ACCSCIENCE PUBLISHING

Artificial Intelligence in Health

ORIGINAL RESEARCH ARTICLE

Enhancing patient safety through integrated sensor technology and machine learning for bed-based patient movement detection in inpatient care

Jonathan Mayer[1], Rejath Jose[1], Molly Bekbolatova[1], Chris Coletti[2], Timothy Devine[2], and Milan Toma[1]*

[1]Department of Osteopathic Manipulative Medicine, New York Institute of Technology College of Osteopathic Medicine, Old Westbury, New York, United States of America

[2]Ferrara Center for Patient Safety and Clinical Simulation, New York Institute of Technology College of Osteopathic Medicine, Old Westbury, New York, United States of America

Abstract

The occurrence of inpatient falls and new-onset seizures are common complications during hospital stays, posing risks to patient safety and potentially leading to prolonged hospital stays and further complications. Given the constraints on medical staff's ability to provide constant monitoring due to their workload, the implementation of a sensor device equipped with machine learning capabilities to recognize and prevent these events becomes imperative. This study utilized data acquired through the Movella Xsens sensor, which detects real-time motions and 3D movements, in conjunction with the PyCaret machine-learning algorithm. Adult-sized and infant-sized mannequins were used to assess the algorithm's ability in predicting specific movements associated with breathing, seizures, rolling to the right side, rolling to the left side, rolling off the bed from the left, and rolling off the bed from the right. The study achieved an overall 89% accuracy rate in detecting each specific movement using the combination of PyCaret and Xsens sensors. The application of PyCaret alongside Xsens sensors demonstrates promising results in accurately detecting movements, thereby mitigating falls and post-seizure complications in an inpatient setting, consequently improving patient safety. Further exploration of this technology holds the potential to revolutionize healthcare delivery by incorporating it into a trigger alert system capable of promptly warning medical staff of urgent situations through real-time capture and analysis of potentially harmful motions.

Keywords: Inpatient falls; Sensor device; Machine learning; Patient safety; Movement detection

***Corresponding author:**
Milan Toma
(tomamil@tomamil.com)

Citation: Mayer J, Jose R, Bekbolatova M, Coletti C, Devine T, Toma M. Enhancing patient safety through integrated sensor technology and machine learning for bed-based patient movement detection in inpatient care. *Artif Intell Health.* 2024;1(2): 132-143. doi: 10.36922/aih.2790

Received: January 19, 2024

Accepted: March 27, 2024

Published Online: April 23, 2024

1. Introduction

During inpatient hospital stays, falls serve as prominent starting points for numerous significant afflictions in patients. Annually, the documentation records up to 1 million inpatient hospital falls, with nearly 250,000 causing various injuries and 11,000 even leading to fatalities.[1] Inpatient hospital falls not only concern patients on an individual level but also pose detrimental challenges to hospital administrations and insurance companies, as they not only lead to injuries and increased risk of fatal events but simultaneously extend hospital stays and inflate medical care costs. According to the Center for Disease Control's National Center for Injury Prevention and Control, unintended injuries are responsible for more years of potential life lost than any other cause of death; among the reported 3.4 million unintended injuries, 72,000 are attributed to falls.[2] While inpatient hospital falls are objectively viewed as preventable events thus far, there remains a dearth of effective preventive measures.[3,4] Current methods include providing patients with educational videos on fall prevention, deploying various forms of bed alarms, and employing video monitoring in patient rooms.[5,6] Falls occur for a variety of reasons. Accidental falls occur when patients slip, trip, or encounter other environmental factors. Anticipated physiological falls can be best described as falls experienced by patients predisposed to falling, influenced by factors such as previously recorded falls, an inaccurate self-assessment of capabilities, the presence of intravenous lines or saline locks, or the use of an ambulatory aid.[7,8] Anticipated physiological falls constitute the majority of inpatient falls, while unanticipated physiological falls are less frequent and unpredictable.[8] This study aims to address the longstanding challenge of hospital patient falls by presenting a solution.

Hospital-onset seizures, defined as seizures occurring in hospitalized patients not admitted for seizure-related incidents and lacking a history of seizures, represent jarring occurrences often associated with extended hospital stays and heightened medical care requirements.[9] A previous study investigating hospital-onset seizures identified 218 patients, revealing that 33% experienced generalized tonic-clonic seizures, while metabolic derangements accounted for 25% of the remaining cases.[9] In addition, the study discovered a higher incidence of mortality among patients experiencing hospital-onset seizures compared to those with preexisting histories of seizures, with rates of 19% and 5%, respectively.[9] Thus, hospital-onset seizures typically manifest as new-onset and often recur, coinciding with elevated mortality rates. The results gathered in this study on patient seizures propose a potential novel safety measure for early seizure detection and swift intervention.

PyCaret is a low-code, open-source machine learning library within Python designed to streamline coding efforts while increasing the time available for analysis. Its application has extended to evaluating turnaround time, a critical performance indicator in medical diagnostic laboratories.[10] In addition, PyCaret has demonstrated promise in studies focusing on histological variants of bladder and urothelial carcinomas.[11] Notably, PyCaret has aided in predicting the evolution of mild cognitive impairment to Alzheimer's disease.[12] In recent years, machine learning has seen growing use in the healthcare industry with the objective of enhancing patient results and fostering more effective and tailored care practices.[13,14] In this study, PyCaret was used to classify data surrounding both simulated patient falls and simulated patient seizures, specifically classifying data relating to six particular motions: breathing, seizures, rolling to the right side, rolling to the left side, rolling off the bed from the left, and rolling off the bed from the right. This study serves as an innovative approach to fall prevention that has not been previously implemented in hospitals. Using the Movella Xsens motion sensors to continuously gather continuous data on patient movements and employing machine learning algorithms to classify said data, trigger alert systems for hospital staff may be developed. The goal is to prevent adverse hospital events such as falls and hospital-onset seizures, thereby leading to better patient outcomes and improved patient safety.

