automation

MDPI

Article

Automatic Classification of Gait Patterns in Cerebral Palsy Patients

Rodrigo B. Ventura [1], João M. C. Sousa [1,*], Filipa João [2], António P. Veloso [2] and Susana M. Vieira [1]

[1] IDMEC, Instituto Superior Técnico, Universidade de Lisboa, 1049-001 Lisboa, Portugal; rodrigo.boal.ventura@tecnico.ulisboa.pt (R.B.V.); susana.vieira@tecnico.ulisboa.pt (S.M.V.)
[2] CIPER, Faculdade de Motricidade Humana, Universidade de Lisboa, 1495-688 Cruz-Quebrada-Dafundo, Portugal; filipajoao@fmh.ulisboa.pt (F.J.); apveloso@fmh.ulisboa.pt (A.P.V.)
* Correspondence: jmsousa@tecnico.ulisboa.pt

Abstract

The application of wearable sensors coupled with diagnostic models presents one of the most recent advancements in automation applied to the medical field, allowing for faster and more reliable diagnosis of patients. Nonetheless, such applications pose a complex challenge for traditional intelligent automation (combining automation and artificial intelligence) methods due to high class imbalances, the small number of subjects, and the high dimensionality of the measured data streams. Furthermore, automatic diagnostic models must also be explainable, meaning that medical professionals can understand the reasoning behind a predicted diagnosis. This paper proposes an intelligent automation approach to the diagnosis of cerebral palsy patients using multiple kinetic and kinematic sensors that record gait pattern characteristics. The proposed artificial intelligence framework is a multi-view fuzzy rule-based ensemble architecture, in which the high dimensionality of the sensor data streams is handled by multiple fuzzy classifiers and the high class imbalance is handled by a cost-sensitive training algorithm for estimating a fuzzy rule-based stack model. The proposed methodology is first tested on benchmark datasets, where it is shown to outperform comparable benchmark methods. The ensemble architecture is then tested on the cerebral palsy dataset and shown to outperform comparable ensemble architectures, particularly on minority class predictive performance.

Keywords: cerebral palsy; gait patterns; fuzzy modeling; imbalanced datasets; multi-view ensemble learning; cost-sensitive learning

Academic Editor: Eyad H. Abed

Received: 21 May 2025
Revised: 20 October 2025
Accepted: 6 November 2025
Published: 9 November 2025

Citation: Ventura, R.B.; Sousa, J.M.C.; João, F.; Veloso, A.P.; Vieira, S.M. Automatic Classification of Gait Patterns in Cerebral Palsy Patients. *Automation* **2025**, *6*, 71. https://doi.org/10.3390/automation6040071

1. Introduction

Intelligent automation, integrating automation and artificial intelligence, has had a massive impact in recent years on a wide variety of complex tasks. Despite the undeniable success of artificial intelligence methodologies, they still have some limitations, such as being computationally expensive, requiring large amounts of data, and yielding black box-models. These types of models are not explainable in the sense that such models do not present the reasoning behind a certain decision or prediction. This is a key issue in making them easily understandable by human experts, a sine qua non condition in fields where human decision is mandatory, such as in healthcare.

Explainability [1] and interpretability [2] are two of the most relevant problems currently faced by artificial intelligence. Intelligent automation must ensure transparency in AI-driven decisions and maintain robust changes in automated systems. Its relevance,

however, is dependent on the application of the models and the field-specific requirements. One such field is healthcare and, in particular, the design of diagnostic models that can help medical professionals with a more precise diagnosis of patients. The automatic diagnostic model must also present its reasoning, which can be reviewed by medical professionals [3].

Beyond the interpretability of diagnostic models, the medical field also presents challenging limitations to current data-based modeling approaches, particularly with respect to the characteristics of the data used to obtain the models [4]. Medical data can be very limited in the amount of samples available to train models, considering the practical difficulties of collecting patient data. Furthermore, the available datasets are often highly imbalanced, particularly when studying rare diseases. Thus, the number of healthy patients is usually much larger than the number of sick patients. This problem, known as class imbalance [5], poses a very challenging task for many traditional classification methods, as these tend to completely disregard minority classes, which are usually the classes of the greatest interest [6], particularly when creating a diagnostic model [7].

Many of the recent applications of intelligent data-based modeling to diagnostic models are based on the use of wearable sensors to collect patient data, as well as other technologies that describe relevant aspects of the patient's condition [8]. The use of sensors and their measurements has allowed for a more automated description of the patient's condition, instead of traditional diagnostic procedures that would heavily rely on the experience of trained medical professionals. The advantages of the application of sensors and different monitoring techniques to diagnose patients are that the diagnosis is more precise and that the justification for a certain diagnosis is easier to quantify and share between medical experts.

However, this approach also creates more challenges for data-based modeling approaches. In order to correctly describe patient conditions, multiple sensors and measurements must be obtained simultaneously, thus generating highly dimensional data streams that must be processed to obtain a diagnostic prediction [9]. Although the problem of high-dimensional data is well known and studied (often referred to as the curse of dimensionality), it becomes even more challenging when the number of data samples is scarce.

This paper proposes an automatic diagnostic model for patients suffering from spastic diplegia, a form of cerebral palsy, using multiple kinetic and kinematic sensors to describe the patient's gait cycle, which corresponds to the sequence of body movements that occur when walking. The dataset used in this work consists of multiple joint angles and moment measurements obtained from motion capture cameras and force platforms. Considering the characteristics of the available data, it is clear that the problem addressed in this paper presents the challenges often encountered when applying intelligent automation in medical applications, in particular in the development of diagnostic models.

A data structure using multiple sources (sensors in this case) is often approached using ensemble learning architectures, in which a base model is trained using each view of the data (sensor data stream). The predictions of the base models are then combined by a stack model to obtain a single prediction corresponding to the output of the ensemble model. Thus, multi-view ensemble architectures are able to handle highly dimensional data and generally improve model performance, especially when all the data views have a much lower dimensionality than the total number of sample features, since each base model is trained on relatively low-dimensional feature vector.

Furthermore, to meet the explainability requirements, fuzzy rule-based modeling was chosen as the intelligent data-based modeling approach to train the base models of the ensemble. Fuzzy modeling methods train fuzzy inference systems that contain multiple fuzzy rules that relate the input features to the output class prediction. Fuzzy rules are

structured as if–then statements and allow for a clear interpretation of the model reasoning for classifying a sample as belonging to a certain class and can be directly represented in a graphical form that is easily understandable by humans.

Many different fuzzy rule modeling approaches have been proposed in the literature. Regarding the problem addressed in this work, the Zero-Order Autonomous Learning Multi-Model (ALMMo-0) classifier was chosen [10]. The choice of the ALMMo-0 classifier is due to the simplicity of its training algorithm, which is very lightweight in terms of computing requirements and does not require the optimization of any hyper-parameters, which is a problem present in many traditional fuzzy modeling methods.

Furthermore, and despite its generally good performance on imbalanced classification problems, this paper also proposes a modification of the original learning algorithm of the ALMMo-0 classifier, referred to as ALMMo-0 (W), which introduces a cost-sensitive local weighting element to the minority class rules. The rule weights are adjusted in order to compensate for the majority-class bias and improve detection of cerebral palsy subjects. The proposed local weight adjustment is formulated as an optimization problem, defined as the maximization of an adequate classification performance metric. In order to keep the optimization procedure lightweight, the proposed modification also proposes a simple heuristic to reduce the number of weights to be optimized, as well as their search value intervals. In the proposed modification, the optimization problem is solved using Bayesian optimization, considering its well-known adequacy for hyper-parameter tuning in machine learning models. Thus, Bayesian optimization is used to fine-tune the weights to improve the minority class detection.

The remainder of this paper is structured as follows: Section 2 presents a brief summary of the main concepts that are relevant as the basis of the work presented in this article; Section 3 details the theoretical formulation of the ALMMo-0 classifier in order to make this paper self-contained; Section 4 presents the intelligent modeling architecture proposed here and the theoretical formulation of its components; and Section 5 discusses the performance of the proposed ensemble approach and compares it with benchmark methods. Finally, in Section 6, a brief summary of the conclusions is presented, as well as possible future lines of research.

2. Background

2.1. Cerebral Palsy

Cerebral palsy is a group of permanent movement disorders in the development of movement and posture caused by non-progressive disturbances in the developing fetal or infant brain [11]. Classified as a neuro-developmental condition, it begins in early childhood and persists throughout the lifespan, being one of the most common movement disorders in children.

Cerebral palsy is characterized by abnormal motor development, muscle tone, and coordination. It may also cause speech and sleep disorders, as well as pain associated with severe muscle spasms and stiff joints.

Currently, there are no cures for cerebral palsy, and the existing treatments focus on managing the different symptoms and improving individual autonomy and quality of life. Common approaches include physical, occupational, and speech therapies. Different forms of surgical intervention and medication may also be used, as well as assistive technologies such as orthotic devices, mobility aids, and speech-generating devices.

Diagnosis is usually based on physical examination and generally assessed at a young age. Diagnostic tools include continuous observation of the patient, primarily focusing on their motor development and cognitive skills. Further diagnostic tools include neuroimaging techniques such as magnetic resonance imaging (MRI).

The most common type of cerebral palsy is known as spastic diplegia, which affects the motor cortex of the brain, which controls voluntary movement. Spastic diplegia is characterized by muscle tightness in the legs, hips, and pelvis, thus having negative effects on the walking functions of the patient. Spastic diplegia affects each patient differently in terms of their walking function. Therefore, correct characterization of spastic diplegia is essential to correctly select treatment options, such as tendon surgery and orthotic devices.

Spastic diplegia may present different gait patterns, which are the sequence of movement patterns exhibited by the lower limbs when walking. These patterns are classified according to the pelvic tilt angular displacement, as well as the sagital patterns of the hip, knee, and ankle [12]. Sagittal gait patterns are classified as true equinus, jump gait, apparent equinus, crouch gait, or asymmetrical. According to [13], these classifications are used as a biomechanical basis that links spasticity, musculoskeletal pathology, and the appropriate intervention management.

Regarding the specific pattern true equinus, it is characterized by the existence of calf spasticity, where the ankle is plantarflexed throughout the stance phase and the hips and knees are extended. The management strategies are usually BTX-A (toxin botulinic) injections to the calf to control spasticity; lengthening of the gastrocnemius for contracture management; or the prescription of a solid or hinged AFO (ankle–foot orthosis) for orthotic management.

Historically, gait patterns have been identified by direct observation by a medical expert, often leading to qualitative descriptions and conflicting classification criteria. However, a continuous effort has been made by the medical community to develop a classification system that is clinically more useful and based on quantitative criteria.

More recently, with the increasing usage of wearable sensors, measuring devices, and data-based modeling tools in the medical field, gait patterns have been identified using kinematic and kinematic measurement devices, such as cameras and force platforms. The measurements obtained from the various sensors can be used for estimating a three-dimensional model of the lower limbs during the gait cycle, as shown in Figure 1. The work presented in this paper is focused on this recent diagnostic methodology and proposes a modeling approach to the identification of a gait pattern known as true equinus, often found in spastic diplegia patients.

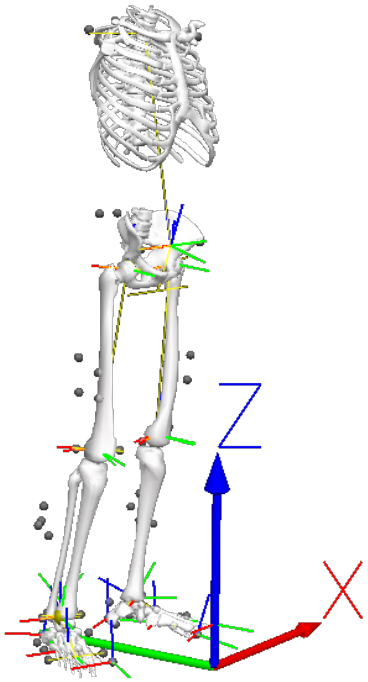

Figure 1. Three-dimensional joint-segment body model estimated from the sensor measurements for each test subject in order to compute the various body joint angles and moments.

