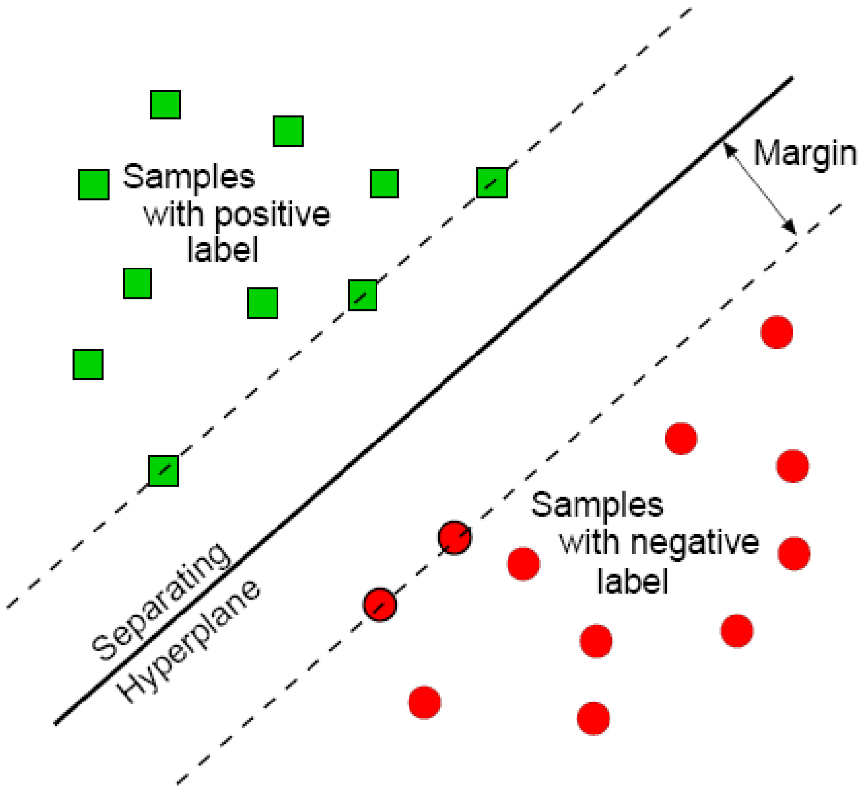
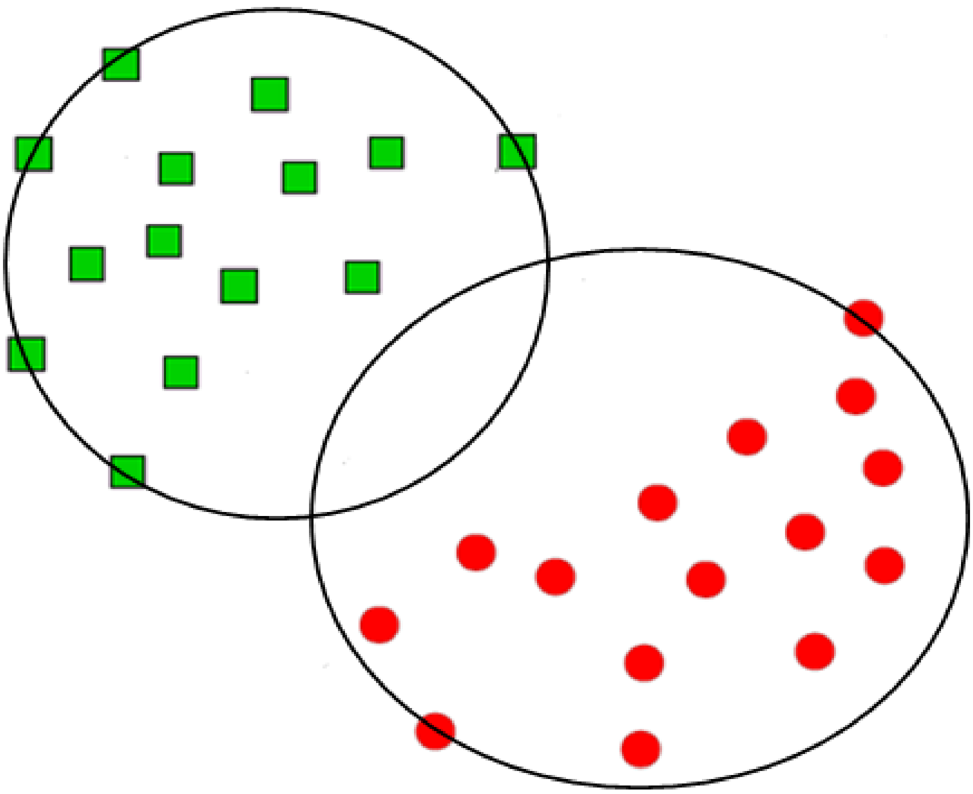


