Subspace-Based Dynamic Selection: A Proof of Concept Using Protein Microarray Data

Alexandre Maciel-Guerra
School of Computer Science
The University of Nottingham
Nottingham, UK
Alexandre.MacielGuerra@nottingham.ac.uk

Grazziela P. Figueredo
The Advanced Data Analysis Centre
School of Computer Science
The University of Nottingham
Nottingham, UK
Grazziela.Figueredo@nottingham.ac.uk

Eliane Marti
Clinical Research and Veterinary Public Health
University of Bern
Bern, Switzerland
eliane.marti@vetsuisse.unibe.ch

Marcos J. C. Alcocer
School of Biosciences
The University of Nottingham
Sutton Bonington, UK
Marcos.Alcocer@nottingham.ac.uk

Jamie Twycross
School of Computer Science
The University of Nottingham
Nottingham, UK
Jamie.Twycross@nottingham.ac.uk

***Abstract*—Traditional dynamic selection methods fail to perform effectively when data dimensionality increases. In addition, those methods do not provide any insights into important features in the data, as the regions of competence used for data classification are always constituted by the same set of features. In this paper, we propose a two-stage framework based on subspace clustering using a Gaussian Based Estimator, followed by a k-Nearest subspace search mechanism to overcome these limitations of dynamic selection. The idea of subspace allows for regions of competence with different numbers of instances and dimension sizes. Our hypothesis is that by using our framework, we will achieve comparable results to the state-of-the-art dynamic selection, with the benefit of producing a model that helps to understand the importance of sets of features for the patterns found within the data. We test our approach to a high dimensional microarray data of insect bite hypersensitivity in horses. Results show that our approach is comparable to traditional dynamic selection methods in terms of accuracy. In addition, it facilitates the interpretability of the feature importance for each class of the dataset.**

***Index Terms*—dynamic selection, subspace clustering, Gaussian kernel density estimator, nearest subspace search, protein microarrays, insect bite hypersensitivity**

I. Introduction

In the last decade, a number of researchers have shown that for low-dimensional data, dynamic selection (DS) outperforms single robust classifiers and traditional combination methods, such as majority voting, bagging and boosting [1]–[6]. For high-dimensional data, however, Maciel-Guerra *et al.* (2019) showed that DS methods fail to perform as well, because the k-Nearest Neighbours (k-NN) approach, commonly adopted to define regions of competence, deteriorates as the number of dimensions increases. As stated by Cruz *et al.* (2017) [1], the mechanisms for a proper determination of the region of competence for DS are underdeveloped; and advances in the area could lead to increased performance of DS methods for high-dimensional data.

In data sets with high feature spaces, many dimensions are irrelevant and can directly impact in the quality of the regions of competence retrieved [7], [8]. Feature selection and extraction methods have been employed to remove irrelevant features to improve those regions quality [7], [8]. However, in high-dimensional data, a phenomena called *local feature relevance* occurs, i.e., different subsets of features are relevant for distinct regions of competence [7]. Therefore, traditional feature selection and extraction methods using all features to determine their individual importance might not be suitable. Instead, subspace clustering methods narrow their search and are able to elucidate clusters within multiple, possible overlapping subspaces of features and/or samples [8]. Subspace clustering is an extension of traditional clustering methods that attempts to find clusters in different regions of the feature space in terms of features and samples [8].

The focus of this paper is therefore to incorporate subspace clustering methods into the DS framework to tackle high dimensional problems. The rationale behind DS techniques is that not every classifier is an expert in predicting all unknown samples; instead, each classifier or a combination of classifiers of the pool is an expert in different regions of the feature space where the test samples are located [1]–[3]. In practical problems, different query samples have particular classification difficulties and may be located in distinct subspace regions. Hence, it is intuitive to think that adopting different classifiers to predict the pattern of different test samples may increase the performance of a multiple classifier system [1], [21]. Moreover, by using the concepts of subspace clustering into the DS framework, it would be possible to know which features are more important for the classification of each test pattern.

Therefore, we propose a novel framework for DS methods called Subspace-Based Dynamic Selection (SBDS). By incorporating a subspace clustering method, we hypothesise

978-1-7281-6926-2/20/$31.00 ©2020 IEEE

that it is possible to increase the performance of DS methods and improve knowledge discovery in high-dimensional small-instance data sets. To accomplish this, we use the main characteristics of the DS framework and integrate it with the concepts of subspace clustering. SBDS can be divided into two steps: (1) subspace clustering based on a Gaussian Kernel Density Estimator (GKDE) to find all one-dimensional clusters and, subsequently, a merging procedure to find all subspaces; (2) k-Nearest Subspace Search (k-NSS) to find the k most similar subspaces in relation to each test sample. Despite the large number of papers published in DS, to the best of our knowledge, there is no comprehensive study available combining subspace clustering with DS. We hope that the proposed SBDS framework achieves higher performance when compared with DS methods in terms of accuracy.

In this work, we use an equine insect bite hypersensitivity (IBH) data set as a case study to test our proposed SBDS framework. This disease is a well-characterised immune response, that involves IgE antibodies to ordinary salivary proteins from insects, with a known aetiology and fully determined clinical symptoms, as described in Marti *et al.* (2015) [22]. The IBH data set used has a smaller number of samples (196) when compared to the number of features (384). The proposed SBDS framework achieved higher results when compared with DS methods.

This paper is organised as follows. Section II provides background on the main concepts of this paper and depicts related work associated with high-dimensional data. Section III introduces our proposed SBDS framework and Section IV describes the methodology used. Experimental results using the IBH data set are shown in Section V. Section VI outlines the conclusions and future work.