This study directly tackles the significant challenge posed by inpatient falls and hospital-onset seizures, occurrences that not only jeopardize patient safety but also impose considerable costs on healthcare systems. Despite existing preventative measures, these events remain a concern. In response, this study introduces a novel approach by utilizing the Movella Xsens sensors alongside the PyCaret machine learning algorithm to predict and potentially prevent such incidents. The sensor device detects real-time motions, while the PyCaret algorithm classifies these movements to recognize patterns associated with risk events. This integrated approach was tested using mannequins, demonstrating an 89% accuracy in movement detection. The findings suggest the potential of this technology to serve as an effective alert system, thereby advancing patient safety by enabling timely interventions by medical staff.

2. Materials and methods

This section provides an overview of the experimental data collection process, operationalization of sensors and movements to obtain relevant data, data preprocessing for machine learning analysis, and utilization of the PyCaret machine learning library to establish, analyze, and evaluate classification models. It emphasizes the metrics used to determine the success and reliability of the models in predicting different types of patient movement, ultimately contributing to the study's goal of improving patient safety through early detection of fall or seizure events.

In conducting this study, a methodology was utilized to replicate the movements associated with inpatient falls and seizures as follows: Adult-sized and infant-sized mannequins were employed to represent a range of patient demographics, ensuring that the collected movement data spanned various relevant physiologies for our algorithm's predictive capabilities. The use of such mannequins allows for consistent and repeatable movement simulations, which are crucial for machine learning applications. The rationale behind selecting the PyCaret machine learning library is twofold. First, PyCaret's low-code environment significantly streamlines the development process, thereby facilitating a more efficient exploration of different predictive models. Second, it offers a comprehensive suite of evaluation metrics and algorithms suitable for both binary and multiclass classification problems, making it particularly well-suited for the complex task of classifying the nuanced movements indicative of potential falls or seizures. This adaptability and ease of use render PyCaret highly suitable for health-care settings, where rapid and accurate decision-making is paramount for patient safety.

This section also provides detailed descriptions of the approaches used for data collection, preprocessing, and the setup of the machine learning model as follows:

(i) Data collection: This segment describes the use of Xsens DOT sensors to gather real-time motion data reflecting 3D orientations in space, which is capable of detecting Euler angles in the X-, Y-, and Z-axes. It discusses the placement of the sensor and the mechanics of its securement to the mannequin's chest. The methodology elucidates the specific movements imitated by the mannequins (e.g., breathing, seizures, rolls, and falls) to collect diverse movement data while distinguishing between the use of adult and infant mannequins for different movements.

(ii) Data preprocessing: This section outlines the process of managing raw datasets, which involves the segregation of collected data into subsets correlating to specific movements of interest. Focus is given to the significance of the Euler angle points in the X-axis and the quantification method for capturing distinct movement data, including simulated rolling and falling off a bed by a mannequin.

(iii) Machine learning model setup: Details are provided on the utilization of PyCaret, a supervised machine-learning module for the study, emphasizing its streamlined workflow and five key steps: setup, compare models, analyze model, save model, and prediction. The process of setting up PyCaret, providing data, labeling the target, and ensuring reproducibility through session IDs is described. Detailed information about the dataset, including data shape before and after transformations and division into training and test sets, is provided.

(iv) Evaluation metrics: PyCaret-generated metrics such as accuracy, area under the curve (AUC), recall, precision, F1 score, and others are introduced. The definitions and significance of these metrics for model evaluation are articulated, including the intricacies of how performance metrics, such as accuracy, precision, recall, and F1 score, are calculated. Other important metrics, such as the receiver operator characteristic curve (ROC), AUC, Cohen's Kappa, Matthews correlation coefficient (MCC), and training time (TT), are discussed, explaining each metric's value range and its implications for the model's predictive performance.

2.1. Data collection

The data were collected using the Xsens DOT sensors capable of capturing Euler angles in the X-, Y-, and Z-axes, also known as roll, pitch, and yaw, respectively, to depict real-time 3D orientation in space. In addition, these sensors have demonstrated efficacy in capturing distinctive data related to various patient movements.[15] As illustrated in Figure 1, one sensor was placed on the mannequin's chest, specifically at the center of the sternum, to evaluate the aforementioned six movements of interest. The sensor was securely affixed to the mannequin using duct tape arranged in a cross-shaped configuration. Subsequently, it was wirelessly connected through Bluetooth through the Movella Dot App. The application allows for continuous streaming and data collection once initiated by the user. The sensors were activated and deactivated for each overall movement of interest, with the collected data immediately transferred to the device connected to the sensor. To collect the data on breathing and seizures, an adult mannequin was used. Conversely, an infant mannequin was used to collect data on the remaining four movements. Data pertaining to breathing involved the mannequin performing one full cycle of tidal volume inhalations and exhalations continuously for 3 min. Seizure data were collected by inducing a seizure in the mannequin for 10 min. For the collection of data on rolling to the side,

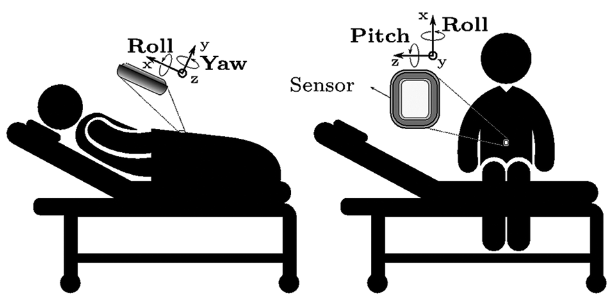

Figure 1. Depiction of the sensor placement and the respective angular motions. Image created using Inkscape

the infant mannequin was initially positioned in the supine position and then rolled approximately 90 degrees to the left side and back to the original supine position, a procedure repeated 100 times. Similarly, the same procedure was performed for rolling to the right side. Data regarding falling off the bed from the left side were collected by initially positioning the infant mannequin in the supine position and subsequently rolling it beyond its left side until it fell off the bed, a procedure repeated 100 times. Given the near-identical nature of dropping the mannequin from its right side, this movement was not performed.