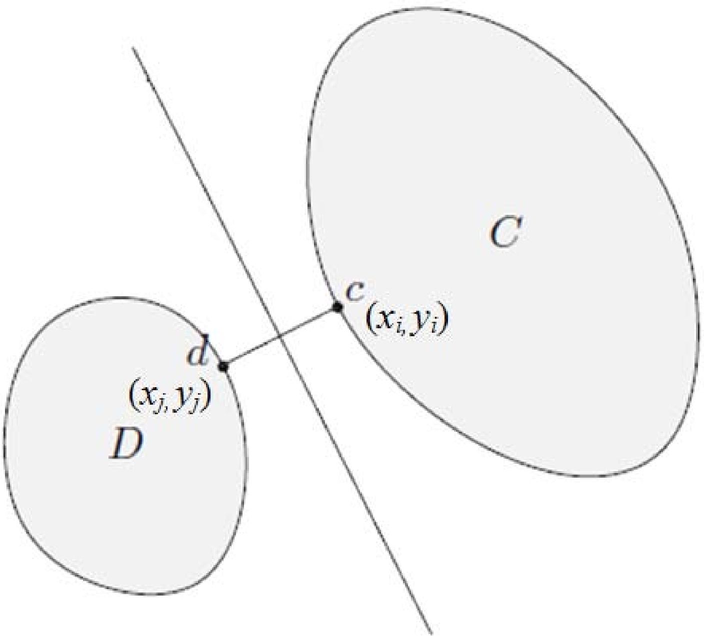
Figure (4): One-Class Hyper-Ellipsoid Classifier constructional phases [27].



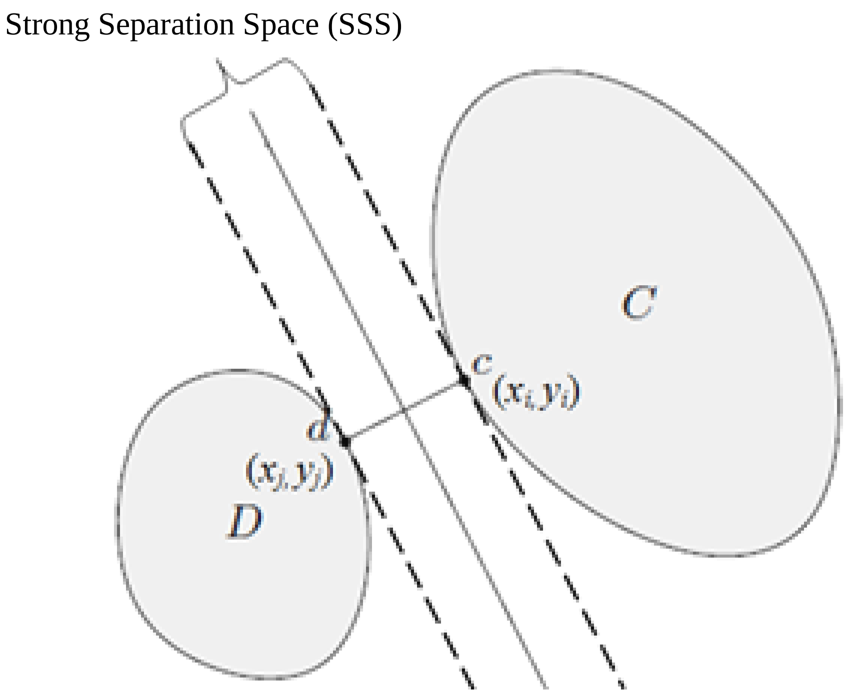
(a): Sample sketch to (SVC) [2]



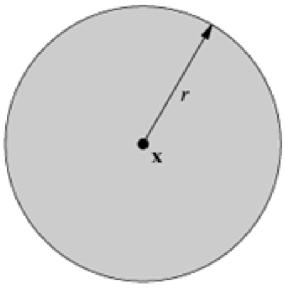
(b): Overlapped Enclosed Ellipsoid Spaces
Figure (5): Enclosed Ellipsoid Spaces fail to accomplish SVC tasks



(a) Strong Separating Hyperplane



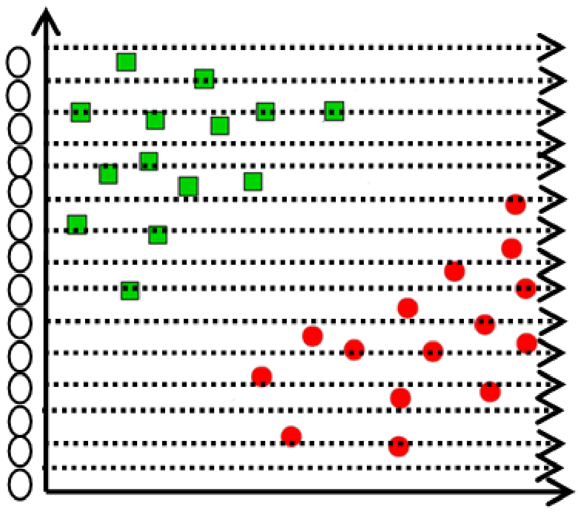
(b) Supporting Hyperplanes
Figure (6): Strong Separating Hyperplane and Supporting Hyperplane) [33].



