Prototype Reduction Algorithms Comparison in Nearest Neighbor Classification for Sensor Data: Empirical Study

Paul Rosero-Montalvo[1,2], Diego H. Peluffo-Ordóñez[1,3], Ana Umaquinga [1], Andrés Anaya[4], Jorge Serrano[5], Edwin Rosero[1], Carlos Vásquez[1], Luis Suaréz[1]
[1] Universidad Técnica del Norte - Ecuador,
[2] Instituto Tecnológico Superior 17 de Julio- Ecuador,
[3] Universidad de Nariño - Colombia,
[4] Universidad Tecnológica de Pereira-Colombia,
[5] YachayTech - Ecuador.

***Abstract*—This work presents a comparative study of prototype selection (PS) algorithms. Such a study is done over data-from-sensor acquired by an embedded system. Particularly, five flexometers are used as sensors, which are located inside a glove aimed to read sign language. Measures were taken to quantify the balance between classification performance and reduction training set data (QCR) with k neighbors equal to 3 and 1 to force the classifier (kNN) to the maximum. Two tests were used: (a)the QCR performance and (b) the embedded system decision in real proves. As result the Random Mutation Hill Climbing (RMHC) algorithm is considered the best option to choose in this data type with removed instances at 87% and classification performance at 82% in software tests, also the classifier kNN must be with k=3 to improve the classification performance. In a real situation, with the algorithm implemented. The system makes correct decisions at 81% with 5 persons doing sign language in real time.**

***Index Terms*—prototype selection,knn,sensor data**

I. Introduction

The nearest neighbor classifier is one the most used techniques for recognition tasks for their implementation simplicity. However, this classifier technique suffers some drawbacks in embedded electronics systems such as: high storage requirements, low classifier performance, low noise tolerance, among others. One of the most useful solutions consists of reducing training data set by means of selecting relevant prototypes [1].

The prototype reduction techniques (PRT) are data pre-processing methods which objective is to reduce training set for generating better representative examples and enhance the nearest neighbor rule [2]. Many prototype selection (PS) methods exist in the literature with different properties, this techniques can be classified into two different approaches, which are known as prototype selection (choosing a subset of the original training data) and prototype generation (new artificial prototypes) [3]. The prototype selection taxonomy categorize the algorithms in three aspects: Direction of search, type selection and evaluation search. For sensor networks where the information is very variant, the second aspect (Type of selection) has been chosen, this techniques seeks to retain border points (Condensation), central points (Edition) or some other set points (Hybrid). In this way, the training set inside the embedded system is considerably reduced, with focus in improving the classifier performance [4].

The most of the sensors need an analog-digital conversion (ADC) to traduce environment signals, human signals,among others in bits that the microprocessor can read. Consequently, each microcontroller has 2^n conversors. This depends on the model, cost and size of the chip. Most cases the ADC has a 10 bit resolution, that means the decimal valor between 0 to 1023. The multiavariant data collected must be analyzed to reduce the computational performance and RAM memory cost [5].

The present work makes an analysis of the different algorithms of selection of prototype with the criterion of type of selection applied to a wearable with 5 sensors flexometer that obtain data of the numbers in sign language. In order to determine the best method in relation to the computational cost, the ease of implementation, the number of possible labels and the final size of the training matrix. Finally, the performance of the classifier k Nearest Neighbor was tested with each matrix obtained with the prototype selection algorithms to determine the best PS algorithm of this data type. As result, Random Mutation Hill Climbing (RMHC)is chosen because it has a large capacity to eliminate redundant data (87%) and its error percentage is balanced (0.82%). An algorithm to consider is Condensed Nearest Neighbor (CNN) since it counts on a more adjusted percentage of error since it balances of better form the training matrix, however it doesn't remove data points in a efficient way.

The rest of this paper is structured as follows: Section 2 exposes the taxonomy to type selection algorithms. Section 3 describes the methods and algorithms used for data analysis.