II. Background

The quantity of data collected from multiple sources have increased greatly in the past decade, particularly in medicine and life sciences, which brings challenges and opportunities. Heterogeneity, scalability, computational time and complexity are some of the challenges that impede progress to extract meaningful information from data [23], [24]. Moreover, biomarker discoveries in medical data sets, including protein microarrays, can help in the diagnosis of diseases such as allergies. We believe that approaches such as DS can help better classifying and understanding important features in high dimensional data, such as microarrays, as further discussed next.

A. Dynamic Selection

Multiple Classifier Systems (MCS) are a very active area of research with recent studies demonstrating its advantages over a single robust classifier [1], [2], [25]. MCS are essentially composed of three major stages: (1) **pool generation**, (2) **selection** and (3) **integration**. In the **pool generation stage**, the main goal is to train a pool of classifiers that are both accurate and diverse, *i.e.* the classifiers must have a low error rate (accurate) and two classifiers must make different errors on new samples (diverse). In the **second stage** the goal is to select a single or an ensemble of classifiers from the pool of classifiers. This stage can be divided into two groups: static (classifiers are fixed for all unknown test samples) and dynamic selection (selects a different set of classifiers for each test sample). The **final stage** consists of combining the outputs of the selected ensemble of classifiers according to a combination rule [1]–[3].

TABLE I
DS methods investigated

Name	Selection criteria	DS Method	Region of Competence	Reference
Classifier Rank (CR)	Ranking	DCS	k-NN	[9]
Modified Classifier Rank (MCR)	Ranking	DCS	k-NN	[10]
Overall Local Accuracy (OLA)	Accuracy	DCS	k-NN	[10]
Local Class Accuracy (LCA)	Accuracy	DCS	k-NN	[10]
A Priori	Probabilistic	DCS	k-NN	[11]
A Posteriori	Probabilistic	DCS	k-NN	[11]
Multiple Classifier Behaviour (MCB)	Behaviour	DCS	k-NN	[12]
Modified Local Accuracy (MLA)	Accuracy	DCS	k-NN	[13]
DES - k-Means (DES-kMeans)	Accuracy & Diversity	DES	k-Means	[14], [15]
DES - k-Nearest Neighbour (DES-kNN)	Accuracy & Diveristy	DES	k-NN	[14], [15]
KNORA - Eliminate (KNORA-E)	Oracle	DES	k-NN	[5]
KNORA - Union (KNORA-U)	Oracle	DES	k-NN	[5]
DES - Exponential (DES-EXP)	Probabilistic	DES	All training samples	[16]
DES - Randomised Reference Classifier (DES-RRC)	Probabilistic	DES	All training samples	[4]
DES - Minimal Difference (DES-MD)	Probabilistic	DES	All training samples	[17]
DES - Kullback-Leibler Divergence (DES-KL)	Probabilistic	DES	All training samples	[18]
DES - Performance (DES-P)	Probabilistic	DES	All training samples	[18]
KNOP - Eliminate (KNOP-E)	Behaviour	DES	k-NN	[19]
KNOP - Union (KNOP-U)	Behaviour	DES	k-NN	[19]
Meta-Learning - DES (Meta-DES)	Meta-learning	DES	k-NN	[6]
Dynamic Selection on Complexity (DSOC)	Accuracy & Complexity	DCS	k-NN	[20]

Table I shows the most important DS methods found in the literature in terms of their selection criteria. DS techniques can select either a single classifier (*Dynamic Classifier Selection*) or an ensemble of classifiers (*Dynamic Ensemble Selection*) based on their competence level to predict the label of a test sample. The competence is estimated considering only the samples of a local region of the feature space. For DS methods using k-NN, as indicated in the table I the region of competence size is k; for approaches not adopting k-NN, all data is used to make the prediction. The majority of DS techniques relies on a k-NN algorithm and the quality of the neighbourhood can have a huge impact on the performance of DS methods [1]–[3].

B. High-Dimensional Data

Exploring associations, making reliable predictions and extracting information are some of the problems that are yet to be solved in high dimensional data [23], [26]. According to Verleysen and François (2005) [27], traditional machine learning techniques were often created having in mind intuitive properties and examples in low-dimensional data sets. However, when tackling high-dimensional data, collinearity, numerical instability, overfitting, model instability are some of the known problems that can occur. Moreover, high-dimensional spaces have geometrical properties that are not intuitive [27], for instance:

- *Distance concentration*: within very high dimensional spaces, the distance between all data instances become almost equal, making, therefore, nearest neighbours to be unable to distinguish between "near" or "far" data points [28].
- *Hubness*: Let $D \subset \mathbb{R}^d$ be a set of d-dimensional points and $N_k(\vec{\mathbf{x}})$ the number of *k-occurrences* of each point $\vec{\mathbf{x}} \in D$, i.e., the number of times a point $\vec{\mathbf{x}}$ appears among the k-NN of all points in D, according to some distance metric [28]. According to Radovanovic *et al.* (2010) [28], as d increases, the distribution of N_k becomes considerably skewed to the right, resulting in the appearance of *hubs*, i.e., points that are "popular" nearest neighbours.

Therefore, these properties of high-dimensional spaces (distance concentration and hubness) can directly affect machine learning application, specially the ones that deal with distance metrics such as k-NN. Our hypothesis is that subspace clustering methods can overcome these issues, since they are able to select a subset of features and samples.

