Is the Unsupervised Method "Isolation Forest" for Fall Detection of Comparable Accuracy to Supervised Methods?

Nirav Kakaiya, Gokul Rakesh, Joel Atchohe
University of Nottingham

***Abstract*—Falls are a leading cause of injury, particularly among elderly populations, making reliable fall detection systems a key area of research. Traditional approaches rely heavily on supervised machine learning models trained on labelled data, which can be time-consuming and expensive to obtain. This study investigates whether an unsupervised method, Isolation Forest, can achieve comparable performance to supervised models such as Random Forest, Support Vector Machine (SVM), and Multi Layer Perceptron (MLP) classifiers, using time-series features extracted from wearable sensor data. We evaluate models on a publicly available dataset from Figshare, applying statistical feature engineering and class balancing with SMOTE. To benchmark unsupervised performance, we also explore KMeans clustering and assess alignment with true labels using the Adjusted Rand Index. Our results show that the unsupervised Isolation Forest algorithm does not achieve performance comparable to supervised models for fall detection. Supervised classifiers, including Random Forest and SVM, demonstrated consistently high balanced accuracy, while Isolation Forest showed only limited discriminatory power, as reflected by a relatively low ROC AUC score. In light of these results, we further explore the potential of combining unsupervised methods such as Principal Component Analysis with K-Means clustering to support data labelling and reduce dependence on annotated datasets.**

I. Introduction

This project makes use of Sensor-Based Fall Detection Dataset, recorded using a device attached to the participants [1]. With 8,953 Activities from 29 Subjects, it includes 2,791 falls and 6,162 Activities of Daily Living (ADL). Each CSV file contains the following columns:

AccX, AccY, AccZ, Magnitude, GyroX, GyroY, GyroZ, Temperature, and *Altitude*.

AccX, AccY and AccZ measure the acceleration along the X, Y, and Z axes, while Magnitude measures the size of the overall acceleration vector. All of this is measured using an accelerometer that is part of the device. GyroX, GyroY and GyroZ are the angular velocity measurements, read by the gyroscope within the device, in each respective axis. The dataset is structured in folders that categorise the activity, and each file name represents the date and time of the start of the file, in a `YYYYMMDD_HHMMSS` format. Each file holds around 800 records of the above types, with the data recorded at around 100 Hz, meaning each comma-separated value (CSV) file holds 8 seconds or sensor data.

Unsupervised methods like Isolation Forest offer the potential to detect falls without requiring extensive labelled datasets, which could make fall detection systems more scalable and accessible. However, there is limited research directly comparing the accuracy of unsupervised approaches against traditional supervised methods. By investigating whether Isolation Forest can achieve comparable performance against a set of supervised models, using a variety of testing methods, this study aims to explore the viability of more flexible, less resource-intensive approaches to fall detection.

We believe the chosen research question is primarily exploratory with some inferential aspects as we explore a possibility that is not well established in the current literature, and once all testing is complete we may make inferences on which method performs better based on the data we have found and used. The question adheres to all the Specific, Plausible, Answerable, Interesting, Novel (SPAIN) characteristics as follows:

- Specific: It clearly focuses on comparing the accuracy of the Isolation Forest method with a defined set of supervised models for fall detection.
- Plausible: There is clear evidence of supervised machine learning models performing and accurately identifying falls therefore investigating the potential of the unsupervised "Isolation Forest" model is not irrelevant.
- Answerable: The question is feasible to investigate using available fall detection datasets and established machine learning frameworks as well as produce quantifiable results by evaluating model performance metrics such as accuracy, precision, recall, and F1-score.
- Interesting: The investigation addresses a practical challenge in fall detection reducing reliance on costly labelled data, making it relevant to both academic research and real-world healthcare applications.
- Novel: While supervised methods have been widely studied, there is limited direct, academic evaluation of unsupervised approaches like Isolation Forest in this domain, offering the opportunity to contribute new insights to the field.

