Towards Detecting Freezing of Gait Events Using Wearable Sensors and Genetic Programming

Adane Nega Tarekegn(✉), Faouzi Alaya Cheikh, Muhammad Sajjad, and Mohib Ullah

Department of Computer Science, Norwegian University of Science and Technology (NTNU), Gjøvik, Norway
adane.n.tarekegn@ntnu.no

Abstract. Freezing of gait (FOG) is one of the most common manifestations of advanced Parkinson's disease. It represents a sudden interruption of walking forward associated with an increased risk of falling and poor quality of life. Evolutionary algorithms, such as genetic programming (GP), have been effectively applied in modelling many real-world application domains and diseases occurrence. In this paper, we explore the application of GP for the early detection of FOG episodes in patients with Parkinson's disease. The study involves the analysis of FOG by exploiting the statistical and time-domain features from wearable sensors, followed by automatic feature selection and model construction using GP. Efforts to use data from wearable sensors suffer from challenges caused by imbalanced class labels, which affect the task of GP model development. Thus, the cost-sensitive approach is incorporated into GP to tackle the imbalanced problem. The standard metrics, such as sensitivity, specificity, and F1-score, were used for testing the final model. With 30 repetitions, the average performance of the GP model has shown promising results in detecting the occurrence of FOG episodes in Parkinson's disease.

Keywords: Freezing of Gait · wearable sensors · Genetic programming · Cost-sensitive learning · Parkinson's diseases · Machine learning

1 Introduction

Predictive models are designed to support medical staff and patients with decisions for screening and diagnosing, early intervention and prevention of diseases, providing patient risk stratification, or making lifestyle changes [1]. Building a clinical predictive model requires data that are representative of a specific population or domain and reliably recorded within the time frame of interest for the prediction. Such models are generally defined as either likelihood of disease or disease group classification, detection or identification of disease cases, the diagnostic or prognostic, likelihood of response or risk of recurrence [2]. This

L. Rutkowski et al. (Eds.): ICAISC 2023, LNAI 14125, pp. 274–285, 2023.
https://doi.org/10.1007/978-3-031-42505-9_24

study focuses on building a predictive model for detecting freezing of gait (FOG) in Parkinson's disease (PD). PD is a complex and progressive brain disorder, which is characterized by a combination of motor symptoms such as bradykinesia, resting tremor, rigidity, freezing of gait, and non-motor symptoms, such as psychiatric disorders, pain, and fatigue [3]. Among these, FOG is one of the most common and disturbing motor manifestations in the advanced stages of PD. It is a common gait impairment or activity disorder often characterized by the inability to walk and severe difficulty in locomotion with an increased risk of falling. FOG can be defined as a "brief, episodic absence or marked reduction of forward progression of the feet despite having the intention to walk" [4]. The diagnosis and treatment of FOG are challenging for healthcare professionals due to the heterogeneity of the patient, and FOG may not manifest during hospital visits [4]. Wearable technologies based on inertial sensors and machine learning have been widely used for their automatic detection and predictions through different sensor placements on the human body. Such predictions can be important in quantifying the characteristics of gait disorders and freezing events in PD. Despite the promising results with the detection of FOG using machine learning techniques, there are still open issues for improvement in terms of stability and generalization capability to implement real-time FOG detection systems using machine learning techniques and/or evolutionary algorithms.

Evolutionary algorithms, such as genetic programming (GP), have been successful in automatically evolving variable-length computer programs to solve practical problems [5,6]. Unlike Blackbox methods such as neural networks, GP has the advantage of being human-friendly and providing an explicit mathematical formula as its output. However, like other machine learning models, the performance of GP can be highly affected by the presence of imbalanced class labels in the data that may lead to high-false negative rates. The resampling methods, such as under-sampling and over-sampling, are popular approaches for dealing with an imbalanced data problem. However, both approaches can cause either the loss of important information or adding irrelevant classification data that can affect the prediction accuracy for minority examples in the imbalanced dataset [7]. Thus, in this study, cost-sensitive learning in GP (CSGP) is proposed to alleviate the imbalanced problem and predict FOG episodes in PD. The main motivation for using GP over traditional machine learning techniques is twofold. Firstly, GP performs an implicit feature selection automatically, discovering relationships among variables and producing fully explorable models. Secondly, GP can present its result by generating interpretable models in the form of parse trees or mathematical equations, which are relatively easy to explain.

