Available online at http://www.journalijdr.com

IJDR

ISSN: 2230-9926

International Journal of Development Research
Vol. 11, Issue, 08, pp. 49769-49772, August, 2021
https://doi.org/10.37118/ijdr.22739.08.2021

RESEARCH ARTICLE OPEN ACCESS

METHODOLOGY FORM TYPE CLASSIFICATION AND STEPPING IN BAROPODOMETRIC SYSTEMS

Ernande F. Melo*, W. Muller, Bruno Simões C Ferreira, Kleycson dos Santos N. Júnior, Manoel SS Azevedo and Andreza Bastos Mourão

School of Technology (EST) – University of the State of Amazonas (UEA) – Manaus – AM – Brazil

ARTICLE INFO

Article History:
Received 20th May, 2021
Received in revised form
11th June, 2021
Accepted 28th July, 2021
Published online 29th August, 2021

Key Words:
Baropodometric Systems, Sensors, Machine Learning, Classification, Neural Networks, Cost and Processing Time.

Corresponding author: Ernande F. Melo,

ABSTRACT

Baropodometric systems are using the area of Artificial Intelgence (AI), more specifically machine learning to classify type and step from data collected by sensors. We observed, in most of the works found in the literature, the use of MLP-type Neural Networks for the classification process, which requires a large amount of data and a high computational cost and processing time. This article proposes a methodology that goes in the opposite direction, that is, low data volume with low processing time and cost, in addition to a dynamic configuration of the classification environment, through the insertion or removal of modules, according to quantity and quality of the data collected by the sensors.

Citation: **Ernande F. Melo, W. Muller, Bruno Simões C Ferreira, Kleycson dos Santos N. Júnior, Manoel SS Azevedo and Andreza Bastos Mourão. *2021.*** "Methodology form type classification and stepping in baropodometric systems", *International Journal of Development Research*, 11, (08), xxxxx-xxxxx.

INTRODUCTION

ML applications are reaching an increasing number of knowledge areas, in particular Baropodometric Systems for type classification and stepping for the identification of postural problems in human beings [1][2]. The literature shows the use of MLP-type Neural Networks [3][4] for classification in most Baropodometric Systems. Neural Networks, although they present a good performance in the classification process, are "black box" solutions, they require a large knowledge base, Deep Learning [5][6] in addition to a high cost and processing time, being unfeasible for use in embedded systems, which are lighter and more practical to work with data collected by sensors. This work proposes an alternative for classification that requires a smaller knowledge base, low cost and processing time, in addition to allowing its use in embedded systems. Another feature of this model is the flexibility of using modules, depending on the quality and quantity of data collected by the sensors. Figure 1 illustrates the proposed model. The model is composed of a statistical module (Correlation) [7] [8], two Machine Learning (KNN) [9] [10] and Decision Tree [11] [12], a Graph module, where an approach is made. graphic and positional data collected by the sensors. Each module produces a ranking, in the divide and conquer strategy, and the final result is given by the higher frequency of partial results. In Section II a description of the modules is presented. Section III describes an application of the model. Section IV describes the results. In Section V the conclusions are presented.

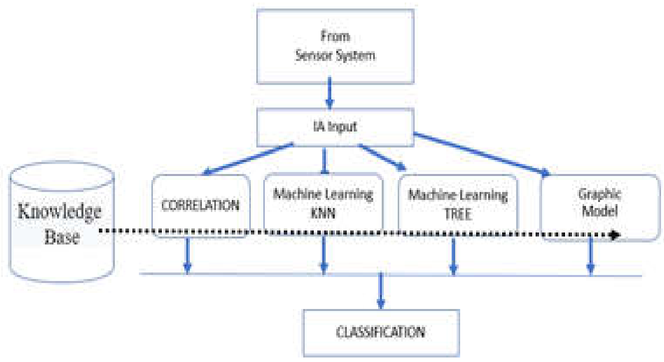

Figure 1. Proposed Model. Four modules. Each module produces a ranking, in the divide and conquer strategy, and the final result is given by the higher frequency of partial results