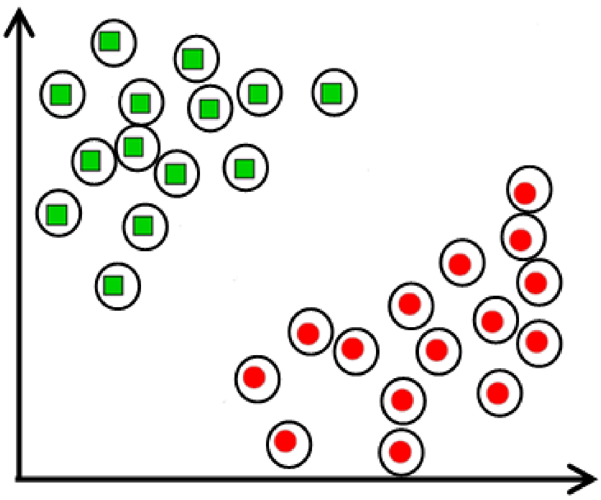
(a): Closed Disk $\overline{D}(\mathbf{x}, r)$ [27]

(x) Represent the Disk central point.

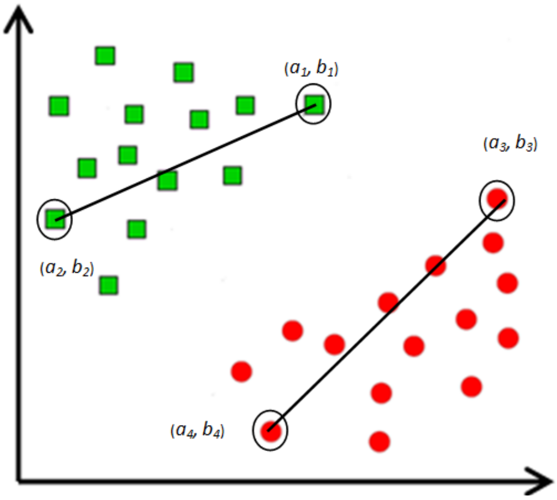
(r) Represent the Disk radius.



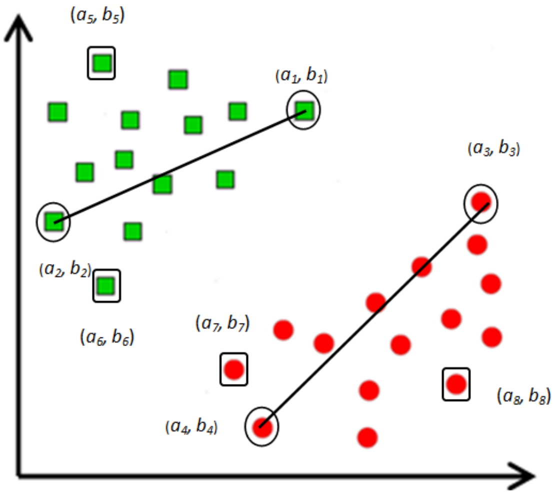
(b): Scanning process to cognize identical Vectors



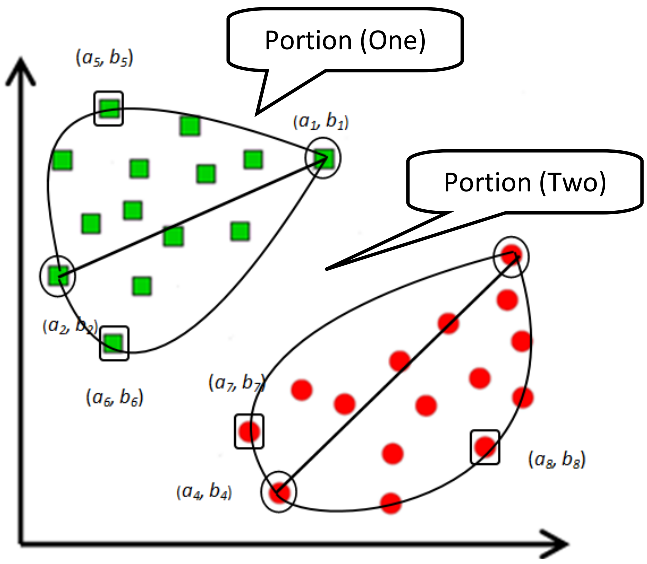
(C): cognized Vectors type
Figure (7): Cognition Phase using Closed Disk



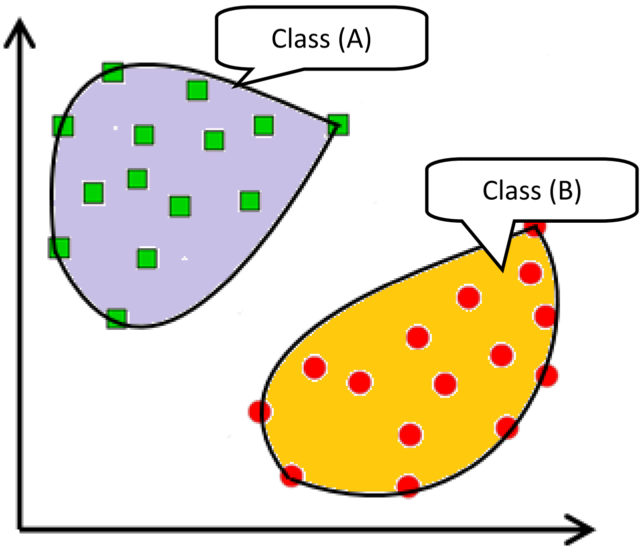
(a): Scoring the two Vectors having the longest distance; and drawing line between them



