Visualization of Multidimensional Data Using a Support Vector Machine

Anatoly P. Nemirko
Department of biotechnical systems
Saint Petersburg Electrotechnical University "LETI"
St. Petersburg, Russia
apn-bs@yandex.ru

***Abstract*— A visualization-related problem is considered during linear analysis of two classes in multidimensional feature space. Coordinate linear transformation results in finding two orthogonal axes. When the classes are projected on them, they are the furthest from each other. Proximity of the classes is assessed based on the minimum distance criterion between their convex hulls. Such criterion allows depiction of cases for complete separability and random outliers. A support vector machine is applied to obtain orthogonal vectors of reduced space, which ensures that a weight vector is obtained determining the minimum distance between the convex class hulls for the classes separable linearly. Algorithms having reduction, compression and displacement of convex hulls are utilized for intersecting classes. Experimental studies are dedicated to application of the studied visualization machines to analyze biomedical data.**

Keywords*— *multidimensional data visualization; machine learning; support vector machinse; biomedical data analysis

I. INTRODUCTION

Despite the rapid development of the neural network approach to image recognition, there is still a vast application domain, for which class description is characterized by multidimensional feature space and the search for solutions in this space. There are many of these problems, especially in biology and healthcare. To resolve these problems, it is important for the researcher to know the extent for class intersection, and to try building the best separating surface. When operating in multidimensional space, the class intersection area is invisible, and the decisive rules are built based on some theoretical hypotheses. However, there are frequent cases, when a detailed study of the intersection area is especially important, for instance, in the cases of the high cost of diagnostics errors, or when detecting outlier points that fail to fall within the description of some biological species. The problem arises of satisfactory display of the intersection area in 2-dimensioned space. This issue may be named as visualization of classes on a plane.

Statistical methods for dimension reduction and visualization are usually applied for this purpose: principal component analysis (PCA) [1], and method of mapping to a plane [2, 3, 4] based on the Fisher's linear discriminant (FDA) [5]. Unfortunately, these methods fail to yield a comprehensive picture for the class intersection area and to ensure displaying of cases for complete separability or random outliers [2].

When projecting multidimensional classes onto a plane, errors occur due to information losses, which are manifest as the occurrence of additional experimental points in the class intersection area. The task here is to find a projection of the classes on the plane, where the number of experimental points that fall within the intersection area on the plane would be the same as in multidimensional space (their smaller number is impossible).

In this paper, class intersection is viewed as the intersection of their convex hulls. Thus, the class intersection area is viewed as the intersection area for their convex hulls both in multidimensional space, and on a plane. The minimum distance between their convex hulls is viewed as the space transformation criterion. Then, the visualization problem is formulated as follows. Find that dimensionality subspace 2, in the orthogonal projections onto which the distance between convex hulls of classes would be the minimum.

II. THE USE OF RECOGNITION PROCEDURES

Assume $\mathbf{x}_i, i = 1, 2, ..., N$ are the vectors in n-dimensional feature space of learning set X. They belong to one of two classes ω_1, ω_2. The linear recognition problem is to find the hyperplane $g(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + w_0 = 0$, which would optimally classify all the vectors related to the learning set, where $\mathbf{w}$ is the weight vector, w_0 is the scalar threshold value. Here, $\mathbf{w}$ is sought so that the hyperplane $g(\mathbf{x})$, which is perpendicular to $\mathbf{w}$ would separate the classes the best way (in terms of the learning criterion). Therefore, it may be considered as the first feature for visualization classes on a plane. The second feature is sought on the plane perpendicular to $\mathbf{w}$ for the learning criterion (the same or another). Next, the recognition procedure for support vector machines is mainly considered [6, 14]. It is known that in case of non-intersecting classes, the support vector machine (SVM) calculates the minimum distance between the convex hulls of classes and its corresponding weight vector $\mathbf{w}$ [15].

For the SVM method, with linearly separable classes, the problem of finding the optimum hyperplane is formulated as

This research was supported by RFBR, research projects Nos 18-07-00264 and 16-01-00159

978-1-5386-5824-6/18/$31.00 ©2018 IEEE

$$\begin{cases} \|\mathbf{w}\|^2 \to \min \\ y_i\left(\mathbf{w}^T\mathbf{x}_i + w_0\right) \geq 1, \quad i = 1,2,...,N \end{cases}, \qquad (1)$$

where y_i – class indicator for each $\mathbf{x}_i$ equal to $+1$ for ω_1 and to -1 for ω_2. This is a problem of convex quadratic programming (for $\mathbf{w}, w_0$) in a convex set with consideration of the totality of linear inequalities. Its solution looks like

$$\mathbf{w} = \sum_{i=1}^{N} \lambda_i y_i \mathbf{x}_i \quad ,$$
$$w_0 = \mathbf{w}^T\mathbf{x}_i - y_i, \quad \lambda_i > 0$$

where λ_i - Lagrange multipliers.