2.2. Data preprocessing

The raw dataset for each respective movement, collected by the sensors, was immediately gathered following the completion of repeated movements. These datasets were divided into six specific movements of interest, each ranging from approximately 18,000 to 35,000 data points containing the Euler angle points in the X-, Y-, and Z-axes. Of particular interest were the Euler angle points along the X-axes exclusively. To isolate the data for each movement, the raw dataset needed to be subdivided to capture data for every 100 movements. Approximately 200 data points were needed to represent one complete movement. For example, the information of one complete roll to the left side consists of 200 points, capturing the transition from the starting supine position to rolling the mannequin to its left side and back to the starting supine position. This collection of 200 points was repeated to ensure a clear delineation of values for each of the 100 movements. This data segmentation process was similarly applied to the other movements of interest. To mimic the mannequin dropping from the right side, the values obtained from dropping the mannequin from the left side were negated. Once the data for each completed movement were collected, it was transferred to one single Excel sheet for further analysis. From the raw datasets, 100 movements were collected for each roll to the right side, roll to the left side, and seizures. Ninety-five movements were collected for each dropping off the bed from the left and right sides, while 89 movements were collected for breathing.

2.3. Pycaret setup

The classification module in PyCaret is a supervised machine learning module designed for classifying elements and aiming to predict categorical class labels that are discrete and unordered. It can handle both binary and multiclass problems, finding applications in various scenarios. The typical workflow in PyCaret for classification consists of five steps: setup, compare models, analyze model, save model, and prediction. The first step, "Setup," initializes the training environment and creates a transformation pipeline. It requires two mandatory parameters, "data" and "target," and offers several optional parameters for customization. The user provides the data in a cohesive fashion with the target labeled appropriately, typically in comma-separated values file (CSV) format. Since this is a classification model, the target is a categorical variable represented numerically (i.e., "Roll right" is 0, "Roll left" is 1, "Drop right" is 2, "Drop left" is 3, "Breathing" is 4, and "Seizure" is 5). The code base for the Google Colaboratory notebook is available at: Sensor_Classification.ipynb.

Once the setup is executed successfully, it displays an information grid with experiment-level details (Table 1). The session ID is a pseudo-random number (123 in this case) used as a seed for reproducibility in all functions throughout the PyCaret pipeline, ensuring consistent results when running the same code with the same session ID. The target refers to the column in the dataset (the CSV file) that will be predicted. In this case, the target is designated "Predict." The target type specifies the nature of the target variable, which in this case is "Multiclass," indicating that the target variable has multiple distinct classes for multiclass classification. The original data shape shows the dimensions of the dataset before any transformations, with 579 rows and 203 columns. Similarly, the transformed data shape also has 579 rows and 203 columns, indicating that the dataset was not modified during the setup process. The transformed training set shape indicates that the training dataset contains 405 rows and 203 columns after preprocessing, which was used to train the machine learning models. The transformed test set shape indicates that the test dataset contains 174 rows and 203 columns after preprocessing, which was used to evaluate the performance of the trained models. Therefore, a split of 70% for training and 30% for testing was used.

Table 1. Experiment setup details

No.	Description	Value
0	Session ID	123
1	Target	Predict
2	Target type	Multiclass
3	Original data shape	(579, 203)
4	Transformed data shape	(579, 203)
5	Transformed train set shape	(405, 203)
6	Transformed test set shape	(174, 203)
7	Numeric features	202
8	Preprocess	True
9	Imputation type	Simple
10	Numeric imputation	Mean
11	Categorical imputation	Mode

Numeric features represent columns with numerical values, encompassing both continuous and discrete data. In this context, the dataset comprises 202 numeric features, with the "Predict" feature serving as a categorical target variable (Table 1). The value "True" for preprocessing indicates that preprocessing steps are applied to the data during the setup process. For the current study, preprocessing steps known as "LabelEncoder" and "SimpleImputer" were applied. The label encoder is a preprocessing step applied to convert categorical target variables (if any) into numerical format. It transforms categorical labels into integer values, making them suitable for training the machine learning model. The simple imputer is also a preprocessing step applied to handle missing values in the dataset. It fills in missing values using simple strategies such as the feature's mean, median, or most frequent value. In cases where numeric features have missing values, the "mean" imputation method is used, replacing the missing numeric values with the mean of the corresponding feature. Conversely, for categorical features with missing values, the "mode" imputation method is used, replacing the missing categorical values with the mode (most frequent category) of the corresponding feature.

The "Compare Models" function trains and evaluates the performance of all available estimators using cross-validation, providing a scoring grid with average cross-validated scores. To analyze the performance of a trained model on the test set, the "plot_model" function can be used. It offers different plot types, such as confusion matrix and AUC, for assessing model performance. In certain cases, re-training the model may be required for plotting specific visualizations. Finally, the model with the entire pipeline is saved on disk for future use, especially for prediction of unseen data.

Hence, the typical workflow in PyCaret for a classification task involves several steps, beginning with the "Setup." During "Setup," the user initiates the training environment by defining the dataset (data) and the variable to be predicted (target). In this case, the target refers to various movements like "Roll right," "Roll left," "Drop right," "Drop left," "Breathing," and "Seizure," encoded numerically from 0 to 5, respectively. The following is how PyCaret handles the classification workflow:

- Session ID: In the setup stage, specifying a session ID as a pseudorandom number (e.g., 123) serves as a seed for all randomness within the pipeline, ensuring that the experiment is reproducible. This setup process implies that the random division of data into folds when applying cross-validation or the random selection of data points if any undersampling or oversampling is performed would yield consistent results each time the code is run with the same session ID.
- Data format and preparation: The input data are provided as a CSV file, which is a standard, easy-to-work-with data format. The data include both the features (e.g., sensor readings) and the target. The features are the inputs that the model will learn from, while the target is the output category that the model is trained to predict.
- Target variable: The target variable is categorical, meaning that it does not have a natural order or numerical value; the assigned numbers are just labels for the classes. As the target is represented numerically, each number corresponds to a discrete category of patient movement, and the model learns to predict these categories.
- Workflow steps: The typical workflow in PyCaret for classification consists of five steps: setup, compare models, analyze model, save model, and prediction.
  - (i) Setup: This crucial first step initializes the analysis environment by setting up the data and defining the target. It also performs basic processing like handling missing values, encoding categorical variables, normalizing the data, and potentially feature engineering.
  - (ii) Compare models: This step systematically trains and evaluates different machine learning models using the preprocessed data, subsequently ranking them according to a chosen evaluation metric, usually accuracy for classification tasks.
  - (iii) Analyze model: For the chosen model, its performance metrics, decision boundary, feature importance, confusion matrix, and other insights are analyzed to understand how well the model works. This step provides information about the classifier's behavior under various conditions through ROC curves, precision-recall curves, and classification errors, allowing the user to deeply interrogate specific models and understand areas for improvement.
  - (iv) Save and predict model: With the model saved, predictions can be made on new data that the model has not seen before. This is the ultimate goal of the machine learning workflow—applying the constructed model to make accurate classifications on real-world data.