(b): detect the most distant vector from that line

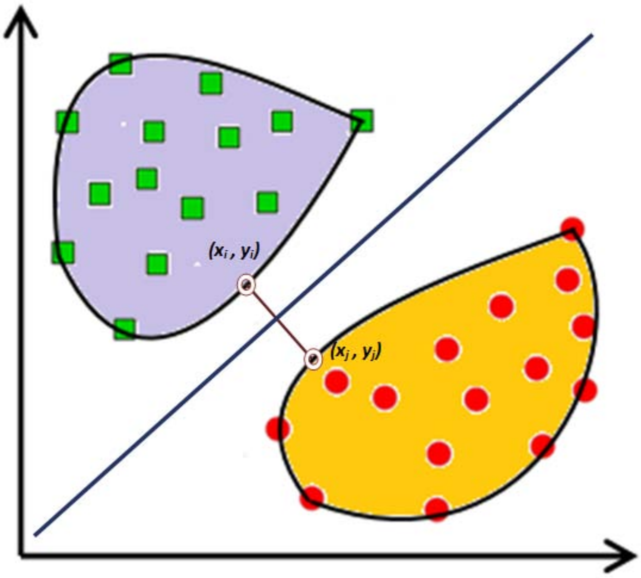


(c): Enclosing identical Vectors with asymmetric curves.

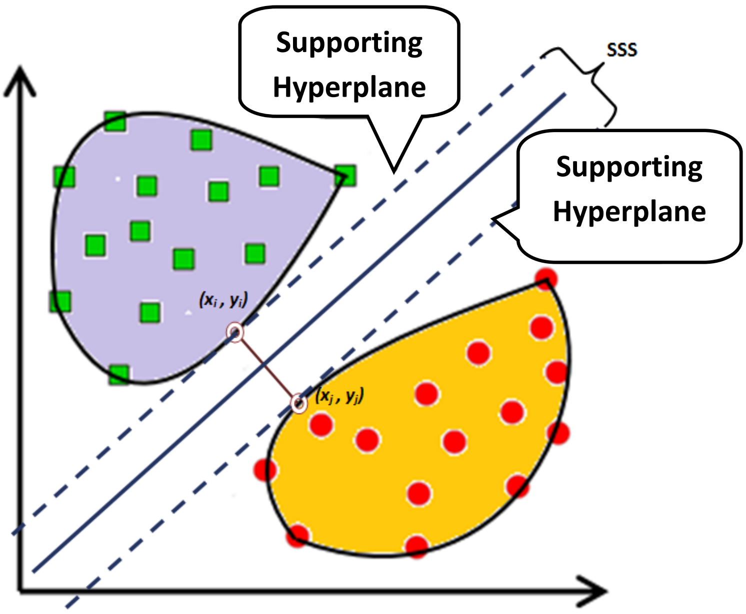


(d): Enclosing class portions with a unique color.