MATERIAL AND METHODS

The context in which the modules that make up the classification system in Baropodometric projects are developed is as follows:

- Data were collected from 13 sensors distributed in insole, called raw data.
- Each individual (U) participating in the experiment presents a set of 700 samples from each of the 13 sensors and their respective classification of type and footfall. Table I shows the classification summary for all individuals participating in the experiment and table II shows the data for a given participant (U1)

The classification, Table I, and the availability of data for each participant in the experiment Table2, are described in [2]. For the classification, as already mentioned, the MPL Neural Networks [1] [2] were used.

Sx and Sy represent the standard deviation, respectively, of the x and y variables.

For raw data (without covariance or standard deviation) the Pearson correlation coefficient (r) is given by Equation 2.

$$r = \frac{n(\Sigma xy) - (\Sigma x)(\Sigma y)}{\sqrt{[n\Sigma x^2 - (\Sigma x)^2][n\Sigma y^2 - (\Sigma y)^2]}} \quad \text{Eq.(2)}$$

An estimate to classify the degree of correlation depends on the problem addressed, but in general, it can be classified as described in Table III

Table I. Classification of the type and step for the 15 participants of the Baropodometric experiment

Nome	U1	U2	U3	U4	U5	U6	U7	U8	U9	U10	U11	U12	U13	U14	U15
Tipo	Cavo	Chato	Chato	Normal	Cavo	Cavo	Normal	Normal	Cavo	Normal	Cavo	Normal	Neutro	evemente Chat	Cavo
Pisada	Pronado	Pronado	Pronado	Neutra	Supinado	Supinado Leve	Neutro	Neutro	Supinado	Neutra	Pronado	Neutra	Normal	Pronado	Neutra

Dara: Source: [2]

Table II. Sample data collected from the 13 sensors for user U1. Raw data

1	sensor 1	sensor2	sensor 3	sensor 4	sensor 5	sensor 6	sensor 7	sensor8	sensor 9	sensor 10	sensor 11	sensor 12	sensor 13	Class
2	2.21	2.22	2.21	2.21	2.21	2.31	2.36	2.21	2.2	2.24	2.32	2.24	2.43	11
3	2.2	2.21	2.2	2.21	2.21	2.3	2.36	2.2	2.2	2.23	2.31	2.24	2.42	11
4	2.21	2.22	2.21	2.21	2.22	2.31	2.36	2.21	2.21	2.24	2.32	2.24	2.43	11
5	2.21	2.21	2.21	2.22	2.21	2.31	2.36	2.21	2.21	2.24	2.32	2.24	2.43	11
6	2.21	2.21	2.21	2.22	2.21	2.31	2.39	2.21	2.21	2.25	2.31	2.24	2.43	11
7	2.2	2.22	2.2	2.21	2.21	2.31	2.37	2.21	2.21	2.25	2.32	2.24	2.43	11
8	2.2	2.21	2.21	2.21	2.21	2.31	2.37	2.2	2.21	2.25	2.31	2.24	2.43	11
9	2.21	2.21	2.21	2.21	2.21	2.31	2.36	2.21	2.21	2.25	2.32	2.24	2.43	11
10	2.21	2.21	2.21	2.22	2.21	2.31	2.37	2.21	2.21	2.24	2.31	2.24	2.43	11
11	2.21	2.21	2.22	2.22	2.21	2.31	2.38	2.21	2.21	2.25	2.31	2.24	2.43	11
12	2.21	2.21	2.22	2.21	2.21	2.31	2.37	2.2	2.21	2.25	2.31	2.24	2.43	11
13	2.21	2.21	2.22	2.22	2.21	2.3	2.38	2.21	2.21	2.25	2.31	2.24	2.43	11
14	2.21	2.21	2.21	2.22	2.21	2.31	2.35	2.21	2.21	2.24	2.32	2.24	2.43	11
15	2.21	2.21	2.21	2.22	2.21	2.31	2.39	2.2	2.21	2.24	2.32	2.24	2.43	11
593	2.21	1.77	2.25	2.22	2.24	2.28	2.36	2.38	2.22	2.17	2.31	2.18	2.42	11
594	2.2	1.93	2.15	2.23	2.23	2.28	2.37	2.2	2.38	2.65	2.44	2.19	2.42	11
595	2.19	2.07	2.21	2.21	2.23	2.28	2.35	2.07	2.22	2.19	2.22	2.22	2.42	11
596	2.19	2.14	2.2	2.21	2.21	2.27	2.35	2.15	2.21	2.21	2.27	2.2	2.42	11
597	2.19	2.17	2.2	2.21	2.21	2.28	2.34	2.2	2.21	2.22	2.28	2.2	2.42	11
598	2.19	2.18	2.2	2.21	2.2	2.28	2.35	2.2	2.2	2.22	2.28	2.2	2.42	11
599	2.04	2.34	2.45	2.21	2.18	2.28	2.35	2.23	2.2	2.21	2.27	2.2	2.41	11
700	2.28	2.75	1.95	2.19	2.14	2.28	2.35	2.2	2.03	2.17	2.29	2.2	2.42	11