II. Literature Review

Falls are a major health concern, particularly for elderly populations, making accurate fall detection systems critically important. One of the most serious health risks posed to older people is falls. According the World Health Organisation (WHO), falls are the second most common cause of unintentional deaths and serious injuries [2]. While most existing fall detection methods rely on supervised machine learning techniques trained on labelled data, collecting and labelling sufficient high-quality data can be time-consuming, costly, and impractical in real-world settings[3].

There is one existing paper that we could find, proposing a fall detection system using isolation forest [4]. However, this is for a wheelchair fall detection system, meaning a sensor-based fall detection system using isolation forest for fully mobile older people is relatively novel.

Additionally, use of the isolation forest model can greatly reduce the time complexity [5], so its use in fall detection systems is appropriate. It can also significantly improve the outlier detection rate, meaning it may rival or even overtake some supervised methods in accuracy.

Supervised, wearable device-based methods for fall detection have been extensively studied due to their practicality and direct monitoring capabilities. A relatively recent review [6] discussed the viability and potential drawbacks of various exisitng methods.

For example, Pierleoni et al. [7] proposed a high-reliability wearable device that combines a magnetometer, gyroscope, and accelerometer to enhance the accuracy of fall detection compared to single-sensor systems. Although their system demonstrated promising results, it lacked comprehensive data presentation and a thorough discussion of existing challenges.

Similarly, Baga et al. [8] developed a modular system architecture aimed at managing neurodegenerative diseases using minimized wearable sensors. Their work emphasized security concerns and provided solutions to overcome them, although it did not extensively discuss methods for accurate fall detection or quantification. These examples highlight the diversity of approaches in wearable-based fall detection and point toward the ongoing need for improving detection reliability, minimizing false alarms, and addressing usability issues for elderly users.

III. Methodology

This section outlines the key methods employed in developing our data pipeline, designed to enable a fair and consistent comparison between the Isolation Forest model and a selected group of supervised learning models. The pipeline incorporates systematic hyper-parameter tuning for all models, with results documented for different parameter configurations. Our approach emphasizes modularity, ensuring that components of the pipeline can be easily modified and updated following iterative cycles of data collection and evaluation. This design facilitates efficient experimentation and continuous improvement throughout the study.

To mitigate common biases that often affect data collection, several measures were taken in the selection of our dataset. First, to reduce publication bias, we chose a dataset that has not yet been used in any published studies, as indicated by its low engagement metrics on Figshare with only 245 views, 45 downloads, and zero citations at the time of writing. Figshare is a well-known platform for sharing and citing research outputs, so these numbers suggest minimal prior exposure. To address historical bias, we note that the dataset was uploaded on January 27th, 2025, making it a relatively recent release. Sampling bias is also reduced, as the dataset is both large in size and collected from 29 diverse participants, suggesting a more representative sample of real-world conditions. Finally, experimenter bias and selection bias are unlikely to affect our results, since the dataset does not make any assumptions about model performance and does not favour either supervised or unsupervised approaches ensuring our evaluation remains neutral with respect to our research question.

$$|\mathbf{x}| = \sqrt{x_1^2 + x_2^2 + x_3^2}$$

Fig. 1. L2 norm formula

A. Data Wrangling

The dataset was substantial, approximately 1.06 GB in size, with falls and main ADL folders (walking and standing up) containing 2–3 thousand CSV files each. Each CSV file represented an 8-second recording containing 800 records, implying a sample rate of 100 Hz. Initially, we considered splitting the data into 1-second windows in order to generate a larger number of samples and capture shorter, localized movement patterns more effectively. This would enable more granular analysis of falls versus non-falls and potentially improve the performance of both supervised and unsupervised models.

However, when splitting continuous 8-second recordings into multiple 1-second windows, the resulting segments are not truly independent, as they originate from the same fall or non-fall event and exhibit temporal correlation. Since many supervised machine learning models assume that each training sample is independent and identically distributed, violating this assumption may introduce data leakage and artificially inflate performance metrics.