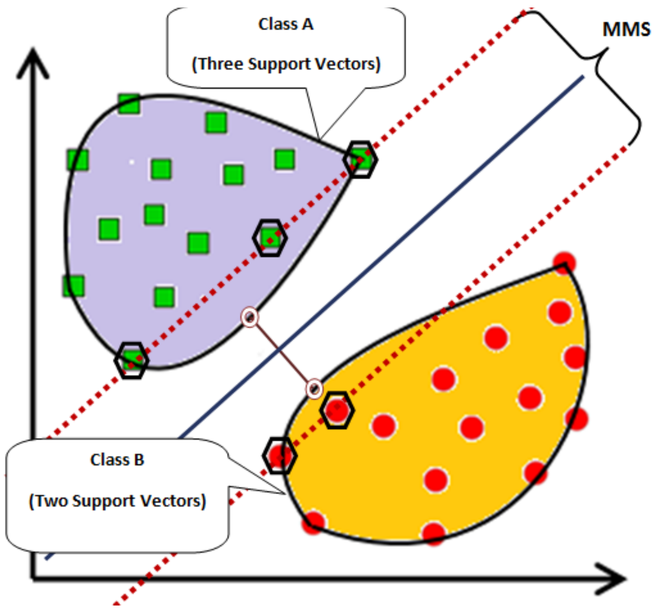
Figure (8): Enclosing Phase



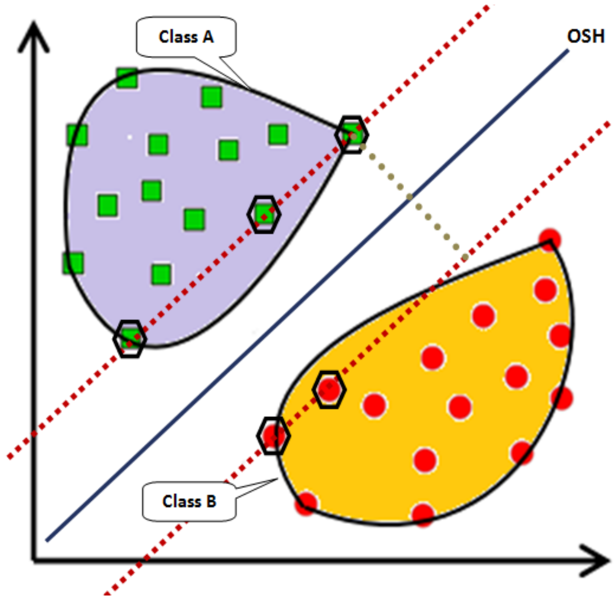
(a): Providing Strong Separation Hyperplane (SSH)



(b): Supporting Hyperplanes Provides Strong Separation Space (SSS)



(c): Detecting Support Vectors and providing (MMS)



(d): Locating (OSH)

Figure (9): Detecting Support Vectors, providing (MMS), and Locating (OSH).

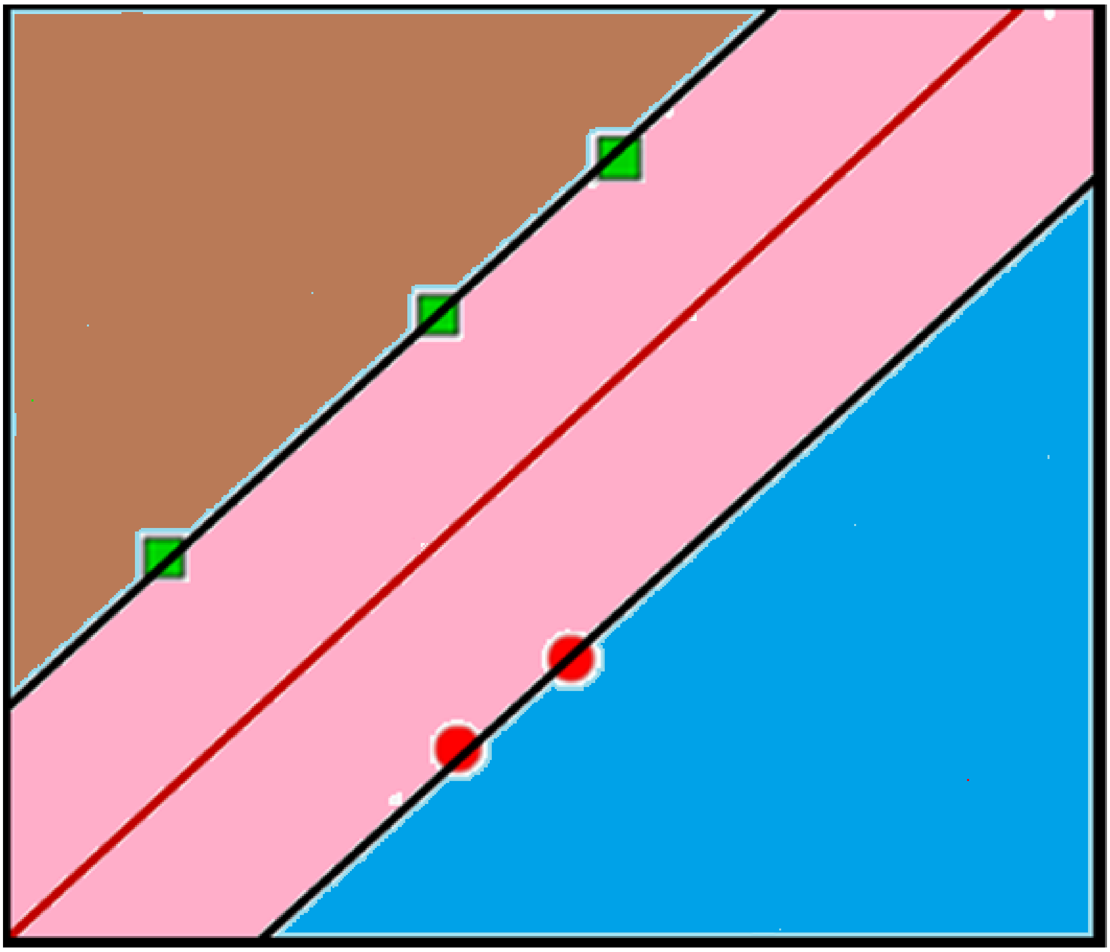
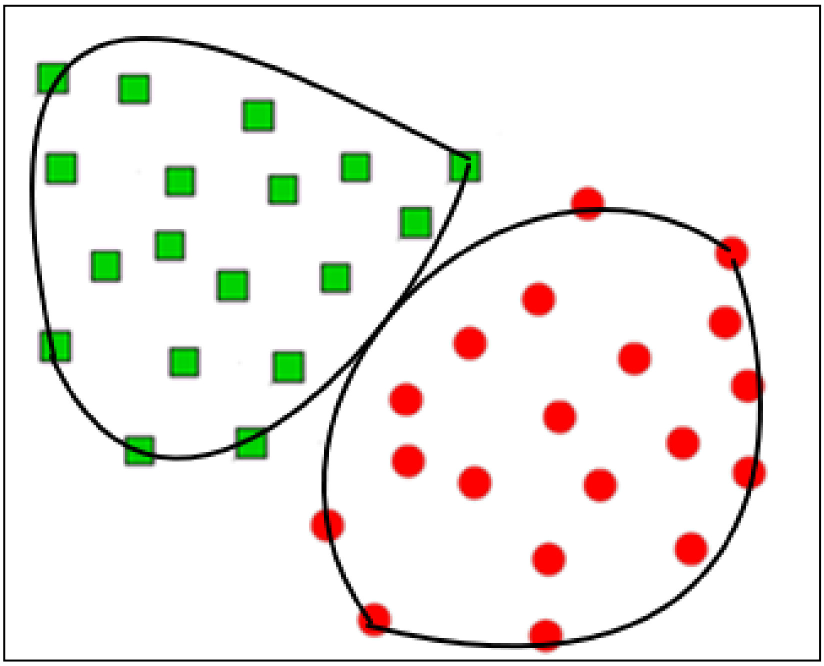
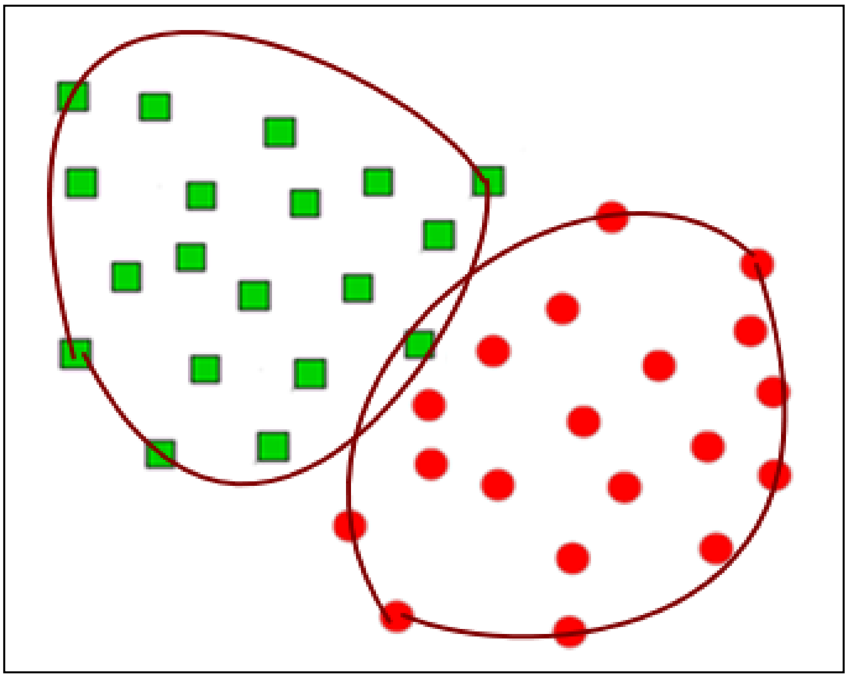


