



Features Selection for Fall Detection Systems Based on Machine Learning and Accelerometer Signals

Carlos A. Silva^{1,2}(✉), Rodolfo García–Bermúdez¹(✉), and Eduardo Casilari²(✉)

¹ Universidad Técnica de Manabí, Manabí, Ecuador
{carlos.silva, rodolfo.garcia}@utm.edu.ec

² Departamento de Tecnología Electrónica, Universidad de Málaga, Málaga, España
ecasilari@uma.es

Abstract. Fall among older people is a major medical concern. Fall Detection Systems (FDSs) have been actively investigated to solve this problem. In this sense, FDSs must effectively reduce both the rates of false alarms and unnoticed fall. In this work we carry out a systematic evaluation of the performance of one of the most widely used machine learning supervised algorithm (Support Vector Machine) when using different input features. To evaluate the impact of the feature selection, we use Area Under the Curve (AUC) of Receiver Operating Characteristic (ROC) Curve as the performance metric. The results showed that with four features it is possible to obtain acceptable values for the detection of falls using accelerometer signals obtained from the user's waist. In addition, we also investigate if the impact of selecting the features based on the analysis of a dataset different from the final application framework where the detector will be operative.

Keywords: Fall detection system · Supervised algorithm · Accelerometer · Wearables · Support vector machine · Feature selection

1 Introduction

The World Health Organization (WHO) has estimated a growth in the number of people over 60 years of age worldwide who annually suffer a fall from 688 million in 2006 to two billion by 2050 [1]. Falls are one of the main causes of morbidity and mortality among elderly as they can provoke from damage to wrists to hip fractures or even traumatic brain injuries [2].

Automatic Fall Detection Systems (FDSs) are being actively investigated in order to solve this problem. An FDS can be defined as a binary classification system which must constantly decide if the movements executed by the subject under monitoring correspond to a possible fall or if, instead, they are originated by conventional Activities of Daily Living (ADLs).

The FDSs are traditionally categorized into two groups: context-aware and wearable-based. The first group utilizes image analysis from video-camera recording, Kinect-like devices and/or other environmental sensors, such as pressure sensors or acoustic

sensors. Context-aware methods present not-negligible installation and service costs as well as remarkable operational limitations due to privacy issues, camera blind spots, low device video resolution, little lighting and other visual artifacts [3]. In contrast, wearable FDSs can be transported as an extra garment or seamlessly integrated via software into conventional personal devices (such as smartwatches or even smartphones).

Due to the difficulties of systematically testing a FDS with actual falls experienced by elderly people, current FDS prototypes are typically trained, shaped and evaluated based on the movements (in particular, mimicked falls) normally generated in a controlled laboratory environment. This aspect makes us question the effectiveness of a certain FDS not only when it is applied on real world falls but also when it is extrapolated to detect falls with a dataset different from that used to configure the classifier.

FDS algorithms can be grouped into threshold-based techniques and detectors implemented on supervised learning or Machine Learning (ML) strategies. One of the issues for FDSs based on thresholds is finding an appropriate value for the threshold able to accurately differentiate falls from ADLs [4]. Therefore, supervised learning algorithms are commonly preferred. In this regard, a key element for the definition of any supervised learning is the election of the features with which input data are characterized.

This work carries out a systematic evaluation of the impact of the selection of the input features. The goal is to assess if the selection of features based on the analysis of a certain dataset is a good choice when a different dataset (with different movements and users) is considered. For this purpose, the study focuses on the behavior of a popular machine learning algorithm (Support Vector Machine -SVM-) when it is applied to several public datasets containing accelerometer signals from ADLs and falls.

The remainder of this paper is organized as follows: Sect. 2 reviews related works. Section 3 describes the experimental framework, including the dataset and input features selection. Section 4 presents and discusses the results. Finally, Sect. 5 recapitulates the conclusions of our work.

2 Related Works

An FDS error causes either a false negative or a false positive. The first case implies not detecting a real fall while the second provokes an unnecessary alarm as an ADL is misinterpreted as a fall. An excessive number of false positives or false alarms can cause users to consider the system to be ineffective and useless [5].

SVM is one of the most employed techniques used to implement the fall detection algorithms in wearable systems (refer, for example, to [6] or [7] for two recent reviews on detection strategies for FDSs). Many works have shown that SVM can outperform other typical ML strategies [8, 9].