978-1-5386-3894-1/17/$31.00 ©2017 IEEE

Section 4 introduces the tests and system results. Finally, Section 5 gathers some final remarks as conclusions and future work.

II. Prototype Selection Taxonomy

The taxonomy studies the principles,methods and purposes of classification. More tan 50 PS has been proposed int he literature. Prototype selection taxonomy with type selection criterion define each category shown as follows: (A) Condensation, (B) Edition and (C) Hybrid.

A. Condensation

These techniques try to storage the points closest to the decision limits, of this way is intended to preserve the accuracy of the training system. With respect to training set, is in smaller quantities since there are less border points than internal in the majority of cases [6][7].

B. Edition

Algorithms eliminate boundary points that are considered as noise or do not agree with their neighbors. However, they do not remove internal points from each data set. The classifier's accuracy is high but the reduction of training set is smaller [7][8].

C. Hybrid

The hybrid methods try to find a small set of training data while maintaining classifier performance. For it, The algorithms can eliminate internal or border points [6][8].

At the Fig. 1 shows the PS methods in year development relation with the type selection criterion.

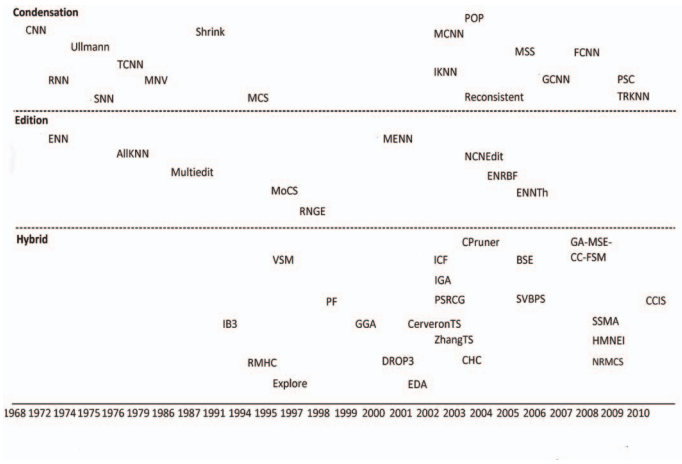

Fig. 1. Prototype selection map with type selection criterion

According to [1], where they perform tests with different data sets and following criteria: storage reduction, noise tolerance, generalization accuracy and time requirements. The following methods shown in table 1 had the best performance in each established criterion.

TABLE I
Best PS methods according [1], Type Selection: Condensation(C), Edition (E) and Hybrid(H)

Complete name	Abbr. Name	Reference	Type Selection
CHC Evolutionary algorithm	CHC	[4]	H
Generational Genetic Algorithm	GGA	[9]	H
Random Mutation Hill Climbing	RMHC	[10]	H
Decremental Reduction Optimization Procedure 3	DROP3	[1]	H
Reduced Nearest Neighbor	RNN	[11]	C
Modified Condensed Nearest Neighbor	MCNN	[11]	C
Fast Condensed Nearest Neighbor	FCNN	[12]	C
All-KNN	AllKNN	[13]	E
Condensed Nearest Neighbor	CNN	[13]	E
Edited Nearest Neighbor	ENN	[12]	E

III. Materials and Methods

This section is divided into two parts: In one hand the materials, which will detail the electrical elements used and the techniques of data acquisition. On the other hand, the methodology proposed for the PS comparison.

A. Materials

The electronics materials are: five sensor flex, one Arduino Lilypad with Atmega328 8-bit microcontroller, one LiPo battery and semiconductor wire. For data acquisition we recollected only five numbers of sign language [14]. The Fig. 2 shows the numbers in sign language.