Figure (10): (SVC) in its final phase.



(a): Contiguous Case



(b): Conjoined Case

Figure (11): Rare inescapable cases

References:

[1] P. Y. Hao and J. H. Chiang, "A Fuzzy Model of Support Vector Regression Machine", International Journal of Fuzzy Systems, Vol. 9, No. 1, 2007.

[2] R. R. Ade, D. S. Medhekar, and M. P. Bote, "Heart Disease Prediction System Using SVM and Naive Bayes", International Journal of Engineering Sciences & Research Technology, pp. 1343-1348, ISSN: 2277-9655, 2013.

[3] D. Oran, "Study on Proximal Support Vector Machine as a Classifier", a Thesis of Master of Technology in Electronics and Instrumentation, Department of Electronics and Communication Engineering National Institute of Technology, Rourkela, Orissa 769008, India, 2012

[4] H. Xue, Q. Yang, and S. Chen, "Data mining and Knowledge Discovery Series - The Top Ten Algorithm in Data Mining - Support Vector Machines", pp. 37-59, Chapman & Hall Book, CRC Press Taylor & Francis Group, LLC, ISBN 978-1-4200-8964-6, USA, 2009

[5] S. Gazzah and N. B. Amara, "Neural Networks and Support Vector Machines Classifiers for Writer Identification Using Arabic Script", The International Arab Journal of Information Technology, Vol. 5, No. 1, pp. 92-101, 2008.