In 2006, Tong Zhang et al. [10] already proposed a method based on the One-class SVM algorithm. The algorithm was tested with the movement signals of the human body using a tri-axial accelerometer. The tests were carried out with 600 samples (from 6 categories) recorded from 12 volunteers. They obtained between 80 and 100% correct detection rate using the following six features: 1) the interval between the beginning of falling and the beginning of the reverse impact, 2) the average acceleration, 3) the variance of acceleration before the impact, 4) the interval of the reverse impact, 5) the average acceleration, and 6) the variance of acceleration after the impact.

Paulo Salgado et al. [11] tested the SVM algorithm with Gaussian Kernel function using accelerometers signals collected with a smartphone located on the volunteer's torso using four features to determine the falls: 1) angular position, 2) angular rate, 3) angular acceleration and 4) radius curve. The tests showed a detection rate of 96%.

Medrano et al. [12] evaluated several supervised classifiers, including a SVM model with a Radial Basis Function kernel, using the accelerometer signals recorded with a smartphone transported by 10 volunteers, who generated 503 samples of simulated falls and 800 ADLs. Authors concluded that SVM clearly outperformed the other algorithms. Regarding to the information fed into the classifier, they have restricted this work to discriminate the acceleration shape during falls using the raw acceleration values.

Santoyo-Ramón et al. [13] used the UMAFall dataset containing data of accelerometers signals of 19 volunteers. These authors evaluated four algorithms: SVM, k-Nearest Neighbors (KNN), Naive Bayes and Decision Tree using six features: 1) Mean signal magnitude vector, 2) The maximum value of the maximum variation of the acceleration components in the three axes, 3) the standard deviation of the signal magnitude vector, 4) The mean rotation angle, 5) mean absolute difference, 6) the mean module acceleration. They concluded that the SVM algorithm achieved the performance in the detection of falls. In a subsequent work [14] they investigated the most significant features of the accelerometer signals, using the ANOVA tool for the statistical analysis of the performance. They evaluated the algorithms with three datasets, UMAFall [15], SisFall [16], and 'Erciyes' repository by Özdemir et al. [17].

In this work we focused on evaluating the importance of a set of candidates features for seven datasets commonly employed in the related literature. Random Forests (RFs) are a common tool for feature selection in machine-learning classifiers as their easy interpretability allows directly weighting the importance of each candidate feature to represent and characterize the data [18]. Thus, by using a RFs model [19] we determine which are the most relevant statistical features for each dataset. Then, we assess the performance of SVM depending on the dataset employed to select the feature set.

3 Process Description for Testbed

3.1 Data Bases Selection

The tests were carried out with publicly available datasets with accelerometer signals obtained on the volunteers' waist. In the literature, eleven public datasets, presented in Table 1, were found with this criterion. The table indicates the complete number of samples and falls for each dataset.

To characterize each movement sample in the datasets, the features are calculated by focusing on a fixed time interval (an 'observation window') within every sample. As a fall is associated to a sudden peak in the acceleration magnitude caused by the impact against the floor, this window is selected around the instant (± 0.5 s) where the maximum value of the acceleration module occurs. For our research we discarded FARSEEING, SMotion, UR Fall and DLR datasets, as no falls (or an insignificant number of falls) were found after applying this windowing techniques to the corresponding traces. In the rest of the datasets, we also ignored those samples in which the acceleration peak was found in the first or last period of 0.5 s of the whole time series (so that a complete

observation window of 1 s cannot be properly defined). The datasets and the number of valid samples finally employed in our tests are shown in Table 2.

Table 1. Public datasets with accelerometer signals captured on volunteers' waist.

	Name and reference	Total number of samples	Falls
1	TST Fall [20]	264	132
2	UMAFall [15]	746	208
3	SisFall [16]	4050	1798
4	Erciyes [17]	3297	1821
5	UP Fall [21]	559	255
6	IMUFD [9]	600	210
7	FallAIID [22]	6605	1722
8	DLR [23]	1017	56
9	UR Fall detection [24]	70	30
10	FARSEEING [25]	22	22
11	SMotion [26]	364	2

Table 2. Employed datasets after applying the windowing technique.