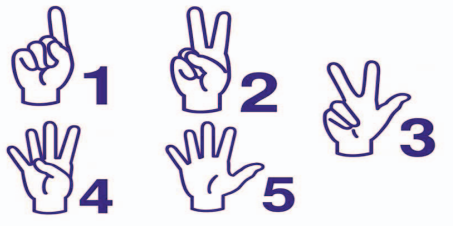

Fig. 2. Sing language numbers off one until five

The data acquisition was done with 5 people who handle the sign language, each of them did perform for 5 times the same sign for 2 minutes with different variations in the sensors flex of the fingers. The system performs a reading every 2 seconds of the 5 analogue converters. Therefore, data matrix **S** with **m**x**n** dimensions, where **m**= 1000 and **n**= 6 (one for number label). Fig. 3 shows the position of the flex sensors in each of the fingers and the Arduino Lilypad in the middle as microprocessor of the system.

B. Data Analysis Methodology

At the Fig. 4 shows the comparative PS methodology for sensor data. As first step, data was divided in two parts: training set(matrix **S** with **p**=750 and **n**= 6, where **p**<**m**) and test set with 250 points (50 for each position). The PS

Fig. 3. Electronic design system with sensor flex inside the right glove

algorithms were tested into the training set, to know the percentage of reduction of data of each one, later we validated the selection of prototypes applying a classifier algorithm multi label (kNN). Using a confusion matrix, we determine the performance of each database generated by the PS as show in Fig. 4.

The different software used that contains the PS algorithms are: R, Python and Keel.

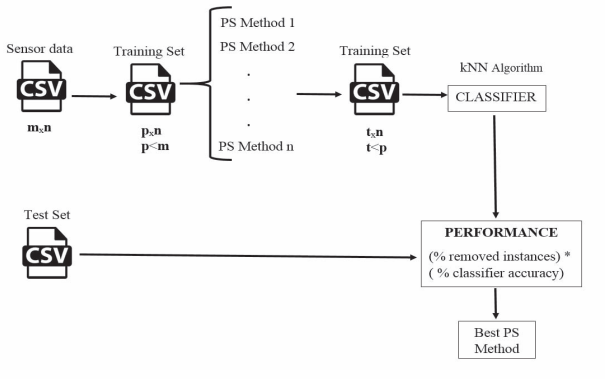

Fig. 4. Methodology designed for find the best PS algorithm

IV. Results

The results were divided into two approaches. The first is in the determination of the best PS algorithm performed outside the electronic system and the second is the tests of the electronic system already with the training matrix defined and see how the classifier works under real conditions [15].

A. PS algorithm selection

The results were divided in relation to the different tests and analyzes of each of the bases generated by the selection of prototypes, all methods use the same test set data. The tableII shows the results of the number of instances removed and the effectiveness percentage of the classifier kNN.

By having a better view of the data, a stage of reduction of dimensionality was realized by the analysis of main components (PCA) and to be able to graph the data set in 2 dimensions [16][17]. In the Fig. 5 The entire data set is shown differentiated each cluster by a different color.

TABLE II
Results of removed instances and classifer accuracy

Abbr. Name	Remov. Inst	%R. I.	Acc. Class
CHC	727	97.55%	14%
GGA	705	94%	19%
RMHC	652	87.00%	18%
DROP3	445	59.33%	10.8%
RNN	509	67.86%	12.8%
MCNN	320	42.66%	13.2%
FCNN	270	36.0%	12.0%
AllKNN	109	14.53%	10.4%
CNN	509	67.86%	12.0%
ENN	101	13.46%	9.6%