Generally the problem of finding the minimum distance between convex hulls of classes (NPP - nearest point problem) is solved also by other algorithms: SK-algorithm (Schlesinger–Kozinec algorithm) [7], MDM-algorithm (Mitchell–Demyanov–Malozemov algorithm) [8]. These algorithms may also be utilized to obtain weight vector for class visualization.

III. Case of Linearly Inseparable Classes

In case of linearly inseparable classes A and B, instead of criterion for the minimum distance between their convex hulls Conv(A) and Conv(B), one can use the minimum penetration depth criterion, or criterion for mutual intersection of classes *D*, which is utilized in collision detection problems [9, 16]. This criterion is defined as the minimum value, by which B should be displaced in any direction to avoid intersection between Conv(A) and B. Finding such direction yields the required vector $\mathbf{w}$. Following the same strategy, one can conclude that in case of linearly inseparable classes, the distance between them may be defined as the minimum distance, to which Conv(A) should be displaced in any direction in order for A and B to intersect (Fig. 1).

There are algorithms for finding the penetration depth for 2D and 3D cases [9, 16], but it is difficult to find the penetration depth for the *n*D cases. This problem is significantly simplified if the projections of convex hulls (or the classes themselves) are considered on some direction in multidimensional space. In more general case, the minimum distance between the convex hulls and the penetration depth of one hull to another at their intersection may be found by trying various directions in multidimensional space and investigating the extreme points of the class projections on these directions.

Assume that A' and B' are class projections on some direction in multidimensional space. Then, by excluding the confluent option for complete inclusion of one set into another, the proximity degree (including intersection degree) *D* may be defined as the following procedure:

$a1 = \min(A')$; $a2 = \max(A')$;
$b1 = \min(B')$; $b2 = \max(B')$;
if $a1 < b1$
  $D = a2 - b1$;
else
  $D = b2 - a1$;
end

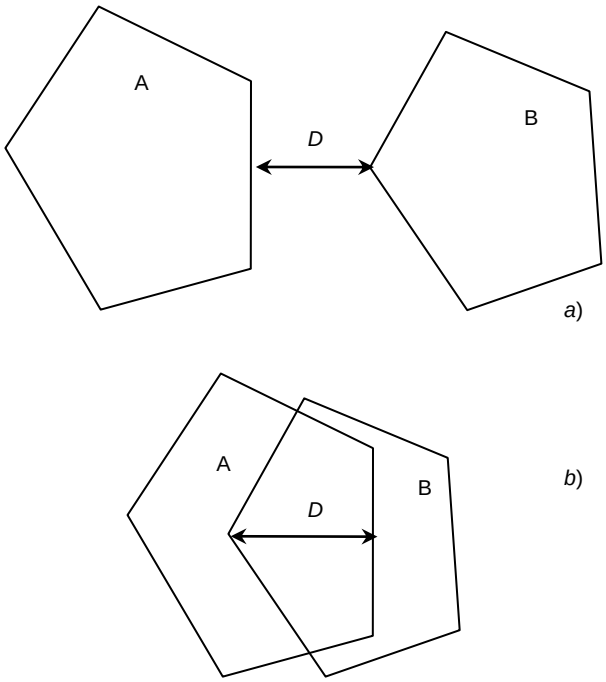

Fig. 1. Determining the proximity for two convex hulls to each other: a) *D* – minimum distance between A and B; b) *D* – minimum penetration depth between A and B

If the sets intersect, *D* will have a negative sign. To find the minimum value for *D* for two classes, that direction $\mathbf{w}$ needs to be found in multidimensional space, for which *D* would the minimum.

Many classification methods solve the problem of inseparable classes using either minimizing classification errors or involving procedures for minimizing such errors. The resulting weight vector generally fails to coincide with the penetration depth vector required to obtain a visual image on a plane.