	Name	Final number of used samples	Final number of used falls
1	TST Fall	263	132
2	UMAFall	604	186
3	SisFall	4474	1797
4	Erciyes	3147	1819
5	UP Fall	505	234
6	IMUFD	591	210
7	FallAIID	1661	465

3.2 Feature Selection

In this work, we use the six 'candidate' features described in the work [14]. Additionally, we added seven extra features (see [27] for a more complete formal definition). For all the samples (ADLs or falls), the thirteen features were calculated taking into account only the triaxial signals captured by the accelerometer (located on the waist) that are provided by the seven datasets under study. The considered features are:

1. Mean Signal Magnitude Vector of the acceleration vector (*Mean SMV*).

2. The maximum value of the maximum variation of the acceleration components in the three axes (*Max diff*).
3. The standard deviation of the Signal Magnitude Vector (*Std SMV*).
4. The mean rotation angle (*Mean rotation angle*).
5. The mean absolute difference between consecutive samples of the acceleration magnitude (*Mean absolute diff*).
6. The mean acceleration of the magnitude of the vector formed by the acceleration components that are parallel to the floor plane (*Mean body inclination*).
7. Maximum value of the acceleration magnitude (*Max SMV*).
8. Minimum value of the acceleration magnitude (*Min SMV*).
9. Third central moment (skewness or bias) of the acceleration magnitude (*Skewness SMV*).
10. Fourth central moment (kurtosis) of the acceleration magnitude (*Kurtosis SMV*).
11. The mean of the autocorrelation of the acceleration magnitude within the observation window (*Mean autocorrelation*).
12. The standard deviation of the autocorrelation module the acceleration magnitude within the observation window (*Std autocorrelation*).
13. The frequency value at which the maximum of the Discrete Fourier Transform (DFT) of the acceleration magnitude is detected (*Freq DFT max*).

3.3 Classification of the Candidate Features

The capability of the different candidate features to characterize the signals was evaluated and classified through Random Forests classification [19], which fits a number of decision tree classifiers on various sub-samples of the dataset and uses averaging to improve the predictive accuracy while controlling over-fitting. For this study, one hundred trees were used. As a result, a matrix is obtained (see Table 3) with the percentage of importance of each feature for each database. As it can be observed, the relevance of each feature as a discrimination element strongly varies depending on the considered dataset.

From the previous data, aiming at defining a global classification of the features, we compute a global ranking by calculating the mean of the percentage importance value of each feature for each dataset. Figure 1 shows (in descending order) the error bar graphic with represents the mean and the standard deviation of the obtained values.

Table 3. Features importance percentage value for each dataset.

Feature	Erciyes	FallAIID	IMUFD	SisFall	TST Fall	UMAFall	UP Fall
Kurtosis SMV	0.90	3.08	0.55	1.86	0.41	0.37	0.00
Max diff	20.98	17.08	13.32	9.93	5.41	12.84	18.20
Max SMV	25.36	19.07	17.09	14.92	6.26	12.75	15.03
Freq DFT max	0.14	10.36	15.24	1.12	26.96	2.41	0.61
Mean body inclination	1.60	19.84	1.41	1.47	7.41	9.35	0.00
Mean autocorrelation	14.95	7.39	16.02	20.27	4.13	11.96	12.40
Mean rotation angle	1.32	0.83	1.85	7.53	6.59	7.58	6.23
Mean absolute diff	0.49	0.34	0.30	0.15	0.03	3.54	5.43
Mean SMV	0.09	5.36	5.68	0.83	17.23	0.78	1.39
Min SMV	3.43	3.09	0.03	1.17	1.24	4.40	5.85
Std SMV	12.14	5.69	21.36	15.27	13.69	16.36	18.84
Skewness SMV	6.75	5.70	1.31	9.53	1.25	0.89	1.75
Std autocorrelation	11.86	2.15	5.85	15.95	9.39	16.77	14.27

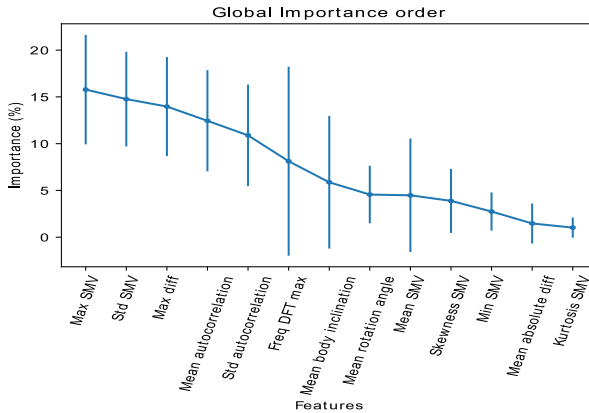


Fig. 1. Mean importance of the features.