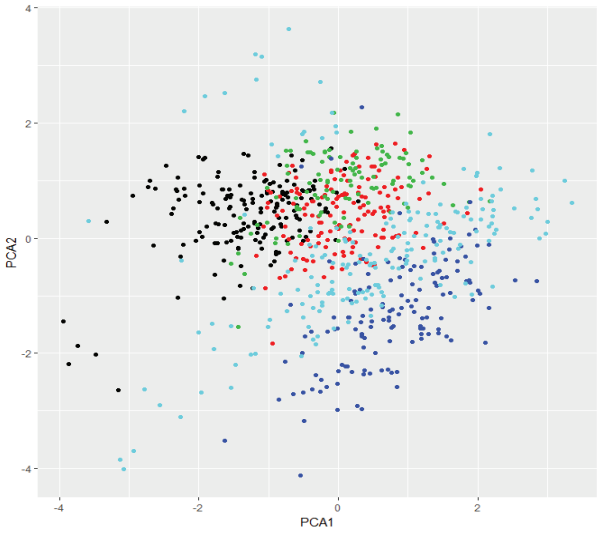

Fig. 5. All points in two dimensions (PCA)

The results of Hybrid PS algorithms applied in training data called matrix **S** are presented in Fig. 6

The results of Condensed and Edited PS algorithms applied in training data called matrix **S** are showed in Fig. 7

To determine the most adequate PS algorithm applied to sensor data, measures were made from the quantification of balance between classification performance and reduction of training data (QCR), through multiplicate the removed instances performance and classifier accuracy with k=3, indicated at table IV.

In order to maximize the training data set, an additional experiment is performed by using the classifier kNN with neighbor k = 1. For to know the algorithm that best set of data choosed for each PS algorithm for a classification task.

In order to know if the classifier performance (k=3) is homogeneous, Table V shows the error percentage of each cluster for the considered the best PS algorithms.

Once performed the QCR with tests with the different K neighbors we has conclusions of methods like MCNN, FCNN are do not have eliminate data in a large percentage and do not have a good classifier performance either. For its part AllKNN and CNN have the highest performance in the classifier (89 and 86 porcent respectively) but their percentage

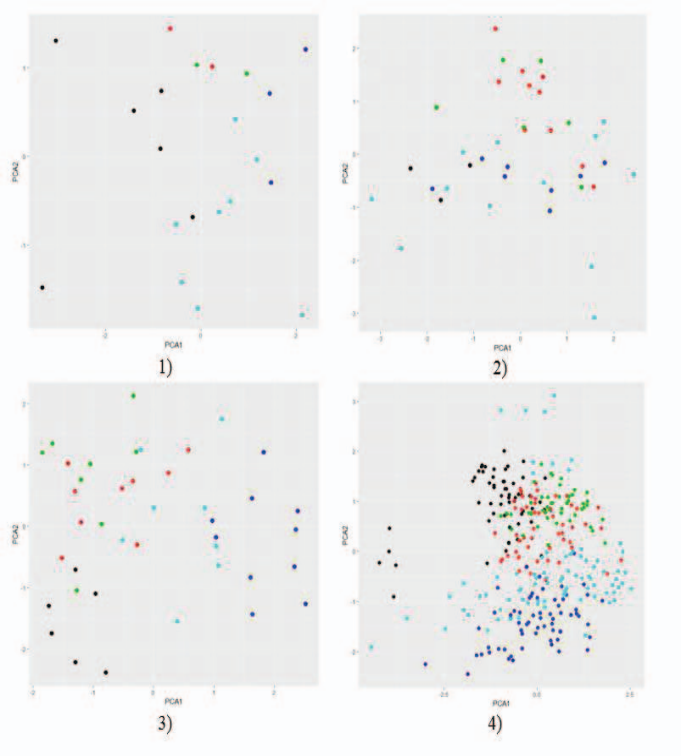

Fig. 6. Hybrid PS algorithms applied PCA in training matrix **S**, (1) HCH, (2) GGA, (3) RMHC and (4) DROP3

TABLE III
MEASURE OF QUANTIFICATION OF BALANCE BETWEEN CLASSIFICATION PERFORMANCE AND REDUCTION OF TRAINING DATA (QCR)IN PS ALGORITHMS