In general, the ensemble classifiers provide better classification accuracy than individual classifiers [29]. However, many ensemble methods proposed in the literature are not much accurate in high-dimensional biomedicine data, according to the recent survey made by Meshram and Shinde (2015) [29].

C. Subspace Clustering

Most traditional clustering algorithms attempt to find clusters using similarity measures based on distance metrics [7], [8], [30]. Moreover, these methods use the whole set of features to compute the similarities [7], [8], [30]. Feature selection and extraction methods can be a good choice to decrease the dimensionality of the data set. Nonetheless, according to Tian and Gu (2019) [30] some features might only work for a subset of samples and appear as noise for the rest of the samples, and this phenomenon is more common in high dimensional data sets.

To deal with this problem, subspace clustering has been used to elucidate clusters in different subsets of features and samples, as shown in Parsons *et al.* (2004) [8], Kriegel *et al.* [7] and Muller *et al.* (2009) [31]. Clusters determined in subspaces can reduce the computational cost and provide a more relevant information regarding local structures of the feature space, which can assist establishing the most important features relevant to the end point investigated [7], [8], [30]. For microarray studies, for instance, the goal is to understand which biomarkers (features) are relevant to the biological activity.

Muller *et al.* (2009) [31] divided subspace clustering methods into three groups:

- Cell-based: divide the data space into grid cells with a threshold and search for subspace clusters depending on the count of points in each cell [30], [31].
- Density-based: define clusters as dense regions separated by sparse regions. Since density estimation is based on distance between objects, the methods in this class compute distance by taking only the relevant dimensions.
- Clustering-oriented: these methods are similar to traditional clustering, where parameters such as the number of clusters, their average dimensionality, or other statistical oriented properties are required to establish groupings.

The proposed SBDS framework uses a density-based subspace clustering method. More specifically, it uses a Gaussian Kernel Density Estimator (GKDE) to find all one-dimensional clusters and subsequently it uses the same merging procedure proposed by Tian and Gu (2019) [30] to find all the subspace clusters. The GKDE methods was chosen due to its simplicity in defining one-dimensional clusters.

D. Nearest Subspace Search

According to Basri *et al.* (2011) [32], the Nearest Subspace Search problem is defined as follows: let $\mathcal{S}^1, \mathcal{S}^2, \cdots, \mathcal{S}^n$ be a collection of subspaces in $\mathcal{R}^d$, each with an intrinsic dimension d_S retrieved from a data set with m samples and d features. Given a query $\mathcal{Q}$ in $\mathcal{R}^d$, the distance between the query and the i^{th} subspace is $dist(\mathcal{Q}, \mathcal{S}^i)$. We seek the subspace $\mathcal{S}^*$ that is the nearest to the query, i.e., $\mathcal{S}^* = \arg\min_i dist(\mathcal{Q}, \mathcal{S}^i)$.

Basri *et al.* (2011) [32] proposed a mathematical approach to calculate an approximate distance between points and subspaces from subset dimensions to tackle high dimensional data. Their experiments indicate that an approximate nearest subspace can be located faster than the exact nearest subspace, with little loss of accuracy, in large databases.

In 2015, Hund *et al.* [33] introduce the concept of Subspace Nearest Neighbour Search (SNNS). It aims at finding query-dependent subspaces for nearest neighbour search, i.e. to find the nearest neighbours of query in a relevant subspace. The paper proposes three questions: (1) ”What is a relevant subspace for a given query?”; (2) ”How can we computationally extract this relevant information?”; and (3) ”How can we adapt ideas from subspace clustering, outlier detection, or feature selection for SNNS?”. Their proposed model for SNNS attempts to address those questions by assuming axis-parallel subspaces and defining a relevant subspace iff the following holds: ”a set of objects a, b, c are NN of the query q in a subspace S, iff a, b and c are NN of q in *all* dimensions of S”.

The three questions proposed by Hund *et al.* (2015) [33] to SNNS will be used as a motivation to define our Nearest Subspace Search algorithm on SBDS.

E. Related Work on IBH Protein Microarray Classification

Protein microarrays are an example of these data sets. They are a powerful tool in allergy diagnosis, as they monitor the interactions between the immune system and allergies. The intensity of the immune response is measured by the fluorescence observed, which is proportional to the concentration of antibodies in each spot on the microarray. In this section, we review the approaches used for the IBH dataset. Further details on the data set used can be found on Section IV-A

In 2015, Marti *et al.* [22] studied the influence of allergen-specific IgE against insect bites in horse sera to understand the causes of IBH. The authors demonstrated using the fluorescence of protein microarrays and a Partial Least Square Discriminant Analysis (PLSDA) that healthy and allergic horses could be highly differentiated. In addition, they were able to automatically selected 31 features using the variable importance in projection scores. Among those features were different *Culicoides sp.* salivary proteins which are in agreement with clinical knowledge about IBH [22].

In 2019, Maciel-Guerra *et al.* [34], investigated the use of DS methods on the same protein microarray data used by Marti *et al.* (2015) [22]. The authors have compared DS results with traditional machine learning methods before and after feature selection using a wrapper with backward elimination embedded with a regularised extreme learning machine. The DS methods did not have a increase in performance and most of them produced statistically similar results. In addition, traditional machine learning methods outperformed DS methods.