[6] C. Cortes and V. Vapnik, " Support-Vector Networks", Machine Learning, Vol.20, pp. 273-297, Kluwer Academic Publishers, Netherlands, 1995.

[7] R. A. S. Hassan, A.F. Hegazy, and A. A. Badr, "Optimize Support Vector Machine Classifier based on Evolutionary Algorithm for Breast Cancer Diagnosis", International Journal of Computer Science and Network Security, VOL.10 No.12, pp. 85-90, 2010.

[8] M. Aghamohammadi and M. Analoui, " Multi-Level Support Vector Machine", World of Computer Science and Information Technology Journal, Vol. 2, No. 5, pp. 174-178, ISSN: 2221-0741, 2012

[9] L. Hamel, "Visualization of Support Vector Machines with Unsupervised Learning", IEEE 1-4244-0623-4/06, 2006.

[10] S. S. Keerthi, O. Chapelle, and D. DeCoste, "Building Support Vector Machines with Reduced Classifier Complexity", Journal of Machine Learning Research, Vol.7, pp.1493–1515, 2006.

[11] S. Gunn," Support Vector Machines for Classification and Regression", ISIS Technical Report, Image Speech and Intelligent Systems Group, University of Southampton, 1998.

[12] S. Tong and D. Koller, "Support Vector Machine Active Learning with Applications to Text Classification", Journal of Machine Learning Research, pp. 45-66, 2001.

[13] D. Huson, "Support Vector Machines", Grundlagen der Bioinformatik, SS'09, pp. 73-80, 2009.

[14] J. Wang, "Support vector machines based on K-means Clustering for real-time business intelligence systems", Int. J. Business Intelligence and Data Mining, Vol. 1, No. 1, pp. 54-64, 2005.

[15] D. Anguita, A. Ghio, I. A. Lawal, and L. Oneto, "A Heuristic Approach to Model Selection for Online Support Vector Machines", ROKS, pp. 77-78, 2013.

[16] B. Debojit, J. Hitesh, A. Manoj K, and R. Balasubramanian , "Study and Implementation of a Non-Linear Support Vector Machine Classifier", International Journal of Earth Sciences and Engineering, , Volume 04, No 06 SPL, ISSN 0974-5904, pp. 985-988, 2011.

[17] C.F. Lin and S.D. Wang. "Fuzzy support vector machines", Journal of IEEE Trans Neural Networks, 13(2), pp. 464-471, 2002.

[18] D. Nguyen and T. Ho, "An Efficient Method for Simplifying Support Vector Machines ", Proceedings of the 22nd International Conference on Machine Learning, Bonn, Germany, 2005.

[19] C. Fatichah and M. R. Widyanto, "Active Learning Support Vector Machine for Reducing Support Vectors", International Conference on Instrumentation, Communication and Information Technology, pp. 290-293, Bandung, Indonesia, 2007.

[20] X. Wei, "Enclosing Machine Learning - Encyclopedia of Data Warehousing and Mining, Second Edition", IGI Global, ch115, pp. 744-750, DOI: 10.4018/978-1-60566-010-3, USA, 2009.

[21] H. S. Yazd and S. Effati, "Vehicle Type Recognition Using Probabilistic Constraint Support Vector Machine", First Joint Congress on Fuzzy and Intelligent Systems, Ferdowsi University of Mashhad, Iran, 2007.

[22] A. Oliyaei and N. Salmasi, "An Efficient Hybrid Algorithm For Multi-Class Support Vector Machines", International Conference on Computer Science and Information Technology, 2011.

[23] H. Bhavsar and M. H. Panchal, "A Review on Support Vector Machine for Data
Classification", International Journal of Advanced Research in Computer Engineering & Technology, Volume 1, Issue 10, pp. 185-189, ISSN: 2278 – 1323, 2012.

[24] O. Ivanciuc, "Applications of Support Vector Machines in Chemistry", Review in Computational Chemistry, Vol. 23, Wiley-VCH, Weinheim, pp. 291-400, 2007.

[25] S.C. Chen, Y.L. Zhu, D.Q. Zhang, and J. Yang, "Feature extraction approaches based on matrix pattern: MatPCA and MatFLDA", Pattern Recognition Letters 26, pp. 1157-1167, 2005.

[26] S. S. Keerthi, S. K. Shevade, C. Bhattacharyya, and K. R. K. Murthy, "A Fast Iterative Nearest Point Algorithm for Support Vector Machine Classifier Design", IEEE Transaction on Neural Network, Vol. 11, NO. 1, pp. 124-136, 2000.

[27] A. Okabe, B. Boots, K. Sugihara, and S. N. Chiu, "Spatial Tessellations", 2nd Ed., Chapter One, John Wiley & Sons, LTD, ISBN 0-471-98635-6, England, 2000.

[28] S. B. Kotsiantis, " Supervised Machine Learning: A Review of Classification Techniques", Informatica Vol. 31, pp. 249-268, 2007.

[29] M. Pal and P. M. Mather, "Support Vector classifiers for Land Cover Classification", Image Processing & Interpretation, Map India Conference, GISdevelopment.net, 2003.