The training and test datasets are created during the setup, with PyCaret automatically splitting the input data into these subsets. The typical default splits allocate 70% of the data for training and 30% for testing. The session ID ensures consistency in any randomization during this process across sessions. Therefore, the transformed train set shape and test set shape reported after setup refers to the shapes of these datasets post- and pre-processing and splitting. The workflow encapsulated by the PyCaret setup supports the end-to-end process of building and deploying classification models. In the study context, the models are tasked with classifying patient movements based on sensor data. The workflow enables the use of sophisticated machine learning algorithms without requiring the user to dive deep into the algorithmic complexities associated with each model. Consequently, researchers and practitioners can focus more on interpreting the results and less on managing the workflow mechanics.

2.4. Evaluation metrics

Pycaret provides a range of metrics, including precision, recall, F1 score, accuracy, AUC, Cohen's Kappa, MCC, and TT. Accuracy, as defined in Equation I, represents the proportion of correctly classified values out of the total number. Precision, as per Equation II, is computed as the ratio of true positive instances to all predicted positive instances, where a higher precision score indicates fewer false positive predictions. Recall, defined in Equation III, assesses the ability to identify actual positives and is also known as sensitivity. The F1 score, as per Equation IV, synthesizes precision and recall into a single value between 0 and 1; higher scores indicate better performance in both areas, while lower scores suggest poor precision or recall.

$$\text{Accuracy}=\frac{\text{True Positive}+\text{True Negative}}{\text{True Positive}+\text{False Positive}+\text{True Negative}+\text{False Negative}} \tag{I}$$

$$\text{Precision}=\frac{\text{True Positive}}{\text{True Positive}+\text{False Positive}} \tag{II}$$

$$\text{Recall}=\frac{\text{True Positive}}{\text{True Positive}+\text{False Negative}} \tag{III}$$

$$\text{F1 score}=\frac{2}{1/\text{Precision}+1/\text{Recall}}=\frac{2\times\text{Precision}\times\text{Recall}}{\text{Precision}+\text{Recall}} \tag{IV}$$

The ROC evaluates the difference in the rates of true positive rate and false positive rate results across different decision thresholds.[16] The AUC serves as an indicator of the model's effectiveness, allowing for comparison of performance across various models.[17-19] An AUC equal to one indicates a perfect model, while an AUC exceeding 0.5 indicates that the model's classification capability outperforms random guessing and possesses predictive value. An AUC of 0.5 signifies that the model's classification capacity is equivalent to random guessing, devoid of predictive value. An AUC lower than 0.5 suggests a classification capacity worse than random guessing; however, if a reverse prediction is conducted, it is superior to random guessing. The collection of all sample points forming a line constitutes an ROC curve.[15]

Cohen's Kappa, often known as "Kappa," is a statistical measure used to assess the agreement between predicted and actual classes while also accounting for the level of agreement beyond what would occur by chance. This metric holds particular significance when dealing with imbalanced datasets, as it considers chance-based agreement. Kappa values range from −1 to 1, where 1 signifies perfect agreement, 0 denotes chance-based agreement, and values below 0 indicate predictions worse than random. Meanwhile, MCC serves as another metric for assessing the quality of binary and multiclass classifications; it takes into account true positives, true negatives, false positives, and false negatives, making it useful in scenarios involving imbalanced datasets. Similar to Cohen's Kappa, the MCC also ranges from −1 to 1: a value of 1 indicates flawless prediction capability, while a value of zero represents predictions at random; anything below zero suggests predictive performance worse than random guessing. Finally, TT refers to the duration taken by a specific machine learning model to train on the dataset, typically measured in seconds. This metric offers valuable insight into the time required to train a particular model.

3. Results

A total of 15 machine-learning classification models were tested using Pycaret (Table 2). These models included Light Gradient Boosting Machine (LIGHTLGBM), Extra Tree Classifier (ET), Extreme Gradient Boosting, Random Forest Classifier, Gradient Boosting Classifier, Decision Tree Classifier, K Neighbors Classifier, Naive Bayes, Linear Discriminant Analysis, Logistic Regression, Support Vector Machine - Linear Kernel, Ridge Classifier, AdaBoost Classifier, Quadratic Discriminant Analysis, and Dummy Classifier (DUMMY). DUMMY makes predictions that ignore the input features, serving as a simple baseline for comparison against more complex classifiers.

The LIGHTLGBM model exhibited the highest accuracy, recall, precision, F1 score, Kappa, and MCC. Specifically, the accuracy, recall, precision, F1 score, Kappa, and MCC of the LIGHTLGBM model were 0.89, 0.89, 0.90, 0.89, 0.87, and 0.87, respectively, with an AUC of 0.98. The performance metrics of the LIGHTLGBM model closely resembled those of the ET model, with the ET model displaying slightly lower accuracy, recall, precision, F1, Kappa, and MCC, but marginally higher AUC on the ROC curve (Figure 2). In addition, the confusion matrix

Table 2. Comparisons of accuracy, AUC, recall, precision, F1 score, Kappa, and MCC for different machine learning classifier models

