Decision Boundaries in the sense of Naive Bayesian for Multidimensional cases (Naïve Decision Surface Network)

Dhiadeen M. Salih

Electrical Engineering Department, College of Engineering, Kirkuk University, Kirkuk, Iraq.

dhiadeen@uokirkuk.edu.iq

Abstract

Naive Bayesian classifier is a fundamental statistical method that assents the conditional independence of features values by minimizing the probability errors within the classes. In practice, Naive Bayesian classifier often violated assumptions and is not robust to the noise with multidimensional cases. A useful way to signify classifier is through discriminant functions where the classifier assigns a feature vector to divide the feature space into decision surfaces separated by multidimensional boundaries. In this work, Naïve Decision Surface Network is proposed to build on discriminant quadratic functions that obtained for a multiclass, multi features problems. The action all of covariance, variance and correlation possibilities are addressed. An example is illustrated to demonstrate the computational and analytical simplifications and the results showed less classification rate error.

Keywords: Bayesian classifiers, Naïve Decision Surface Network, Decision Boundaries.

حدود القرار بمنظور نظرية ناييف بايزين للحالات متعددة الأبعاد (شبكة ناييف لاسطح القرار)

ضياء الدين محمد صالح

قسم الهندسة الكهربائية، كلية الهندسة، جامعة كركوك، كركوك، العراق.

dhahirgeo1958@gmail.com

الملخص

مصنف نايف بايزيين هو طريقة إحصائية أساسية توثق الاستقلالية المشروطة لقيم السمات وتعدد المصنف الأمثل من خلال تقليل أخطاء الاحتمال داخل التصنيف. في الممارسة العملية، نايف بايزيين غالبا ما تنتهك الافتراضات وليس قوية للضوضاء مع الحالات متعددة الأبعاد. وهناك طريقة مفيدة للدلالة على المصنف هي من خلال وظائف تمييزية حيث يعين المصنف متجه العناصر لتقسيم مساحة العنصر إلى أسطح قرار مفصولة بحدود متعددة الأبعاد. في هذا العمل، نقترح شبكة نايف لسطح القرار لبناء وظائف التربيعية التمييزية التي تم الحصول عليها مشاكل التصنيف لمتعددة الابعاد ومتعددة الميزات. ويتناول العمل التباين وإمكانيات الارتباط التحليلي. تم استخدام مثال يوضح امكانية التصنيف للطريقة المقترحة والنتائج أظهرت معدل خطأ اقل لتحديد الاصناف.

الكلمات الدالة: : نظرية بايزين، تصنيف بطريقة ناييف بايسس، زهرة النبق، مصنف بطريقة شجرة القرار.

1. Introduction

Bayesian classifier is a statistical approach that quantifies the trade-offs between various decisions using probabilities. The Bayesian strategies for pattern classification minimize the expected risk, in another word just to get as not many misclassifications as possible. The probability functions for each group outline decision regions in which boundaries with the highest probability of misclassification assigns each value of data set to one of the available classes [1]. Such a procedure will divide the input space into decision regions Rk as shown in Fig. 1 (b), such that if all points falls in region R1 it will be assigned to a specific class. The boundaries between decision regions called decision boundaries and if regions Ri and Rj happen to be contiguous, then they separated by a decision surface in the multidimensional feature space as shown in Fig. 1 (a), otherwise decision regions could contain some number of disjoint regions [2].
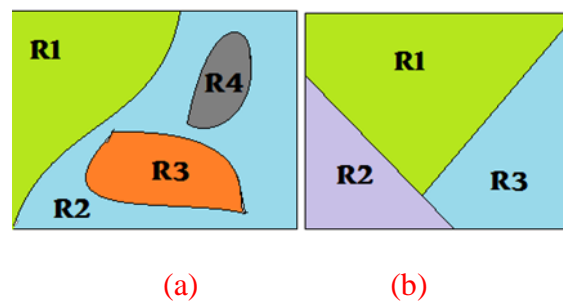

(a)   (b)

Fig. 1: The variations possible in some of decision region structure.