2 Preliminaries and Related Works

2.1 Wearable Sensors-Based FOG Detection

Wearable sensors have been used in the detection and analysis of FOG episodes to characterize their severity and to enable the application of rhythmic auditory cueing. Rhythmic Auditory Stimulation (RAS) is applied to produce a rhythmic ticking sound upon detection of a FOG episode. Several research papers proposed wearable systems based on motion sensors for the detection and treatment of FOG with auditory stimulation [8]. Marc B et al. [9] proposed a wearable system for FOG detection that provides the rhythmic auditory signal. They used accelerometers placed on the ankles to evaluate the frequency components of motion. Authors in [10] proposed a machine learning algorithm for online FOG detection and treatment using a smartphone as a wearable device. They tested several learning methods, including Naive Bayes, k-Nearest Neighbor, Random Forests, decision trees, and others with different sensor locations and temporal windows size to optimize FOG detection accuracy and latency. The results obtained demonstrate the potential of machine learning algorithms. A recent review on the application of machine learning and wearable sensors for FOG detection is presented in [11].

However, there are still challenges to studying FOG events in real life, as FOG events are influenced by several factors, such as the state of the patient's medication (On and Off), the severity of PD and other personal factors. Moreover, datasets are usually imbalanced that require augmentation strategies, more robust and accurate machine-learning methods are also required. This study explores the possible advantages of genetic programming (GP) in the detection and prediction of FOG events in PD.

2.2 Genetic Programming

Genetic programming (GP) is a part of evolutionary algorithms that apply heuristic search principles inspired by natural evolution to the problem of finding an optimal solution through parameter optimization [12]. The term evolution refers to an artificial process analogous to the biological evolution of living organisms in accordance with Darwin's theory of evolution by natural selection [13]. GP can be expressed as a domain-independent approach that genetically breeds a population of computer programs to solve a problem [14]. GP is a search and optimization algorithm that iteratively transforms a population of computer programs into a new generation of programs using various genetic operators. The most commonly used operators are crossover, mutation, and reproduction. The crossover operator recombines randomly chosen subtrees among the parents and creates a new program for the new population. The mutation operator replaces a randomly chosen subtree with a randomly generated tree, while the reproduction operator replicates a selected individual to a new population. GP finds the solution to the problem in the form of programs or functions.

2.3 Highlights of This Article

– This paper proposed the application of cost-sensitive GP (CSGP) and wearable sensors for the detection of FOG events in Parkinson's disease.
– We investigate how well the existing cost-sensitive learning approaches in machine learning perform if they are applied in a GP-based classifier.
– GP performed optimal feature extraction automatically, discovering relationships among variables and generating interpretable models in the form of a

tree-like structure. This shows that GP maintains its substantial advantage over traditional machine learning algorithms.
- Our experimental results highlighted that CSGP successfully performed automatic FOG detection with high efficiency and excellent performance.

3 Methods and Materials

3.1 Dataset Description

In this paper, we used the publicly available Daphnet dataset developed for FOG detection from wearable sensors attached to the leg, thigh, and trunk of PD patients [9]. A total of 10 participants performed three basic walking activities: (1) walking back and forth in a straight line, (2) random walking with a series of initiated stops, (3) walking simulating activities of daily living, such as entering and leaving rooms, and walking to the kitchen. These activities were performed based on two sessions to replicate a normal daily walking routine. During the first session, the wearable system collected all the data and conducted online FOG detection without RAS feedback. In the second session, the same procedure was followed. However, the RAS feedback was activated. The sensors recorded three-dimensional (3D) accelerations at a sampling frequency of 64 Hz and transferred their data to a wearable computer, which was attached to the trunk of the subjects (along with the third sensor) and provided RAS upon the detection of a freezing episode.

3.2 Data Preprocessing

The data collected by the wearable sensors has been cleaned, filtered, and standardized, resulting in a dataset that is consistent and suitable for analysis and model training processes.