Table 4 shows the features ordered by the mean importance percentage. This ranking does not coincide with that provided in [14], which used the ANOVA analysis. In that work authors estimated that the best global variables were: 1) Std SMV, 2) body inclination, 3) max diff, 4) rotation angle, 5) absolute difference and 6) mean SMV.

Table 4. Global feature ranking.

Order	Feature	Mean importance (%)
1	Max SMV	15.78
2	Std SMV	14.76
3	Max diff	13.97
4	Mean autocorrelation	12.45
5	Std autocorrelation	10.89
6	Freq DFT max	8.12
7	Mean body inclination	5.87
8	Mean rotation angle	4.56
9	Mean SMV	4.48
10	Skewness SMV	3.88
11	Min SMV	2.74
12	Mean absolute diff	1.47
13	Kurtosis SMV	1.02

4 Results and Discussion

In this section, we analyze the performance of the SVM algorithm when different criteria to select the input features are followed. The goal is to evaluate if the performance of the ML technique degrades when it is individually applied to a certain dataset, but the selection of the features is based on the analysis of a different repository. In this framework we also consider the case when the most relevant features are selected according to the mean case (following the global ranking presented in the previous tables).

Before training and testing, all the features values were scaled following a typical Z-score normalization [28].

4.1 Model Selection

For the study, we utilized Python and the implementation of the SVM algorithm provided by the *scikit-learn* package [29]. The SVM algorithm was evaluated with the following kernels: linear, polynomial, Radial Basis Function (RBF) and Sigmoid. The RBF kernel offered the best results, so the following tests were based on SVM configured with this kernel.

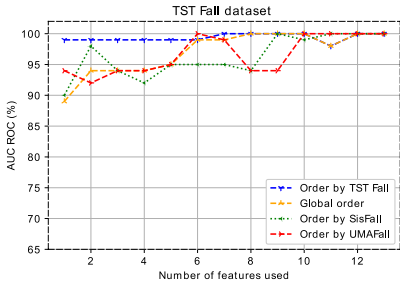
When training an SVM with the RBF kernel, two parameters must be considered: C and $gamma$. The parameter C , common to all SVM kernels, trades off misclassification of training examples against simplicity of the decision surface. The $gamma$ parameter in turn defines the influence of a single training example [29]. The optimal values for the C and $gamma$ parameters were set through a “fit” and a “score” method provided by *scikit-learn* package [30].

4.2 Model Training and Evaluation

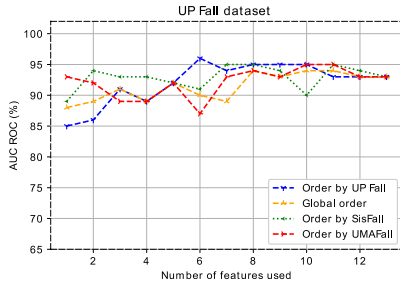
We evaluated the performance of the SVM algorithm with a variable number of input features (from 1 to 13), taking into the rank obtained for both the global and particular analysis of the datasets. Model training is achieved with 70% of falls and 70% of the ADL while testing is carried out with the remaining 30% of samples. Whenever a new test is triggered, an extra feature is added according to 1) the rank established by the dataset under analysis ('self-order'), 2) the global rank, 3 and 4) the rank defined by the analysis of two particular datasets (SisFall and UMAFall, as they have been frequently utilized by the related literature).

As a performance metric, we use Area Under the Curve (AUC) of Receiver Operating Characteristic (ROC) Curve. The AUC provides an aggregate measure of performance across all possible classification thresholds [31]. Figure 2 shows the ROC AUC values of the evaluations for each dataset for the four feature selection criteria and for the 13 possible combinations of features. Results show that, for almost all datasets and for a low number of features, basing the selection on the global analysis of all datasets leads to a certain increase of the performance of the detection algorithm, when it is compared with the case in which just a single dataset is considered as the unique reference (SisFall and UMAFall) to rank and select the features. This performance increase in the detection algorithm is more remarkable with the FallAID dataset.