In the SVM, this problem is solved by minimizing classification errors, which is also not the best solution in terms of minimizing penetration depth. For linearly inseparable classes in the SVM, expression (2) is utilized instead of (1).

$$\begin{cases} \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{N}\xi_i \to \min\limits_{\mathbf{w},w_0,\xi_i} \\ y_i\left(\mathbf{w}^T\mathbf{x}_i + w_0\right) \geq 1 - \xi_i, \quad i = 1,2,...,N \\ \xi_i \geq 0, \quad i = 1,2,...,N \end{cases}, \qquad (2)$$

where variables $\xi_i \geq 0$ reflect the error value on objects $\mathbf{x}_i$, $i = 1, 2, ..., N$, while C is the parameter settings of the method that allows you to adjust the relationship between the maximization of the width of the separating margin and the minimization of the total error. The problem is solved similarly to the problem for the case of linearly separable classes.

IV. The Use of Modified SVM Methods

Universal methods for resolving the problem regardless class intersection conditions are proposed by transforming convex hulls into reduced convex hulls (RCH) [10] and scaled convex hulls (SCH) [11], which reduces the problem to analysis of linearly separable classes. We proposed a similar procedure for the offset convex hull (OCH), which results in displacement of any element of a single class by a permanent value towards the difference vector for their centroids. The problem with separated classes is then solved, whereupon a reverse displacement is performed.

V. Experimental Studies

The class intersection degree after their mapping to the plane was assessed by the number *g* of members of the training samples of both classes falling in the intersection area, i.e., $g = (n_1 + n_2)/(N_1 + N_2)$, where n_1, n_2 is the number of points related to the 1-st and 2-nd classes falling in the convex hull intersection area; N_1, N_2 is number of members of the training sample related to the 1-st and 2-nd classes. It is obvious that $0 < g < 100$ %.

The first experiment in 4-dimensioned data visualization used 2 classes of Fisher's irises [12]: '*virginica'* and '*versicolor'*. Each class consists of 50 specimens measured by 4 attributes: length and width of sepal, length and width of petal. The following table shows class intersection results after their projection on a plane using various algorithms.

TABLE I. CLASS INTERSECTION ON A PLANE FOR VARIOUS ALGORITHMS

Algorithm	$N_1 + N_2$	n_1	n_2	*g %*
PCA	50 + 50	2	6	8
FDA	50 + 50	1	2	3
SVM	50 + 50	1	0	1

These results show that for the purposes of visualization of a 2-class problem set in multidimensional feature space, out of the three methods investigated, the best is the SVM method. It yields the minimum class intersection when they are displayed on plane. However, this method requires parameters selection in every individual case.

The second problem investigated is diagnostics of breast cancer. Data are taken from Breast Tissue [13] base. They consist of 106 specimens of breast tissue measured by 9 parameters of tissue impedance. The data are verified by 6 classes of mammary neoplasms, out of which 2 classes were selected for our experiments: breast carcinoma (malignant neoplasm) 21 cases, and mammary fibroadenoma (benign neoplasm) 15 cases. Input data were normalized by average value and dispersion. Fig. 2 shows the outcome from applying PCA algorithm to these data. Data visualization the outcome using the SVM algorithm illustrated in Fig. 3 showed their full linear separability.

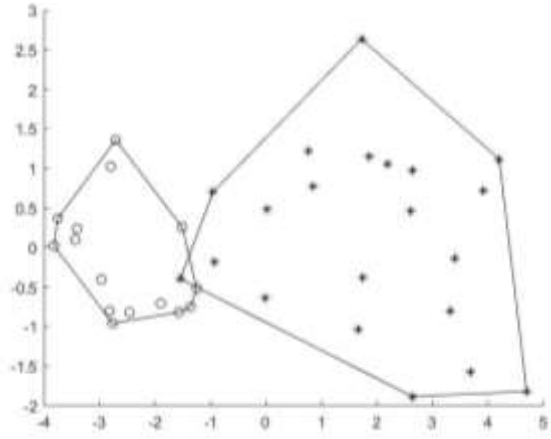

Fig. 2. Display of mammary breast neoplasm classes on a plane using the PCA method. On the left is the fibroadenoma class, the carcinoma class is on the right. The X-axis is the first weight vector, the Y- is the second one. It can be seen that the convex hulls of the classes intersect