- **Signal Filtering:**the triaxial linear accelerations signals are made up of several components, and thus there can be inherent noise components. As a result, the signals were filtered to remove the noise using a median filter and a third-order low-pass Butterworth filter with a 15 Hz cut-off.
- **Normalization:** the aim of normalization is to change the values of features in the dataset to a common scale, without distorting differences in the ranges of values. For this purpose, Min-max normalization was used to transform each acceleration signal into the range between 0 to 1 with mean and standard deviation.
- **Data Segmentation:** Data segmentation is the process of dividing sensor signals into partially overlapping windows. Before training a GP and other machine learning models, raw time-series data recorded from wearable sensors are split into temporal segments. The sliding approach is frequently used and has been demonstrated to be useful for handling flowing data. In this paper, we used a windowing function with a window length of 4 s with an overlap of 0.5 s. A window was labelled as a FOG window if more than 50% of the samples were labelled as FOG.

– **Feature Extraction and Selection:** After segmenting the data, statistical and time-domain features were extracted and used for model development, as suggested by Mazilu et al. [15]. A total of 54 features were obtained and the most relevant ones were automatically selected using an embedded automatic feature selection mechanism in GP. The importance of each feature was based on the number of references in all models generated during the GP iterations. This approach allows the model to focus on the most significant information and improve performance.

3.3 Cost-Sensitive GP

Most machine learning classifiers, including GP, assume that misclassification costs (false negative and false positive) are set balanced [16]. In most of the real-world domains, however, this assumption is not true. Usually, in the medical domain, the proportion of healthy patients is larger than unhealthy patients, i.e., many datasets of medical diagnosis exhibit imbalanced class distribution [17]. In these FOG datasets, 9% of the data was recorded as FOG events, and the remaining 90% of the data was recorded as normal locomotion. When using the GP model in this dataset, a very low sensitivity was observed in the rare class. Consequently, FOG events belonging to the minority class are misclassified more than those belonging to the majority class. Traditionally in GP, classification problems are solved with accuracy as a fitness function. However, when the data of the problem to be solved contains an imbalanced class distribution, only accuracy cannot be a good option since maximizing the accuracy will naturally lead to classifying everything as the majority class and does not give acceptable results. Therefore, one of the techniques to overcome this biasedness and overfitting, and at the same time to build a good classifier for imbalanced data, is cost-sensitive learning. Cost-sensitive learning focuses on providing a higher cost for predicting a specific class, such as any misclassification of a class will penalize (cost more) the classifier. A cost matrix, which is similar to the confusion matrix, describes the cost for misclassification in a particular scenario and gives a higher cost to misclassifying rather than correctly classifying the instance. In this paper, the minority classes are denoted by '+', and the majority classes are denoted by '-'. Let C (i, j) be the cost of misclassifying an instance from 'class i' as 'class j', and C (i, i) denotes the cost of correct classification (zero cost). The cost matrix can be computed as shown in Table 1, where C (-, +) and C (+, -) correspond to the costs associated with a false negative (FN) and a false positive (FP), respectively.

Table 1. Cost Matrix

		Predicted class	
		FOG	No FOG
Actual Class	FOG	C(+,+)	C(-,+)
	No FOG	C(+,-)	C(-,-)

Table 1 shows an example of a cost matrix where classifying a person with 'FOG' as having 'No FOG' will cost the classifier a penalty equal to C(-,+). A penalty cost equals C(+,-) can also be defined for classifying a normal person as having the FOG event. Defining high-cost values gives more importance to that class. Therefore, it helps in building a classifier for imbalanced data and can help in training the classifier to detect the class with a smaller number of instances and avoid the overfitting problem. This paper applies a cost-sensitive approach in GP (CSGP) to build a predictive model for detecting the presence of FOG in patients, capable of minimizing the expected misclassification costs and errors through a fitness function. Therefore, the fitness function can be calculated as shown in equation (3) which aims to choose the classifier with the minimum misclassification cost (total cost):

$$Totalcost = C(-,+) \times FN + C(+,-) \times FP$$

where FP denotes the total number of false positives and FN is the total number of false negatives. FP is the number of instances that were classified as having the FOG, and actually, they have not. FN is the number of instances that have FOG but were classified as not having the FOG event.