III. Subspace-Based Dynamic Selection (SBDS)

In this section, we introduce the proposed SBDS framework (Fig. 1) which aims to merge different concept of DS, subspace clustering and nearest subspace search. The objective of this framework is to improve feature relevance in high-dimensional small-instances data sets while maintaining or increasing the performance when compared with DS methods. The advantage of SBDS over DS is the use of subspace clustering to search through the feature and sample spaces for relevant clusters (equivalent to regions of competence in DS). By training different classifiers in distinct subspaces, we ensure that each classifier (or a combination of classifiers) is an expert in a different region of the feature space [1]–[3].

The proposed approach generates possible clusters for each individual dimension using a Gaussian Kernel Density Estimator (GKDE). Subsequently, a merging process is conducted to combine the one-dimensional clusters to form the subspaces. Finally, a classifier is trained on each subspace, and a 7-nearest subspace search is conducted to find the most similar subspaces to each unknown test sample. The majority voting

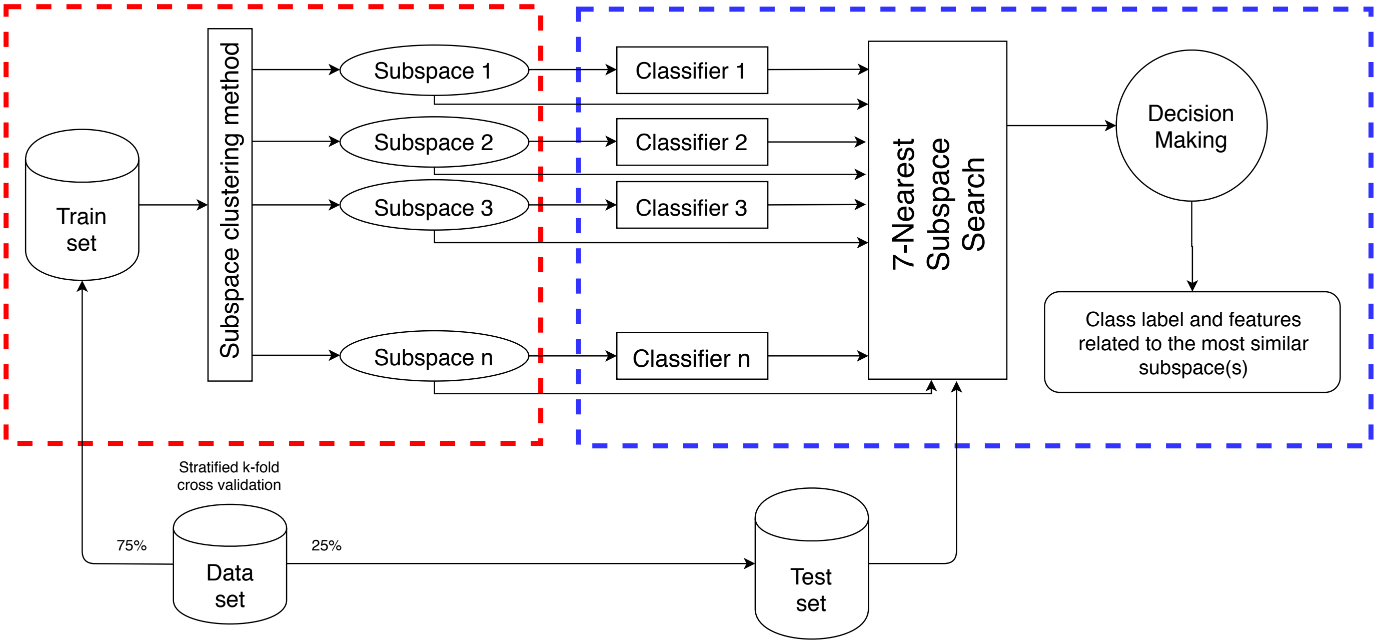
Train set
Subspace clustering method
Subspace 1
Subspace 2
Subspace 3
Subspace n
Classifier 1
Classifier 2
Classifier 3
Classifier n
7-Nearest Subspace Search
Decision Making
Class label and features related to the most similar subspace(s)
Stratified k-fold cross validation
75%
25%
Data set
Test set

Fig. 1. Proposed novel SBDS framework. The red dashed square indicates the Step 1 (subspace clustering) and the blue dashed squared indicates Step 2 (k-Nearest Neighbour Search and decision making).

of these 7 classifiers will give the final prediction for each test sample. Further details of each stage are given next.

A. Data generation

Initially the data is normalized using a Min-Max normalization approach, which scales the data between 0 and 1 for each feature. Then data is randomly divided into 75% for the training set and 25% for the testing set using a stratified k-fold cross validation to preserve the proportion of samples for each class.

B. Find One-Dimensional Clusters

Given a training data set, we determine the clusters in each dimension using a GKDE. A cluster is defined by a local maximum of the estimated density function, i.e. all points near the local maximum are assigned to the same cluster (Fig. 2).

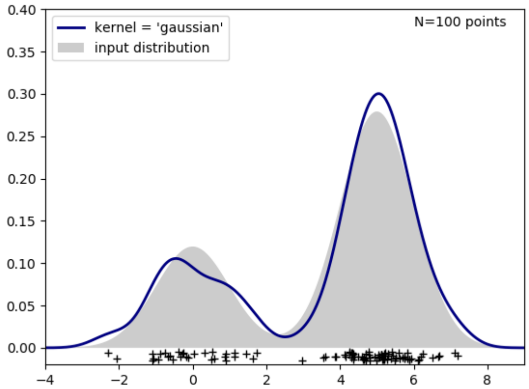

Fig. 2. Example of a Gaussian Kernel Density Estimator to find one-dimensional clusters.