The triaxial acceleration and gyroscope values are ratio-scale measurements, enabling statistical features such as mean, standard deviation, minimum, maximum, range, and signal magnitude area (SMA) to be calculated as summary statistics to generate new features for each of the existing features. The remaining features include the magnitude (L2 norm, see figure 1) of the acceleration values, detected temperature at the time of recording and the detected altitude. For these features, the SMA is not calculated. This results in 48 features being generated for the refined dataset, with one value per feature calculated for each CSV file. While this approach reduces the granularity of the data, it is a deliberate choice to manage the large volume of available data and to avoid data leakage. Each record in the refined dataset corresponds to a single CSV file, which contains 8 seconds of sensor data, as previously mentioned. By processing data at the file level, we also avoid the need to handle duplicate or missing values within individual columns. Additionally, the use of summary statistics (e.g., mean, standard deviation, range) helps to smooth out local fluctuations and irregularities, reducing the impact of outliers and eliminating the need for explicit outlier removal when generating the new features. However, CSV files that are entirely empty are excluded from the dataset for simplicity.

B. Data Pre-processing

Prior to model implementation, several preprocessing steps were necessary to ensure data quality. Initial exploratory analysis identified a problematic entry containing an empty CSV file (269 KB), which was subsequently removed. Outlier detection was performed using z-score thresholds to identify extreme values that could potentially compromise model performance.

Outlier visualisation via heatmaps revealed distinct patterns between fall and non-fall classes. Approximately 16.5% of records contained outliers with z-scores exceeding 3, with falls exhibiting more pronounced outlier patterns in acceleration and gyroscope measurements. After careful consideration, a z-score threshold of 6 was selected to eliminate only extreme outliers while preserving the natural variability present in fall events, as approximately 1.8% of records contained outliers when using this threshold which we deemed acceptable for the given practical purposes.

Feature correlation analysis employed Pearson correlation to identify redundant features. The correlation heatmap revealed several feature pairs with high correlation coefficients (> 0.9), including AccX_range vs AccX_max (0.913) and Magnitude_range vs Magnitude_max (0.996). To reduce dimensionality while preserving information content, 16 features with high correlation but lower relevance to the target variable were removed, reducing the feature set from 48 to 32 dimensions.

For model training and evaluation, an 80/20 stratified train-test split was implemented to ensure proportional class representation in both sets. To address the class imbalance in the training set, Synthetic Minority Over-sampling Technique (SMOTE) was applied to generate synthetic fall instances based on existing samples. This balanced the training data to equal representation (4,889 instances per class) while maintaining the original distribution in the test set to reflect real-world conditions.

C. Model implementation

1) Supervised learning approaches: Three supervised classification algorithms were implemented and evaluated to establish baseline performance for fall detection:

We constructed Random Forest classifiers with varying configurations to determine optimal model complexity. The tested parameter sets included: default settings with 100 estimators; increased estimators (200) with max_depth=10; and further increased estimators (300) with reduced max_depth=5.

Support Vector Machines with RBF kernels were implemented using different hyper-parameter combinations to explore the trade-off between margin width and classification error. Configurations included: C=1.0 with gamma='scale'; C=10.0 with gamma=0.01; and C=0.1 with gamma=1.

Multi-layer Perceptron neural networks with varying architectures were developed to capture complex non-linear patterns in the data. Tested configurations included: single hidden layer (50 neurons) with max_iter=300; two hidden layers (100, 50) with alpha=0.001 and max_iter=500; and two equal hidden layers (100, 100) with alpha=1e-5 and max_iter=1000.

2) Unsupervised learning approach: To address our primary research question, Isolation Forest was implemented as an unsupervised anomaly detection technique. This algorithm identifies anomalies (falls) by isolating observations rather than profiling normal points, potentially making it suitable for imbalanced datasets where falls represent the minority class.

The contamination parameter, indicating the expected proportion of anomalies, was varied (0.1, 0.15, 0.25, 0.35, 0.45, 0.5) to analyse its impact on detection performance. This parameter effectively controls the decision threshold between normal activities and anomalous events.

3) Evaluation Methodology: Given the class imbalance, balanced accuracy was selected as the primary evaluation metric for supervised models. This metric equally weights performance on both classes, preventing bias toward the majority class (ADLs):

Balanced Accuracy = (Sensitivity + Specificity) / 2.