Many of the models used today for classification such as Feed-Forward Neural Networks, Support Vector Machines, Nearest Neighbour classifiers, and Decision Trees are generally manifolds with boundaries and run by constructing difficult decision regions in the input space .These decision regions can be few or many, convex or concave, have high or small volumes, etc. Thus it allows a qualitative research to demonstrate how it partitioned into decision regions by applying different classification strategies with different parts input space [2]. Since bayes classification suffers from complexity with multidimensional problems, Naive bayes classifier approach raised stand on Bayesian theorem, where it is particularly appropriate when the inputs dimension is high. For simple decision problems; the decision surfaces separating optimal bayes decision regions are more complicated than the decision surfaces of the Naive Bayes model [3]. The classification as dividing the input space into decision regions and visualizing the decision surfaces can be achieved by studying the relationship between classification models and decision boundaries. The users can see the

distance between data and their decision boundaries and they may know whether the classifier will lead to over-fitting or not [4]. The rule of discriminant function is maximizing the posterior probability to minimize the error rate for both the discrete and continuous problems. The only distinction is that probabilities used as a replacement for of probability densities [5]. In this paper, we derive a general partitioning function for Naive Bayesian decision surface of three featured data set classes. In the next section we illustrate an example for three class's data set. The conclusion on results will be in the last.

2. Bayesian Discriminant Function

Bayesian theorem is essentially an expression of conditional probabilities where, conditional probabilities represent the probability of an event occurring given evidence. Under estimating probability density functions (pdf's), it is important to define four general terms related to our subjects. The first is the prior or a priori distribution which implies of how a particularly the system formed. However, a uniform distribution can be used to model the prior probability as Gaussian model. Secondly, the likelihood is simply the probability of a specific class given the random variable. The third is a posteriori probability that it states the probability of an event occurring given evidence and this specifically what results from the Bayes rule. Finally, the evidence $P(X)$ in bayes theorem is usually considered a scaling term that stated for a feature vector X [6]:

$$P(C_j|X) = \frac{P(X|C_j)P(C_j)}{P(X)} \tag{1}$$

where $P(C_j|X)$ posteriori for C_j (j of classes), and X is the evidence given(feature vector). The likelihood function $P(X|C_j)$ of C_j with respect to X, (i.e., the probability that X belongs to C_j), and the prior probability $P(C_j)$ reflects knowledge of the element frequency aimed at instances of a class. While, the evidence:

$$P(X) = \sum_j P(X|C_j) P(C_j) \tag{2}$$

The decision rule is:

$$for\ all\ \ j \neq i\ Decide\ \ C_j\ \ if\ \ \ \ \ \ \ \ P(C_j|X) > P(C_i|X) \tag{3}$$

$$for\ all\ \ j \neq i\ Decide\ \ C_i\ \ if\ \ \ \ \ \ \ \ P(C_j|X) < P(C_i|X) \tag{4}$$

Hence, the classification processes done by estimating the membership of an inspection in a class based on the features observation. However, we might not always be making the best decision, so there must be a function that says to us which deed to take for every possible observation which is the risk function [7]. The overall risk given our interpretation for each action is computed as sum of associated risk of all the states, weighted by $\gamma(r|C_j)$ the probability of occurrence of each state.

$$R(r|X) = \sum_j \gamma(r|C_j)\, P(C_j\,|X) \tag{5}$$

The rule to minimize risk and the corresponding error rate can be done by maximizing the posterior probability [8]. Thus, for the minimum error probability case that separated by a decision surface in the multidimensional feature space this described by the equation (6),

$$P(C_j\,|X) - P(C_j\,|X) = 0 \tag{6}$$

However, from a mathematical point of view, there are equivalent functions $D_j(X)$ and $D_i(X)$ considered as discriminant functions for all classes such that :

$$D_j(X) = f\left(P\left(C_j|X\right)\right), \qquad D_i(X) = f(P(C_i|X)) \qquad j \neq i$$

Consequently, the decision tests in (3) and (4) is now stated as

Classify X in C_i if $\qquad D_i(X) > D_j(X) \quad \forall\ j \neq i$

Classify X in C_j if $\qquad D_i(X) < D_j(X) \quad \forall\ j \neq i$

In another word, there are decision surfaces that separate contiguous regions, and described by,

$$D_{ji} \equiv D_j - D_i = 0 \qquad i,j\ =\ 1,2\dots\dots K \qquad \forall\ j \neq i \tag{7}$$

where, K is the total classes number

3. Decision Surface

Now we assume that the likelihood functions of C_j with respect to X in the d dimensional feature space follow the general multivariate normal (Gaussian) density because of its computational tractability and the truth that it models sufficiently a large number of cases,