C. Merging Procedure

After finding all one-dimensional clusters, a merge process is conducted to obtain the subspace clusters. Here we adopt a general method based on Tian and Gu (2019) [30]. The authors proposed to first merge similar clusters with different subspaces by using the Jaccard coefficient. In addition, if a cluster is contained in another cluster (in terms of samples) they must be merged [30]. The summary of the merging process is the same as the one proposed by Tian and Gu (2019) [30] and is described as follows:

Step 1 : set an empty set $\mathcal{DB}$

Step 2 : choose one dimension that has not been merged and determine its clusters using a GKDE

Step 3 : choose one cluster of the current dimension

Step 4 : compare the selected cluster with all subspace clusters in $\mathcal{DB}$ to find if there is a similar cluster by computing the Jaccard coefficient (Equation 1). If no similar subspace exists, go to Step 7

Step 5 : merge the chosen cluster with its similar one

Step 6 : compare the selected cluster with all subspace clusters in $\mathcal{DB}$ to find if it is contained in another subspace (Equation 2). If it is merge them and go to Step 8

Step 7 : add the selected subspace to $\mathcal{DB}$ and compare the select cluster with all subspaces in $\mathcal{DB}$ to find if there is one that contains it. If there is, merge them

Step 8 : if all cluster of current dimension are selected, go to Step 9. Otherwise go to Step 3

Step 9 : if all dimensions are merged, return $\mathcal{DB}$. Otherwise go to Step 2.

The Jaccard coefficient to measure the similarity between two subspaces is defined as:

$$J(E,F) = \frac{|E \cap F|}{|E \cup F|} \tag{1}$$

where E and F are the samples of two subspaces. Besides, the containment relationship of E and F is defined as:

$$\begin{aligned} C_1(E,F) &= \frac{|E \cap F|}{|E|} \\ C_2(E,F) &= \frac{|E \cap F|}{|F|} \end{aligned} \tag{2}$$

if $C_1(E,F)$ is close to 1 and $C_2(E,F)$ gets a smaller value, E is contained in F.

After the merging process, we train one classifier per subspace cluster, iff the subspace contains samples for more than one class. The next step is to find the nearest subspaces to each test sample in order to make their prediction.

D. 7-Nearest Subspace Search

For each unknown test sample, we need to determine the 7-nearest subspaces. This value is the same as the one used on most papers in DS to define the size of the region of competence. This step measures the similarity between a point and a subspace.

We first calculate the centroid $\mathcal{C}_i$ for each subspace $\mathcal{S}^i$, by averaging each feature for all instances in $\mathcal{S}^i$. Subsequently, we measure the average Euclidean distance (d_{Sc}) between all points in $\mathcal{S}^i$ and $\mathcal{C}_i$. We also calculate the Euclidean distance (d_{Tc}) between the test sample $\mathcal{Q}$ (using only the dimensions within $\mathcal{S}^i$) and $\mathcal{C}_i$.

The ratio between the two distances d_{Tc} and d_{Sc} is calculated to verify whether the instance $\mathcal{Q}$ belongs to the subspace. However, we observed that this ratio does not remain constant as the dimensionality increases. High dimensional data produces higher distance values, which adds bias toward the k-nearest low dimensional subspaces, since we are selecting the k smallest ratios.

To prevent this bias the ratio needs to be multiplied by a function of the dimension, in a way that the ratio between $\mathcal{Q}$ and all subspaces are comparable.

Equation 3 therefore gives the final value that will be used to compare the point-to-subspace similarities. The multiplier factor $\frac{1}{1+\sqrt{(dim)}*\ln(dim)}$ was found empirically. Finally, the 7 smallest ratios are selected.

TABLE II
ACCURACY, SENSITIVITY AND SPECIFICITY RESULTS OF THE IBH DATA SET

Classifiers	Accuracy	Sensitivity	Specificity
SBDS	$\mathbf{0.7986 \pm 0.0455}$	0.9270 ± 0.0389	0.4027 ± 0.1199
CR	0.7217 ± 0.0659	0.8171 ± 0.0747	0.4277 ± 0.1133
MCR	0.7612 ± 0.0522	0.8936 ± 0.0512	0.3527 ± 0.1605
OLA	0.7245 ± 0.0493	0.8405 ± 0.0551	0.3666 ± 0.1151
LCA	0.7177 ± 0.0582	0.8288 ± 0.0765	0.3750 ± 0.1268
A Priori	0.7231 ± 0.0631	0.8234 ± 0.0727	0.4138 ± 0.1317
A Posteriori	0.7599 ± 0.0222	$\mathbf{0.9811 \pm 0.0253}$	0.0777 ± 0.0984
MCB	0.7265 ± 0.0684	0.8315 ± 0.0884	0.4027 ± 0.1219
MLA	0.6844 ± 0.0474	0.7801 ± 0.0623	0.4055 ± 0.1268
DES-kMeans	0.7299 ± 0.0547	0.7990 ± 0.0661	$\mathbf{0.5166 \pm 0.1280}$
DES-kNN	0.7442 ± 0.0473	0.8676 ± 0.0546	0.3638 ± 0.1402
KNORA-E	0.7361 ± 0.0666	0.8180 ± 0.0754	0.4833 ± 0.1298
KNORA-U	0.7803 ± 0.0437	0.9234 ± 0.0503	0.3388 ± 0.1234
DES-EXP	0.7578 ± 0.0598	0.8585 ± 0.0867	0.4472 ± 0.1603
DES-RRC	0.7768 ± 0.0477	0.9189 ± 0.0563	0.3388 ± 0.1359
DES-MD	0.7578 ± 0.0598	0.8585 ± 0.0867	0.4472 ± 0.1603
DES-KL	0.7626 ± 0.0607	0.9009 ± 0.0772	0.3361 ± 0.1352
DES-P	0.7782 ± 0.0389	0.9072 ± 0.0472	0.3805 ± 0.1069
KNOP-E	0.7211 ± 0.0639	0.8171 ± 0.0757	0.4250 ± 0.1125
KNOP-U	0.7823 ± 0.0429	0.9351 ± 0.0455	0.3111 ± 0.1137
Meta-DES	0.7401 ± 0.0582	0.8387 ± 0.0727	0.4361 ± 0.1192
DSOC	0.7694 ± 0.0508	0.9045 ± 0.0566	0.3527 ± 0.1547