For the unsupervised Isolation Forest, ROC AUC score was utilised to evaluate its ranking capability independent of threshold selection. This metric assesses how well the model's anomaly scores distinguish between falls and ADLs across all possible thresholds.

All evaluations employed 5-fold stratified cross-validation to ensure robust performance estimates. This approach maintains class proportions in each fold and provides more reliable assessments than single train-test splits. For each model configuration, both best and average performance across folds are reported.

IV. RESULTS

The supervised models demonstrated strong classification performance, with Random Forest achieving the highest balanced accuracy (BA). Table 1 presents the detailed results across all model configurations.

Contamination rate	Confusion matrix	ROC AUC score
0.10	[[1158, 64], [443, 98]]	0.5643
0.15	[[1119, 103], [401, 140]]	0.5872
0.25	[[1037, 185], [317, 224]]	0.6313
0.35	[[936, 286], [255, 286]]	0.6473
0.45	[[822, 396],[180, 354]]	0.6689
0.50	[[784, 434],[160, 374]]	0.6720

TABLE II

TABLE 2: UNSUPERVISED MODEL PERFORMANCE

Model	Configuration	Best BA	Average BA
Random forest	Default	0.9827	0.9807
	max_depth=10, n_estimators=200	0.9809	0.9765
	max_depth=5, n_estimators=300	0.9435	0.9381
SVM (RBF Kernel)	C=1.0, gamma = 'scale'	0.9754	0.9719
	C=10.0, gamma = 0.01	0.9775	0.9737
	C=0.1, gamma=1	0.5000	0.5000
MLP Classifier	hidden_layers=(50), max_iter=300	0.9710	0.9641
	hidden_layers=(100,50), alpha=0.001, max_iter=500	0.9757	0.9650
	hidden_layers=(100,100), alpha=1e-5, max_iter=1000	0.9626	0.9592

TABLE I

TABLE 1: SUPERVISED MODELS PERFORMANCE

The Random Forest classifier achieved the highest performance with a best balanced accuracy of 0.9827 and an average of 0.9807 across folds. The SVM with RBF kernel followed closely with a best balanced accuracy of 0.9775 for the optimised configuration, while the poorly parameterised third configuration (C=0.1, gamma=1) performed no better than random guessing. The MLP classifier demonstrated competitive performance with a best score of 0.9757 for its second configuration.

Confusion matrices for the optimal configuration of each supervised model revealed excellent classification capabilities for both ADLs and falls, with minimal misclassification.

In contrast, the Isolation Forest exhibited notably lower performance as indicated in Table 2.

The Isolation Forest results demonstrate that as the contamination parameter increases, the ROC AUC score improves, with the highest value of 0.6720 achieved at a contamination rate of 0.50. However, based on the confusion matrices, as contamination increases, the True Negative and False Positive rates decrease and increase respectively, while the inverse is true for True Positive and False Negative cases. This indicates that allowing the algorithm to identify a larger proportion of data points as anomalies enhances its ability to detect falls, though it remains significantly below the performance of supervised models.

A. *Feature Importance Analysis*

The Random Forest classifier provides an intrinsic measure of feature importance based on contribution to decreasing impurity across decision trees. Figure 2 displays the top 10 features ranked by importance score.

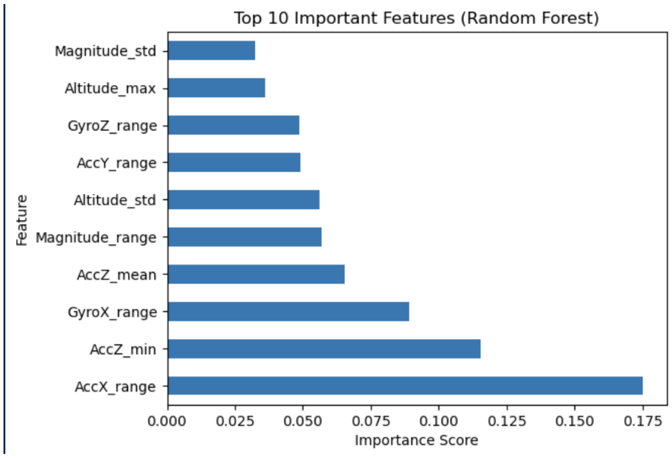

Fig. 2. Feature importance from random forest

The most informative features for fall detection were: AccX_range (0.175) - Range of acceleration in X-axis; AccZ_min (0.125) - Minimum acceleration in Z-axis; GyroX_range (0.095) - Range of angular velocity in X-axis; AccZ_mean (0.075) - Mean acceleration in Z-axis; and Magnitude_range (0.060) - Range of vector magnitude (see figure 2).