Abbr. Name	%R. I.	Acc. Class	QCR,k=3
CHC	0.97	0.86	0.83
GGA	0.94	0.81	0.77
RMHC	0.87	0.82	0.71
DROP3	0.59	0.90	0.53
RNN	0.68	0.87	0.59
MCNN	0.42	0.87	0.36
FCNN	0.36	0.88	0.32
AllKNN	0.15	0.90	0.13
CNN	0.68	0.88	0.60
ENN	0.13	0.91	0.11

TABLE IV
BEST PS METHODS ACCORDING [1], TYPE SELECTION: CONDENSATION(C), EDITION (E) AND HYBRID(H)

Abbr. Name	%R. I.	Acc. Class	QCR,k=1
CHC	0.97	0.78	0.75
GGA	0.94	0.80	0.75
RMHC	0.87	0.85	0.73
DROP3	0.59	0.88	0.51
RNN	0.68	0.84	0.57
MCNN	0.42	0.80	0.3
FCNN	0.36	0.81	0.29
AllKNN	0.15	0.91	0.13
CNN	0.68	0.83	0.56
ENN	0.13	0.90	0.117

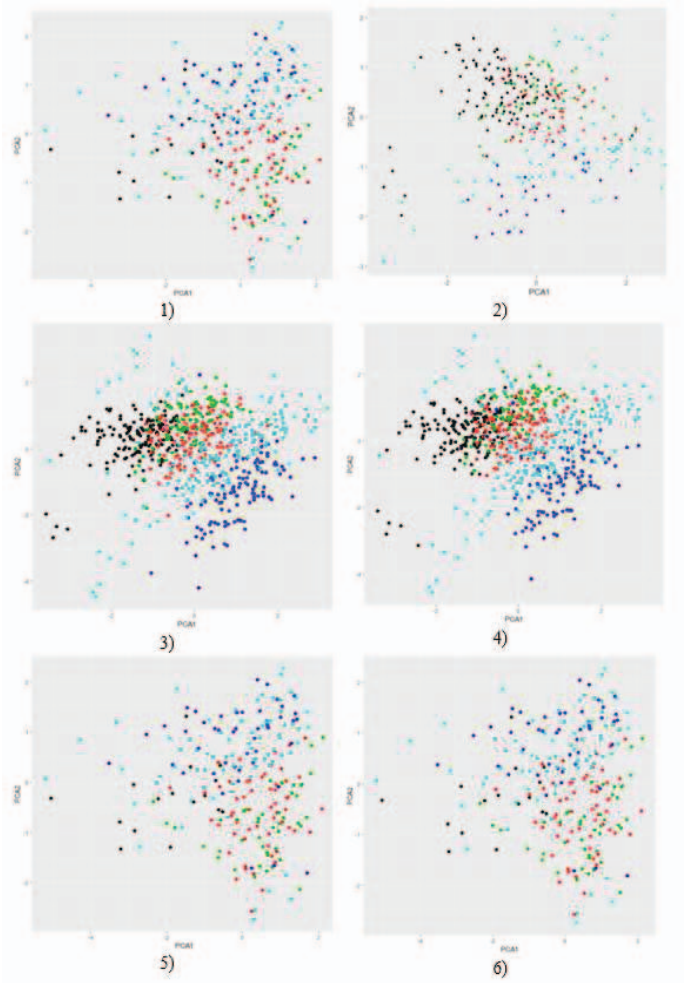

Fig. 7. Condensed and Edited PS algorithms applied PCA in training matrix **S**, (1) RNN, (2) MCNN,(3) FCNN, (4) AllKNN, (5) CNN, (6) ENN

TABLE V
ERROR PERCENTAGE OF EACH CLUSTER WITH PS ALGORITHM TRAINING DATA WITH K=3

Abbr. Name	Cl.1	Cl.2	Cl.3	Cl.4	Cl.5
CHC	1.85	17.4	17.81	44.11	60.6
GGA	7.69	34.8	12.9	27.7	37.5
RMHC	10.9	42.5	25	14.25	22
RNN	5.55	22.5	20.68	14.63	14.81
CNN	5.45	21.42	17.24	14.63	13.20

of data reduction is very low, where they do not surpass 20%. DROP3 is an average algorithm in reduction and classification, its drawback is in the processing time required, in relation to embedded systems where its processor is limited or in the increase of the size of database which could cause problems in response times. GGA and RNN have high averages of reduction of prototypes and classification, but in comparison with CHC or RMHC do not have the same performance.