Model	Accuracy	AUC	Recall	Precision	F1	Kappa	MCC	TT (s)
LIGHTLGBM - Light Gradient Boosting Machine	0.8937*	0.9830	0.8937*	0.9017*	0.8926*	0.8723*	0.8744*	3.0400
ET - Extra Trees Classifier	0.8912	0.9856*	0.8912	0.8976	0.8899	0.8693	0.8710	0.2650
XGBOOST - Extreme Gradient Boosting	0.8765	0.9821	0.8765	0.8859	0.8742	0.8517	0.8544	2.0500
RF - Random Forest Classifier	0.8763	0.9853	0.8763	0.8875	0.8737	0.8514	0.8545	0.7430
GBC - Gradient Boosting Classifier	0.8738	0.9799	0.8738	0.8882	0.8717	0.8485	0.8522	13.181
DT - Decision Tree Classifier	0.8273	0.8969	0.8273	0.8493	0.8271	0.7927	0.7974	0.0750
KNN - K Neighbors Classifier	0.7946	0.9456	0.7946	0.8236	0.7943	0.7534	0.7602	0.0570
NB - Naive Bayes	0.7773	0.9589	0.7773	0.7919	0.7739	0.7331	0.7373	0.0480
LDA - Linear Discriminant Analysis	0.7701	0.9248	0.7701	0.7890	0.7690	0.7239	0.7282	0.1420
LR - Logistic Regression	0.7182	0.8895	0.7182	0.7932	0.7175	0.6613	0.6735	1.4500
SVM - SVM – Linear Kernel	0.6270	0.0000	0.6270	0.6821	0.5873	0.5511	0.5844	0.1190
RIDGE - Ridge Classifier	0.5676	0.0000	0.5676	0.6386	0.5372	0.4803	0.5079	0.0910
ADA - Ada Boost Classifier	0.5136	0.7910	0.5136	0.4010	0.4147	0.4120	0.5165	0.5200
QDA - Quadratic Discriminant Analysis	0.3955	0.6350	0.3955	0.5045	0.3514	0.2706	0.3064	0.1170
DUMMY - Dummy Classifier	0.1729	0.5000	0.1729	0.0299	0.0510	0.0000	0.0000	0.0420

Note: *Highest value.
Abbreviations: AUC: Area under the curve; MCC: Matthews correlation coefficient; TT: Training time.

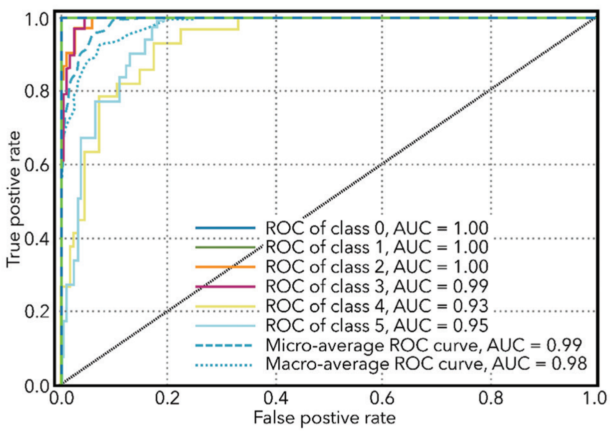

Figure 2. Area under the curves for Light Gradient Boosting Machine (LIGHTLGBM) classifier. Image created with Inkscape
Abbreviations: AUC: Area under the curves; ROC: Receiver operator characteristic curve.

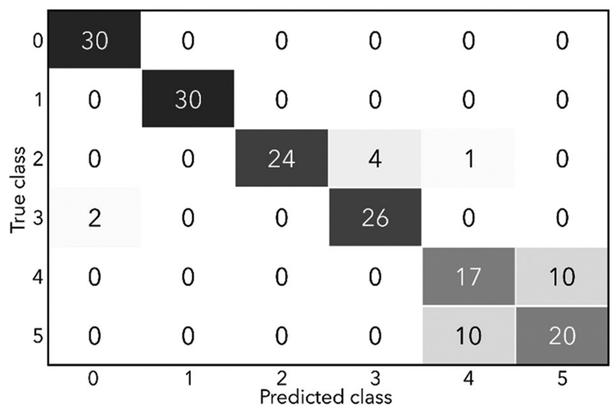

Figure 3. Confusion matrix for the Light Gradient Boosting Machine (LIGHTLGBM) classifier. Image created with Inkscape

generated by Pycaret depicted predictions in the testing split for all categories (Figure 3). A confusion matrix serves as a tool to visualize the performance of a classification model. The diagonal elements of the matrix denote the number of correct predictions for each class, while the off-diagonal elements indicate the number of incorrect predictions, where the model predicts a different class from the actual label. In this study, the confusion matrix is a 6×6 matrix, reflecting the six classes encoded from 0 to 5, for the LIGHTLGBM classifier used.

The numbers on the diagonal can be interpreted as follows: 30 correct predictions for class 0 ("Roll right"); 30 correct predictions for class 1 ("Roll left"); 24 correct predictions for class 2 ("Drop right"); 26 correct predictions for class 3 ("Drop left"); 17 correct predictions for class 4 ("Breathing"); and 20 correct predictions for class 5 ("Seizure"). These numbers indicate that the LIGHTLGBM classifier exhibits the best performance at detecting "Roll right" and "Roll left" movements, as these classes boast the highest number of correct predictions (30 each). The non-zero off-diagonal elements that are 10 or lower represent instances of misclassification of a movement by the model.

For example, a value of 10 in the off-diagonal position indicates that 10 instances of "Breathing" were incorrectly predicted as "Seizure" by the classifier. The implications of these findings from the confusion matrix could be twofold: Classes 0 and 1 are well-recognized by the model, possibly because these movements possess distinct characteristics that are easily discernible by the sensors. Conversely, classes 4 and 5 have the lowest instances of correct prediction (17 and 20, respectively), suggesting that recognizing "Breathing" and "Seizure" movements presents greater difficulty for the model. This challenge could stem from similarities in the patterns of these movements or subtle characteristics that were not captured by the sensors or model features. Misclassifications (off-diagonal numbers) highlight areas where the model confuses one movement for another. Although all off-diagonal numbers are 10 or lower, indicating a decent level of precision, these errors could carry significant consequences depending on the clinical importance of the movements. For instance, mistaking breathing for a seizure (or vice versa) could lead to inappropriate medical interventions or a failure to respond promptly to an actual seizure.