$$P(X|C_j) = \frac{1}{\sqrt[n]{2\pi}\sqrt{|\Sigma_j|}} \exp\left(-\frac{1}{2}(X-\mu_j)^T \Sigma_j^{-1}(X-\mu_j)\right) \quad , \quad j = 1, 2 \dots K \tag{8}$$

where, $\mu_j = E[X]$, is the mean value of the C_j class, Σ_j is the covariance matrix, and n is the feature space dimension. However, because of the exponential form is involved densities, it is preferable to work with the following $ln(.)$ (monotonic) logarithmic function [10]. Therefore and by subsist (1) into (6) we get,

$$ln\ P(X|C_j)P(C_j) = ln\ P(X|C_i)P(C_i) \tag{9}$$

Here, $P(X)$ omitted since it exits in both side, and using (7), (8) and (9) we get the quadratic discriminant surface:

$$D_{ji} = \frac{1}{2}(X-\mu_j)^T \Sigma_j^{-1}(X-\mu_j) - \frac{1}{2}(X-\mu_i)^T \Sigma_i^{-1}(X-\mu_i) + \frac{1}{2}\ln\left(\frac{|\Sigma_j|}{|\Sigma_i|}\right) - \ln\left(\frac{P(C_j)}{P(C_i)}\right) \tag{10}$$

The decision surfaces in general are formed in hyper-quadrics custom (i.e., ellipsoids, parabolas, hyperbolas). However, for multi-dimensional features d and multi-dimensional classes k, we get N of quadratic discriminant functions where;

$$N = \sum_{k=1}^{n}(k-1) \tag{11}$$

3.1. Naïve Bayesian Decision Surface

The terms of independence and conditional independence are outlined significantly in the Bayesian world. Independence basically ensures that two (or more) random variables are not dependent on one another, where the over all probability can be described mathematically by the Joint Probability [11]:

$$P(A, B, C) = P(A)P(B)P(C)$$

The concept of Naïve Bayes Decision Surface is the ability to act as a classification that can be handling an arbitrary number of independent variables whether continuous or categorical. For instance, a set of statistically independent variables (features), $X = \{x_1, x_2, x_3 \dots, x_n\}$ posterior probability for the event Cj among a set of possible outcomes $C = \{C_1, C_2, C_3 \dots, C_K\}$, under this assumption, we can write using Baye's rule:

$$P(C_i|X) = P(C_i) \prod_n P(x_n|C_i) \tag{12}$$

Her we can decompose the likelihood to a product of terms for feature vector X since that the variables are statistically independent as Naive Bayes theorem assumes. Although, this assumption is not always accurate, it simplifies the classification task dramatically because it allows the likelihood $P(X|C_i)$ to be calculated separately for each variable. Furthermore it turns out that the naive-Bayes classifier can be very robust to violations of its independence assumption. This has been reported to be implemented well for many real-world data sets and can be modeled with different density functions [12]. In this paper the normal Gaussian model implemented for the same reasons mentioned above.

4. Naïve Decision Surface Network

The proposed scheme can be described by Illustrating multiclass and multi feature data attributes as a decision boundary surface in a network form such that can be obtained by applying the case of equality with detrimental where the pattern can be assigned to either of any two classes [9]. Thus the decision surface can equivalently be based on (10) for minimum-error-rate classification and written as:

$$Dji = P(C_j) \prod_n P(x_n|C_j) - P(C_i) \prod_n P(x_n|C_i) = 0 \tag{13}$$

The Fig. 2 shows the proposed Naïve decision surface network demonstrated by means of three layer network structures where the input nodes represented by a set of data features represented as, $X = \{x_1, x_2, x_3 \dots, x_n\}$ for each class.
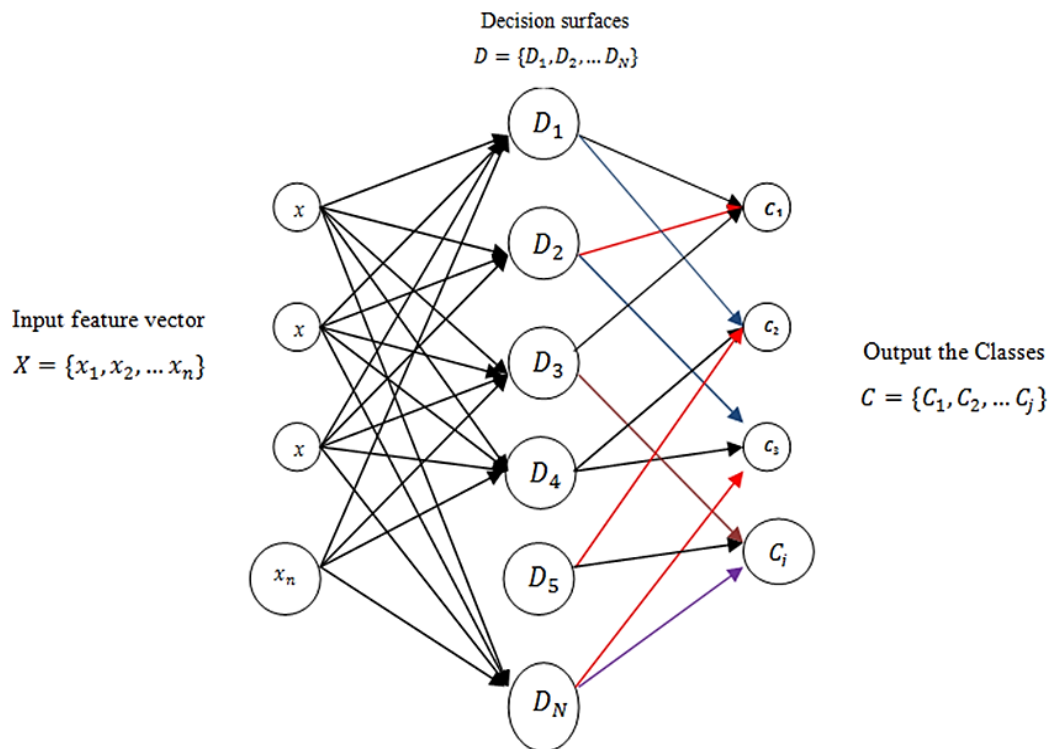
Fig. 2: Naïve Decision Surface Network.

The middle layer is consisting of discriminant functions Dji employed in the each layer nods that should be connected to the output layer nods with output weights. These weights distributed to the output nodes such that each one will be tuned according learning algorithm. Note that the output nodes represent the classes.

4.1 The Naïve Decision Surface Network Algorithm

The algorithm for the proposed network has two phases; the learning phase and the testing phase. The data set for each phase are different to avoid over fitting problem. The learning algorithm can be summarized as below;

Set   K = the No, of Classes   (C_j)   j=1, 2, 3,....K

Set the features vector   X={x_1 , x_2 , ...x_n }

Set N the quadratic discriminant functions using Eq. (11)

For each class Cj   Do

Calculate   Standard deviation σ_{C_jn} and μ_{C_jn} for the feature vector x_n in the class C_j,

Mean value μ_j of the C_j class, and, Σ_j the covariance matrix,

Mahalanobis Distance J(X)

The likelihood $P(X|C_i)$, Eq (12)

Stop

   for j=1 to K

     for i=j+1 to K

          Find the Discriminant functions Dji using Eq. (16)

$$D_j(X) = f\,(\,P\,(\,C_j|X\,)\,)\,, \qquad D_i(X) = f\,(P(C_i|X)$$

    End

   End

Construct Naïve decision surface network of three layer network structure where,

   The input nodes = length of the features vector, $X = \{x_1, x_2, x_3\ldots, x_n\}$,

   The middle layer nodes = N consisting of discriminant functions Dji in each nods,

   The output nodes = K (No. of the classes),

Initializing the network Parameters,

   All the input weights set to one

Learning procedure for the output weights

   for j=1 to K

     for i=j+1 to K

        Classify X in C_i if $D_i(X) > D_j(X)$ $\forall\, j \neq i$

        Assign the highest Mahalanobis distance = W_{ij}

        Classify X in C_j if $D_i(X) < D_j(X)$ $\forall\, j \neq i$

        Assign the highest Mahalanobis distance = W_{ji}

    End

   End

The network can be learned initially by the learning data sets where the weights should take two forms only: the active denoted by 1 (*highest Mahalanobis distance*) and inactive denoted by 0, and at least two weights should be active for one class in output nodes in order to be classified in testing phase with test data sets. Remember that, for multiclass and multi feature data set we can obtain partitioning the feature space into $R1\ldots.Rn$ regions, and so discriminant functions in the middle layer equal to *N* as in (11).