3.4 Proposed Model Framework

The model development process follows an approach that is comprised of a combination of two phases, as shown in Fig. 1. The first phase includes data preprocessing and parameter setup. In the beginning, we performed data preprocessing, which includes noise reduction, missing data filling, data normalization, and filtering. Data segmentation is also required to convert multi-dimensional sensor data into sample data in suitable conditions for model training. Then feature extraction and feature selection were applied to each window segment. For GP, the main parameters, such as population size, number of generations, crossover rate, and mutation rate, have been configured. The second phase includes the development of a GP model using cost-sensitive learning based on the preprocessed dataset and GP parameters from phase one. Following this, the sample data are separated into training and test data for model training and evaluation. Finally, the fitness score in terms of accuracy, precision, recall, F1-score, and confusion matrix is used to validate the final model. The overall GP model development framework is evolved using a fitness measure that includes a series of steps.

4 Experimental Settings

In this section, an experimental setup is established for the proposed approach to demonstrate the performance of GP for FOG analysis and detection. To evaluate the generated model, several experiments were performed using HeuristicLab

3.3.16 framework [18] by splitting the datasets into training and testing. Taking into account that the data is imbalanced, stratified sampling is applied in splitting the data in order to preserve the ratio of each class in the training and testing parts. The proportions are 75% and 25% for the training and testing, respectively, where the training set is used to train the model and includes both input data and the corresponding expected output, and the testing set includes only input data that is being used to assess how well GP is trained and to estimate model properties.

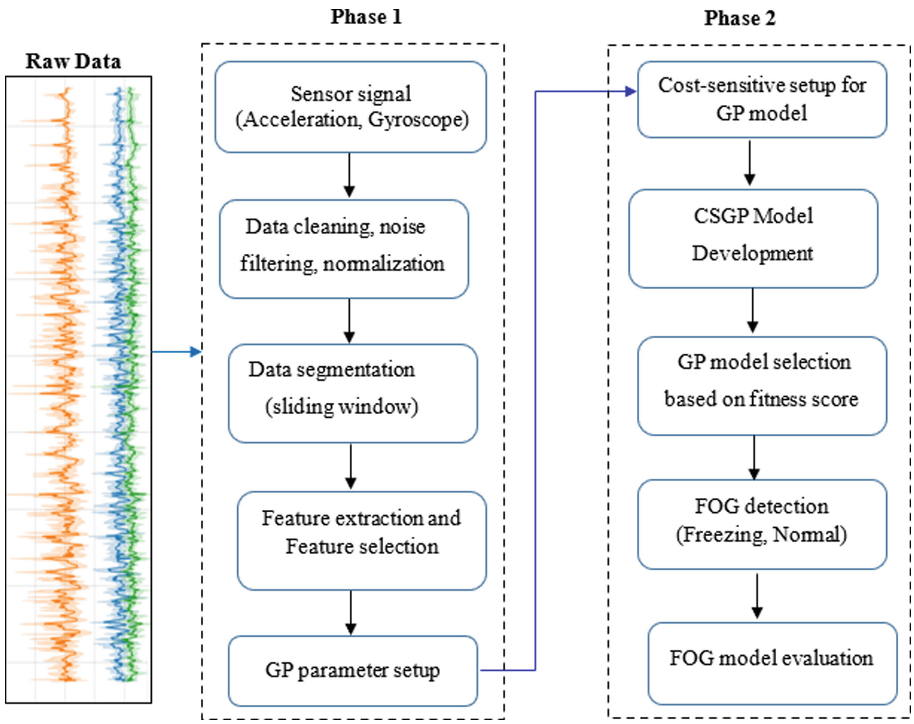

Fig. 1. CSGP model development workflow

4.1 GP Parameter Setup

In GP, setting the control parameters is an important first step to manipulate data and obtain better results. For this, a systematic experimentation process was conducted to tune the parameters of GP using different population sizes (i.e., 100, 330, 500, and 1000). For mutation and crossover rates, GP experimented at 2%, 5%, 10% and 15% for mutation, and 85%, 90%, and 95% for crossover. Due to the stochastic nature of GP, 30 runs were performed in all problems, each with a different random number generator seed. The selection mechanism has been the tournament selection and the maximum tree depth set to the default value. GP requires that further control parameters be specified. In this experiment, the best performance was obtained with the parameter values listed in Table 2.