Overall, despite the misclassifications, the numbers suggest a higher number of correct predictions across all classes than incorrect ones, indicating the promising potential of the classifier. However, it is essential to identify the clinical consequences of each type of misclassification to prioritize improvements in the classifier's performance.

4. Discussion

This study, conducted with data collected using Xsens DOT sensors and analyzed utilizing PyCaret, showcases the potential benefits of using artificial intelligence methods in practical, real-life applications. The results indicate an accuracy rate of over 89% in predicting the movements of inpatients based on the provided data. The failure to achieve a perfect accuracy rate from the LIGHTLGBM classifier is attributed to its difficulty in distinguishing between breathing and seizures, with prediction values for these movements approximately 63% and 67%, respectively. This challenge likely stems from the absence of chest movement during seizures, as opposed to the common occurrence of head movements.[20] With that being said, one potential solution to this problem is to place an additional sensor on the mannequin's head to detect seizure movements. The data gathered from this sensor on seizures can be used as the evaluation metrics, while the sensor placed on the chest will solely gather data on the other movements of interest.

While promising, the results of this study do highlight several additional limitations, beyond the one previously mentioned, all of which fortunately are addressable with appropriate tools. One such limitation concerns the utilization of duct tape to secure the monitor to the mannequin's chest. Since duct tape is impractical for live patients, an alternative solution, such as a strap-like device, could be introduced to ensure the monitor remains securely in place on the patient's chest without causing discomfort. Another limitation pertains to the battery life of the sensor device. Throughout the study, it was observed that the sensor's battery depleted to 50% capacity after approximately 3 h of use. This limitation could be resolved by having the patient's nursing team replace the sensor at the 6-h mark. The replaced sensor can then be recharged and reused for subsequent monitoring sessions.

Inpatient falls pose a potential risk of prolonged hospital stays, further injuries, and various complications for many hospitalized patients in recovery. Hospital-onset seizures also present a concern, as they are difficult to predict and can significantly worsen a patient's overall well-being. Early recognition of these events, preferably before their occurrence, is crucial for ensuring patient safety.[21,22] While one-to-one observation is an effective way to monitor high-risk patients, it is not feasible for all admitted patients.[23-26] In addition, while video monitoring appears to be a viable alternative solution, it can raise privacy concerns and incur substantial startup costs and resource investment to install cameras in multiple rooms.[27] The utilization of Euler angle measurements through the Movella Xsens sensors, in conjunction with machine learning algorithms, offers a potential solution to these problems by enabling the detection and classification of specific movements in real time. While there are limitations to using these sensors in practical settings that must still be addressed, this study presents promising results and lays a foundation for further research in this area. By accurately predicting ongoing motion, this novel approach can be incorporated into a trigger alert system for the medical staff, allowing for swift intervention before a fall or seizure occurs. This proactive approach will significantly reduce the risk of adverse events, lower complication rates, and ultimately improve overall patient outcomes.[28-32]

This study presents several limitations. The use of a controlled environment with a limited number of mannequins to replicate patient movements does not fully capture the variability observed in actual patient populations. To better simulate real-world conditions, expanding the study's scope to include a larger and more diverse group of patients across various health-care settings would provide a more comprehensive understanding of sensor capabilities and algorithm performance. In addition, sensors may have limitations in accurately detecting subtle or intricate patient movements, leading to potential false positives or negatives. Therefore, implementing a multimodal sensor approach or utilizing more advanced sensor technologies could enhance the accuracy and reliability of movement detection. Moreover, while machine learning models may perform well under experimental conditions, they might be less effective in real-world applications due to issues such as overfitting or difficulties in interpreting complex data. Thus, employing sophisticated machine learning techniques such as deep learning—which can capture complex patterns—and methods to ensure model robustness against overfitting—such as dropout or data augmentation—is necessary.

In addition, transitioning a model from research settings to widespread clinical use could present challenges due to infrastructure or resource limitations. Developing an adaptable solution necessitates close collaboration with healthcare technology providers while ensuring compatibility across various facility infrastructures. Healthcare providers may hesitate to embrace new technology due to integration challenges with existing workflows or feeling overwhelmed by constant alerts. It is crucial to create technology that seamlessly integrates into current hospital systems and processes. Alert systems need to prioritize important events to minimize unnecessary interruptions and prevent staff fatigue from excessive alerts. Regular maintenance procedures and backup systems must be established to guarantee continuous functionality. Using durable and low-maintenance hardware can also reduce the occurrence of malfunctions. Implementing advanced sensor systems and machine learning models may require an additional investment from healthcare institutions. Therefore, conducting cost-benefit analyses is essential to illustrate the long-term savings associated with reducing patient falls and seizures, such as shorter hospital stays and fewer medical interventions, thereby making a compelling case for investing in technology integration.

The sensor technology collects data on patients' movements, which is then analyzed by a machine learning algorithm to detect falls or other significant events. This system can operate on a closed network within the hospital, without necessarily requiring an external real-time data network connection for its day-to-day functioning. However, it is important to clarify that for the system to be effective in real-world applications; it should enable real-time processing of the sensor data so that immediate alerts can be sent to the medical staff in case of a detected fall or seizure, regardless of whether it is connected to an external network or operates on an internal network. The goal is to ensure timely interventions and improved patient safety, which can be achieved through on-site data processing and alert mechanisms. The actual network requirements would depend on the specifics of the system implementation and operational logistics, such as sensor placement and management of data privacy and security.

In addition, introducing the sensors and machine learning system into a real-world health-care setting would raise considerations regarding data sharing and protection. Implementation would need to comply with data protection regulations, such as the Health Insurance Portability and Accountability Act in the United States or the General Data Protection Regulation in the European Union, which govern the privacy and security of patient data. The hospital's information technology infrastructure would need to ensure adequate measures are in place for data encryption, secure access protocols, and potential anonymization of patient data to prevent unauthorized use or disclosure. Furthermore, the legal aspects of data handling would require careful planning to ensure patient consent is obtained where necessary, and there is transparency in how patient data is used and protected. Failure to adequately address these concerns could potentially jeopardize patient privacy and expose the health-care facility to legal and regulatory risks.