Source:[2]

Description of Modules: The proposed classification system has 4 modules. A statistical module (correlation), two inductive machine learning modules, KNN and Tree, and a graphical module.

Correlation: The term correlation means two-way relationship and is used in statistics to designate the degree of force that holds two sets together (X, Y). In the context of this work, strength refers to the degree of linear dependence, measured by the correlation coefficient and (X, Y) the data sets refer to the data set generated by the sensors and that make up the knowledge base of the proposed model (X), and the input data set to sort (Y). The correlation will be stronger the closer the coefficient is to –1 or +1, and will be weaker the closer the coefficient is to zero. We then do the type of classification and step by evaluating the calculation of the correlation coefficient. For this, we use the calculation of the Pearson linear correlation coefficient (r) given by Equation 1.

$$r = \frac{S_{xy}}{S_x S_y} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}} \quad \text{Eq (1)}$$

Where:
SXY_{is} the covariance.

Machine Learning: Figure 2 shows the scheme of a ML, which basically consists of a knowledge base and a learning algorithm. Data are structured in attributes, and in the case of supervised learning, the class of each set of attributes [9] [10][11]. A part of this data is used in the training of the learning algorithm and another part is used to test the performance evaluation of the LM. The more data used in training the better the performance of the LM and the more data in training can be obtained with the increment of the base.

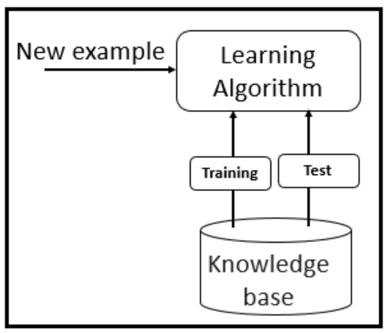

Figure 2. Schematic representation of a LM. with a knowledge base with one dataset for training and one for testing, and a learning algorithm

Table III. Estimate for ranking the degree of correlation

degree of correlation Absolute Value	*Classification*
0.9 -1.0	very strong
0.7 - 0.9	strong
0.5 (0.7)	moderate
0,3 - 0,5	weak
0.0 - 0.3	null

Among the learning algorithms, the most popular are: KNN, Decision Tree, SVM and ANN [9] [10] [11]. For the experiment to be shown in this article, we chose the KNN and Decision Tree for its lightness in both the algorithm and the knowledge base, transparency, flexibility and speed.