5. Simulation Example

In our example, we choose Iris flower data set (*setosa*, *versicolor*, and *virginica*) to illustrate the proposed classification scheme because it has been applied as a benchmark in Matlab classification demos in Statistics. For each of the species, 50 observations with four features: sepal length, sepal width, petal length, and petal width are recorded (see Iris.dat MATLAB). Since our feature attributes are independent, the normal distribution pdf can be used for likelihood:

$$P(x|C_i) = \frac{1}{\sqrt{2\pi}\sigma_{xci}} \, exp\left(-0.5 \left(\frac{x - \mu c_i}{\sigma_{xci}}\right)^2\right) \tag{14}$$

By using (13) and (14) the discriminant function D_{ji} will be:

$$D_{ji} = \ln\left(\frac{P(C_j)}{P(C_i)}\right) + \sum_1^n \ln P(x_n|C_j) - \sum_1^n \ln P(x_n|C_i) \tag{15}$$

In common,

$$D_{ji} = \ln\left(\frac{P(C_j)}{P(C_i)}\right) + \sum_1^n \ln(\frac{1}{\sigma_{C_jn}}) - 0.5\left(\frac{x_n - \mu_{C_jn}}{\sigma_{C_jn}}\right)^2 - \sum_1^n \ln\left(\frac{1}{\sigma_{C_in}}\right) - 0.5\left(\frac{x_n - \mu_{C_in}}{\sigma_{C_in}}\right)^2 \tag{16}$$

where, σ_{C_jn} is the standard deviation for the feature vector x_n in the class C_j and μ_{C_jn} is the mean of features vector x_n for the class C_j. To visualize the plotting of the decision boundary surfaces the reduction for feature dimension size by one is applicable using Mahalanobis distance. We get:

$$\mu_{setosa} = \begin{bmatrix} 5.006 \\ 3.428 \\ 1.462 \\ 0.246 \end{bmatrix}, \mu_{versi} = \begin{bmatrix} 5.9360 \\ 2.77 \\ 4.26 \\ 1.326 \end{bmatrix}, \Sigma = \begin{bmatrix} 0.4118 & -0.0633 & 0.7557 & 0.2864 \\ -0.0633 & 0.2292 & -0.4182 & -0.1545 \\ 0.7557 & -0.4182 & 2.1012 & 0.8023 \\ 0.2864 & -0.1545 & 0.8023 & 0.3194 \end{bmatrix}$$

Mahalanobis Distance:

$$J(X_m) = (\mu_1 - \mu_2)^T \Sigma^{-1}(\mu_1 - \mu_2)$$

$$J(wxy) = \left(\begin{bmatrix} 5.006 \\ 3.428 \\ 1.462 \\ 0.246 \end{bmatrix} - \begin{bmatrix} 5.9360 \\ 2.77 \\ 4.26 \\ 1.326 \end{bmatrix}\right)^T \begin{bmatrix} 0.4118 & -0.0633 & 0.7557 & 0.2864 \\ -0.0633 & 0.2292 & -0.4182 & -0.1545 \\ 0.7557 & -0.4182 & 2.1012 & 0.8023 \\ 0.2864 & -0.1545 & 0.8023 & 0.3194 \end{bmatrix}^{-1} \left(\begin{bmatrix} 5.006 \\ 3.428 \\ 1.462 \\ 0.246 \end{bmatrix} - \begin{bmatrix} 5.9360 \\ 2.77 \\ 4.26 \\ 1.326 \end{bmatrix}\right)$$

$$= 3.7988$$

Since there are only 4 possible triplets out of 4 features, the Mahalanobis distance will be calculated for all four, and the triplet that give the highest distance will be selected.

$$wxy = 3.7988$$

$$wxz = 3.7776$$

$$wyz = 3.7759$$

$$xyz = 3.8140$$

From the calculated distances, the xyz triplet gives the highest Mahalanobis distance, which means that the combination of x, y, and z features is the best to discriminate between the two classes. However, we assume that the prior probabilities are equal: $P(setosa) = P(versi) = P(virgi)$ for sake of simplicity .The mean μ_{setosa}, μ_{versi} and the covariance Σ_{setosa}, Σ_{versi} matrices are obtained for *setosa* and *versicolor* respectively,

$$\mu_{setosa} = \begin{bmatrix} 3.428 \\ 1.462 \\ 0.246 \end{bmatrix}, \quad \mu_{versi} = \begin{bmatrix} 2.770 \\ 4.260 \\ 1.326 \end{bmatrix}$$

$$\Sigma_{setosa} = \begin{bmatrix} 0.144 & 0.0117 & 0.0093 \\ 0.0117 & 0.0302 & 0.0061 \\ 0.0093 & 0.0061 & 0.0111 \end{bmatrix}, \Sigma_{versi} = \begin{bmatrix} 0.0985 & 0.0827 & 0.0412 \\ 0.0827 & 0.2208 & 0.0731 \\ 0.0412 & 0.0731 & 0.0391 \end{bmatrix}$$