[30] A. B. Hasan, T. S. Kiong, J. K. S. Paw, and A. K. Zulkifle, "Support Vector Machines Study on English Isolated-Word-Error Classification and Regression", Research Journal of Applied Sciences, Engineering and Technology 5(2): 531-537, ISSN: 2040-7459; E-ISSN: 2040-7467, Maxwell Scientific Organization, 2013.

[31] M. Papadonikolakis and C.S. Bouganis, " A Heterogeneous FPGA Architecture for Support Vector Machine Training", 18th IEEE Annual International Symposium on Field-Programmable Custom Computing Machines, pp. 211-214, 2010.

[32] X. Wei, Y. Li, D. Liu, and L. Zhan. "Mahalanobis Support Vector Machines Made Fast and Robust - New Advances in Machine Learning", Yagang Zhang (Ed.), pp. 227-250, InTech, ISBN: 978-953-307-034-6, China, 2010.

[33] S. Boyd and L. Vandenberghe, "Convex Optimization", Cambridge University Press, ISBN: 0 521 83378 7, UK, 2004.

[34] G. B. Thomas, M. D. Weir, J. R. Hass, F. R. Giordano, "Thomas' Calculus", 11th Ed., Pearson, Inc, Publishing as Pearson Addison-Wesley, ISBN: 9780321198006, 2005.

[35] H. Sutopo, "Bresenham's Lines Algorithm Visualization Using Flash", International Journal of Computer Theory and Engineering, Vol. 3, No. 3, pp. 422 – 426 , 2011

[36] H. S. Ridha, "A Plain Algorithm Exhibits the Benefit of Lagrange Interpolation Polynomial to Fashion a Best Fit Proctor Curve Drawing Out its Upshot", the 11th Annual Scientific Conference of the University of Babylon, 2009.

[37] M. S. H. Khayal, A. Khan, S. Bashir, F. H. Khan, and S. Aslam, " Modified New Algorithm For Seed Filling ", Journal of Theoretical and Applied Information Technology, Vol. 26, No.1, Publication of Little Lion Scientific R &D, ISSN: 1992-8645, E-ISSN: 187-3195, pp. 28-32, Pakistan, 2011.

[38] H. S. Ridha, "A Flexible Algorithm For Constructing Two-Dimensional Voronoi Diagram", Thesis of Master of Science in Computer Science, Department of Computer Science, College of Science, University of Basrah, Iraq, 2001.

[39] I. H. Witten and E. Frank, "Data Mining - Practical Machine Learning Tools and Techniques", 2nd

إنشاء مصّنِف الموجِهات الساندة بالإعتماد على الموجِه الساند الذهبي

هشام صالح رضا
جامعة البصرة – كلية طب الأسنان – فرع العلوم الأساسية
العراق – البصرة
zhesham60@hotmail.com

الخلاصة

من اجل زيادة سرعة المعالجة لتطبيقات التلقين الآني المتمثِلة بمطلبها الملحَ لتقليل عدد الموجٍهات الساندة؛ كرِس هذا البحث لإظهار خوارزمية متينة لإنشاء مصّنِف الموجِهات الساندة الذائع الصيت، وذلك من خلال اغتنام موجٍه داعِم واحد من كل نوع من انواع الشواهد المدرّبة. اطلقنا على ذلك الموجِه اسم (الموجِه الساند الذهبي).

تبنّت خوارزميتنا وسائل رياضية اساسية في اطوارها الإنشائية. تبدأ الخوارزمية بتطبيق ميكانكيتها الهجينة للتطويق من اجل تطويق مجموعتين من الشواهد (الموجِهات) المرسومة بأفضل الفضاءات المثالية المماثلة لتوزيع تلك الشواهد والتي لاتتداخل مع بعضها بعضاً. تعتبر هذه الميكانيكية كنقطة انطلاق لأرشادنا مباشرة الى فصل تلك الفضاءات بمستوي فاصل متين. من جانبي هذا المستوي المتين سوف تطلق فضاءات ساندة لتستقر عند اول موجٍه مكتشف؛ الذي اطلقنا عليه اسم الموجه الساند الذهبي. يعتبر كل فضاء ساند مع موجهه الساند المحرز القاعدة الأساسية لبناء الحد الأكبر للفضاء المفصول؛ والذي له القدرة على توفير العمومية ليس فقط للشواهد المدربّة، بل للشواهد المستقبلية التي لها نفس توزيع الشواهد المدّربة. اخيراً سيتوسط الفضاء المنصف المثالي ذلك الحد الأكبر ليحذف كل الشواهد المتبقية.

تمت مناقشة حالات نادرة لامفرَ من حدوثها مدعومة بحلول متواضعة كمقترحات للأعمال المستقبلية. تم عرض اشكال مصورة وافرة للتأكيد على مصداقية خوارزميتنا.

كلمات مفتاحية : مكائن الموجِهات الساندة، مصّنِف الحد الأكبر، تدريب الماكنة، التدريب الأحصائي، التنقيب عن البيانات.