These results highlight that lateral (X-axis) acceleration range and vertical (Z-axis) minimum acceleration are particularly important indicators of falls. This aligns with the biomechanics of falling, where sudden changes in acceleration patterns, particularly in the direction of gravity, are characteristic of fall events.

B. *Effect of Contamination Parameter on Isolation Forest Performance*

The contamination parameter in Isolation Forest significantly affected detection performance. Figure 3 illustrates the positive relationship between contamination rate and ROC AUC score, with an almost linear improvement as contamination increases from 0.10 to 0.35 and a slower rate of improvement with contamination rates from 0.45 to 0.50.

This trend can be explained by examining the confusion matrices at different contamination levels. At lower contamination rates (0.10), the model identifies fewer instances as anomalies, resulting in high false negatives (443 falls classified as ADLs). As the contamination rate increases to 0.35, more instances are classified as anomalies, improving fall detection (reducing false negatives to 255) at the cost of increased false positives (from 64 to 286 ADLs misclassified as falls).

The trade-off is evident: higher contamination values improve fall detection sensitivity but reduce specificity. For critical applications like fall detection, where missing a fall event (false negative) may have more severe consequences than a false alarm (false positive), higher contamination settings may be justified despite the overall lower performance compared to supervised approaches.

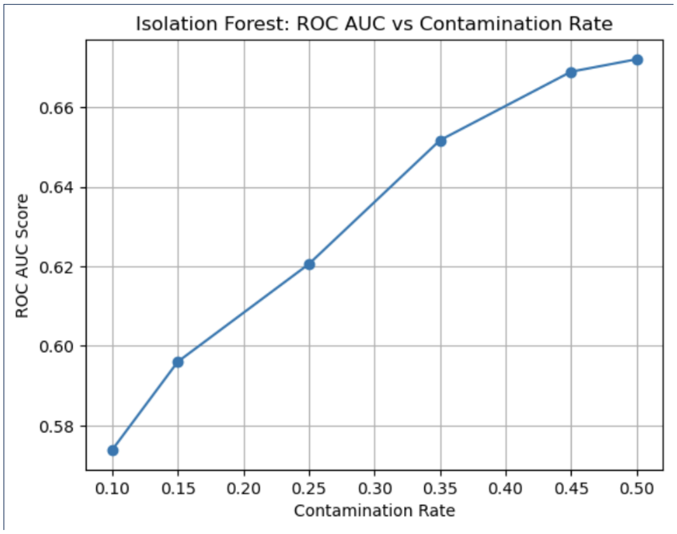

Fig. 3. ROC AUC vs Contamination Rate

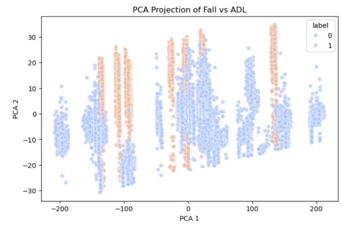

Fig. 4. PCA projection of Fall vs ADL

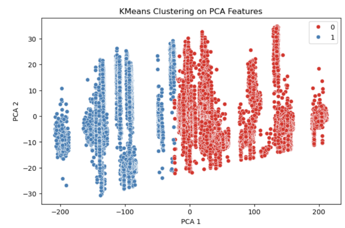

Fig. 5. KMeans Clustering on PCA Features

C. Dimensionality reduction and visualization

Principal Component Analysis (PCA) was applied to visualise the separation between falls and ADLs in lower-dimensional space. The first two principal components captured approximately 42% of the total variance in the dataset.