Since $ln \frac{P(setosa)}{P(versic)} = 0$, therefore and from (10) we get:

$$D_{1,2} = \frac{1}{2}\left[(X_{setosa} - \mu_{setosa})^T \Sigma_{setosa}^{-1}(X_{setosa} - \mu_{setosa}) - (X_{versi} - \mu_{versi})^T \Sigma_{versi}^{-1}(X_{versi} - \mu_{versi})\right] + ln \sqrt{\frac{|\Sigma_{setosa}|}{|\Sigma_{versi}|}}$$

Here, we will get a quadratic equation for the decision boundary between *setosa* and *versicolor* as stated below and graphed in the Fig. 3.

$$D_{1,2} = -5.42\,x^2 - 0.613\,xy + 11.8\,xz + 1.37\,x + 12.9\,y^2 + 2\,yz - 24.5\,y + 11\,z^2 - 6.92\,z + 11.8$$
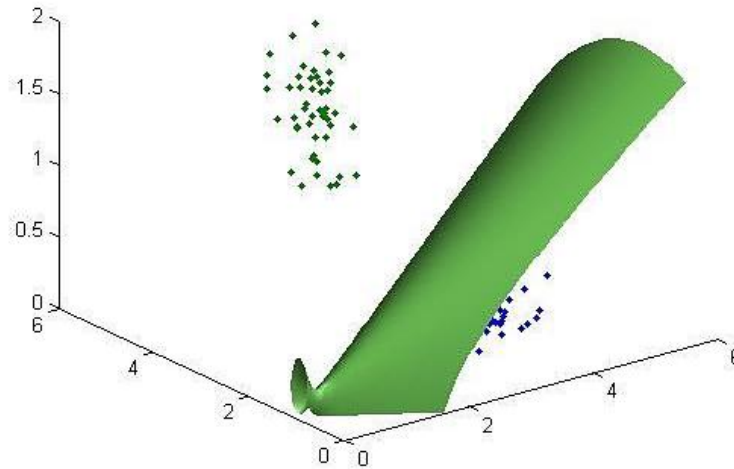
Fig. 3: Baye's decision surface between *setosa* and *versicolor*.

Moreover, In our case we will perform pair of classes since the discriminant function should cover all the classes without duplicate , in contrast, since number of classes is *n=3* then number of discriminant function *N=3* see (11).

Here, the proposed Naïve Decision Surface Network (NDSN) will deal with each feature individually for each class and according to (16), if we assume that *setosa set =A and versicolor=B* , then we get the discriminant function $D_{1,2}$ between them will be as below,

$$D_{1,2} = ln(\frac{P(A)}{P(B)}) + \left(ln\left(\frac{1}{\sigma_{A1}}\right) - 0.5\left(\frac{x_1-\mu_{A1}}{\sigma_{A1}}\right)^2 + ln\left(\frac{1}{\sigma_{A2}}\right) - 0.5\left(\frac{x_2-\mu_{A2}}{\sigma_{A2}}\right)^2 + ln(\frac{1}{\sigma_{A3}}) - 0.5\left(\frac{x_n-\mu_{A3}}{\sigma_{A3}}\right)^2 \right) -$$

$$- \left(ln\left(\frac{1}{\sigma_{B1}}\right) - 0.5\left(\frac{x_1-\mu_{B1}}{\sigma_{B1}}\right)^2 + ln(\frac{1}{\sigma_{B2}} - 0.5\left(\frac{x_2-\mu_{B2}}{\sigma_{B2}}\right)^2 + ln(\frac{1}{\sigma_{B3}}) - 0.5\left(\frac{x_n-\mu_{B3}}{\sigma_{B3}}\right)^2 \right)$$