Table 2. GP Control Parameter Settings

Parameter	Value
Population Size	1000
Maximum number of generations	100
Crossover probability	0.90
Mutation probability	0.15
Selection method	Tournament selection
Termination condition	Max generation
Tree initialization	Ramped half and half
Genetic operators	Crossover, mutation
Elites	1

4.2 Performance Evaluation

In GP, the fitness function defines a measure to calculate the accuracy of a solution by comparing the predicted class labels with the actual class labels. In a binary classification problem, the outcome of classification performance can be represented by a confusion matrix, as shown in Table 3. The predictive model was evaluated using common statistical parameters such as sensitivity, specificity, accuracy, and F1-score, which are based on the true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN). Sensitivity (true positive rate) is the ratio of the proportions of positive patient cases that are accurately predicted divided by the total number of actual positive cases. Specificity (true negative rate) is calculated as the number of negative case predictions divided by the total number of actual negative cases. Accuracy measures the number of all correct predictions divided by the total number of samples. However, the overall accuracy is known to be unsuitable for classification with unbalanced data [19]. Measuring the individual classification accuracy of the minority and majority classes separately using sensitivity and specificity can avoid this learning bias when evaluating model performance in unbalanced class scenarios. The F1-score is used as the harmonic mean of precision and recall [20]. It provides the most reliable evaluation of the model's prediction performance while considering the worst-case prediction scenario for a classifier.

5 Results and Discussions

In analyzing GP for FOG detection, the most fundamental aspect is to know the number of samples that are classified correctly and those which are classified

Table 3. Confusion matrix

	Predicted Positive Class	Predicted Negative Class
Actual Positive Class	True Positive (TP)	False Negative (FN)
Actual Negative Class	False Positive (FP)	True Negative (TN)

incorrectly. In cost-sensitive GP (CSGP), this task is handled by evaluating the quality of the generated model using sensitivity and specificity. The quality of GP is generally called the fitness of a solution candidate. The fitness of the proposed CSGP model was tested using different penalty cost matrices starting from 3 until 13 by representing the penalty cost matrices using the expressions [1:1], [1:3], [1:5], [1:7], [1:9],[1:11] and [1:13]. These costs are increased by +2 and came from extensive experiments with different penalty costs, where 13 was the last penalty cost because sensitivity values increase significantly with larger costs. The results of the CSGP with five different cost matrices on the Daphnet FOG dataset are compared using sensitivity and specificity as fitness functions, as shown in Fig. 2, which shows how the sensitivity of CSGP increases with the different penalty cost matrices, where [1:1] represents the performance of the original data classification.

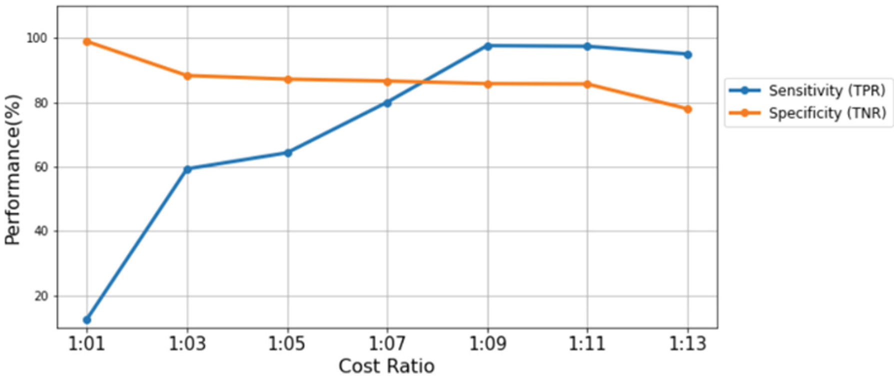

Fig. 2. The performance of CSGP on FOG detection at various penalty costs.

The increase in sensitivity values reflects the high detection of subjects with FOG but also the high misclassification of the normal subjects. It is clear from the figure that the sensitivity fitness function detects more FOG events with higher penalty costs, while specificity relatively drops with higher costs. The sensitivity values start at 13% for the [1:1] penalty cost matrix and reach 97.6% with the [1:9] penalty cost matrix. Then it falls to reach 95% with [1:13] penalty cost matrix. CSGP using the cost penalty matrix of [1:9], achieved the highest performance with a sensitivity value equal to 97.6% and an accuracy value equal to 96%. However, it achieved the lowest specificity of 85.8%. In addition, the population dynamics of CSGP across generations are also evaluated based on mean squared error (MSE). Using MSE on the selected cost matrix, the average fitness of the best solution per generation is calculated based on results stored from 30 runs of GP. Figure 3 shows the median MSE on the five simulations at 100 generations.