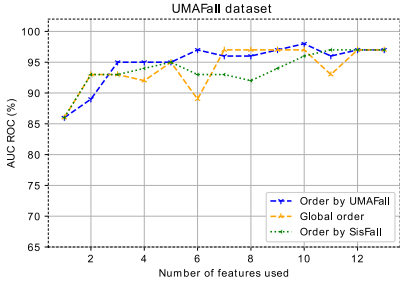
Another important aspect when dealing with ML strategies is the reduction of the number of features, as a lower dimension of the features may ease the implementation of the algorithm in low cost devices with limited computation resources. In this regard, from Fig. 2 we observe that an increase of the number of features beyond 4 or 5 does not imply a significant improvement of the performance metric if the selection of features is based on the global knowledge. The values of AUC are tabulated in Table 5 for the case in which the four most relevant features are selected to characterize the data. The table also shows the sensitivity (Se) and specificity (Sp) values (for the optimal configuration of the SVM), which are also basic metrics to evaluate binary classifiers.



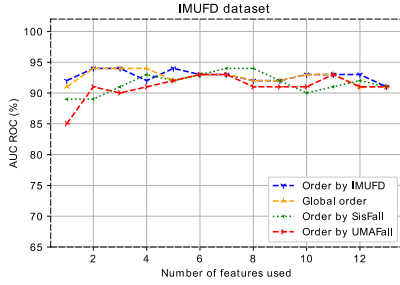
(a) TST Fall dataset



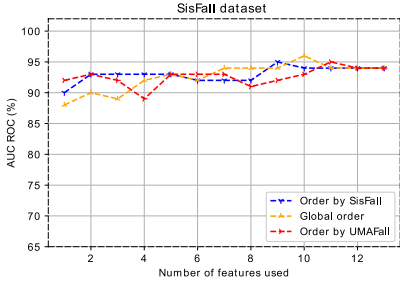
(e) UP Fall dataset



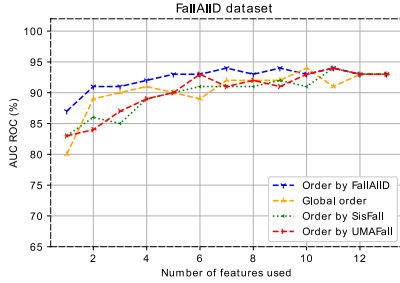
(b) UMAFall dataset



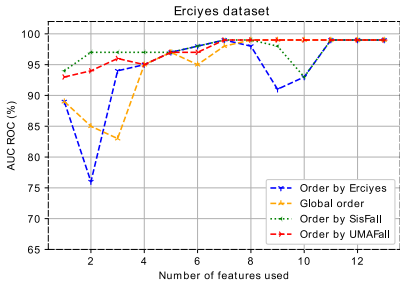
(f) IMUFD dataset



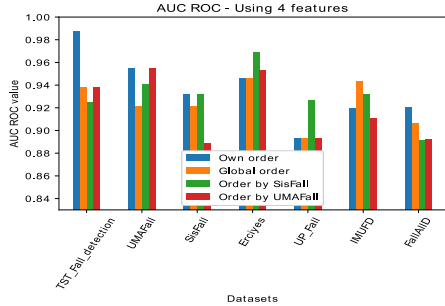
(c) SisFall dataset



(g) FallAIID dataset



(d) Erciyes dataset



(h) Results using 4 features

Fig. 2. AUC ROC for different feature selection criteria and number of features.

Table 5. Sensitivity, specificity and AUC ROC for each dataset applying the 4 most significant features according to the global ranking, the order recommended by SisFall and the order derived from the analysis of each dataset.