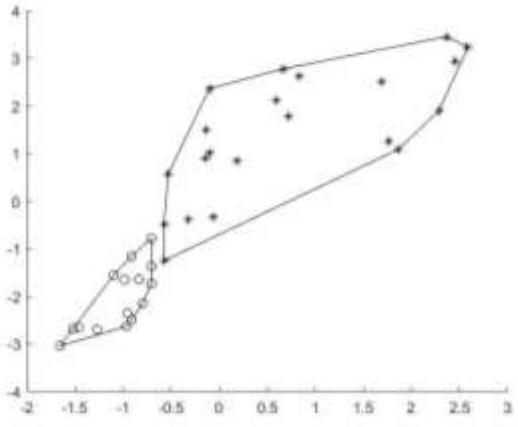

Fig. 3. Display ofmammary breast neoplasm classes on a plane using the SVM method. Mutual arrangement of classes and axes is the same as in Figure 2. Classes are completely linearly separable

VI. Conclusion

To display the class intersection area from multidimensional space onto a plane, one can use the proximity criterion for their convex hulls *D*. For non-intersecting classes, this criterion takes shape of minimizing the distance between two convex hulls. For intersecting classes, it is transformed into minimizing the degree of their mutual intersection *D*. The *D* criterion is automatically implemented when utilizing the SVM method for linearly separable classes. For linearly nonseparable classes, it is expedient to utilize the SVM method as approximate solutions with transformations of RCH, SCH and OCH types. The last one is the simplest. Instead of the SVM it is permissible here to utilize other nearest point algorithms NPP. Generally, to find the optimum *D* values, search procedures need to be utilized. Of the three display methods,

the SVM method was the best. It yielded the minimum class intersection when they were projected on the plane.

REFERENCES

[1] Jolliffe, I.T.: Principal Component Analysis. 2nd ed., New York: Springer-Verlag, 2002. 487 p.

[2] Nemirko A.P.: Transformation of feature space based on Fisher's linear discriminant. Pattern Recognition and Image Analysis, vol. 26(2), pp.:257–261 (2016).

[3] Manilo L.A., Nemirko A.P.: Recognition of biomedical signals based on their spectral description data analysis. Pattern Recognition and Image Analysis, vol. 26(4), pp. 782–788 (2016).

[4] Maszczyk T. and Duch W. Support Vector Machines for visualization and dimensionality reduction. Lecture Notes in Computer Science, Vol. 5163, 346-356, 2008.

[5] Duda R.O., Hart P.E., Stork D.G.: Pattern Classification. New York: Wiley, 2001. 659 p.

[6] C. Cortes and V.N. Vapnik, "Support vector networks," Mach. Learn., vol. 20, no. 3, pp. 273–297, Sep. 1995.

[7] V. Franc and V. Hlavác, "An iterative algorithm learning the maximal margin classifier," Pattern Recognit., vol. 36, no. 9, pp. 1985–1996, Sep. 2003.

[8] B.F. Mitchell, V.F. Demyanov, V.N. Malozemov, Finding the point of a polyhedron closest to the origin, SIAM J. Control 12 (1974) 19–26.

[9] R. Weller, New Geometric Data Structures for Collision Detection and Haptics, Springer Series on Touch and Haptic Systems, DOI 10.1007/978-3-319-01020-5_2, © Springer International Publishing Switzerland 2013.

[10] Mavroforakis M., Sdralis M., Theodoridis S. "A geometric nearest point algorithm for the efficient solution of the SVM classification task," IEEE Transactions on Neural Networks, Vol. 18(5), pp. 1545–1550, 2007.

[11] Zhenbing Liu, J. G. Liu, Chao Pan, and Guoyou Wang. A Novel Geometric Approach to Binary Classification Based on Scaled Convex Hulls, IEEE Transactions on Neural Networks. 2009. Vol. 20, No. 7. pp. 1215-1220.

[12] Iris Data Set. UCI Machine Learning Repository. Available at: https://archive.ics.uci.edu/ml/datasets/iris (accessed 26 April 2018)

[13] Breast Tissue Data Set. UCI Machine Learning Repository. Available at: http://archive.ics.uci.edu/ml/datasets/breast+tissue (accessed 26 April 2018)

[14] V.N. Vapnik, Statistical Learning Theory. New York: Wiley, 1998.

[15] Bennett K.P. and Bredensteiner E.J., Duality and geometry in SVM classifiers, in Proc. 17th Int. Conf. Mach. Learn., 2000, pp. 57–64.

[16] Ming C. Lin, Dinesh Manocha, and Young J. Kim, COLLISION AND PROXIMITY QUERIES. In the Handbook of Discrete and Computational Geometry, J.E. Goodman, J. O'Rourke, and C.D. Tóth (editors), 3rd edition, CRC Press, Boca Raton, FL, 2017.