Generally, the standard deviation for the feature vector of the two classes can be considered that they are not equal $\sigma_{A1=}\sigma_{A2=}\sigma_{A3=}\sigma_1 \neq \sigma_{B1=}\sigma_{B2=}\sigma_{B3} = \sigma_2$. Fig. 4 shows the Naive decision surfaces of the case.
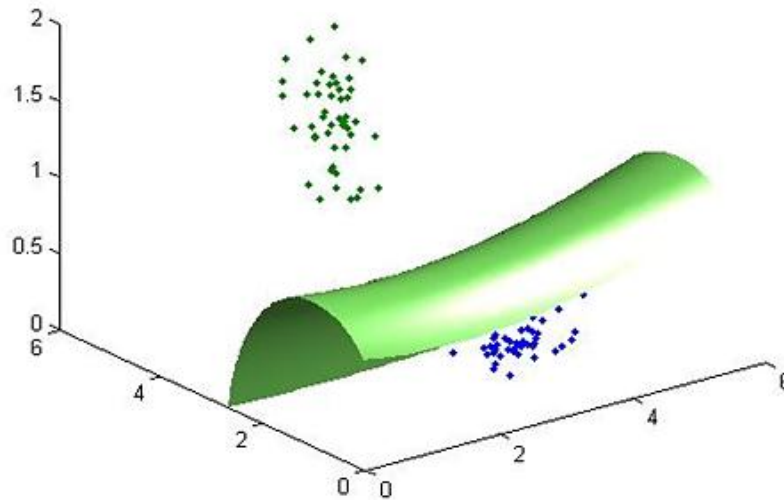
Fig. 4: The naive decision surfaces between *setosa* and *versicolor*.

Moreover, for the other two decisions surface, the same procedure can be applied to get $D_{1,3}$ and $D_{2,3}$ as showed in the Fig. 5.
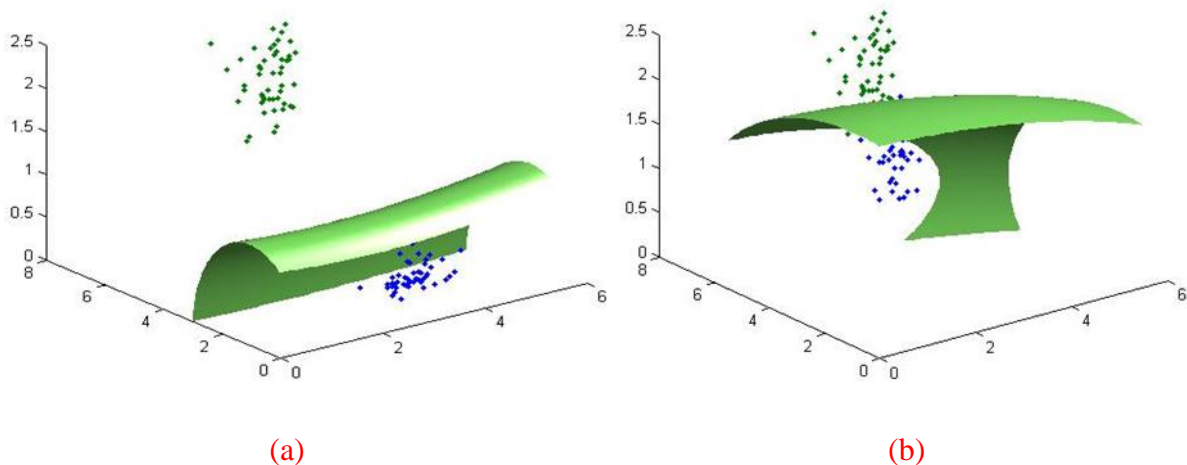

(a)   (b)

Fig. 5: The naïve decision surfaces between (a) virginica and versicolor, (b) virginica and setosa.

Now, the simulation example of the Naïve Decision Surface Network (NDSN) on our dataset started with cross-validation to prevent over fit problems were 2, 5, and 10 fold methods is applied for each class. The training data is applied first then the test data used to demonstrate the performance of our proposed method by calculating the classification error rate as below,

$$Error\ Rate = \sum (bad)/N \quad \%$$

The comparison with other well-known methods in Matlab Statistics Toolbox functions namely, Linear Discriminant Analysis (LDA), Quadratic Discriminant Analysis (QDA), and Decision Tree. The result recorded in the Table 1 and where the test error on an independent fisher iris flower data set is summarized.