The best PS methods are CHC and RMHC for sensors data . For the following reasons, CHC has a very high data reduction capacity (97%) and does not lose the performance of the

classifier (0.85%), the disadvantage is that it does not balance for each data label. This results in the loss of classification effectiveness in some clusters. RMHC is chosen because it has a large capacity to eliminate redundant data (87%) and its error percentage is balanced (0.82%). An algorithm to consider is CCN since it counts on a more balanced percentage of error since it balances of better form the training matrix, however it doesn't remove data points in an efficient way.

B. System performance

With the CHC and MRHC training set inside in the microcontroller memory, five people used the glove, each user did 20 numbers (of the one to five) in sign language and via serial communication the system decide which number do the person. The system time decision with knn algorithm implemented was almost 2 seconds for each decision. CHC matrix has the classifer performance at 78%, the reason is that CHC matrix has only two data in cluster two. For this reason with k neighbors =3 the system fail in several times. For other hand RMHC matrix has at least five data for each cluster. The performance was at 81%. The higher inconvenient in this test was the hand size, this aspect influence in the flexion of the sensor when the user make the sign language.

V. Conclusion and Future Work

The embedded systems are coming to be important devices to help or acclimatization daily our activities, this entails, that these it must be intelligent and learn from persons. The limitations lies in their computational resources. In this way, if the system choose only the sufficient data for giving the correct answers with minimal error, we are reducing the battery, cpu, ram memory among others. PS algorithms es a great importance to training data set suitable.

The PS algorithms are a data cleaning tool very efficient, each of them has a different criteria for data selection. In this case, when the data are multivariate and all data points has the same value range, the CHC algorithm anc RMHC are the most convenient algorithms with the removed instance criteria and classifier performance. CNN had the best average error, but it did not reduce the training set efficiently. In test with embedded systems RMHC showed to be the best PS choose.The classifier kNN for make correct decision should be k=3, in test with k=1 the accuracy low in each PS algorithm.

If our criteria is more in focus in classifier performance and we have the suitable storage, CNN should be your PS algorithm. However in this system when only have a 2 Kilobytes of memory, the training set size is very important.

The training matrix size causes a processing time, algorithms like DROP or GGA this time can be a strong aspect to use them, since is very significant in relation with the others PS algorithms. Our matrix has a medium size (600 points) and delayed in returning the new matrix in almost 15 minutes, the others PS algorithms did not exceed the minute.

As future work will be to implement a PS algorithm within an embedded system, so that individuals can select their own training data.

Acknowledgment

The authors would like to thank to Universidad Técnica del Norte and Instituto Tecnológico 17 de Julio-Yachay for giving the authorization of electronic labs to use.

References

[1] S. Garcia, J. Derrac, J. Cano, and F. Herrera, "Prototype selection for nearest neighbor classification: Taxonomy and empirical study," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 3, pp. 417–435, 2012.

[2] I. Triguero, J. Derrac, S. Garcia, and F. Herrera, "A taxonomy and experimental study on prototype generation for nearest neighbor classification," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 42, no. 1, pp. 86–100, 2012.

[3] B. Krawczyk, I. Triguero, S. Garca, M. Woniak, and F. Herrera, "A first attempt on evolutionary prototype reduction for nearest neighbor one-class classification," in *2014 IEEE Congress on Evolutionary Computation (CEC)*, 2014, pp. 747–753.