$$\mathcal{R} = \frac{d_{Tc}}{d_{Sc}} * \frac{1}{1 + \sqrt{dim} * \ln(dim)} \tag{3}$$

where dim is the dimension of the subspace.

The predictions are given by the 7 classifiers associated with each one of the 7-nearest subspaces, and a majority voting is used to define the label of the test sample.

IV. Methodology

A. The Insect Bite Hypersensitivity Data Set

A total of 196 horses comprising 49 non-affected (healthy) controls and 147 IBH-affected horses are included in the study. A complex protein microarray containing 384 extracts and pure proteins from a wide range of protein families (e.g. fruit, dairy, seeds, pollen, fungi, insects, fish) is assembled essentially as described by Marti *et al.* (2015) [22]. The data set does not contain missing values and is pre-processed according to the scheme described by Vigh-Conrad *et al.* (2010) [35]. The authors normalise the data by correcting the autofluorescence in both red and green channels [35]. They assume that for each spot, the red channel intensity (R) is the sum of the fluorescence of the second antibody - IgE (R_{IgE}) and autofluorescence (R_{AF}); while the green channel intensity (G), since is not affect by the second antibody, is, therefore, equal to its autofluorescence (G_{AF}). On slides with buffer only, they observed that $R_{AF} = mG_{AF} + b$, in other words, a linear relationship exists between the red and green channels [35]. R_{AF} and G_{AF} were, therefore, obtained by applying linear models for each allergen separately, and the resulting value of R_{AF} was subtracted from R to obtain R_{IgE}. By using this normalisation, the final intensities are centered at 0. Finally, the data is further normalised for each feature to have a range between 0 and 1.

B. Experimental Design

For evaluating the results we employ accuracy, sensitivity and specificity for each classifier. The experiment is carried out using 30 replications. For each replication, the datasets are randomly divided as 75% for training and 25% for testing. These divisions are performed preserving the proportion of samples for each class by using the stratified k-fold cross validation function in the *scikit-learn* [36] library.

The same DS methods used in Maciel-Guerra *et al.* (2019) [34] are listed in Table I. More information about each method can be found on their respective reference. Similarly to Maciel-Guerra *et al.* (2019) [34], 11 decision trees are used to compose the pool of classifiers. The size of the region of competence is set to 7 for all techniques based on k-NN.

V. Results

In our experiment, SBDS is compared with some state-of-art machine learning methods and some of the most import DS methods in the literature using a protein microarray data set. Table II shows accuracy, sensitivity and specificity results for all techniques mentioned on Section IV-B. The numbers after the "±" symbol are standard deviation. From the obtained results in Table II it is relevant to observe that all methods have learned better the majority class. Moreover, our proposed SBDS framework performed better in terms of accuracy (0.7986) than all DS methods. *A posteriori* had the highest sensitivity (0.9811) and DES-kMeans the highest specificity (0.5166).

These results indicate that SBDS is able to achieve similar results by having an embedded subspace clustering method. This allows us to verify which features were selected in each subspace to predict the label of the unknown test sample. Therefore, our method poses a great advantage in comparison

TABLE III
FREQUENCY OF PROTEINS THAT WERE SELECTED IN THE 7-NEAREST SUBSPACE SEARCH

Allergome name	Latin name	Appearance
Api g [Root]	Apium graveolens	81.70%
Cul n 10.03	Culicoides nubeculosus	78.88%
Cul o2P	Culicoides obsoletus	77.78%
Mal d [Fruit]	Malus domestica	77.78%
Cul o 7	Culicoides nubeculosus	76.98%
Cul o 7	Culicoides nubeculosus	73.89%
Cul o 7	Culicoides nubeculosus	71.40%
Culicidae Cul E	culicidae	71.24%
C0145	Culicoides obsoletus	64.07%
Cul ob 8	Culicoides obsoletus	61.81%
Mus xp	Musa x paradisiaca	60.80%
Pru p 3	Prunus persica	57.96%
Bos d 4	Bos domesticus	57.93%
Cul o1P	Culicoides nubeculosus	57.64%
Cul o 7	Culicoides nubeculosus	56.63%
Cul n 4	Culicoides nubeculosus	56.05%
Cor a 9	Corylus avellana	55.65%
Culicidae Cul C	culicidae	54.18%
Cul o 1	Culicoides nubeculosus	53.64%
Culicidae Cul D	culicidae	53.04%
Cul o 4	Culicoides obsoletus	50.80%

to DS methods in terms of giving this additional information of which are the most important features for each sample.

Table III shows which features were most selected by the 7-Nearest Subspace Search over all test samples in the 30 iterations. It is important to notice that 16 out of the 21 proteins are related to the *Culicoides sp.* allergome family that are clinically the cause of IBH in horses.