The PCA scatter plot revealed partial separation between falls and ADLs, with some distinct clusters but also significant overlap regions. This visualisation helps explain why supervised models, which can learn complex decision boundaries, outperform unsupervised methods like Isolation Forest that rely primarily on statistical outlier detection.

K-means clustering was applied to the PCA-reduced data to assess whether unsupervised clustering could naturally separate the classes. The Adjusted Rand Index (ARI) between K-means cluster assignments and true labels was 0.195, indicating limited but non-random correspondence. This moderate ARI value suggests that while falls do have distinguishable characteristics in the feature space, the separation is not clean enough for straightforward unsupervised classification.

An autoencoder is an unsupervised neural network architecture designed to learn a compressed representation of input data. The model is trained to minimize its reconstruction (from the decoder) error typically using mean squared error between the input and its reconstruction, learning to retain the most essential features of the data. Autoencoders are particularly useful for non-linear dimensionality reduction, offering an alternative to traditional methods such as Principal Component Analysis (PCA). The autoencoder consisted of a fully connected feedforward neural network. The encoder compressed the input feature set into a 2-dimensional latent representation using two linear layers with ReLU activations, and the decoder reconstructed the original input from this compressed space. The network was trained using mean squared error (MSE) loss over 50 epochs with the Adam optimizer, using mini-batch gradient descent.

After training, the encoder was used to generate 2D latent representations of the dataset. These were then passed to the K-means clustering algorithm to evaluate whether this non-linear feature space yielded better class separability. The Adjusted Rand Index (ARI) comparing the cluster assignments to the true labels was 0.179, slightly lower than the PCA-based clustering result. This indicates that while the autoencoder learned some structure in the data, the non-linear compression did not significantly improve the ability of unsupervised clustering to distinguish between falls and ADLs. As with PCA, some class-specific patterns are likely present, but not sufficiently distinct for clean, unsupervised separation.

V. Discussion

A. Supervised vs unsupervised performance

Our results demonstrate a substantial performance gap between supervised and unsupervised approaches for fall detection. The best supervised model (Random Forest) achieved

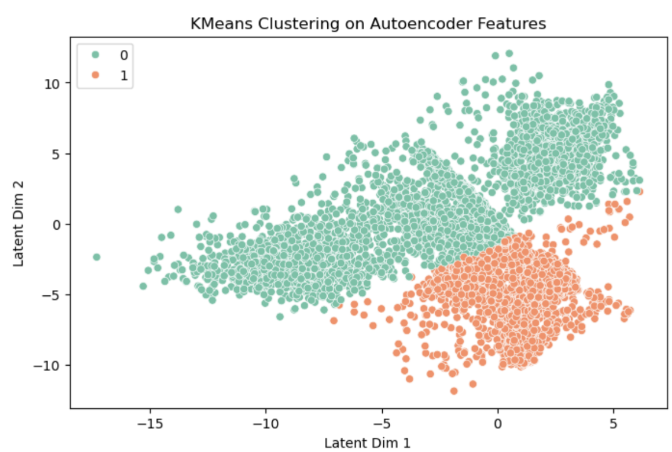

Fig. 6. KMeans clustering on Autoencoder features

a balanced accuracy of 0.9827, while the best unsupervised approach (Isolation Forest with contamination=0.35) reached an ROC AUC of 0.6473.

This performance disparity can be attributed to fundamental differences in how these approaches function. Supervised models benefit from learning specific patterns that differentiate falls from ADLs during training, using labelled examples to optimise decision boundaries. In contrast, Isolation Forest operates without knowledge of true labels, relying solely on identifying statistical outliers in the feature space.

Despite its lower performance, Isolation Forest's ability to detect falls with an AUC of 0.6720 (significantly above the random baseline of 0.5) suggests it has practical utility in scenarios where labelled training data is unavailable or expensive to obtain. Such situations might include initial deployment phases before sufficient labelled data is collected, personalised systems that must adapt to individual movement patterns, and detection of novel or rare fall types not represented in training data.

The improvement in performance with increasing contamination parameter also suggests that domain knowledge about the expected frequency of falls can be incorporated to optimise unsupervised detection.