KNN Learning Algorithm: *KNN (K-Nearest Neighbors*) [12] [13] is a supervised learning algorithm that classifies inputs using a prediction method that uses the distance between the current input and its k nearest neighbors in the training set to define what will be the result of its prediction and consequent classification. These distance measures can be: Euclidean, Manhattan and Minkowski distance [13]. Given two vectors X= ($x_1, x_2,, x_3, \ldots, x_n$) and Y=($y_1, y_2, y_3,$ .. Yn) the Minkowski distance is defined *by Equation 3, as:*

$$D(X,Y) = \left(\sum_{i=1}^{n} |x_i - y_i|^p\right)^{1/p} \qquad \text{Eq.(3)}$$

If p=1 the distance is Euclidean, if p=2 it is Manhattan

With the chosen distance, the algorithm calculates the distance of the new instance with the training data, and gathers the k closest instances. If k=3, for example, the 3 closest instances are selected. Once the instances are selected, the classification is determined such that the class with more representatives in the neighborhood ends up being the class of the new instance. Figure 2 illustrates the process of choosing the class for 3 and 6 neighbors.

Decision tree learning algorithm: Decision Tree is an induction learning algorithm. The decision tree induction process has the function of recursively partitioning a set of training until each subset obtained from this partitioning contains cases of a single class [14].

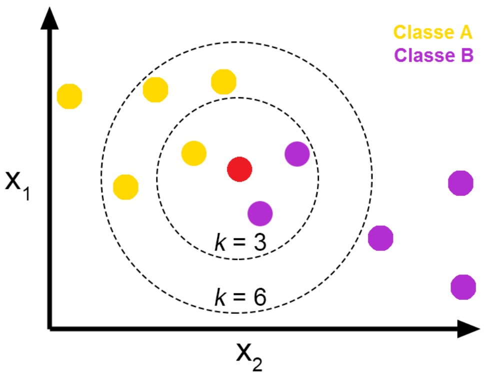

Figure 2. Choice of two possible class for 3 and 6 closest neighbours

A decision tree takes as input an object or situation described by a set of attributes, which in this case are the readings of the set of sensors distributed in insoles and returns a decision on the classification of the type and tread. The classification tree is the result of asking an ordered sequence of questions, and the questions asked at each step in the sequence depend on the answers to the previous questions. The sequence ends in a preview of the class. The starting point of a classification tree is called the root node and consists of the entire learning set, and is at the top of the tree. A node is a subset of the attribute set, and it can be terminal or non-terminal. A non-terminal node is a node that splits into child nodes. Such division is determined by a condition on the value of a single attribute, which will divide the examples according to the condition, in other nodes. A node that does not split is called a terminal node, and it is assigned a class. Each example in the set falls into one of the terminal nodes [14]. Figure 3 illustrates the decision tree classification.

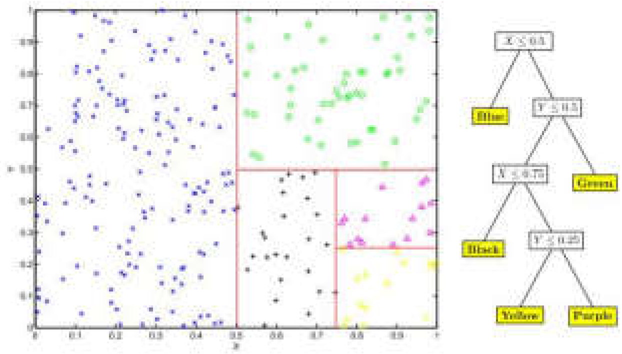

Figure 3. Example of partitions in decision trees, Source [14].

Graphic Template: Often the set of attributes for the various knowledge base entries have a strong linear dependence, that is, despite the various pre-processing resources, it is not possible to find a linear separation between them, which makes the classification process difficult. new examples. In this case, the dimensional change of the problem can minimize the classification errors of new examples. Figure 4 illustrates the dimensional shift from numerical data to a graphical dimension. Data is viewed from a visual perspective. Each rectangle in the insole design represents the intensity of a sensor.