[4] J. R. Cano, F. Herrera, and M. Lozano, "Using evolutionary algorithms as instance selection for data reduction in kdd: an experimental study," *IEEE Transactions on Evolutionary Computation*, vol. 7, no. 6, pp. 561–575, Dec 2003.

[5] P. R. Montalvo, S. Nunez, S. Realpe, V. Alvear, L. Beltran, and C. Rosado, "Internet de las cosas y redes de sensores inalambricos:review," vol. 73, pp. 31–37.

[6] C. Domeniconi, J. Peng, and D. Gunopulos, "Locally adaptive metric nearest-neighbor classification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 9, pp. 1281–1285, Sep 2002.

[7] S. Pan, J. Wu, X. Zhu, and C. Zhang, "Graph ensemble boosting for imbalanced noisy graph stream classification," *IEEE Transactions on Cybernetics*, vol. 45, no. 5, pp. 954–968, May 2015.

[8] S.-W. Kim and B. J. Oommen, "Creative prototype reduction schemes: a taxonomy and ranking," in *IEEE International Conference on Systems, Man and Cybernetics*, vol. 7, Oct 2002, pp. 6 pp. vol.7–.

[9] G. Salvador, C. Jose, and H. Francisco, "A memetic algorithm for evolutaionary prototype selection selection: A scaling up approach," *Pattern Recognition*, pp. 2693–2709, 2008.

[10] L. I. Kuncheva, "Editing for the k-nearest neighbors rule by a genetic algorithm," *Pattern Recogn. Lett.*, vol. 16, no. 8, pp. 809–814, Aug. 1995. [Online]. Available: http://dx.doi.org/10.1016/0167-8655(95)00047-K

[11] G. Gates, "The reduced nearest neighbor rule (corresp.)," *IEEE Transactions on Information Theory*, vol. 18, no. 3, pp. 431–433, May 1972.

[12] P. Hart, "The condensed nearest neighbor rule (corresp.)," *IEEE Transactions on Information Theory*, vol. 14, no. 3, pp. 515–516, May 1968.

[13] J. Ullmann, "Automatic selection of reference data for use in a nearest-neighbor method of pattern classification (corresp.)," *IEEE Transactions on Information Theory*, vol. 20, no. 4, pp. 541–543, Jul 1974.

[14] P. Rosero-montalvo, D. Jaramillo, S. Flores, D. Peluffo, V. Alvear, and M. Lopez, "Human Sit Down Position Detection Using Data Classification and Dimensionality Reduction," *Advances in Science, Technology and Engineering Systems Journal*, vol. 2, no. 3, pp. 749–754, 2017. [Online]. Available: http://astesj.com/v02/i03/p95/

[15] S. Nunez-Godoy, V. Alvear-Puertas, S. Realpe-Godoy, E. Pujota-Cuascota, H. Farinango-Endara, I. Navarrete-Insuasti, F. Vaca-Chapi, P. Rosero-Montalvo, and D. H. Peluffo, "Human-sitting-pose detection using data classification and dimensionality reduction," in *2016 IEEE Ecuador Technical Chapters Meeting (ETCM)*, Oct 2016, pp. 1–5.

[16] P. Rosero-Montalvo, P. Diaz, J. A. Salazar-Castro, D. F. Pena-Unigarro, A. J. Anaya-Isaza, J. C. Alvarado-Perez, R. Theron, and D. H. Peluffo-Ordonezez, *Interactive Data Visualization Using Dimensionality Reduction and Similarity-Based Representations*, 2017, pp. 334–342.

[17] P. D. Rosero-Montalvo, D. F. Peña-Unigarro, D. H. Peluffo, J. A. Castro-Silva, A. Umaquinga, and E. A. Rosero-Rosero, *Data Visualization Using Interactive Dimensionality Reduction and Improved Color-Based Interaction Model*. Cham: Springer International Publishing, 2017, pp. 289–298.