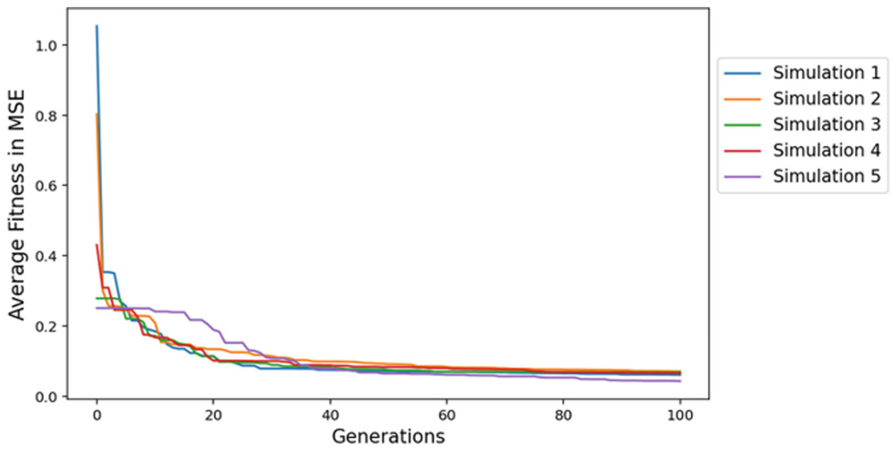

Fig. 3. CSGP evolution plots using mean squared error (MSE).

Table 4. Performance comparison of GP and machine learning techniques

Models	Accuracy	Recall	Precision	F1-Score
GP	96.0	**97.6**	85.8	**98.0**
RF	97.9	81.0	–	0.75
XGB	85.7	85.0	–	92.0
LR	97.0	82.0	–	86.0
DT	81.4	81.0	–	89.00

The median fitness refers to the average of the fitness scores across the entire population. The evolution of the error in average fitness reveals the ability of CSGP to learn the relationship between variables. There is a constant decrease in the test error across generations, indicating that no overfitting is occurring. The final model produced by GP includes the best features selected during the evolutionary process. These variables are the most frequent variables which were the most relevant for the detection of FOG events. Table 4 presents the performance comparison of GP with other benchmark machine learning classifiers, namely random forest (RF), extreme gradient boosting (XGB), logistic regression (LR) and decision tree (DT). The comparison is based on accuracy, recall, precision, and F1-score. Our results showed that GP was able to show competitive performance in the detection of FOG events compared to the traditional machine learning models.

6 Conclusions

This paper is an investigation of genetic programming with a cost-sensitive approach (CSGP) for early detection of freezing of gait (FOG) events in Parkinson's disease (PD). CSGP used various penalty costs for prediction errors ranging from 3 to 13 with a step of 2, and each is represented as a penalty cost matrix. After configuring the different cost matrixes for the CSGP algorithm using the FOG dataset, several experiments with 30 runs were conducted by adjusting the parameters. The proposed approach was evaluated with the well-known GP classification fitness functions, including accuracy, sensitivity, specificity, precision and mean squared error. The results showed that the proposed approach CSGP achieved better detection of FOG episodes using higher penalty costs. From the results, it is evident that CSGP demonstrated substantial potential as a method for the automated development of clinical prediction models for detection and prediction purposes. Overall, the results are encouraging, and further studies can be investigated to extend and optimize the findings for FOG studies and other medical problems using cost-sensitive evolutionary algorithms.

Acknowledgments. This work was supported by European Union through the Horizon 2020 Research and Innovation Programme in the context of the ALAMEDA project under grant agreement No GA 101017558.