B. Feature analysis insights

The feature importance analysis revealed that motion characteristics in specific axes are particularly relevant for fall detection. The predominance of range measurements (AccX_range, GyroX_range, Magnitude_range) among the top features suggests that the magnitude of change in acceleration and rotation is more informative than absolute values.

The high importance of AccZ_min aligns with biomechanical understanding of falls, where a sudden decrease in vertical acceleration occurs during the free-fall phase before impact. Similarly, the importance of X-axis measurements (lateral motion) highlights that falls often involve sideways movement patterns distinct from normal ADLs.

These findings have practical implications for sensor placement and feature engineering in fall detection systems. Sensors positioned to accurately capture lateral and vertical acceleration changes would likely provide the most discriminative data. Additionally, feature extraction algorithms should prioritise range calculations and minimum values, particularly for vertical acceleration.

C. Practical implications

Our findings have several practical implications for real-world fall detection systems:

Model Selection Trade-offs: While supervised models offer superior accuracy, they require substantial labelled training data. In applications where such data is limited or where detecting novel fall patterns is critical, combining supervised models with anomaly detection approaches like Isolation Forest may provide a more robust solution. This could be achieved through a semi-supervised learning approach where an unsupervised model such as Isolation forest is used to assign pseudo-labels to an unlabelled set of data, then the supervised models can be trained using the pseudo-labelled data.

Class Imbalance Handling: The effectiveness of SMOTE in improving supervised model performance demonstrates the importance of addressing class imbalance. Real-world systems should incorporate similar techniques to ensure balanced training, as falls will typically be rare events compared to normal activities.

Feature Engineering Focus: The identified important features suggest that developers should focus on extracting range, minimum, and mean values from accelerometer and gyroscope data, particularly in the X and Z axes. This could potentially reduce computational requirements by limiting feature extraction to the most informative measurements.

Contamination Parameter Tuning: For unsupervised approaches, careful tuning of the contamination parameter based on the expected fall frequency and the relative costs of false positives versus false negatives is essential. Higher contamination values may be appropriate for applications where missing a fall event has severe consequences.

Limitations: Despite strong performance in our controlled dataset, real-world deployment would face additional challenges including device positioning variability, sensor drift, and individual movement pattern differences. These factors might reduce the performance gap between supervised and unsupervised approaches in practical applications.

Future research should explore hybrid approaches that leverage the strengths of both supervised and unsupervised methods, potentially using anomaly detection as a complementary system to identify falls that supervised models might miss due to their novelty or unusual characteristics.

VI. Conclusion and Future Work

A. Conclusion

This study investigated the comparative performance of supervised and unsupervised machine learning approaches for fall detection using wearable sensor data. Our primary research question, whether Isolation Forest is comparable to supervised models for fall detection, has been comprehensively addressed through rigorous empirical evaluation.

The findings demonstrate that supervised approaches, particularly Random Forest, significantly outperform unsupervised methods in fall detection accuracy. The best Random Forest model achieved a balanced accuracy of 0.9827, setting a high benchmark for classification performance. Support Vector Machines with optimised parameters and Multi-layer Perceptron neural networks also demonstrated excellent performance, with balanced accuracies of 0.9775 and 0.9757 respectively. These results confirm the effectiveness of supervised learning in discriminating between falls and activities of daily living when sufficient labelled data is available.

In contrast, Isolation Forest, the unsupervised anomaly detection technique, achieved a maximum ROC AUC score of 0.6720 with a contamination parameter of 0.50. While this performance is substantially lower than that of supervised methods, it remains significantly above random classification, indicating that unsupervised approaches can detect falls to a moderate degree without prior knowledge of labels. The incremental improvement in performance with increasing contamination parameter values highlights the importance of appropriate threshold selection in anomaly detection systems.

In summary, while supervised models remain the preferred approach when labelled data is available, unsupervised methods like Isolation Forest offer a viable alternative in scenarios where obtaining labelled training data is impractical or prohibitively expensive. These findings contribute to the growing body of knowledge on machine learning applications in healthcare and safety monitoring, particularly in the context of fall detection systems.