	Dataset	Global order			Self-order			SisFall order		
		AUC ROC	<i>Se</i>	<i>Sp</i>	AUC ROC	<i>Se</i>	<i>Sp</i>	AUC ROC	<i>Se</i>	<i>Sp</i>
1	TST Fall	93.75	100.0	87.50	98.75	100.0	97.50	92.50	97.50	87.50
2	UMAFall	92.16	87.50	96.83	95.44	96.43	94.44	94.05	92.86	95.24
3	SisFall	92.08	89.26	94.90	93.16	90.19	96.14	93.16	90.19	96.14
4	Erciyes	94.55	97.62	91.48	94.55	97.62	91.48	96.86	97.99	95.74
5	UP Fall	89.30	85.92	92.68	89.30	85.92	92.68	92.63	90.14	95.12
6	IMUFD	94.29	92.06	96.52	91.91	87.30	96.52	93.14	88.89	97.39
7	FallAIID	90.60	89.29	91.92	92.06	88.57	95.54	89.16	85.00	93.31

5 Conclusions

This paper has assessed the capacity of the detection algorithms intended for wearable FDSs to extrapolate conclusions obtained with a certain dataset when they are applied to data obtained with other users and movements. The study is based on the analysis of seven public datasets that contain recorded accelerometer signals captured on the volunteers' waist and the use of AUC ROC as a performance validation metric.

In particular, we analyzed the performance of the SVM algorithm when up to thirteen accelerometer-based features are considered to feed the classifier. Results reveal the difficulties of low dimensional algorithms to detect falls when the selection criteria are merely based on a previous study of a single dataset. This fact highlights the importance of considering a variety of datasets to configure, train and test any fall detection algorithm, as most current datasets have remarkable limitations in terms of the number of experimental subjects and the typology of movements with which they were generated.

On the other hand, the results also show with a small set of well selected features (in this work four features seem to be enough) an acceptable performance can be obtained for fall detection. The results suggest that using a greater number of features to determine if a fall occurs just slightly improves detection effective ratio at the cost of increasing the complexity of the algorithms, which may hamper its implementation on the low cost wearable devices that produce the fall detection decision in real-time.

Funding Information. Research presented in this article has been partially funded by FEDER Funds (under grant UMA18-FEDERJA-022), Universidad de Málaga, Campus de Excelencia Internacional Andalucía Tech and Asociación Universitaria Iberoamericana de Postgrado (AUIP).

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

References

1. Yoshida, S.: A global report on falls prevention epidemiology of falls. World Health Organization (2007)
2. Stevens, J.A., Corso, P.S., Finkelstein, E.A.: The costs of fatal and nonfatal falls among older adults. *Inj. Prev.* **12**, 290–295 (2006)
3. Vallabh, P., Malekian, R.: Fall detection monitoring systems: a comprehensive review. *J. Ambient. Intell. Humaniz. Comput.* **9**(6), 1809–1833 (2017). <https://doi.org/10.1007/s12652-017-0592-3>
4. Yacchirema, D., de Puga, J.S., Palau, C., Esteve, M.: Fall detection system for elderly people using IoT and ensemble machine learning algorithm. *Pers. Ubiquitous Comput.* **23**(5), 801–817 (2019)
5. Igual, R., Medrano, C., Plaza, I.: Challenges, issues and trends in fall detection systems. *Biomed. Eng. Online* **12**(1), 66 (2013)
6. Ramachandran, A., Karuppiah, A.: A survey on recent advances in wearable fall detection systems. *BioMed Research International*, vol. 2020. Hindawi Limited (2020)
7. Rastogi, S., Singh, J.: A systematic review on machine learning for fall detection system. *Comput. Intell.* **4**, 1–24 (2021)
8. Hou, M., Wang, H., Xiao, Z., Zhang, G.: An SVM fall recognition algorithm based on a gravity acceleration sensor. *Syst. Sci. Control Eng.* **6**(3), 208–214, September 2018
9. Aziz, O., Musngi, M., Park, E.J., Mori, G., Robinovitch, S.N.: A comparison of accuracy of fall detection algorithms (threshold-based vs. machine learning) using waist-mounted tri-axial accelerometer signals from a comprehensive set of falls and non-fall trials. *Med. Biol. Eng. Comput.* **55**(1), 45–55 (2016). <https://doi.org/10.1007/s11517-016-1504-y>
10. Zhang, T., Wang, J., Xu, L., Liu, P.: Fall Detection by Wearable Sensor and One-Class SVM Algorithm, pp. 858–863. Springer, Berlin, Heidelberg (2006). https://doi.org/10.1007/978-3-540-37258-5_104
11. Salgado, P., Afonso, P.: Body Fall Detection with Kalman Filter and SVM, pp. 407–416. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-10380-8_39
12. Medrano, C., Igual, R., Plaza, I., Castro, M.: Detecting falls as novelties in acceleration patterns acquired with smartphones. *PLoS One* **9**(4), e94811 (2014)
13. Santoyo-Ramón, J., Casilari, E., Cano-García, J., Santoyo-Ramón, J.A., Casilari, E., Cano-García, J.M.: Analysis of a smartphone-based architecture with multiple mobility sensors for fall detection with supervised learning. *Sensors* **18**(4), 1155, April 2018
14. Santoyo-Ramón, J.A., Casilari-Pérez, E., Cano-García, J.M.: Study of the detection of falls using the svm algorithm, different datasets of movements and ANOVA. In: Rojas, I., Valenzuela, O., Rojas, F., Ortuño, F. (eds.) *IWBIO 2019. LNCS*, vol. 11465, pp. 415–428. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-17938-0_37
15. Casilari, E., Santoyo-Ramón, J.A., Cano-García, J.M.: Analysis of public datasets for wearable fall detection systems. *Sensors (Switzerland)*, **17**(7), 1513, July 2017
16. Sucerquia, A., López, J.D., Vargas-bonilla, J.F.: SisFall: a fall and movement dataset. *Sensors* **19**(52), 1–14 (2017)
17. Özdemir, A.T.: An analysis on sensor locations of the human body for wearable fall detection devices: principles and practice. *Sensors (Switzerland)* **16**(8), 1161 (2016)
18. Dewi, C., Chen, R-C.: Control, and undefined. Random forest and support vector machine on features selection for regression analysis. *ijcic.org* (2019)
19. sklearn.ensemble.RandomForestClassifier—scikit-learn 0.24.1 documentation. <https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html>. Accessed 16 Apr 2021