2.2. Fuzzy Systems

Fuzzy systems are knowledge-based systems that commonly use fuzzy logic to solve real-world problems. They often incorporate vague or imprecise (human) information, using linguistic variables and if–then rules [14]. Fuzzy inference systems are an artificial intelligence technique that maps a given input to an output by applying fuzzy logic inference. This technique is normally composed using fuzzy rules, structured as if–then statements [15]. Fuzzy rules are inherently able to model uncertainty, since their antecedents (if part of the rules) are modeled using fuzzy sets, which are groups of elements that have a membership degree between 0 and 1. These fuzzy sets are mathematically defined using membership functions and can be obtained from data. Furthermore, fuzzy sets are linked to linguistic terms (in natural language), allowing the interpretation of the fuzzy rules and, therefore, of the fuzzy model.

Classical approaches to fuzzy rule data-based modeling have been used in a wide range of applications since their development. The first type of classic fuzzy model is Mamdani FIS, where both antecedents and consequents are defined by fuzzy sets. The second type of classic fuzzy modeling approach is Takagi-Sugeno FIS, in which the consequents of the fuzzy rules are represented by crisp functions.

Despite showing good performance on a wide variety of problems while still allowing for easier interpretation of the rules by human experts, fuzzy systems pose some challenges related to their training procedure, such as often requiring fine-tuning of the model parameters, as well as requiring the optimization of hyper-parameters, such as a fuzzy clustering algorithm and its clustering parameters. Another challenge of fuzzy systems that has been addressed is the online training of fuzzy models. Such approaches allow for fuzzy models to adjust their rule parameters in real time in order to adapt to non-stationary environments and abrupt changes in the data stream.

One of the most promising groups of fuzzy modeling approaches that has consistently shown good performance when addressing the aforementioned challenges is AnYa-type fuzzy rule-based systems, see, e.g., [16]. Such systems are structurally similar to the traditional fuzzy systems, with the main difference being that their fuzzy sets are defined from clouds, instead of fuzzy clusters.

The aforementioned approach to train fuzzy models and define their antecedent parameters from data clouds does not require model structure initialization nor hyper-parameter tuning, simplifying the training algorithm, reducing the required computational power, and allowing for online training and good performance in non-stationary environments. Furthermore, despite the lightweight training approach, AnYa-type fuzzy systems have shown good performance on a wide range of applications and problem types.

Examples of fuzzy rule-based systems of AnYa type include eClass0 [16], SOFIS [17], ALMMo [18], and ALMMo-0 [10], which is used in this paper to due to its simplicity and overall good performance.

The relatively simple model structure has also made AnYa-type fuzzy systems highly adequate in deep fuzzy applications based on multi-layered ensembles of fuzzy systems, allowing for highly dimensional problems such as image recognition to be approached using fuzzy rule-based modeling. Such types of ensemble architectures are fairly recent advancements with a lot of related work published and have allowed for entirely fuzzy rule-based systems to obtain performances comparable to those of benchmark deep learning approaches. Some examples of recently proposed deep fuzzy approaches based on this ensemble architecture include the SOFEmsemble [19] and MEEFIS [20] deep neuro-fuzzy architectures.

Fuzzy rule-based ensemble models have been shown to be particularly adequate for data-based modeling applications in healthcare, such as ICU patient monitoring [21] and

personalized medicine [22]. The problem at hand, i.e., classification of gait patterns, usually requires human expert intervention, making fuzzy systems very suitable to address the problem at hand.

2.3. Ensemble Learning

Ensemble learning refers to a group of machine learning approaches based on training multiple models (known as base models) and combining individual predictions to obtain a final ensemble model with improved predictive performance. Ensemble learning methods can generally use any learning algorithm to train the base models, are classified as meta-algorithms, and also allow for base models trained using different algorithms (heterogeneous ensembles). Moreover, ensemble learning includes parallel approaches, in which the base models are trained independently of each other (thus allowing simultaneous training), and sequential approaches, in which each new base model is trained using the predictions of the previous base model and sequentially improves the performance.

Ideally, the ensemble models must achieve performances significantly better than the performances of each base classifier. Otherwise, the ensemble approach offers no advantage in terms of performance compared to the best base model alone. Furthermore, the improvements in predictive performances of ensemble models depend on the diversity of its base learners and not necessarily on their individual predictive performances. Another critical factor that affects the performance of ensemble models is the approach used to combine the outputs of the base models into a single prediction. Common approaches include simple schemes such as winner takes all, majority voting, and weighted majority voting, as well as data-based modeling methods to combine the individual predictions.

One of the most recent and promising ensemble learning approaches is known as multi-view ensemble learning, which has been successfully applied in a wide range of data-based modeling tasks. Multi-view ensemble approaches consist of training each base model on a distinct set of features (known as a view). The outputs of each base model are then combined into a single prediction, similarly to the aforementioned bagging ensembles and using similar methods.

Multi-view ensemble approaches include different types of problems, with different levels of complexity. Generally speaking, the most complex problems are the ones that are not naturally structured as multiple data views and thus require the feature sets to be determined, often using an iterative search approach. Therefore, datasets that are naturally structured as different sources (each representing a set of features) are usually well suited to be approached using multi-view ensemble methods, including problems with multiple source types (known as multimodal learning), such as video, sound, and text.

Taking into account the structure of the problem approached in this work, see Section 2.1, the proposed intelligent approach is structured as a multi-view ensemble architecture. Thus, it can handle high-dimensional problems that would pose challenges to a single base model, while also taking advantage of the data structure in terms of predictive performance, as well as interpretability.

2.4. Model Performance on Imbalanced Datasets

Imbalanced datasets are classification datasets in which some classes (majority classes) have a significantly larger number of samples than the other classes (minority classes). This poses a challenge to most data modeling approaches, since they typically assume that all classes are equally represented. This often results in models with good predictive performances on the majority classes, but with very poor performances on the minority class, which may even be completely ignored. Here, we consider binary classification problems, since they represent most of the real application problems and are more commonly

studied in the literature. Furthermore, multi-class problems can always be decomposed into multiple binary problems using one-vs-all or one-vs-one decomposition, at the cost of requiring multiple models to be trained.

The usual convention applied to binary classification on imbalanced datasets is that the majority class corresponds to the negative samples, with target values of 0, and the minority class corresponds to the positive samples, with target values of 1. The class imbalance is usually quantified by the imbalance ratio IR, defined as the ratio of the majority samples to the minority samples. The predictions of a binary classifier fall into four possible cases. For a negative prediction, it corresponds to a true negative (TN) if the prediction is correct or to a false positive (FP) if the sample is actually negative. For a positive prediction, it corresponds to a true positive (TP) if the prediction is correct or to a false positive (FN) if the sample is actually positive.

In the case of imbalanced datasets, false negatives are normally more critical than false positives since they occur when the classifier does not detect a minority class sample. Such errors are particularly important if one thinks of positive labels in medical diagnosis or fault detection applications, in which the end result is a sick patient being classified as healthy or damaged equipment being classified as operational.

Therefore, model performance on imbalanced datasets is improved if the number of false negatives is reduced, while not excessively increasing the number of false positives. In order to correctly evaluate the model performance, the classification metric must consider this compromise, as well as the imbalance ratio of the problem.

Performance of the positive class is usually described by the ratio of true positive predictions to the total number of positive samples (known as recall, or sensitivity) or to the total number of positive predictions (known as precision). Thus, recall penalizes false negatives, while precision penalizes false positives. In practice, one seeks to increase recall while not overly decreasing precision.

However, the performance in the negative class must still be considered by the metric and its relevance should be proportional to the imbalance ratio. Many different metrics have been proposed and studied in the literature. Here, we present some of the most relevant and commonly used.

Balanced accuracy, defined as the arithmetic mean of recall and specificity, gives equal weight to both classes, thus being a common alternative to the usual accuracy score, which only considers the total number of correct predictions, regardless of the classes. One similar metric is the geometric mean, which instead considers the root of the product of recall and specificity, therefore penalizing lower recall values resulting from a higher number of false negative predictions. Another well-known metric is the F1-score, which is defined as the root of the product of recall and precision and quantifies the compromise between minimizing false negatives and not increasing false positives.

Plenty of other metrics have been applied in the literature, such as Cohen's Kappa Coefficient and the Matthews correlation coefficient, which are well known, robust, and relevant for different problems and imbalance ratios. However, please note that the predictive performance is highly dependent on the model application and its project parameters, thus meaning that recall and precision remain relevant and crucial as model choosing criteria.

2.5. Approaches to Imbalanced Datasets

Considering the prevalence of imbalanced datasets, as well as their frequency in relevant applications, a wide variety of approaches have been proposed in the literature [23], seeking to mitigate the effects that class imbalance has on model performance. Here, we provide a broad overview of the different approach types, as well as their advantages and limitations, within the context of this paper.

In general terms, approaches to imbalanced datasets can be divided into data-level approaches and algorithm-level approaches [24]. Data-level approaches refer to methods that modify the training data, while algorithm-level approaches refer to methods that modify the learning algorithm of the classifier. Generally speaking, data-level approaches can be applied with any classification modeling method since only the data samples are modified, while algorithm-level approaches are specific to a learning algorithm.

Data-level approaches include resampling techniques [25], which work by modifying the imbalance ratio of the data by either removing majority-class samples or adding minority-class samples. Resampling techniques include random resampling methods (such as random oversampling [26] and random undersampling [27]) and also generative techniques, such as SMOTE [28], Tomek Links [29], and ADASYIN. Another group of approaches includes feature selection methods [30], which work by selecting a subset of the original data features. The resultant subset not only allows for dimensionality reduction and thus improves model performance, but also contains the most relevant features of the problem, thus allowing for better inter-class discrimination and improved minority class detection. Although feature selection has been extensively applied to different high-dimensional problems, its usage in addressing class imbalance has also achieved promising results [31].

Algorithm-level approaches generally work by making the learning algorithm more sensitive to minority-class samples. These are usually specific to the algorithm. Although such approaches are typically specific to a modeling method, the more general and well-known approaches belong to a group of techniques known as cost-sensitive learning methods. Cost-sensitive learning approaches work by assigning a larger weight to the minority class and penalizing incorrect predictions on its samples [32]. Cost-sensitive approaches can be divided into direct approaches, which incorporate cost-sensitive mechanisms within the learning algorithm itself [33], and indirect approaches, which convert cost-insensitive classifiers without modifying the actual training algorithm. Examples of direct approaches are methods that modify the objective function that guides the training procedure (which is usually formulated assuming equal class distribution), thus changing the learned model parameters and reducing the majority class bias of the trained model. Examples of indirect approaches include threshold moving techniques, which work by tuning the decision threshold value (used by regression models to transform the numerical output to a class label) and maximize a chosen classification metric.