VI. CONCLUSIONS AND FUTURE WORK

This paper presents a novel framework, extending DS methods to select classifiers based on subspace clustering called SBDS. It first searches for one-dimensional clusters using a GKDE and then a merging procedure is conducted to generate subspace clusters. A classifier is then trained on each subspace. Finally, a 7-nearest subspace search is conducted to determine which classifiers will be used to make the prediction for each unknown test sample. SBDS can be classified into a density-based approach and find subspace clusters efficiently.

Accurate diagnosis of a disease is vital for a successful treatment. Moreover, precision diagnostic of each separate individual allows doctors to give personalised treatment. The proposed SBDS algorithm brings this possibility by incorporating a subspace clustering method into the DS framework. SBDS achieved the highest accuracy over all DS methods, but did not achieve the highest sensitivity and specificity. Moreover, it has the advantage of showing which features were selected for the classification of each test sample.

Future work will be conducted by verifying if similar results occur with other high-dimensional data sets and how it is possible to improve the nearest subspace search by investigating other methods. Moreover, we intend to investigate quality measures for subspaces to be able to select only interesting ones.

REFERENCES

[1] R. M. Cruz, R. Sabourin, and G. D. Cavalcanti, “Dynamic classifier selection: Recent advances and perspectives,” *Information Fusion*, vol. 41, no. Supplement C, pp. 195 – 216, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1566253517304074

[2] R. M. Cruz, H. H. Zakane, R. Sabourin, and G. D. Cavalcanti, “Dynamic ensemble selection vs k-nn: why and when dynamic selection obtains higher classification performance?” in *The Seventh International Conference on Image Processing Theory, Tools and Applications (IPTA)*, Montreal, Canada, 2017.

[3] A. S. Britto, Jr., R. Sabourin, and L. E. S. Oliveira, “Dynamic selection of classifiers - a comprehensive review,” *Pattern Recogn.*, vol. 47, no. 11, pp. 3665–3680, Nov. 2014. [Online]. Available: http://dx.doi.org/10.1016/j.patcog.2014.05.003

[4] T. Woloszynski and M. Kurzynski, “A probabilistic model of classifier competence for dynamic ensemble selection,” *Pattern Recogn.*, vol. 44, no. 10-11, pp. 2656–2668, Oct. 2011. [Online]. Available: http://dx.doi.org/10.1016/j.patcog.2011.03.020

[5] A. H. R. Ko, R. Sabourin, and A. S. Britto, Jr., “From dynamic classifier selection to dynamic ensemble selection,” *Pattern Recogn.*, vol. 41, no. 5, pp. 1718–1731, May 2008. [Online]. Available: http://dx.doi.org/10.1016/j.patcog.2007.10.015

[6] R. M. Cruz, R. Sabourin, G. D. Cavalcanti, and T. I. Ren, “Meta-des: A dynamic ensemble selection framework using meta-learning,” *Pattern Recognition*, vol. 48, no. 5, pp. 1925 – 1935, 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0031320314004919

[7] H.-P. Kriegel, P. Kröger, and A. Zimek, “Clustering high-dimensional data: A survey on subspace clustering, pattern-based clustering, and correlation clustering,” *ACM Transactions on Knowledge Discovery from Data*, vol. 3, no. 1, pp. 1–57, 2009.

[8] L. Parsons, E. Haque, and H. Liu, “Subspace clustering for high dimensional data: A review,” *SIGKDD Explorations*, vol. 6, pp. 90–105, 2004.

[9] M. Sabourin, A. Mitiche, D. Thomas, and G. Nagy, “Classifier combination for hand-printed digit recognition,” in *Document Analysis and Recognition, 1993., Proceedings of the Second International Conference on*, Oct 1993, pp. 163–166.

[10] K. Woods, W. P. Kegelmeyer, and K. Bowyer, “Combination of multiple classifiers using local accuracy estimates,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 4, pp. 405–410, Apr 1997.

[11] G. Giacinto and F. Roli, “Methods for dynamic classifier selection,” in *Proceedings 10th International Conference on Image Analysis and Processing*, 1999, pp. 659–664.

[12] ——, “Dynamic classifier selection based on multiple classifier behaviour,” *Pattern Recognition*, vol. 34, 11 2002.

[13] P. C. Smits, “Multiple classifier systems for supervised remote sensing image classification based on dynamic classifier selection,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 40, no. 4, pp. 801–813, Apr 2002.

[14] R. G. F. Soares, A. Santana, A. M. P. Canuto, and M. C. P. de Souto, “Using accuracy and diversity to select classifiers to build ensembles,” in *The 2006 IEEE International Joint Conference on Neural Network Proceedings*, 2006, pp. 1310–1316.

[15] M. C. P. de Souto, R. G. F. Soares, A. Santana, and A. M. P. Canuto, “Empirical comparison of dynamic classifier selection methods based on diversity and accuracy for building ensembles,” in *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*, June 2008, pp. 1480–1487.

[16] T. Woloszynski and M. Kurzynski, “On a new measure of classifier competence applied to the design of multiclassifier systems,” in *Proceedings of the 15th International Conference on Image Analysis and Processing*, ser. ICIAP ’09. Berlin, Heidelberg: Springer-Verlag, 2009, pp. 995–1004.

[17] B. Antosik and M. Kurzynski, “New measures of classifier competence - heuristics and application to the design of multiple classifier systems,” in *Computer Recognition Systems 4*, R. Burduk, M. Kurzyński, M. Woźniak, and A. Żołnierek, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 197–206.