20. Gasparrini, S., Cippitelli, E., Spinsante, S., Gambi, E.: A depth-based fall detection system using a Kinect® sensor. *Sensors* **14**(2), 2756–2775 (2014)
21. Martínez-Villaseñor, L., Ponce, H., Espinosa-Loera, R.A.: Multimodal database for human activity recognition and fall detection. In: Proceedings of the 12th International Conference on Ubiquitous Computing and Ambient Intelligence (UCAmI 2018), vol. 2, no. 19 (2018)
22. Saleh, M., Abbas, M., Le Jeannes, R.B.: FallAIID: an open dataset of human falls and activities of daily living for classical and deep learning applications. *IEEE Sens. J.* **21**(2), 1849–1858 (2021)
23. Frank, K., Vera Nadales, M.J., Robertson, P., Pfeifer, T.: Bayesian recognition of motion related activities with inertial sensors. In: Proceedings of the 12th ACM International Conference on Ubiquitous Computing, pp. 445–446 (2010)
24. Kwolek, B., Kepski, M.: Human fall detection on embedded platform using depth maps and wireless accelerometer. *Comput. Methods Programs Biomed.* **117**(3), 489–501 (2014)
25. Klenk, J., et al.: The FARSEEING real-world fall repository: a large-scale collaborative database to collect and share sensor signals from real-world falls. *Eur. Rev. Aging Phys. Act.* **13**(1), 8, December 2016
26. Ahmed, M., Mehmood, N., Nadeem, A., Mehmood, A., Rizwan, K.: Fall detection system for the elderly based on the classification of shimmer sensor prototype data. *Healthc. Inform. Res.* **23**(3), 147–158 (2017)
27. Casilari, E., Santoyo-Ramón, J.A., Cano-García, J.M.: On the heterogeneity of existing repositories of movements intended for the evaluation of fall detection systems. *J. Healthc. Eng.* **2020**, 6622285 (2020)
28. sklearn.preprocessing.scale—scikit-learn 0.24.1 documentation. <https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.scale.html>. Accessed 19 Apr 2021
29. Version 0.23.2—scikit-learn 0.24.1 documentation. https://scikit-learn.org/stable/whats_new/v0.23.html#version-0-23-1. Accessed 16 Apr 2021
30. sklearn.model_selection.GridSearchCV—scikit-learn 0.24.1 documentation. https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.GridSearchCV.html. Accessed 16 Apr 2021
31. Clasificación: ROC y AUC | Curso intensivo de aprendizaje automático. <https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc?hl=es-419>. Accessed 19 Apr 2021