3. Zero-Order Autonomous Learning Multiple Model (ALMMo-0) Classifier

The ALMMo-0 [10] classifier is a non-parametric, non-iterative, and fully autonomous AnYa-type fuzzy rule-based (FRB) multi-class classifier. The structure of ALMMo-0 classifiers is composed of one sub-model per class. Each one of the class sub-models is based on multiple AnYa-type fuzzy rules with one data cloud associated with each rule's antecedent. The structure of such sub-models is as shown in (1), where $\mathbf{x}$ is a newly arrived data observation and f_j^i are the R^i clouds of the ith sub-model.

$$IF\left(\mathbf{x}\sim\mathbf{f}_1^i\right)OR\left(\mathbf{x}\sim\mathbf{f}_2^i\right)OR\ \ldots\ OR\left(\mathbf{x}\sim\mathbf{f}_{R^i}^i\right)THEN\left(class^i\right) \tag{1}$$

Each sub-model is trained independently of the others since only its class samples are used to iteratively create the clouds, meaning that the cloud structure for each class sub-model is not affected by the cloud structures of the remaining sub-models. Therefore, the fully trained ALMMo-0 classifier is made up of multiple FRB systems, each with its own set of rules. When classifying a new observation $\mathbf{x}$, each one of the rules j in each one of the class

models i will produce a confidence score, denoted as λ_j^i, which is defined as a function of the distance between the sample $\mathbf{x}$ and a focal point of the cloud $\mathbf{f}_j^i$, as shown in (2).

$$\lambda_j^i = \exp\left(-\frac{1}{2}\left\|\mathbf{x} - \mathbf{f}_j^i\right\|^2\right) \tag{2}$$

In order to output a class prediction $\hat{y}$, one must devise some form of rule based on the confidence scores of each sub-model. In ALMMo-0 classifiers, the winner-takes-all strategy is used. First, the maximum confidence score for each class sub-model $\lambda_{j^*}^i$ is found, and then the winner-takes-all principle is used and the class sub-model with the highest confidence score assigns its label, as shown in (3), where C is the number of class sub-models.

$$\hat{y} = \underset{i=1,2,\ldots,C}{\arg\max}(\lambda_{j^*}^i) \tag{3}$$

This classification strategy means that the data clouds effectively create Voronoi tessellation of the data space, dividing it into different sub-regions that belong to different classes. As mentioned, these data clouds are non-parametric and no prior assumptions about the data are made. Furthermore, the training algorithm that creates the clouds is non-iterative and lightweight, meaning that ALMMo-0 classifiers are particularly adequate for streaming data applications, even though no specific mechanisms to adapt to abrupt changes in the data are proposed. Let $\mathbf{x}_k^i$ be the $k$þsample of the ith class. The sample is first norm normalized, as shown in (4), effectively removing one degree of freedom from the data space and projecting the sample to a unit radius hypersphere centered around the origin.

$$\mathbf{x}_k^i \leftarrow \frac{\mathbf{x}_k^i}{\left\|\mathbf{x}_k^i\right\|} \tag{4}$$

If the sample is the first sample of its class, $\mathbf{x}_1^i$, then the respective sub-model parameters are initialized using (5).

$$\begin{cases} \boldsymbol{\mu}^i \leftarrow \mathbf{x}_1^i \\ X^i \leftarrow \left\|\mathbf{x}_1^i\right\|^2 \end{cases} \tag{5}$$

Here, μ^i is the mean of the sub-model and X^i is the average scalar product of the sub-model. Then, the sample is used to create the first cloud of its class sub-model using (6).

$$\begin{cases} R^i \leftarrow 1 \\ M_1^i \leftarrow 1 \\ \mathbf{f}_1^i \leftarrow \mathbf{x}_1^i \\ X_1^i \leftarrow \left\|\mathbf{x}_1^i\right\|^2 \\ r_1^i \leftarrow r_0 \end{cases} \tag{6}$$

where R^i is the total number of rules in the sub-model, M_1^i is the number of samples used to update the cloud, $\mathbf{f}_1^i$ is the cloud's focal point, X_1^i is the cloud's average scalar product, and r_1^i is the cloud's radius. This radius effectively describes a confidence score threshold around the focal point and its initial value is not a problem-specific parameter. In this article, the initial cloud radius r_0 is assumed to be $r_0 = \sqrt{2(1-\cos(15))}$, as specified in the original article [10]. If the newly arrived sample $\mathbf{x}_k^i$ is not the first sample of its class, the sub-model parameters $\boldsymbol{\mu}^i$ and X^i are recursively updated using (7).

$$\begin{cases} \mu^i \leftarrow \frac{k-1}{k}\mu^i + \frac{\mathbf{x}_k^i}{k} \\ X^i \leftarrow \frac{k-1}{k}X^i + \frac{\left\|\mathbf{x}_k^i\right\|^2}{k} \end{cases} \tag{7}$$

The algorithm then checks for density anomalies in the sub-model, i.e., regions of the data space where the unimodal density is either too high (high concentration of clouds) or too low (low concentration of clouds). This is accomplished by computing the unimodal density between the sample $\mathbf{x}_k^i$ and the sub-model mean $\boldsymbol{\mu}^i$, using (8), and comparing it to the unimodal densities between each cloud focal point in the sub-model $\mathbf{f}_j^i$, which is computed using (9).

$$D(\mathbf{x}_k^i, \boldsymbol{\mu}^i) = \frac{1}{1 + \frac{\left\|\mathbf{x}_k^i - \boldsymbol{\mu}^i\right\|^2}{X^i - \left\|\boldsymbol{\mu}^i\right\|^2}} \tag{8}$$

$$D(\mathbf{f}_j^i, \boldsymbol{\mu}^i) = \frac{1}{1 + \frac{\left\|\mathbf{f}_j^i - \boldsymbol{\mu}^i\right\|^2}{X^i - \left\|\boldsymbol{\mu}^i\right\|^2}} \tag{9}$$

The maximum and minimum focal point densities are compared to the sample density, defining the first condition in the algorithm, shown in (10).

$$\begin{aligned} D(\mathbf{x}_k^i, \boldsymbol{\mu}^i) \; &> \; \max_{j=1,2,\ldots R^i} (D(\mathbf{f}_j^i, \boldsymbol{\mu}^i)) \vee \\ D(\mathbf{x}_k^i, \boldsymbol{\mu}^i) \; &< \; \min_{j=1,2,\ldots R^i} (D(\mathbf{f}_j^i, \boldsymbol{\mu}^i)) \end{aligned} \tag{10}$$

If Condition 1 is verified, then there is a density anomaly (either too high or too low) and a new cloud is created around the sample $\mathbf{x}_k^i$ using (11).

$$\begin{cases} R^i \leftarrow R^i + 1 \\ M_j^i \leftarrow 1 \\ \mathbf{f}_j^i \leftarrow \mathbf{x}_k^i \\ X_j^i \leftarrow \left\|\mathbf{x}_k^i\right\|^2 \\ r_j^i \leftarrow r_0 \end{cases} \tag{11}$$

Otherwise, no density anomaly exists, and a second condition, based on distance instead of density, is checked to assess if a nearby cloud already exists. First, the distances between the sample $\mathbf{x}_k^i$ and each focal point in the sub-model are computed, and then the nearest cloud focal point $\mathbf{f}_{j*}^i$ is found using (12).

$$\mathbf{f}_{j*}^i = \underset{j=1,2,\ldots R^i}{\arg\min} \left(\left\| \mathbf{x}_k^i - \mathbf{f}_j^i \right\| \right) \tag{12}$$

If the distance to the nearest focal point $\mathbf{f}_{j*}^i$ is less than its radius r_{j*}^i, then the sample is considered to be close enough and the cloud is updated. This distance criterion is expressed by Condition 2, as shown in (13).

$$\left\| \mathbf{x}_k^i - \mathbf{f}_{j*}^i \right\| \leq r_{j*}^i \tag{13}$$

If Condition 2 is met, the nearest cloud parameters are updated using (14).

$$\begin{cases} M^i_{j*} \leftarrow M^i_{j*} + 1 \\ \mathbf{f}^i_{j*} \leftarrow \frac{M^i_{j*}-1}{M^i_{j*}} \mathbf{f}^i_{j*} + \frac{\mathbf{x}^i_k}{M^i_{j*}} \\ X^i_{j*} \leftarrow \frac{M^i_{j*}-1}{M^i_{j*}} X^i_{j*} + \frac{\|\mathbf{x}^i_k\|^2}{M^i_{j*}} \end{cases} \tag{14}$$

Otherwise, a new cloud is created using (11). The algorithm then proceeds to the next sample. The complete learning process is summarized in Algorithm 1.

Algorithm 1 ALMMo-0 training algorithm.

```
for each class i do
    for each x_k^i in class i do
        Normalize x_k^i (4)
        if sub-model i is empty then
            Initialize sub-model (5) and (6)
        else
            Update sub-model parameters (7)
            Compute sample density (8)
            Compute focal densities (9)
            if Condition 1 (10) is met then
                Create new cloud (11)
            else
                Find nearest cloud (12)
                if Condition 2 (13) is met then
                    Update nearest cloud (14)
                else
                    Create new cloud (11)
```

4. Proposed Approach

4.1. ALMMo-0 (W)

The ALMMo-0 classifier is fairly robust to majority-class bias imposed by high class imbalances, mainly due to the separate sub-models for each class, which are trained separately and only using their respective samples. Nonetheless, one key behavior that was observed in multiple models was that most false negative predictions occur in borderline regions of the data space, which correspond to the zones of separation between the different class clouds.

In order to improve the minority-class detection performance, we propose to introduce a weighting component to compensate for class bias. The proposed weighting scheme is defined at the local (cloud) level, and not at the global (class model) level, meaning that the confidence score weighting is applied individually to each minority rule, and not simply to the minority-class rule with the highest activation score, which corresponds to the output of the minority-class sub-model.

While a global weighting scheme would be functionally similar to convert the ALMMo-0 model to a regressor, and thus allow for direct application of a simple threshold moving approach, it would also apply the same bias correction to all minority clouds, even in regions of the data space where the inter-class separation is adequately modeled. Moreover, while this would still decrease the bias of the final model, it would also result in an unnecessarily large number of majority samples being mislabeled as negatives.

In order to implement the proposed weighting scheme, a weighted confidence score η_j^i is introduced and defined as shown in (15), where ω_j^i is the actual weight assigned to the jth cloud of the ith class.

$$\eta_j^i = \exp\left(-\omega_j^i \left\|\mathbf{x} - \mathbf{f}_j^i\right\|^2\right) \tag{15}$$

Therefore, the weighting scheme proposes the weighting of the sample–cloud distances, and not the actual confidence scores λ_j^i, meaning that η_j^i remains coherently defined as the maximum and equal to one only when $\mathbf{x}$ coincides with the cloud focal point $\mathbf{f}_j^i$. Another consequence of the chosen weighting scheme is that it affects the original partition characteristics of the data space, generalizing it to a multiplicative weighted Voronoi diagram.

Using the newly defined weighted confidence score, the winner-takes-all prediction criterion is redefined, as shown in (16), where $\eta_{j^*}^i$ represents the weighted confidence score of the nearest cloud of the ith class sub-model.

$$\hat{y} = \underset{i=1,2,\ldots,C}{\arg\max}(\eta_{j^*}^i) \tag{16}$$

Having defined the predictive behavior for a general locally weighted ALMMo-0 classifier in (15) and (16), we can now return to the specific case of an imbalanced binary problem, defined as shown in (17), where N and P are introduced to simplify the notation.

$$\begin{cases} i = \{N = 1, \;\; P = 2\} \\ \omega_j^N = 1 \forall j \in \{1, 2, \ldots, R^N\} \end{cases} \tag{17}$$

Therefore, the majority-class weights ω_j^N remain unchanged and the corresponding weighted confidence scores η_j^N are equal to the original confidence scores λ_j^N. Moreover, this also reduces the number of parameters (weights) that must be optimized, allowing for a faster optimization procedure. Nonetheless, it still requires optimization of all minority cloud weights ω_j^P, which may still result in excessively large training times.

In order to reduce the complexity of the weight optimization task, the proposed approach uses a simple criterion that selects the minority clouds where false negatives occur to be optimized.