[18] T. Woloszynski, M. Kurzynski, P. Podsiadlo, and G. W. Stachowiak, “A measure of competence based on random classification for dynamic ensemble selection,” *Information Fusion*,

vol. 13, no. 3, pp. 207 – 213, 2012. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1566253511000297

[19] P. R. Cavalin, R. Sabourin, and C. Y. Suen, "Dynamic selection approaches for multiple classifier systems," *Neural Computing and Applications*, vol. 22, no. 3, pp. 673–688, Mar 2013. [Online]. Available: https://doi.org/10.1007/s00521-011-0737-9

[20] A. L. Brun, A. S. Britto, L. S. Oliveira, F. Enembreck, and R. Sabourin, "Contribution of data complexity features on dynamic classifier selection," in *2016 International Joint Conference on Neural Networks (IJCNN)*, July 2016, pp. 4396–4403.

[21] J. Xiao, C. He, X. Jiang, and D. Liu, "A dynamic classifier ensemble selection approach for noise data," *Inf. Sci.*, vol. 180, no. 18, pp. 3402–3421, Sep. 2010. [Online]. Available: http://dx.doi.org/10.1016/j.ins.2010.05.021

[22] E. Marti, X. Wang, N. Jambari, C. Rhyner, J. Olzhausen, J. J. Pérez-Barea, G. P. Figueredo, and M. J. C. Alcocer, "Novel in vitro diagnosis of equine allergies using a protein array and mathematical modelling approach: a proof concept using insect bite hypersensitivity," *Veterinary Immunology and Immunopathology*, vol. 167, pp. 171 – 177, 2015.

[23] D. Agrawal, P. Bernstein, E. Bertino, S. Davidson, U. Dayal, M. Franklin, ..., and J. Widom, "Challenges and opportunities with big data: A white paper prepared for the computing community consortium committee of the computing research association," *Computing Research Association*, 2012. [Online]. Available: + http://cra.org/ccc/resources/ccc-led-whitepapers/

[24] C. Ballard and W. Wang, "Dynamic ensemble selection methods for heterogeneous data mining," in *2016 12th World Congress on Intelligent Control and Automation (WCICA)*, June 2016, pp. 1021–1026.

[25] L. I. Kuncheva, "Switching between selection and fusion in combining classifiers: an experiment," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 32, no. 2, pp. 146–156, Apr 2002.

[26] R. Miotto, F. Wang, S. Wang, X. Jiang, and J. T. Dudley, "Deep learning for healthcare: review, opportunities and challenges," *Briefings in Bioinformatics*, p. bbx044, 2017. [Online]. Available: + http://dx.doi.org/10.1093/bib/bbx044

[27] M. Verleysen and D. François, "The curse of dimensionality in data mining and time series prediction," in *Proceedings of the 8th International Conference on Artificial Neural Networks: Computational Intelligence and Bioinspired Systems*, ser. IWANN'05. Berlin, Heidelberg: Springer-Verlag, 2005, pp. 758–770.

[28] M. Radovanovic, A. Nanopoulos, and M. Ivanovic, "Hubs in space: Popular nearest neighbors in high-dimensional data," *J. Mach. Learn. Res.*, vol. 11, pp. 2487–2531, Dec. 2010.

[29] S. B.Meshram and S. M. Shinde, "A survey on ensemble methods for high dimensional data classification in biomedicine field," *International Journal of Computer Applications*, vol. 111, pp. 5–7, 02 2015.

[30] J. Tian and M. Gu, "Subspace clustering based on self-organizing map," in *Proceeding of the 24th International Conference on Industrial Engineering and Engineering Management 2018*, G. Q. Huang, C.-F. Chien, and R. Dou, Eds. Singapore: Springer Singapore, 2019, pp. 151–159.

[31] E. Muller, S. Gunnemann, I. Assent, and T. Seidl, "Evaluating clustering in subspace projections of high dimensional data," *Proc. VLDB Endow.*, vol. 2, no. 1, pp. 1270–1281, 08 2009.

[32] R. Basri, T. Hassner, and L. Zelnik-Manor, "Approximate nearest subspace search," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 2, p. 266–278, Feb. 2011. [Online]. Available: https://doi.org/10.1109/TPAMI.2010.110

[33] M. Hund, M. Behrisch, I. Färber, M. Sedlmair, T. Schreck, T. Seidl, and D. Keim, "Subspace nearest neighbor search - problem statement, approaches, and discussion," in *Similarity Search and Applications*, G. Amato, R. Connor, F. Falchi, and C. Gennaro, Eds. Cham: Springer International Publishing, 2015, pp. 307–313.

[34] A. Maciel-Guerra, G. P. Figueredo, F. J. Von Zuben, E. Marti, J. Twycross, and M. J. C. Alcocer, "Microarray feature selection and dynamic selection of classifiers for early detection of insect bite hypersensitivity in horses," in *2019 IEEE Congress on Evolutionary Computation (CEC)*, June 2019, pp. 1157–1164.

[35] K. A. Vigh-Conrad, D. F. Conrad, and D. Preuss, "A protein allergen microarray detects specific ige to pollen surface, cytoplasmic, and commercial allergen extracts," *PLOS ONE*, vol. 5, no. 4, pp. 1–11, 04 2010. [Online]. Available: https://doi.org/10.1371/journal.pone.0010174

[36] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in python," *Journal of Machine Learning Research*, vol. 12, pp. 2825 – 2830, 2011.