5. Conclusion

The study underscores the potential of combining advanced sensor technology with sophisticated machine learning algorithms to detect and prevent events such as falls and seizures in a hospital setting. Opportunities to enhance the usability and effectiveness of these technologies are evident, particularly in optimizing sensor placement and improving operational logistics, such as battery life management. By addressing these challenges, the approach tested in this study could pave the way for creating robust, real-time monitoring systems that not only alert care providers to potential falls or seizures but also contribute to a broader range of applications in patient care and monitoring. Further exploration and refinement could lead to the development of a more comprehensive solution that mitigates the risks associated with patient falls and seizures, ultimately improving patient outcomes and reducing healthcare costs.

The practical implications of this study, which enhances patient safety with integrated sensor technology and machine learning for bed-based patient movement detection in inpatient care, can significantly affect various aspects of health-care delivery as follows:

(i) Improved patient safety: Accurately detecting movements indicative of potential falls or seizures allows for proactive staff alerts and timely intervention, reducing the incidence and severity of such events and leading to improved patient outcomes.

(ii) Reduced health-care costs: Preventing falls and seizures decreases the average duration of hospital stays and the number of medical interventions, resulting in substantial cost savings for health-care facilities.
(iii) Enhanced monitoring: Continuous and automated monitoring supplements the efforts of health-care staff, allowing them to focus on other critical tasks, knowing that the system will alert them to potential issues with patients.
(iv) Data-driven insights: The data collected by the sensors can provide insights into the most common times or conditions under which falls and seizures occur, facilitating the development of refined care protocols and targeted preventative measures.
(v) Staff efficiency: With a system in place to handle routine monitoring tasks, the staff can allocate their time more efficiently, thereby improving overall productivity.
(vi) Training and education: Data from the sensor technology can serve educational purposes, teaching health-care professionals about patient safety and fall prevention strategies by providing real examples and insights.
(vii) Patient and family peace of mind: Knowing that a sophisticated system monitors their loved ones can provide patients and their families with greater confidence in the care provided by the hospital, potentially improving their overall experience and satisfaction.
(viii) Quality of care metrics: Hospitals can use data from these technologies to demonstrate adherence to patient safety protocols, potentially improving their quality-of-care metrics and accreditation outcomes.
(ix) Improved resource allocation: Predictive analytics enable hospitals to anticipate patient needs and allocate nursing and medical resources more effectively, ensuring that high-risk patients receive more frequent human monitoring compared to low-risk patients.
(x) Legal and compliance benefits: Implementing state-of-the-art technology for patient safety may help health-care facilities comply with legal standards and regulations, potentially reducing the risk of lawsuits associated with inpatient falls and other incidents.
(xi) Scientific advancement: The technology provides a rich data source for further research into patient safety, potentially identifying new risk factors for falls and seizures that were previously unknown.
(xii) Enhanced rehabilitation: For patients recovering from surgery or injury, the technology can monitor rehabilitation progress and ensure that movements are within safe parameters, thereby supporting a more effective recovery process.
(xiii) Broad healthcare applications: Although initially focused on inpatient care, such technology could be extended to other settings such as nursing homes, rehabilitation centers, and even home healthcare, expanding its applications on patient safety beyond the hospital.

Implementing such technology requires a thoughtful approach to address potential challenges such as user acceptance, data security, and ethical considerations. However, the benefits mentioned clearly indicate a profound positive impact on patient care, safety, and hospital operations. In conclusion, the integration of advanced sensor technology and machine-learning algorithms in health-care settings holds immense promise for improving patient safety, thus warranting further research in this technology.

Acknowledgments

None.

Funding

None.

Conflict of interest

The authors declare that they have no competing interests.

Author contributions

Conceptualization: All authors
Investigation: All authors
Methodology: All authors
Writing – original draft: All authors
Writing – review & editing: All authors

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Availability of data

The data generated by the sensors that support the findings of this study are available from the corresponding author on reasonable request.

References

1. LeLaurin JH, Shorr RI. Preventing falls in hospitalized patients: State of the science. *Clin Geriatr Med*. 2019;35(2):273-283.

   doi: 10.1016/j.cger.2019.01.007

2. Centers for Disease Control and Prevention. *Web-based*

Injury Statistics Query and Reporting System (WISQARS). National Center for Injury Prevention and Control, Centers for Disease Control and Prevention (Producer); 2023. Available from: https://www.cdc.gov/ncipc/wisqars [Last accessed on 2023 Dec 22].

3. Eost-Telling C, Yang Y, Norman G, *et al*. Digital technologies to prevent falls in people living with dementia or mild cognitive impairment: A rapid systematic overview of systematic reviews. *Age Ageing*. 2024;53:afad238.

   doi: 10.1093/ageing/afad238

4. Wang Y, Jiang M, He M, Du M. Design and implementation of an inpatient fall risk management information system. *JMIR Med Inform*. 2024;12:e46501.

   doi: 10.2196/46501

5. Dykes PC, Burns Z, Adelman J, *et al*. Evaluation of a patient-centered fall-prevention tool kit to reduce falls and injuries: A nonrandomized controlled trial. *JAMA Netw Open*. 2020;3(11):e2025889.

   doi: 10.1001/jamanetworkopen.2020.25889

6. Zubrinic M, Vrbanic L, Keshavjee S. Remote telemonitoring is associated with improved patient safety and decreased workload of nurses. *JTCVS Open*. 2023;16:493-497.

   doi: 10.1016/j.xjon.2023.09.014

7. Morse JM. *Preventing Patient Falls*. Berlin: Springer Publishing Company; 2008.

8. Mcvey L, Alvarado N, Zaman H, *et al*. Interactions that support older inpatients with cognitive impairments to engage with falls prevention in hospitals: An ethnographic study. J Clin Nurs. 2024;33:1884-1895.