In order to facilitate the computations required by the selection criterion and the optimization procedure itself, we introduce the distance matrix $\mathbf{D}$, defined as shown in (18). Thus, two distances matrices, $\mathbf{D}^N$ and $\mathbf{D}^P$, are computed, containing the pairwise distances between each sample $\mathbf{x}_k$ and each cloud focal point $\mathbf{f}_j$ of the negative and positive classes, respectively.

$$\mathbf{D} = \begin{pmatrix} \|\mathbf{x}_1 - \mathbf{f}_1\| & \ldots & \|\mathbf{x}_K - \mathbf{f}_1\| \\ \vdots & \ddots & \vdots \\ \|\mathbf{x}_1 - \mathbf{f}_R\| & \ldots & \|\mathbf{x}_K - \mathbf{f}_R\| \end{pmatrix} \tag{18}$$

Using the introduced distance matrix, the weighted model prediction $\hat{y}_k$ for sample $\mathbf{x}_k$ can also be rewritten as shown in (19), where $\odot$ represents the column-wise product of the weight vector by its class distance matrix.

$$\hat{y}_k = \arg\min\left(\min_j\left(\mathbf{!^N} \odot \mathbf{D}_{jk}^N\right), \min_j\left(\mathbf{!^P} \odot \mathbf{D}_{jk}^P\right)\right) \tag{19}$$

Returning to the cloud selection criterion, the index set K^* of false negative samples can be defined as shown in (20).

$$K^* = \{k \in K : y_k = P, \hat{y}_k = N\} \tag{20}$$

The index set of the minority clouds nearest to the false negative instances, J_P^*, can then be defined as shown in (21) and (22).

$$J_N^* = \left\{ \arg\min_j \left(\mathbf{D}_{jk}^N \right) : k \in K^* \right\} \tag{21}$$

$$J_P^* = \left\{ \arg\min_j \left(\mathbf{D}_{jk}^P \right) : k \in K^* \right\} \tag{22}$$

The candidate weights that would allow for capturing the sample are then obtained

$$W^* = \left\{ \frac{\mathbf{D}_{ik}^P}{\mathbf{D}_{jk}^N},\ i \in J_P^*,\ j \in J_N^*,\ k \in K^* \right\} \tag{23}$$

The search intervals for each weight are then defined as shown in (24).

$$\begin{cases} \omega_j^P \in \left[W_j^*,\ \ 1 \quad if j \in J_P^* \right] \\ j \in J_P^* \end{cases} \tag{24}$$

Finally, we define the objective function F, as shown in (25), as a function of the target y and the weighted target predictions $\hat{y}$.

$$F = F\left(y, \hat{y}\left(\mathbf{!^N}, \mathbf{!^P} \right) \right) \tag{25}$$

The complete optimization algorithm is presented in Algorithm 2.

Algorithm 2 ALMMo-0-W training algorithm.

```
Train initial model using Algorithm 1
Define search intervals for each weight using (24)
Get initial predictions using (19)
Get initial objective function value using (25)
while Stop criteria not met do
    Find next candidate value for each weight
    Get weighted predictions using (19)
    Get objective function value using (25)
```

4.2. Ensemble Architecture

The proposed ensemble approach is a multi-view ensemble architecture, consisting of multiple base classifiers, each one modeling one of the data streams recorded by each sensor. The predictions of the base models are then used to train a cost-sensitive stack classifier that outputs the final class prediction.

Consider a generic classification problem for a multi-view dataset $\mathcal{D}$ with n samples, m features, and v data views such that $\mathcal{D} = \left\{ \mathbf{X}^{(1)}, \mathbf{X}^{(2)}, \ldots \mathbf{X}^{(v)} \right\}$, where $\mathbf{X}^{(i)} = \left[\mathbf{x}_1^{(i)}, \mathbf{x}_2^{(i)}, \ldots, \mathbf{x}_n^{(i)} \right] \in \mathcal{R}^{n \times m_i}$ represents the i-th data view and m_i its number of features. Therefore, each data view has the same number of samples, but a unique subset of features, such that $m = \sum_{i=1}^{v} m_i$ is the total number of features. For a classification problem with c classes, we define the class

labels as $\mathbf{Y} = \{\mathbf{y}_1, \mathbf{y}_2, \ldots \mathbf{y}_n\} \in \mathcal{R}^{n \times c}$, where $\mathbf{y}_k = [y_{k1}, y_{k2}, \ldots, y_{kc}]$ defines the label of the k-th sample, with $y_{kj} = 1$ if the sample belongs to the j-th class and $y_{kj} = 0$ otherwise.

Having defined the generic problem for a multi-view dataset, we can now define the proposed ensemble structure formulation. Let $\mathcal{B}$ be the set of base classifiers in the ensemble model, as defined in (26), where $f^{(i)}\left(\theta^i, \mathbf{x}_k^{(i)}\right)$ represents the i-th base classifier as a mapping from the k-th sample $\mathbf{x}_k^{(i)}$ to the respective class prediction $\hat{\mathbf{y}}_k^{(i)}$ using the estimated model parameters θ^i.

$$\mathcal{B} = \left\{ f^{(1)}\left(\theta^1, \mathbf{x}_k^{(1)}\right), f^{(2)}\left(\theta^2, \mathbf{x}_k^{(2)}\right), \ldots, f^{(v)}\left(\theta^v, \mathbf{x}_k^{(v)}\right) \right\} \tag{26}$$

The base layer output $\hat{\mathbf{w}}$ can then be defined based on the v class predictions, such that $\hat{\mathbf{w}} = \left[\hat{\mathbf{y}}^{(1)}, \hat{\mathbf{y}}^{(2)}, \ldots, \hat{\mathbf{y}}^{(v)}\right] \in \mathcal{R}^{v \times c}$, with $\hat{w}_{ij}$ defining the i-th classifier prediction for the j-th class. The ensemble base layer can therefore be defined as $\hat{\mathbf{w}} = \mathcal{B}(\mathbf{x})$ defining a mapping from sample $\mathbf{x}$ to the base model prediction. The base model predictions are then used as the input to the stack classifier S, which aggregates the multiple class confidence scores into a final class label prediction $\hat{\mathbf{y}}$, such that $\hat{\mathbf{y}} = \mathcal{S}(\hat{\mathbf{w}})$ with $\hat{\mathbf{y}} \in \mathcal{R}^c$.

The ensemble formulation presented so far is generic and commonly used in the literature for classification problems on similarly structured datasets. As such, the main novelty of the methodology proposed in this paper is the ALMMo-0 (W) algorithm that is used for estimating the cost-sensitive stack classifier. Therefore, the proposed ensemble trains each base model as an ALMMo-0 classifier using Algorithm 1, while the stack model is trained using the proposed ALMMo-0 (W) approach, as presented in Algorithm 2. The proposed ensemble architecture is shown Figure 2. Therefore, only one model (the stack model) requires optimization of the cloud weights, instead of all the base models, thus drastically reducing the computation requirements and consequent training times, while still allowing the final ensemble model to compensate for majority-class bias.

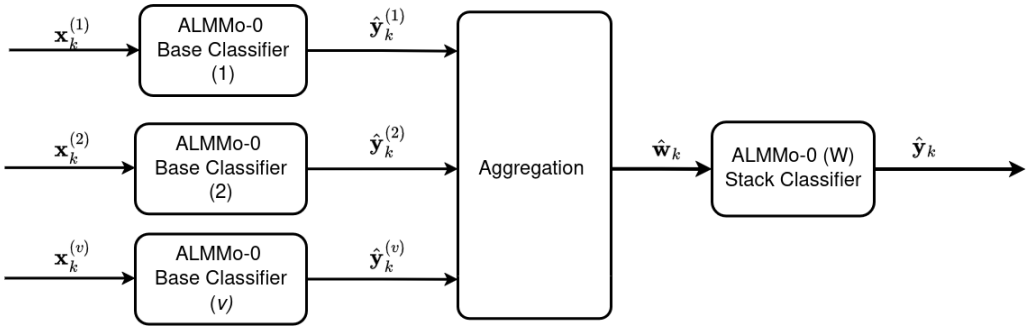

Figure 2. Proposed ensemble model architecture.

5. Results

In this section, the performance of the proposed intelligent architecture is assessed using two sets of tests. In order to evaluate the different elements of the proposed ensemble, the first set of tests evaluates the performance of the ALMMo-0 (W) by itself, and not only as part of the proposed ensemble architecture, on benchmark imbalanced datasets. The second set of tests evaluates the performance of the proposed ensemble architecture by assessing its performance on the intended real-world application of detecting patients with spastic diplegia using gait cycle measurements.

Both sets of tests consider the same group of classification algorithms for performance comparison. This group includes three benchmark methods (decision trees, k-nearest neighbors, and support vector machine) and the original ALMMo-0 classifier. The proposed

ALMMo-0 (W) is also tested with different cost function metrics. Table 1 lists the selected methods and their respective training parameters.

Table 1. Algorithms used in the tests and respective training parameters.

Algorithm	Parameter	Value
ALMMo-0	-	-
ALMMo-0 (W)	Maximum iterations	100
	Optimization metric	Geometric Mean
		F1-Score
		Matthews Correlation Coefficient
Decision Tree	Split quality criterion	Gini impurity
	Split strategy	Best split
K-Nearest Neighbors	K	5
	Distance metric	Euclidean
Support Vector Machine	Regularization parameter	1.0
	Kernel type	RBF
	Kernel Coefficient	2.0

In order to facilitate the discussion of the results and the performance details of the proposed approach, the classification metrics that are used throughout this section to discuss model performance are summarized in Table 2.

Table 2. Relevant classification metrics and respective definitions.

Metric	Definition
Recall	$\frac{TP}{TP+FN}$
Precision	$\frac{TP}{TP+FP}$
Specificity	$\frac{TN}{TN+FP}$
Balanced Accuracy	$\frac{Recall+Specificity}{2}$
Geometric Mean	$\sqrt{Recall+Specificity}$
F1-Score	$2\times\frac{Recall\times Precision}{Recall+Precision}$
Matthews Correlation Coefficient	$\frac{TP\times TN-FP\times FP}{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}$

5.1. Benchmark Datasets

In order to evaluate the performance of the proposed ALMMo-0 (W) approach on a diverse set of problems that are representative of common classification applications, it was decided to use the KEEL dataset repository, a well-known dataset repository that is often

used in the literature for testing model performance on imbalanced problems. The available datasets are mostly imbalanced versions of benchmark datasets that are resampled to increase the imbalance ratio or multi-class datasets that are decomposed into imbalanced binary tasks.

The benchmark procedure that was followed is based on 100 datasets that were selected from the repository, each with different class imbalance ratios, numbers of samples, types of features, and feature dimensionalities. These datasets are already partitioned using five-fold cross validation to standardize the training and test sets used for benchmark tests. Therefore, a total of 500 tests were performed for each algorithm. The selected benchmark datasets are summarized in Figure 3.