Several promising directions for future research emerge from this study:

Hybrid Detection Systems: Developing hybrid systems that combine the strengths of both supervised and unsupervised approaches could enhance overall performance. Such systems might employ supervised models for routine detection while using unsupervised methods to identify novel or unusual fall patterns not represented in training data.

Real-Time Adaptation: Incorporating models that adapt over time to a user's specific movement patterns could improve personalised fall detection accuracy. This might involve online learning or anomaly detection that updates with new data.

B. Future work

Temporal Modelling: Our current approach treats each sensor reading as an independent data point. Future work could explore temporal models such as recurrent neural networks (RNNs) or long short-term memory networks (LSTMs) to capture the sequential nature of fall events, potentially improving detection accuracy.

Transfer Learning: Investigating transfer learning techniques to adapt pre-trained models to new individuals or environments could address the challenge of personalisation in fall detection systems. This approach might reduce the amount of user-specific labelled data required for effective deployment.

These future directions represent significant opportunities to advance the field of fall detection and broader applications of machine learning in healthcare monitoring systems. By addressing these challenges, researchers can contribute to the development of more accurate, reliable, and accessible fall detection technologies that enhance safety and independence for vulnerable populations.

References

[1] Ivan Ursul. "Sensor-Based Fall Detection Dataset with 8,953 Activities from 29 Subjects". In: *FigShare* (Jan. 2025). DOI: 10.6084/m9.figshare.28287482.v1. URL: https://figshare.com/articles/dataset/Sensor-Based_Fall_Detection_Dataset_with_8_953_Activities_from_29_Subjects/28287482.

[2] David Sarabia-Jácome et al. "Highly-efficient fog-based deep learning AAL fall detection system". In: *Internet of Things* 11 (2020), p. 100185. ISSN: 2542-6605. DOI: https://doi.org/10.1016/j.iot.2020.100185. URL: https://www.sciencedirect.com/science/article/pii/S2542660518300829.

[3] T. Fredriksson et al. "Data Labeling: An Empirical Investigation into Industrial Challenges and Mitigation Strategies". In: *International Conference on Product-Focused Software Process Improvement*. Cham: Springer International Publishing, Nov. 2020, pp. 202–216.

[4] Sofia Yousuf and Muhammad Bilal Kadri. "A ubiquitous architecture for wheelchair fall anomaly detection using low-cost embedded sensors and isolation forest algorithm". In: *Computers and Electrical Engineering* 105 (2023), p. 108518. ISSN: 0045-7906. DOI: https://doi.org/10.1016/j.compeleceng.2022.108518. URL: https://www.sciencedirect.com/science/article/pii/S0045790622007339.

[5] Zhangyu Cheng, Chengming Zou, and Jianwei Dong. "Outlier detection using isolation forest and local outlier factor". In: *Proceedings of the Conference on Research in Adaptive and Convergent Systems*. RACS '19. Chongqing, China: Association for Computing Machinery, 2019, pp. 161–168. ISBN: 9781450368438. DOI: 10.1145/3338840.3355641. URL: https://doi.org/10.1145/3338840.3355641.

[6] Nishat Tasnim Newaz and Eisuke Hanada. "The Methods of Fall Detection: A Literature Review". In: *Sensors* 23.11 (2023). https://www.mdpi.com/1424-8220/23/11/5212/pdf?version=1685503233, p. 5212. DOI: 10.3390/s23115212. URL: https://app.dimensions.ai/details/publication/pub.1158523233.

[7] P. Pierleoni et al. "A High Reliability Wearable Device for Elderly Fall Detection". In: *IEEE Sensors Journal* 15.8 (2015), pp. 4544–4553. DOI: 10.1109/JSEN.2015.2423562. URL: https://doi.org/10.1109/JSEN.2015.2423562.

[8] D. Baga et al. "PERFORM: A Platform for Monitoring and Management of Chronic Neurodegenerative Diseases: The Parkinson and Amyotrophic Lateral Sclerosis Case". In: *Proceedings of the 2009 4th International IEEE/EMBS Conference on Neural Engineering*. Antalya, Turkey: IEEE, 2009, pp. 1–5.