   doi: 10.1111/jocn.17006

9. Fields MC, Labovitz DL, French JA. Hospital-onset seizures: An inpatient study. *JAMA Neurol*. 2013;70(3):360-364.

   doi: 10.1001/2013.jamaneurol.337

10. Tsai ER, Demirtas D, Hoogendijk N, Tintu AN, Boucherie RJ. Turnaround time prediction for clinical chemistry samples using machine learning. *Clin Chem Lab Med*. 2022;60(12):1902-1910.

    doi: 10.1515/cclm-2022-0668

11. Evrimler S, Ali Gedik M, Ahmet Serel T, Ertunc O, Alperen Ozturk S, Soyupek S. Bladder urothelial carcinoma: Machine learning-based computed tomography radiomics for prediction of histological variant. *Acad Radiol*. 2022;29(11):1682-1689.

    doi: 10.1016/j.acra.2022.02.007

12. Rye I, Vik A, Kocinski M, Lundervold AS, Lundervold AJ. Predicting conversion to Alzheimer's disease in individuals with Mild Cognitive Impairment using clinically transferable features. *Sci Rep*. 2022;12(1):15566.

    doi: 10.1038/s41598-022-18805-5.

13. Bekbolatova M, Mayer J, Ong CW, Toma M. Transformative potential of AI in healthcare: Definitions, applications, and navigating the ethical landscape and public perspectives. *Healthcare*. 2024;12(2):125.

    doi: 10.3390/healthcare12020125

14. Toma M, Wei OC. Predictive modeling in medicine. *Encyclopedia*. 2023;3(2):590-601.

    doi: 10.3390/encyclopedia3020042

15. Mayer J, Jose R, Kurgansky G, *et al*. Evaluating the feasibility of euler angles for bed-based patient movement monitoring. *Signals*. 2023;4(4):788-799.

    doi: 10.3390/signals4040043

16. Chou CY, Hsu DY, Chou CH. Predicting the onset of diabetes with machine learning methods. *J Pers Med*. 2023;13(3):406.

    doi: 10.3390/jpm13030406

17. Abraham A, Jose R, Ahmad J, *et al*. Comparative analysis of machine learning models for image detection of colonic polyps vs. Resected polyps. *J Imaging*. 2023;9(10):215.

    doi: 10.3390/jimaging9100215

18. Jose R, Syed F, Thomas A, Toma M. Cardiovascular health management in diabetic patients with machine-learning-driven predictions and interventions. *Appl Sci*. 2024;14(5):2132.

    doi: 10.3390/app14052132

19. Jose R, Thomas A, Guo J, Steinberg R, Toma M. Evaluating machine learning models for prediction of coronary artery disease. *Glob Transl Med*. 2024;3(1):2669.

    doi: 10.36922/gtm.2669

20. Freitas ME, Ruiz-Lopez M, Dalmau J, *et al*. Seizures and movement disorders: Phenomenology, diagnostic challenges and therapeutic approaches. *J Neurol Neurosurg Psychiatry*. 2019;90(8):920-928.

    doi: 10.1136/jnnp-2018-320039

21. Song W, Latham NK, Liu L, *et al*. Improved accuracy and efficiency of primary care fall risk screening of older adults using a machine learning approach. *J Am Geriatr Soc*. 2024;72:1145-1154.

    doi: 10.1111/jgs.18776

22. Durán-Vega LA, Santana-Mancilla PC, Buenrostro-Mariscal R, *et al*. An IoT system for remote health monitoring in elderly adults through a wearable device and mobile application. *Geriatrics* (*Basel*). 2019;4(2):34.

    doi: 10.3390/geriatrics4020034

23. Oliver D, Healey F, Haines TP. Preventing falls and fall-related injuries in hospitals. *Clin Geriatr Med*. 2010;26(4):645-692.

    doi: 10.1016/j.cger.2010.06.005

24. Majidi SA, Fakoorfard Z, Safarmohammadi H, Kazemnezhad Leily E. The relationship between moral intelligence and patient safety culture in nurses. *J Caring Sci*. 2023;12: 241-247.

    doi: 10.34172/jcs.2023.30501

25. Hiyama A. Using the analytic hierarchy process to measure nurses' decision-making regarding fall risks and care strategies for fall prevention. *J Nurs Meas*. 2024;31: 1113-1125.

    doi: 10.1891/JNM-2023-0006

26. Orts-Cortés MI, Cabañero-Martínez MJ, Meseguer-Liza C, Arredondo-González CP, de la Cuesta-Benjumea C, Abad-Corpa E. Effectiveness of nursing interventions in the prevention of falls in older adults in the community and in health care settings: A systematic review and meta-analysis of RCT. *Enferm Clín* (*Engl Ed*). 2024;34:4-13.

    doi: 10.1016/j.enfcle.2024.01.001

27. Abbe JR, O'Keeffe C. Continuous video monitoring: Implementation strategies for safe patient care and identified best practices. *J Nurs Care Qual*. 2021;36(2):137-142.

    doi: 10.1097/NCQ.0000000000000502

28. Camp K, Murphy S, Pate B. Integrating fall prevention strategies into EMS services to reduce falls and associated healthcare costs for older adults. *Clin Interv Aging*. 2024;19:561-569.

    doi: 10.2147/CIA.S453961

29. Wabe N, Meulenbroeks I, Huang G, *et al*. Development and internal validation of a dynamic fall risk prediction and monitoring tool in aged care using routinely collected electronic health data: A landmarking approach. *J Am Med Inform Assoc*. 2024:ocae058.

    doi: 10.1093/jamia/ocae058

30. Coleman A. Reducing falls among residents of retirement homes: A DNP project. *Nurse Pract*. 2024;49(4):39-47.

    doi: 10.1097/01.NPR.0000000000000161

31. Lee-Confer J. Strength in arms: Empowering older adults against the risk of slipping and falling-a theoretical perspective. *Front Sports Act Living*. 2024;6:1371730.

    doi: 10.3389/fspor.2024.1371730

32. Xu D, Wang Y, Zhu S, Zhao M, Wang K. Relationship between fear of falling and quality of life in nursing home residents: The role of activity restriction. *Geriatr Nurs*. 2024;57:45-50.

    doi: 10.1016/j.gerinurse.2024.03.006