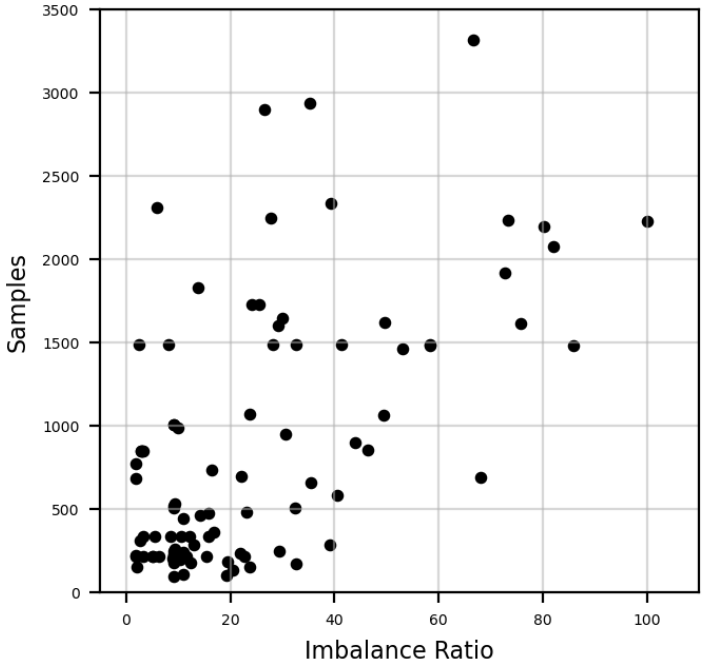

Figure 3. Benchmark dataset distribution in class imbalance ratio and number of samples.

The first set of results is shown in Table 3, which presents the pairwise comparisons of the original ALMMo-0 classifier, as well as the proposed modifications, against the other benchmark datasets. These comparisons are made in terms of balanced accuracy, recall, and precision.

Regarding the original ALMMo-0, it is clear that it generally outperforms the chosen benchmark methods in terms of balanced accuracy and recall, as evidenced by the win counts and respective mean relative improvements. Moreover, the Wilcoxon test shows that these wins are statistically significant for all cases, except for SVM, even though it shows a higher win count and scores. Moreover, the results also show that the original ALMMo-0 performs poorly in terms of precision, despite the Wilcoxon test showing weak statistical significance. Furthermore, the mean relative differences clearly show that the advantage in terms of recall is larger than that of precision in terms of performance. Therefore, the results show that the original ALMMo-0 shows the typical compromise between recall and precision, although it still shows good performance in the majority class, as evidenced by consistently higher balanced accuracy scores.

Table 3. Win–Tie–Loss, mean relative difference, and Wilcoxon signed-rank test *p*-value results of the original ALMMo-0 and proposed ALMMo-0 (W), measured against the benchmark methods in terms of balanced accuracy, recall, and precision.

Compared Against Algorithm on Metric		Algorithm			
		ALMMo-0	ALMMo-0 (W)		
			Weight Optimization Metric		
			Geometric Mean	F1-Score	Matthews Correlation Coefficient
ALMMo-0	Balanced Accuracy	-	62/378/60 +2.1 ± 11.1% (0.05)	52/412/36 +1.7 ± 9.2% (0.00)	53/403/44 +1.8 ± 9.7% (0.01)
	Recall	-	85/415/0 +8.8 ± 22.7% (0.00)	62/438/0 +5.4 ± 17.8% (0.00)	69/431/0 +6.4 ± 19.3% (0.00)
	Precision	-	21/382/97 −3.3 ± 10.5% (1.00)	25/412/63 −1.1 ± 6.0% (1.00)	20/404/76 −1.6 ± 7.2% (1.00)
Decision Tree	Balanced Accuracy	187/161/152 +11.6 ± 32.4% (0.01)	190/159/151 +12.6 ± 33.1% (0.00)	191/160/149 +12.4 ± 33.2% (0.00)	190/160/150 +12.5 ± 33.2% (0.00)
	Recall	173/228/99 +10.8 ± 45.6% (0.00)	199/223/78 +17.6 ± 47.4% (0.00)	187/228/85 +14.7 ± 47.2% (0.00)	191/226/83 +15.7 ± 47.3% (0.00)
	Precision	141/188/171 −2.9 ± 42.1% (0.85)	131/185/184 −4.9 ± 41.2% (0.99)	135/186/179 −3.9 ± 41.4% (0.95)	133/186/181 −4.0 ± 41.4% (0.96)
K-Nearest Neighbors	Balanced Accuracy	179/176/145 +12.0 ± 29.5% (0.00)	191/174/135 +13.0 ± 30.0% (0.00)	192/174/134 +12.9 ± 29.9% (0.00)	189/174/137 +12.9 ± 29.9% (0.00)
	Recall	203/239/58 +18.9 ± 39.1% (0.00)	230/229/41 +25.2 ± 42.0% (0.00)	225/231/44 +23.3 ± 40.8% (0.00)	227/230/43 +23.9 ± 41.3% (0.00)
	Precision	103/216/181 −8.6 ± 36.0% (1.00)	105/210/185 −10.2 ± 36.3% (1.00)	99/215/186 −9.5 ± 35.6% (1.00)	101/212/187 −9.6 ± 35.6% (1.00)
Support Vector Machine	Balanced Accuracy	237/133/130 +22.8 ± 39.7% (0.00)	245/133/122 +23.5 ± 39.5% (0.00)	240/133/127 +23.6 ± 39.7% (0.00)	245/133/122 +23.6 ± 39.8% (0.00)
	Recall	135/217/148 +7.3 ± 46.0% (0.10)	149/222/129 +12.0 ± 46.8% (0.00)	146/221/133 +10.2 ± 46.3% (0.01)	147/220/133 +10.7 ± 46.6% (0.00)
	Precision	260/160/80 +18.2 ± 44.4% (0.00)	258/156/86 +16.7 ± 43.7% (0.00)	259/158/83 +17.6 ± 43.9% (0.00)	258/157/85 +17.5 ± 44.0% (0.00)

Proceeding now to the proposed ALMMo-0 (W), and first comparing it to the original ALMMo-0, the results show improved performance in terms of balanced accuracy and recall, but worse results in terms of precision. Furthermore, this behavior is consistent for the different cost function metrics, with the Wilcoxon tests indicating a significant performance difference. However, the results also show that the different cost functions have different effects on the compromise made between improving recall and not excessively worsening precision. Using the geometric mean as the weight optimization metric seems to consistently

result in the largest recall improvements, at the expense of more severe drops in precision. Moreover, the results also indicate that using the F1-score as the optimization metric results in the most modest improvements in recall, which are matched with a very small drop in precision. The Matthews correlation coefficient seems to result in a middle ground between the other two metrics, even though the win counts and the mean relative differences show that it behaves more closely to the F1-score.

Regarding the pairwise comparisons of the proposed ALMMo-0 (W) against the benchmark methods, the results show exactly the same behavior as discussed for the ALMMo-0 (better balanced accuracy and recall scores, at the expense of worse precision scores), although more exaggerated, which is totally expected, attenuating the performance differences relative to the original ALMMo-0.

Proceeding now to the next set of results, shown in Table 4, the same pairwise comparison procedure is performed, but now in terms of geometric mean, F1-score, and Matthews correlation coefficient. Starting once again with ALMMo-0, against the benchmark methods it shows a consistent and significant better performance in terms of geometric mean, which is expected, considering the already-mentioned better balanced accuracy scores. Furthermore, the results also show that the ALMMo-0 classifier has comparable (or slightly worse) performance in terms of F1-score and Matthews correlation compared to decision trees and K-nearest neighbors, despite clearly outperforming the SVM in all the metrics.

Comparing once again the proposed ALMMo-0 (W) to the original ALMMo-0 classifier, the results show that the proposed modification clearly and consistently outperforms when comparing the geometric mean scores, but generally underperforms in terms of F1-score and Matthews correlation coefficient. While this behavior may seem unexpected, attending to the aforementioned improved performance in the other metrics, and also to the metrics being used in the weight optimization, it is relevant to mention that the optimization procedure itself does not guarantee improved performance in the objective function metric, since the cloud weights are tuned only on the training samples. Moreover, the distinct results obtained for the geometric mean indicate that the proposed ALMMo-0-W tends to aggressively compensate for the majority class by focusing on increasing the recall, and not so much on maintaining precision performance. Similar conclusions can be derived from observing the comparisons against the benchmark methods, which once again show consistently better geometric mean performances, but worse results when measuring the F1-score and Mathews correlation coefficient.

Attending to the aforementioned results, one can derive two key conclusions. Regarding the performance of the ALMMo-0 classifier, the results support the overall better performance on imbalanced datasets, showing consistently better performance when measured against benchmark classification methods of comparable complexity, while also having other important performance advantages in terms of computational requirements and interpretability. Regarding the proposed ALMMo-0 (W) approach, the results also support its adequacy for classification on imbalanced datasets, showing significant improvements over the original ALMMo-0 in terms of the predictive performance on the minority class, while not excessively degrading the predictive performance on the majority class. Furthermore, the results indicate that the choice of the cost function metric has significant impact on the performance of the weighted model, showing that the geometric mean consistently leads to the largest reductions in the number of false positives, at the expense of leading to more false positives, while the F1-score results in a more balanced bias compensation, which may be more adequate in some applications. Choosing the Matthews correlation coefficient seems to allow for a compromise between the two, which may also be more adequate in certain application requirements.

Table 4. Win–Tie–Loss, mean relative difference, and Wilcoxon signed-rank test *p*-value results of the original ALMMo-0 and proposed ALMMo-0 (W), measured against the benchmark methods in terms of geometric mean, F1-score, and Matthews correlation coefficient.

Compared Against Algorithm on Metric		Algorithm			
		ALMMo-0	ALMMo-0 (W)		
			Weight Optimization Metric		
			Geometric Mean	F1-Score	Matthews Correlation
ALMMo-0	Geometric Mean	-	60/381/59 +2.5 ± 12.6% (0.01)	50/412/38 +1.8 ± 9.7% (0.01)	52/403/45 +2.0 ± 10.5% (0.01)
	F1-Score	-	45/380/75 −0.9 ± 11.1% (0.84)	46/412/42 +0.5 ± 7.2% (0.03)	47/405/48 +0.2 ± 7.9% (0.10)
	Matthews Correlation	-	43/378/79 −2.0 ± 14.5% (0.95)	43/385/72 −0.4 ± 9.2% (0.11)	44/379/77 −0.7 ± 10.8% (0.29)
Decision Tree	Geometric Mean	193/167/140 +10.9 ± 40.9% (0.00)	199/165/136 +12.6 ± 40.5% (0.00)	196/166/138 +11.9 ± 41.5% (0.00)	196/166/138 +12.1 ± 41.3% (0.00)
	F1-Score	162/168/170 −0.7 ± 39.5% (0.36)	164/167/169 −0.4 ± 39.2% (0.30)	164/167/169 −0.0 ± 39.6% (0.18)	165/167/168 −0.0 ± 39.5% (0.19)
	Matthews Correlation	175/160/165 −2.2 ± 41.5% (0.39)	163/168/169 −2.4 ± 41.5% (0.37)	164/169/167 −1.7 ± 41.4% (0.26)	165/169/166 −1.8 ± 41.3% (0.28)
K-Nearest Neighbors	Geometric Mean	189/190/121 +16.8 ± 36.9% (0.00)	204/185/111 +18.2 ± 37.3% (0.00)	201/188/111 +17.9 ± 37.1% (0.00)	202/187/111 +17.9 ± 37.1% (0.00)
	F1-Score	147/191/162 +0.5 ± 32.7% (0.56)	159/185/156 +0.6 ± 33.5% (0.40)	158/189/153 +1.2 ± 33.0% (0.28)	158/188/154 +1.1 ± 33.0% (0.32)
	Matthews Correlation	134/176/190 −2.7 ± 35.3% (0.99)	129/193/178 −2.6 ± 35.5% (0.98)	135/194/171 −1.9 ± 34.8% (0.97)	132/193/175 −1.9 ± 34.6% (0.97)
Support Vector Machine	Geometric Mean	243/147/110 +25.8 ± 45.8% (0.00)	253/145/102 +27.2 ± 44.7% (0.00)	247/147/106 +26.8 ± 45.4% (0.00)	250/146/104 +26.9 ± 45.4% (0.00)
	F1-Score	263/148/89 +18.4 ± 42.5% (0.00)	267/146/87 +18.7 ± 41.8% (0.00)	264/148/88 +19.1 ± 42.1% (0.00)	264/147/89 +19.0 ± 42.2% (0.00)
	Matthews Correlation	264/144/92 +16.0 ± 45.2% (0.00)	253/152/95 +16.1 ± 44.5% (0.00)	255/153/92 +16.6 ± 44.4% (0.00)	257/152/91 +16.8 ± 44.1% (0.00)

Finally, we performed a one-way ANOVA test for all methods and for each classification performance metric, thus evaluating if the methods yield significantly different results. The results are presented in Table 5.