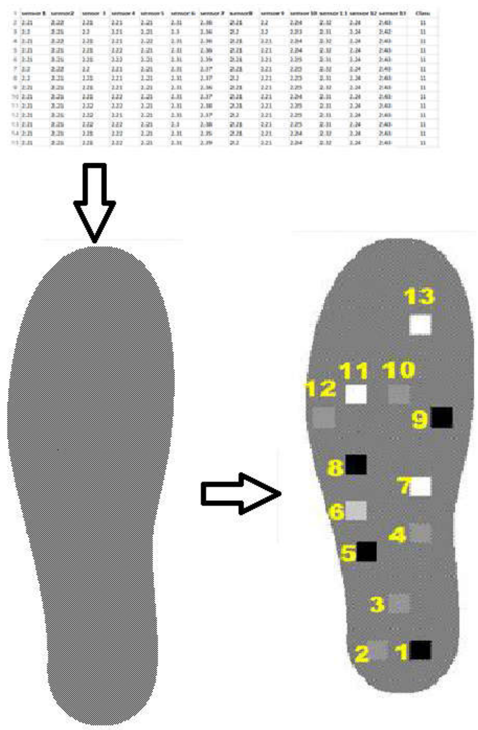

Figure 4. Mapping the data generated by the sensors to a graphical dimension

For the classification, specific image recognition algorithms or KNN and Decision Tree or even brute force can be used.

Application of the Method: The scenario in which the method was applied is described in section II. There are 4 steps to wine tasting:

Steps

Step 1: From the 15 users that participate in the experiments, we chose the first ones that don't repeat type classification and stepping, to compose the knowledge base. Table IV shows the selection (in blue) . 5 users are chosen to compose the base, with their proper ratings.

Table IV. Selection of users to compose the Knowledge Base, in blue

Nome	JJJ	Melissa	Lucas M	Stevan	Rodrigo	Régis	Okida2	Okida	Fabricio	Ismael	Hugo	Mário	Fenato	Wesley	Pontes
Tipo	Cavo	Chato	Cavo	Normal	Cavo	Cavo	Normal	Cavo	Cavo	Normal	Cavo	Normal	Normal	Cavo	Cavo
Pisada	Pronado	Pronado	Pronado	Neutra	Supinado	Supinado	Neutro	Supinado	Supinado	Neutra	Pronado	Neutro	Neutro	Supinado	Neutra

Step 2: Apply modules individually:

Correlation: in this case the correlation is calculated in pairs of attributes from the base and the data entered for classification. Then, the correlation analysis and classification is performed by selecting the maximum correlation.

KNN: With the knowledge base established, new users are classified using the KNN algorithm for 1 nearest neighbor KNN-1.

Decision Tree: Once the knowledge base is established, classification is performed using the Decision Tree algorithm.

Application of the graphic model; after the mapping shown in Figure 4. a comparison is made between the images that make up the knowledge base and the example to be classified. Although the image base is small, the analysis is positional of the intensity of the pixels that represent the sensors.

Step 3: After applying the modules and obtaining the individual classification of each one, select the classification with the highest occurrence.

Development environment: The experiments were carried out: in the following environment.

Hardware: Notebook SONY VAIO Notebook, 4GB, 2.5GHz, 64-bit.
Software: Framework Spyder3, Python 3.7.3, numpy, pandas 1.16.4, scipy.org, sklearn-0.21.2, scikit-learn.org, OpenCV 3.4.

RESULTS

Various combinations of modules were used and the results are described in Table V where various combinations of modules and classification performance are shown. Two simple performance measures were used, one being the number of correct answers in relation to the number of participants who are not in the base composition, in this case, 10. Another measure is the number of correct answers in relation to the total number of participants, who are 15.