Table 1: the classification error rates and the confusion matrix of classifiers, and NDSN method.

Classifiers	Classification Error Rate % (= sum(bad) / N %)	Confusion Matrix		
		setosa	*versicolor*	*virginica,*
Decision Tree	13.3	49	2	1
		0	44	2
		0	2	47
Linear Discriminant Analysis (LDA)	20	49	1	0
		0	36	14
		0	15	35
Quadratic Discriminant Analysis (QDA)	22.3	48	0	1
		0	32	12
		0	2	27
Naïve Decision Surface Network (NDSN)	4.6	50	0	0
		0	48	2
		0	4	46

From Table 1, it is clear that the error rate of the proposed NDSN merhod is best in classification from the other well-known methods. Moreover, the confusion matrix showed that full of 50 of the Iris-setosm data were classified and 48 with 2 errors for Iris-versicolor. While, 46 of Iris-virginica data set were correctly classified with 4 errors.

6. Conclusion

From side of complexity, sometimes the decision boundary surface more helpful to use as classifier instead of working directly with probabilities (or risk functions).The formal definitions of decision boundary surface for Bayes and naïve Bayes are presented in this paper. The visualization of decision boundaries gives a good understanding the distance and decision boundary among decision regions between the data sets. The proposed NDSN method is more convenient when the data sets involved *pdf's* that are complicate and their estimation is not an easy task. Moreover, it is preferable to compute decision surfaces directly by means of alternative costs such that it gives rise to discriminant functions and decision surfaces. The method in general is suboptimal with respect to Bayesian classifiers. An

example is applied to illustrate the superiority of the proposed method over some well-known methods. However, this method can be applied to other datasets for the generalizing purposes. The under fit or over fit problem is possible here but this procedure can helped to overview the classifier behavior whether the classifier will lead to over-fitting or not.

References

[1] Arturo B. and Jorge I. "***The Effect of Repeated Measurements on Bayesian Decision Regions for Class Discrimination of Time-Dependent Biological Systems***", Advances in Artificial Intelligence, Lecture Notes in Computer Science, 5317, 305 (2008).

[2] Ofer M." ***Decision Region Connectivity Analysis: A Method for Analyzing High-Dimensional Classifiers***", Machine Learning, 48, 321 (2002).

[3] Mark A., "***Geometric Implications of the Naive Bayes Assumption: UAI '96: Proceedings of the Twelfth Annual Conference on Uncertainty in Artificial Intelligence***", Reed College, Portland, Oregon, USA, 414 (1996).

[4] ***"Zhiyong Yan CongfuXu Studies on Classification Models Using Decision Boundaries Procceding of 8th IEEE int."*** Conference on Cognitive Information (ICCI'09) , 287(2009).

[5] Qiong W., George M. Garrity, and James M. Tiedje " ***Naïve Bayesian Classifier for Rapid Assignment of rRNA Sequences into the New Bacterial Taxonomy***" Applied and Environmental Microbiology, 73(16), 5261(2007).

[6] Ximeng L.*,* Rongxing L., Jianfeng M., Le C., and Baodong Q." ***Privacy-Preserving Patient-Centric Clinical Decision Support System on Naive Bayesian Classification***" , IEEE Journal of biomedical and health information, 20(2), 655 (2016).

[7] Shuo Xu "***Bayesian Naïve Bayes classifiers to text classification***" Journal of Information Science, 44 (1), 48 (2018).

[8] Sergios T. and Konstantinos K. "***Pattern Recognition***". 4th Edition, Academic Press, Elsevier Inc. (2009).

[9] Christopher M. Bishop "***Pattern Recognition and Machine Learning***" Springer, New York, (2006).

[10] Kuncheva L., Bezdek J. and Duin R., "***Decision Templates for Multiple Classifier Fusion: An Experimental Comparison***", Pattern Recognition, 34, 299 (2001).

[11] Makri1 F. and Psillakis Z.," ***On Limited Length Binary Strings with an Application in Statistical Control***", The Open Statistics & Probability Journal, 8, 1 (2017).

[12] Wafa F., Hasna N., Asma H., Nouha B., Mabrouka M., Ahmed R., and Malika B. "***Using general linear model, Bayesian Networks and Naive Bayes classifier for prediction of Karenia selliformis occurrences and blooms***", Ecological Informatics, 43,12(2018).