Table 5. One-way ANOVA test results for various classification performance metrics.

Metric	F-Statistic	*p*-Value
Balanced Accuracy	42.46	0.000
Recall	60.36	0.000
Precision	11.49	0.000
Geometric Mean	84.67	0.000
F1-Score	18.05	0.000
Matthews Correlation	23.91	0.000

5.2. Spastic Diplegia Dataset

The dataset consists of kinematic and kinetic measurements obtained from computerized clinical gait analysis from a pool of participants (both CP children and typically developed children) that were assessed in the Biomechanics and Functional Morphology Laboratory at FMH. The gait analysis protocol (described in [34]) was approved by and executed in accordance with the Faculty of Human Kinetics Ethics Committee (CEFMH-2/2019).

Each gait cycle measurement consists of four joint angles and three joint moments, each one recorded in space (X, Y, and Z axes), meaning that a total of 21 data streams (each one composed of 101 data points) are used to describe each gait cycle.

The approach proposed in this paper consists of a multi-view ensemble architecture that considers each angle and moment measurement as a data view that is used to train a single base model, meaning that each ensemble model consists of 21 base models. The ensemble prediction is accomplished using a stack model, which is trained on the predictions of each base model. This ensemble structure was used to obtain all the results presented in this section, changing only the learning algorithms used to train the base models and the stack model. The approach proposed in this paper specifically refers to the algorithms used for the training procedure, using the ALMMo-0 classifier to train the base models and using the proposed ALMMo-0 (W) approach to train the stack model.

The modeling approach presented in this article focuses on detecting a specific type of gait cycle anomaly known as true equinus, often found in individuals with spastic diplegia. The dataset used is composed as shown in Table 6, consisting of 25 healthy individuals (control group) with a total of 183 measured gait cycles and 6 individuals suffering from spastic diplegia (patient group) with a total of 27 recorded gait cycles. Taking into account external factors such as noise and equipment failure, not all leg recordings were available or useful, which is also reflected in Table 6.

Table 6. Class distribution for the control (healthy) and patient groups (individuals with spastic diplegia and true equinus gait patterns) and complete samples (all angles and moments are available).

Group	Samples		
	Individuals	Legs	Gait Cycles
Control	25 (81%)	50 (85%)	183 (87%)
Patient	6 (19%)	9 (15%)	27 (13%)

In order to maximize the number of samples available for training the models, each gait cycle (and respective 21 data streams) is considered a sample. Regarding the validation procedure, five-fold cross-validation is used, meaning that five tests are performed for each ensemble setup. However, in order to avoid bias from the samples included in each train–test split, the data were partitioned to ensure that the gait cycles of each individual are only part of either the training set or the test set. Furthermore, despite treating each leg

as an independent set of measurements, data partitioning was also performed to ensure that the two legs are both included either in the training set or the test set. In order to fully accommodate these two partition restrictions, and considering the low number of individuals (particularly in the patient group), the resultant partitions show some variation in the ratio of training and test samples and, in particular, in the class imbalance ratios in the training and test sets. The partitions used for the modeling and testing procedures are shown in Table 7.

Table 7. Number of gait cycles (samples) for each train–test split and respective class distribution and imbalance ratio.

Split	Gait Cycles				Class Imbalance Ratio	
	Control		Patient			
	Train	Test	Train	Test	Train	Test
1	147	36	20	7	7.4	5.1
2	149	34	20	7	7.5	4.9
3	150	33	20	7	7.5	4.7
4	142	41	20	7	7.1	5.9
5	144	39	21	6	6.9	6.5

The results obtained for the different ensemble models are presented in Table 8, which shows the different performance metrics for each ensemble setup. The results clearly show better predictive performance for ensembles that have ALMMo-0 classifiers as base models, as shown by the higher scores across all metrics. Furthermore, it is also clear that the class imbalance present in the dataset does not seem to have a significant impact on the performance of the ensemble models, as evidenced by the high recall scores. In fact, the results even show that ensembles trained with the decision tree and K-nearest neighbors classifiers have better recall scores than precision scores. As discussed before, this behavior would not necessarily be problematic and could even indicate good predictive performance on the minority class. However, observing recall and precision results for ensembles trained with the ALMMo-0 algorithm, it is clear that these not only show better recall scores, but also achieve perfect precision results. One clear exception is the support vector machine ensembles, which also achieve perfect precision results, despite also showing slightly lower recall scores.

Regarding the effects on the ensemble performance of using the proposed ALMMo-0 (W) approach to train the stack model, the results show an effect very similar to the one discussed for the benchmark tests, with significance recall improvements. Furthermore, the effect of the cost function metrics on the minority-class detection performance is similar to the one observed for the benchmark tests, with the geometric mean providing the largest reduction in false negatives, which results in achieving perfect recall scores (at the expense of a large decrease in precision scores). The F1-score and Matthews correlation metrics allow for more modest recall improvements, which is consistent with the results obtained for the benchmark tests.

However, one key difference in the effects that these two metrics have on the weighted model is that they are able to maintain the precision scores of the original ALMMo-0, which is even more impressive when considering that they achieve perfect precision scores, meaning that no false positive predictions are made. Another key difference is that the ALMMo-0 (W) stack models are also able to improve the F1-score and Matthews correlation performances in the testing data, which was not observed in the benchmark datasets. In particular, stack models trained using the proposed ALMMo-0 (W) and the Matthews coefficient as the optimization metric allow for the best overall ensemble performances,

achieving the best scores in balanced accuracy, geometric mean, F1-score, and the Matthews coefficient itself, while also keeping the perfect precision scores of the original ALMMo-0.

Table 8. Balanced accuracy, recall, precision, geometric mean, F1-score, and Matthews correlation coefficient results obtained with the proposed ensemble architecture and using different learning algorithms to train the base models and the stack model of each ensemble. Bold cells mark the best in the column.

Ensemble Algorithms		**Metric**					
Base Model	**Stack Model**	**Balanced Accuracy**	**Recall**	**Precision**	**Geometric Mean**	**F1-Score**	**Matthews Correlation**
ALMMo-0	ALMMo-0	0.943 ± 0.078	0.886 ± 0.156	**1.000 ± 0.000**	0.938 ± 0.085	0.933 ± 0.091	0.929 ± 0.097
	ALMMo-0 (W) (Geometric Mean)	0.924 ± 0.073	**1.000 ± 0.000**	0.700 ± 0.283	0.919 ± 0.080	0.798 ± 0.195	0.767 ± 0.222
	ALMMo-0 (W) (F1-Score)	0.950 ± 0.112	0.900 ± 0.224	**1.000 ± 0.000**	0.941 ± 0.131	0.933 ± 0.149	0.935 ± 0.144
	ALMMo-0 (W) (Matthews Correlation)	**0.967 ± 0.075**	0.933 ± 0.149	**1.000 ± 0.000**	**0.963 ± 0.082**	**0.960 ± 0.089**	**0.956 ± 0.099**
Decision Tree	Decision Tree	0.827 ± 0.140	0.738 ± 0.232	0.668 ± 0.270	0.815 ± 0.154	0.686 ± 0.234	0.630 ± 0.284
K-Nearest Neighbors	K-Nearest Neighbors	0.808 ± 0.127	0.767 ± 0.325	0.437 ± 0.095	0.781 ± 0.157	0.517 ± 0.106	0.488 ± 0.131
Support Vector Machine	Support Vector Machine	0.914 ± 0.093	0.829 ± 0.186	**1.000 ± 0.000**	0.905 ± 0.105	0.897 ± 0.117	0.892 ± 0.119

In order to better understand the performance results for the ensemble models, it is also important to analyze the performances of the individual base models, particularly those trained using the ALMMo-0 algorithm. These results are summarized in Tables 9 and 10, which show the performance metrics for the ALMMo-0 base models trained on the kinematic and kinematic measurements, respectively.

Regarding the base models trained on kinematic features, the results in Table 9 show that there is considerable diversity in the different base model performances, and some models outperform the respective ensemble model in some metrics. In particular, the base models trained on knee joint angles in the X axis show the overall best performances for all metrics, while the base models trained on ankle joint angles in the Y axis are able to achieve perfect recall scores.

However, a key result shown in Table 9 is that no base model is able to outperform the respective ensemble model in all metrics. In particular, no base model achieves perfect precision scores, as was the case for the ensemble models. This result suggests that the ensemble approach is able to combine the different base models in order to improve precision performance. Moreover, since there are multiple base models that outperform the ensemble in recall scores, it seems that the precision improvements come at the expense of

slightly worse recall scores. Therefore, the main advantage of training the stack models using the proposed ALMMo-0 (W) is that they allow for recovering the better recall scores of the ALMMo-0 base models, while also keeping the precision improvements offered by the ensemble architecture itself.

Table 9. Balanced accuracy, recall, precision, geometric mean, F1-score, and Matthews correlation coefficient results for the ALMMo-0 base models trained on kinematic (angle) features. Bold cells mark the best in the column.

Angle Feature		**Metric**					
Joint	**Axis**	**Balanced Accuracy**	**Recall**	**Precision**	**Geometric Mean**	**F1-Score**	**Matthews Correlation**
Ankle	X	0.766 ± 0.152	0.667 ± 0.312	0.397 ± 0.108	0.741 ± 0.173	0.485 ± 0.148	0.425 ± 0.206
	Y	0.961 ± 0.010	**1.000 ± 0.000**	0.386 ± 0.069	0.961 ± 0.010	0.554 ± 0.073	0.595 ± 0.059
	Z	0.754 ± 0.220	0.667 ± 0.312	0.600 ± 0.303	0.737 ± 0.225	0.598 ± 0.243	0.497 ± 0.387
Hip	X	0.964 ± 0.030	0.960 ± 0.055	0.578 ± 0.067	0.963 ± 0.030	0.720 ± 0.062	0.730 ± 0.061
	Y	0.884 ± 0.023	0.847 ± 0.065	0.350 ± 0.069	0.882 ± 0.024	0.491 ± 0.066	0.511 ± 0.055
	Z	0.870 ± 0.135	0.843 ± 0.205	0.656 ± 0.275	0.866 ± 0.140	0.724 ± 0.232	0.678 ± 0.278
Knee	X	**0.985 ± 0.022**	0.980 ± 0.045	0.834 ± 0.099	**0.984 ± 0.023**	**0.899 ± 0.061**	**0.898 ± 0.062**
	Y	0.751 ± 0.109	0.600 ± 0.224	0.520 ± 0.292	0.727 ± 0.121	0.523 ± 0.171	0.473 ± 0.214
	Z	0.923 ± 0.059	0.907 ± 0.130	0.447 ± 0.127	0.921 ± 0.062	0.587 ± 0.112	0.606 ± 0.100
Pelvis	X	0.929 ± 0.065	0.893 ± 0.123	0.557 ± 0.128	0.927 ± 0.069	0.681 ± 0.122	0.686 ± 0.122
	Y	0.661 ± 0.156	0.367 ± 0.217	0.800 ± 0.447	0.540 ± 0.307	0.500 ± 0.289	0.466 ± 0.390
	Z	0.689 ± 0.165	0.548 ± 0.334	0.287 ± 0.144	0.645 ± 0.214	0.373 ± 0.195	0.291 ± 0.244

Regarding the base models trained on the kinetic features, the results in Table 10 show a very similar performance pattern to that just discussed for the kinematic features. Concretely, the base model with the best overall performance is the one trained on the ankle joint moment On the X axis, which outperforms the other base models in terms of balanced accuracy, precision, geometric mean, F1-score, and Matthews correlation coefficient. Moreover, once again, there is only one base model that achieves perfect recall scores, which corresponds to the base model trained on the knee joint moments on the Z axis. Similarly to the results discussed for the base models trained on kinematic features, it is also verified that no base model is able to achieve the perfect precision scores of the respective ensemble models, although multiple base models achieve better recall scores.