Table V. System performance for various composition of the proposed modules

Composition	Classification/10	Rating/15
DT3	0.4	0.6
KNN.	0.4	0.6
Correlation	0.6	0.73
Graphic	0.6	0.73
DT3+KNN+Correlation	0.7	0.8
DT3+KNN+Correlation+Graph	0.8	0.87

The results show that depending on the quality of the type and step data generated by the sensors, the system can be adjusted, with insertion or removal of modules and change of parameters to increase performance. A response time of around 10 seconds was also observed.

DISCUSSION

The purpose of this article is to show an alternative to type and step classification in Baropodometric systems. As observed in the literature, Multi-Layer Perceptron Neural Networks (MLP) are frequently used for classification. These networks require a high computational cost and a knowledge base with a large amount of data, in addition to not being transparent (black box). The data manipulated in this work was provided by the project described in reference [2]. A reference to evaluate the model proposed in this work is that we used only raw data provided by sensors in the insoles, while in the original project [15] it was necessary to add inertial data to improve the performance for classification. So, the results show that it is possible to build a lighter, more flexible, faster and more shippable Baropodometric classification system.

ACKNOWLEDGEMENTS

This article is the result of the RD&I Smart IoT Gait Analysis System project, carried out by the University of the State of Amazonas (UEA), in partnership with Samsung Eletrônica da Amazônia Ltda., using resources from Federal Law nº 8.387/1991, being its dissemination and publicity in accordance with the provisions of article 39 of Decree No. 10.521/2020.

BIBLIOGRAPHICAL REFERENCES

[1] Ramirez-Bautista, Julian Andres et al. Classification of plantar alterations foot by fuzzy cognitive maps against multi-layer perceptron neural network. ELSEVIER Biocybernetics and Biomedical Engineering. 2020
[2] Birth, Lucas Medeiros Souza. Development of a device for step classification using inertial sensors. Federal Technological University of Paraná Ponta Grossa - PR 2019
[3] SILVA, I.; SPATTI, D.; FLAUZINO, R. Artificial neural networks for engineering and applied sciences. São Paulo: Artlibert Editora Ltda, 2010.
[4] WAFAI, L.; et al. Automated Classification of Plantar Pressure Asymmetry during Pathological Gait Using Artificial Neural Network. In: MIDDLE EAST CONFERENCE ON BIOMEDICAL ENGINEERING, Doha. Proceedings. Doha: IEEE, 2014. p. 220 - 223.
[5] CHOLLET, François. Deep Learning with Python. Manning Publications Co. 2018.
[6] Y. LeCun, Y. Bengio, G. Hinton. Deep Learning. Nature 521(7), 436-444. 2015.
[7] AJIzenman, Modern Multivariate Statistical Techniques. Springer, 2008.
[8] MAGALHÃES, MN LIMA, A. Notions of Probability and Statistics. São Paulo. Edusp. 7a. ed. 2015
[9] GÉRON, Aurelien. Hands-on machine learning with Scikit-Learn and TensorFlow: concepts, tools, and techniques to build intelligent systems. O'Reilly Media, Inc.", 2017
[10] KOTISIANTIS SB Supervised machine learning: A review of classification techniques. Emerging Artificial Intelligence. Applications in Computer Engineering. IOS Press Amsterdam, The Netherlands.2007.
[11] HUANG, GB; Wang, DH; Lan, Y. Extreme learning machines: a survey. International Journal of Machine Learning and Cybernetics, Springer, volume 2, no.2, p. 107–122, 2011.
[12]]HU, Li-Yu et al. The distance function effect on k-nearest neighbor classification for medical datasets. SpringerPlus, v. 5, no. 1, p. 1304, 2016.
[13] Documentation of Scikit Learn – Kneighbors Classifier. Available at: https://scikit-learn.org/stable/modules/generated/ sklearn.neighbors.KNeighborsClassifier.html. Visited on 10/08/2021.
[14] Barbosa, Juliana Moreira. Senna, Tiago Garcia .C. Tavares, Andrea Iabrudi. Decision Tree Classification Methods. Federal University of Ouro Preto.

****