References

1. Harrell, F.E.: Multivariable modeling strategies. In: Regression Modeling Strategies. SSS, pp. 63–102. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-19425-7_4
2. Tarekegn, A.N., Ricceri, F., Costa, G., Ferracin, E., Giacobini, M.: Predictive modeling for frailty conditions in elderly people: machine learning approaches. JMIR Med. Informatics. (2020). https://doi.org/10.2196/16678
3. Roca, L.G., Ríos, L.N., Sucarrats, G.M., Medina, H.C., García, D.S.: Parkinson's disease, Kranion (2021). https://doi.org/10.24875/KRANION.M21000002
4. Nutt, J.G., Bloem, B.R., Giladi, N., Hallett, M., Horak, F.B., Nieuwboer, A.: Freezing of gait: moving forward on a mysterious clinical phenomenon. Lancet Neurol. (2011). https://doi.org/10.1016/S1474-4422(11)70143-0
5. Tarekegn, A., Ricceri, F., Costa, G., Ferracin, E., Giacobini, M.: Detection of frailty using genetic programming. In: Hu, T., Lourenço, N., Medvet, E., Divina, F. (eds.) EuroGP 2020. LNCS, vol. 12101, pp. 228–243. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-44094-7_15

6. Tarekegn, A.N., Alemu, T.A., Tegegne, A.K.: A cluster-genetic programming approach for detecting pulmonary tuberculosis. Ethiop. J. Sci. Technol. (2021). https://doi.org/10.4314/ejst.v14i1.5
7. Galar, M., et al.: A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches. IEEE Trans. Syst. Man, Cybern. Part C Appl. Rev. **42**, 463–484 (2012). https://doi.org/10.1109/TSMCC.2011.2161285
8. Punin, C., et al.: A non-invasive medical device for Parkinson's patients with episodes of freezing of gait. Sensors (Switzerland) (2019). https://doi.org/10.3390/s19030737
9. Bächlin, M., et al.: Wearable assistant for Parkinson's disease patients with the freezing of gait symptom. IEEE Trans. Inf Technol. Biomed. (2010). https://doi.org/10.1109/TITB.2009.2036165
10. Mazilu, S., et al.: Online detection of freezing of gait with smartphones and machine learning techniques. In: 2012 6th International Conference on Pervasive Computing Technologies for Healthcare, PervasiveHealth (2012). https://doi.org/10.4108/icst.pervasivehealth.2012.248680
11. Martínez-Villaseñor, L., Ponce, H., Miralles-Pechuán, L.: A Survey on Freezing of Gait Detection and Prediction in Parkinson's Disease. In: Martínez-Villaseñor, L., Herrera-Alcántara, O., Ponce, H., Castro-Espinoza, F.A. (eds.) MICAI 2020. LNCS (LNAI), vol. 12468, pp. 169–181. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-60884-2_13
12. Lissovoi, A., Oliveto, P.S.: On the time and space complexity of genetic programming for evolving Boolean conjunctions. J. Artif. Intell. Res. **66**, 655–689 (2019). https://doi.org/10.1613/jair.1.11821
13. Darwin, C.: The origin of species, by means of natural selection, or the preservation of favored races in the struggle for life. Crayon **7**, 149–150 (1860). https://doi.org/10.2307/25528056
14. O'Reilly, U.-M.: Genetic programming. In: Proceedings of the Fourteenth International Conference Genetic and Evolutionary Computation Conference Companion - GECCO Companion '12, ACM Press, New York, New York, USA, p. 693 (2012). https://doi.org/10.1145/2330784.2330912
15. Mazilu, S., Calatroni, A., Gazit, E., Roggen, D., Hausdorff, J.M., Tröster, G.: Feature learning for detection and prediction of freezing of gait in Parkinson's disease. In: Perner, P. (ed.) MLDM 2013. LNCS (LNAI), vol. 7988, pp. 144–158. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-39712-7_11
16. Thai-Nghe, N., Gantner, Z., Schmidt-Thieme, L.: Cost-sensitive learning methods for imbalanced data. In: Proceedings of the International Joint Conference Neural Networks (2010). https://doi.org/10.1109/IJCNN.2010.5596486
17. Yildirim, P.: Chronic kidney disease prediction on imbalanced data by multilayer perceptron: chronic kidney disease prediction. In: Proceedings - International Computer Software Applications Conference (2017). https://doi.org/10.1109/COMPSAC.2017.84
18. Wagner, S., et al.: Architecture and design of the HeuristicLab optimization environment (2014). https://doi.org/10.1007/978-3-319-01436-4_10
19. Bhowan, U., Zhang, M., Johnston, M.: Genetic programming for classification with unbalanced data (2010). https://doi.org/10.1007/978-3-642-12148-7_1
20. Witten, I.H., et al.: Data mining: practical machine learning tools and techniques. Elsevier (2011). https://doi.org/10.1016/C2009-0-19715-5