Table 10. Balanced accuracy, recall, precision, geometric mean, F1-score, and Matthews correlation coefficient results for the ALMMo-0 base models trained on kinetic (moment) features. Bold cells mark best in the column.

Moment Feature		Metric					
Joint	**Axis**	**Balanced Accuracy**	**Recall**	**Precision**	**Geometric Mean**	**F1-Score**	**Matthews Correlation**
Hip	X	0.907 ± 0.069	0.847 ± 0.139	0.541 ± 0.096	0.902 ± 0.075	0.656 ± 0.096	0.657 ± 0.101
	Y	0.839 ± 0.073	0.727 ± 0.130	0.470 ± 0.191	0.829 ± 0.081	0.562 ± 0.172	0.551 ± 0.171
	Z	0.814 ± 0.086	0.667 ± 0.170	0.454 ± 0.087	0.795 ± 0.109	0.538 ± 0.116	0.523 ± 0.127
Knee	X	0.919 ± 0.044	0.880 ± 0.084	0.502 ± 0.061	0.917 ± 0.045	0.637 ± 0.060	0.643 ± 0.063
	Y	0.911 ± 0.044	0.927 ± 0.083	0.296 ± 0.074	0.910 ± 0.044	0.445 ± 0.082	0.488 ± 0.074
	Z	0.963 ± 0.004	**1.000 ± 0.000**	0.388 ± 0.047	0.962 ± 0.004	0.557 ± 0.047	0.598 ± 0.036
Pelvis	X	0.966 ± 0.026	0.940 ± 0.055	**0.850 ± 0.074**	0.965 ± 0.027	0.891 ± 0.048	0.887 ± 0.049
	Y	0.865 ± 0.051	0.795 ± 0.075	0.733 ± 0.208	0.861 ± 0.053	0.749 ± 0.116	0.706 ± 0.144
	Z	0.852 ± 0.164	0.767 ± 0.325	0.633 ± 0.217	0.830 ± 0.196	0.673 ± 0.239	0.643 ± 0.271

Last but not least, we performed ensemble ablation by removing each of the base classifiers and evaluating the impact on the ensemble predictive performance and respective classification metrics. The results are presented in Table 11.

Taking into account the aforementioned results, we can derive the following three key conclusions. Firstly, the results suggest that the proposed ensemble architecture is adequate for the performance requirements of the problem and the structure of the dataset. This adequacy of the proposed ensemble architecture is independent of the learning algorithms used to obtain the base models and the stack model. This conclusion is supported by the overall good results for the different algorithms, which show that despite the significant performance differences between the different learning algorithms, all the ensemble models are able to achieve satisfactory predictive performance in the minority class (as evidenced by the high recall scores shown in Table 8). Concretely, the ensembles with the ALMMo-0 and SVM base models are shown to be the most adequate, showing good results on both classes, as evidenced by the high scores for all performance metrics shown in Table 8.

The second key conclusion that is derived from the aforementioned results is that the proposed approach (which refers to the specific ensemble setup combining ALMMo-0 base models with a stack model trained using the proposed ALMMo-0 (W)) is shown to outperform the ensemble approaches that train the stack model using the original ALMMo-0 algorithm. In practice, all ensembles with ALMMo-0 (W) stack models are able to significantly improve the recall scores of the respective ensemble model. Moreover, the impact of the different cost function metrics on the overall performance of the ensemble model is shown to be similar to the one observed for the individual ALMMo-0 (W) performances on the benchmark datasets, with the geometric mean metric showing the largest recall

improvements (at the expense of the largest drops in precision), followed by the Matthews coefficient and the F1-score.

Table 11. Ensemble ablation by removal of each one of the base classifiers and respective results.

Removed Base Model			Metrics					
Measure	**Joint**	**Axis**	**Balanced Accuracy**	**Recall**	**Precision**	**Geometric Mean**	**F1-Score**	**Matthews Coefficient**
Angle	Ankle	X	0.907 ± 0.096	0.847 ± 0.150	0.772 ± 0.032	0.897 ± 0.054	0.775 ± 0.124	0.768 ± 0.093
		Y	0.902 ± 0.072	0.839 ± 0.032	0.772 ± 0.089	0.892 ± 0.042	0.774 ± 0.041	0.763 ± 0.032
		Z	0.907 ± 0.065	0.847 ± 0.117	0.767 ± 0.062	0.897 ± 0.068	0.773 ± 0.022	0.766 ± 0.098
	Hip	X	0.902 ± 0.131	0.840 ± 0.021	0.768 ± 0.053	0.892 ± 0.065	0.770 ± 0.143	0.760 ± 0.060
		Y	0.904 ± 0.107	0.843 ± 0.085	0.773 ± 0.061	0.894 ± 0.150	0.775 ± 0.081	0.766 ± 0.083
		Z	0.904 ± 0.039	0.843 ± 0.132	0.766 ± 0.105	0.894 ± 0.080	0.769 ± 0.022	0.761 ± 0.053
	Knee	X	0.901 ± 0.096	0.839 ± 0.064	0.761 ± 0.079	0.891 ± 0.037	0.765 ± 0.109	0.756 ± 0.031
		Y	0.907 ± 0.109	0.849 ± 0.132	0.769 ± 0.112	0.897 ± 0.086	0.775 ± 0.107	0.766 ± 0.102
		Z	0.903 ± 0.063	0.841 ± 0.114	0.771 ± 0.065	0.893 ± 0.075	0.773 ± 0.046	0.763 ± 0.140
	Pelvis	X	0.903 ± 0.102	0.841 ± 0.123	0.768 ± 0.086	0.892 ± 0.041	0.771 ± 0.045	0.761 ± 0.137
		Y	0.909 ± 0.126	0.855 ± 0.130	0.762 ± 0.101	0.902 ± 0.041	0.775 ± 0.062	0.767 ± 0.146
		Z	0.909 ± 0.040	0.850 ± 0.116	0.775 ± 0.041	0.899 ± 0.107	0.778 ± 0.028	0.771 ± 0.138
Moment	Hip	X	0.903 ± 0.112	0.843 ± 0.100	0.768 ± 0.054	0.893 ± 0.057	0.771 ± 0.100	0.762 ± 0.081
		Y	0.905 ± 0.078	0.846 ± 0.120	0.770 ± 0.089	0.895 ± 0.087	0.774 ± 0.092	0.765 ± 0.093
		Z	0.905 ± 0.044	0.847 ± 0.117	0.771 ± 0.054	0.896 ± 0.059	0.774 ± 0.100	0.765 ± 0.026
	Knee	X	0.903 ± 0.148	0.842 ± 0.110	0.769 ± 0.054	0.893 ± 0.129	0.772 ± 0.062	0.762 ± 0.021
		Y	0.903 ± 0.069	0.841 ± 0.110	0.775 ± 0.141	0.893 ± 0.038	0.776 ± 0.023	0.766 ± 0.069
		Z	0.902 ± 0.103	0.839 ± 0.120	0.772 ± 0.136	0.892 ± 0.090	0.774 ± 0.058	0.763 ± 0.124
	Pelvis	X	0.902 ± 0.064	0.840 ± 0.050	0.761 ± 0.117	0.891 ± 0.054	0.765 ± 0.117	0.756 ± 0.067
		Y	0.904 ± 0.028	0.844 ± 0.075	0.764 ± 0.061	0.894 ± 0.024	0.769 ± 0.094	0.761 ± 0.041
		Z	0.905 ± 0.075	0.845 ± 0.106	0.766 ± 0.106	0.895 ± 0.031	0.771 ± 0.075	0.762 ± 0.100

The third and final conclusion concerns the performance compromise between recall scores and precision scores and its relation to the performance of the individual base models. The results presented in Table 8 show a somewhat unexpected performance pattern (not observed in the benchmark results), with recall scores that are consistently higher than precision scores, despite the significant class imbalance ratio present in the dataset. This performance pattern is also observed in the individual base models, as evidenced by the recall and precision results shown in Tables 9 and 10, suggesting that this performance pattern is not caused by the ensemble architecture itself and its effects on the combination of the base model predictions to obtain the overall class prediction. Considering the prevalence of this performance pattern for the different base models and learning algorithms, this is likely caused by the specific characteristics of the dataset itself and, in particular, by the data distributions of the two classes. Concretely, the data distributions of the two classes seem to be characterized by the prevalence of outliers belonging to the majority class inside regions of the data space that are characteristic of the minority class (significantly higher density of minority samples). In the specific context of the considered dataset, this implies that there is a significant number of gait cycles belonging to healthy individuals (majority class) that show some patterns that are characteristic of gait cycles belonging to patients with spastic diplegia (minority class). Furthermore, it is also possible that these patterns are only prevalent in some of the measurements and also that their impact is only significant because of the relatively small number of samples in the dataset. It is also possible that these ambiguous patterns are a limitation of the measuring apparatus, which may not be

able to capture some patterns that are easily detectable by a medical expert observing the gait cycles of the individual.

However, regardless of the actual cause of the higher recall results, the ALMMo-0 stack models are shown to take advantage of the performance of the base model. Concretely, the ALMMo-0 stack models can use the relatively weak precision score performances of the base models to obtain a final ensemble model that achieves perfect precision scores, at the expense of worse recall scores. Furthermore, stack models trained using the proposed ALMMo-0 (W) approach (particularly when using the F1-score and Matthews coefficient as weight optimization metrics) take advantage of the performance improvements obtained by the unweighted ALMMo-0 stack models (also achieving perfect precision scores) while significantly improving the recall scores, leading to final ensemble models that are able to achieve a small number of false negatives (resulting in high recall scores), as well as a small number of false positives (resulting in high precision scores).

Attending to the robust predictive performance shown by the proposed modeling approach, improved minority-class detection capabilities, and explainable model structure, we suggest that our approach would be adequate for various cerebral palsy medical applications, such as wearable rehabilitation devices, real-time patient monitoring, clinical decision-support dashboards, and patient progress tracking.

6. Conclusions

This paper proposed intelligent models, namely ensemble fuzzy rule-based models, to automate the detection of a type of gait cycle anomaly known as true equinus, found in individuals suffering from spastic diplegia. The diagnostic application addressed in this article may help to achieve a more accurate identification of the gait patterns characteristic of cerebral palsy and consequently facilitate correct decision-making regarding the type of intervention to be performed with the patients (surgical, orthotic, or spasticity management).

The resultant data structure (characterized by multiple data streams measured by the instrumented gait analysis) poses challenging restrictions to the applicable modeling approaches, which must be adequate for a dataset characterized by multiple independent data sources, a large class imbalance, and a small sample size of spastic diplegia cases. Furthermore, medical diagnostic intelligent models are also required to be explainable and interpretable, which means that human experts must be able to understand the reasoning behind the model predictions.

In order to address all the restrictions and requirements posed by the addressed problem, this paper proposes a novel approach consisting of a multi-view ensemble architecture to address the challenge posed by the multiple data sources and efficient parallel processing requirements. Moreover, the proposed approach addresses the interpretability requirements by choosing fuzzy rule-based systems as the base models of the ensemble and using the ALMMo-0 classifier to efficiently obtain the fuzzy models. Finally, the challenge posed by the class imbalance is addressed by modeling the ensemble stack model using the proposed ALMMo-0 (W), which modifies the original ALMMo-0 learning algorithm by introducing a weighting scheme to the minority-class rules and a cost-sensitive optimization procedure.

The proposed approach is evaluated in two sets of tests. The first set of tests uses multiple benchmark datasets to evaluate the proposed ALMMo-0 (W) method, showing that it consistently outperforms the original ALMMo-0 and also showing that the improvements in minority-class detection are controllable by the chosen optimization metric. The second set of tests addresses the diagnostic application addressed in this paper, showing that the selected ensemble architecture is adequate for the problem structure and requirements for multiple learning algorithms. Moreover, it also shows that the specific ensemble setup

of the proposed intelligent approach results in significantly better-performing diagnostic models, outperforming the ensemble setups that exclusively use the original ALMMo-0 learning algorithm to obtain the individual models of the ensemble.

Furthermore, it is also shown that there exist significant predictive performance differences between the different base models and their respective data sources. Sagittal joint angle measurements and sagittal ankle joint moment measurements are the most descriptive features of the dataset, resulting in base models with distinctly better performance than base models trained on other features. It is also shown that the ensemble structure is able to combine the strengths of the different base models and improve the overall predictive performance, consistently outperforming the performances of the individual base models. The proposed approach can be integrated into wearable rehabilitation devices for real-time gait monitoring or used in clinical decision-support dashboards to assist healthcare professionals in diagnosing and tracking gait abnormalities.

In future work, it is suggested to apply the proposed approach to problems that have similar data structures and performance requirements, not only on medical diagnostic applications but also in different fields and on data-based modeling problems in the automation field, which pose comparable challenges and requirements. Moreover, it is suggested that the proposed ALMMo-0 (W) approach can be considered individually, and not exclusively as part of the proposed ensemble architecture, as it may be adequate for other types of imbalanced problems and show different results by considering different performance metrics not tested in this paper. Finally, as the methodology was developed for true equinus gait patterns in spastic diplegia individuals, its generalizability to other gait disorders or broader clinical populations remains to be validated.

Author Contributions: Conceptualization, R.B.V., S.M.V., F.J., A.P.V. and J.M.C.S.; methodology, R.B.V., S.M.V., F.J., A.P.V. and J.M.C.S.; software, R.B.V.; validation, S.M.V., F.J., A.P.V. and J.M.C.S.; writing—original draft preparation, R.B.V., S.M.V. and F.J.; writing—review and editing, R.B.V., S.M.V., F.J., A.P.V. and J.M.C.S.; supervision, S.M.V. and J.M.C.S.; project administration, F.J. and A.P.V.; funding acquisition, S.M.V., F.J., A.P.V. and J.M.C.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the Fundação para a Ciência e Tecnologia (FCT) under the AI4Life project scholarship DSAIPA/DS/0054/2019, grant number UIDB/00447/2020, attributed to CIPER–Centro Interdisciplinar para o Estudo da Performance Humana (unit 447; DOI: 10.54499/UIDB/00447/2020), the CPJoyWalk project grant PTDC/EMD-EMD/5804/2020, and the PhD scholarship 2022.14216.BDANA, and through IDMEC, under LAETA Funding, project UID/50022/2025, to whom the authors express their gratitude.

Data Availability Statement: The data in this paper is unavailable due to privacy or ethical restrictions.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Lughofer, E.; Pratama, M. Evolving multi-user fuzzy classifier system with advanced explainability and interpretability aspects. *Inf. Fusion* **2022**, *91*, 458–476. [CrossRef]
2. Gilpin, L.H.; Bau, D.; Yuan, B.Z.; Bajwa, A.; Specter, M.; Kagal, L. Explaining Explanations: An Overview of Interpretability of Machine Learning. *arXiv* **2019**. [CrossRef]
3. Ouifak, H.; Idri, A. On the performance and interpretability of Mamdani and Takagi-Sugeno-Kang based neuro-fuzzy systems for medical diagnosis. *Sci. Afr.* **2023**, *20*, e01610. [CrossRef]
4. Shilaskar, S.; Ghatol, A.; Chatur, P. Medical decision support system for extremely imbalanced datasets. *Inf. Sci.* **2017**, *384*, 205–219. [CrossRef]
5. Johnson, J.M.; Khoshgoftaar, T.M. Survey on deep learning with class imbalance. *J. Big Data* **2019**, *6*, 27. [CrossRef]

6. López, V.; Fernández, A.; García, S.; Palade, V.; Herrera, F. An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics. *Inf. Sci.* **2013**, *250*, 113–141. [CrossRef]
7. Barella, V.H.; Garcia, L.P.; De Souto, M.C.; Lorena, A.C.; De Carvalho, A.C. Assessing the data complexity of imbalanced datasets. *Inf. Sci.* **2021**, *553*, 83–109. [CrossRef]
8. Soares, E.; Angelov, P.; Gu, X. Autonomous Learning Multiple-Model zero-order classifier for heart sound classification. *Appl. Soft Comput.* **2020**, *94*, 106449. [CrossRef]
9. Škrjanc, I.; Iglesias, J.A.; Sanchis, A.; Leite, D.; Lughofer, E.; Gomide, F. Evolving fuzzy and neuro-fuzzy approaches in clustering, regression, identification, and classification: A Survey. *Inf. Sci.* **2019**, *490*, 344–368. [CrossRef]
10. Angelov, P.; Gu, X. Autonomous learning multi-model classifier of 0-Order (ALMMo-0). In Proceedings of the IEEE Conference on Evolving and Adaptive Intelligent Systems, Ljubljana, Slovenia, 31 May–2 June 2017; pp. 1–7. [CrossRef]
11. Rosenbaum, P.; Paneth, N.; Leviton, A.; Goldstein, M.; Bax, M.; Damiano, D.; Dan, B.; Jacobsson, B. A report: The definition and classification of cerebral palsy April 2006. *Dev. Med. Child Neurol. Suppl.* **2007**, *109*, 8–14.
12. Rodda, J.M.; Graham, H.K.; Carson, L.; Galea, M.P.; Wolfe, R. Sagittal gait patterns in spastic diplegia. *J. Bone Jt. Surg. Br.* **2004**, *86*, 251–258. [CrossRef]
13. Rodda, J.; Graham, H.K. Classification of gait patterns in spastic hemiplegia and spastic diplegia: A basis for a management algorithm. *Eur. J. Neurol.* **2001**, *8*, 98–108. [CrossRef]
14. Terano, T.; Asai, K.; Sugeno, M. *Applied Fuzzy Systems*; Academic Press: Cambridge, MA, USA, 2014.
15. Sousa, J.M.C.; Kaymak, U. *Fuzzy Decision Making in Modeling and Control*; World Scientific Inc.: Singapore, 2002.
16. Angelov, P.; Zhou, X. Evolving Fuzzy-Rule-Based Classifiers from Data Streams. *IEEE Trans. Fuzzy Syst.* **2008**, *16*, 1462–1475. [CrossRef]
17. Gu, X.; Angelov, P.P. Self-organising fuzzy logic classifier. *Inf. Sci.* **2018**, *447*, 36–51. [CrossRef]
18. Angelov, P.P.; Gu, X.; Principe, J.C. Autonomous Learning Multimodel Systems from Data Streams. *IEEE Trans. Fuzzy Syst.* **2018**, *26*, 2213–2224. [CrossRef]
19. Gu, X.; Angelov, P.; Zhao, Z. Self-organizing fuzzy inference ensemble system for big streaming data classification. *Knowl. Based Syst.* **2021**, *218*, 106870. [CrossRef]
20. Gu, X. Multilayer Ensemble Evolving Fuzzy Inference System. *IEEE Trans. Fuzzy Syst.* **2020**, *29*, 2425–2431. [CrossRef]
21. Viegas, R.; Salgado, C.M.; Curto, S.; Carvalho, J.P.; Vieira, S.M.; Finkelstein, S.N. Daily prediction of ICU readmissions using feature engineering and ensemble fuzzy modeling. *Expert Syst. Appl.* **2020**, *79*, 244–253. [CrossRef]
22. Salgado, C.M.; Vieira, S.M.; Mendonça, L.F.; Finkelstein, S.; Sousa, J.M. Ensemble fuzzy models in personalized medicine: Application to vasopressors administration. *Eng. Appl. Artif. Intell.* **2016**, *49*, 141–148. [CrossRef]
23. Singh, A.; Purohit, A. A Survey on Methods for Solving Data Imbalance Problem for Classification. *Int. J. Comput. Appl.* **2015**, *127*, 37–41. [CrossRef]
24. Nejatian, S.; Parvin, H.; Faraji, E. Using sub-sampling and ensemble clustering techniques to improve performance of imbalanced classification. *Neurocomputing* **2018**, *276*, 55–66. [CrossRef]
25. More, A. Survey of resampling techniques for improving classification performance in unbalanced datasets. *arXiv* **2016**, arXiv:1608.06048. [CrossRef]
26. Pang, Y.; Chen, Z.; Peng, L.; Ma, K.; Zhao, C.; Ji, K. A signature-based assistant random oversampling method for malware detection. In Proceedings of the 2019 18th IEEE International Conference on Trust, Security and Privacy in Computing and Communications/13th IEEE International Conference on Big Data Science and Engineering, TrustCom/BigDataSE 2019, Rotorua, New Zealand, 5–8 August 2019; pp. 256–263. [CrossRef]
27. Prusa, J.; Khoshgoftaar, T.M.; DIttman, D.J.; Napolitano, A. Using Random Undersampling to Alleviate Class Imbalance on Tweet Sentiment Data. In Proceedings of the 2015 IEEE 16th International Conference on Information Reuse and Integration, IRI 2015, San Francisco, CA, USA, 13–15 August 2015; pp. 197–202. [CrossRef]
28. Fernández, A.; García, S.; Herrera, F.; Chawla, N.V. SMOTE for Learning from Imbalanced Data: Progress and Challenges, Marking the 15-year Anniversary. *J. Artif. Intell. Res.* **2018**, *61*, 863–905. [CrossRef]
29. Ning, Q.; Zhao, X.; Ma, Z. A Novel Method for Identification of Glutarylation Sites Combining Borderline-SMOTE With Tomek Links Technique in Imbalanced Data. *IEEE/ACM Trans. Comput. Biol. Bioinform.* **2022**, *19*, 2632–2641. [CrossRef]
30. Lango, M.; Stefanowski, J. Multi-class and feature selection extensions of Roughly Balanced Bagging for imbalanced data. *J. Intell. Inf. Syst.* **2018**, *50*, 97–127. [CrossRef]
31. Yin, L.; Ge, Y.; Xiao, K.; Wang, X.; Quan, X. Feature selection for high-dimensional imbalanced data. *Neurocomputing* **2013**, *105*, 3–11. [CrossRef]
32. Tao, X.; Li, Q.; Guo, W.; Ren, C.; Li, C.; Liu, R.; Zou, J. Self-adaptive cost weights-based support vector machine cost-sensitive ensemble for imbalanced data classification. *Inf. Sci.* **2019**, *487*, 31–56. [CrossRef]

33. Wang, M.; Lin, Y.; Min, F.; Liu, D. Cost-sensitive active learning through statistical methods. *Inf. Sci.* **2019**, *501*, 460–482. [CrossRef]
34. Ricardo, D.; Teles, J.; Raposo, M.R.; Veloso, A.P.; João, F. Test-Retest Reliability of a 6DoF Marker Set for Gait Analysis in Cerebral Palsy Children. *Appl. Sci.* **2021**, *11*, 6515. [CrossRef]