Progress in Artificial Intelligence manuscript No.
(will be inserted by the editor)

User-Centered fall detection using supervised, on-line learning and transfer learning.

José R. Villar · Enrique de la Cal · Mirko Fañez · Víctor M. González · Javier Sedano

Received: date / Accepted: date

Abstract There is no need to explain the relevance of Fall Detection (FD) in the elderly population: the faster they receive help, the higher the probabilities of recovery. FD has been widely studied and a number of solutions have been proposed in the literature, including commercial products with well-known trademarks. Basically, the majority of the solutions include an event detection method followed by a feature extraction block as the inputs to a classifier that labels the sample as Fall or as Not Fall. Nevertheless, the proposals always rely on user feedback because of the relatively high percentage of false positives. This study proposes two supervised user-centered solutions for FD which are

This research has been funded by the Spanish Ministry of Science and Innovation, under project MINECO-TIN2017-84804-R.

J. R. Villar
Computer Science Department
University of Oviedo (SPAIN)
Tel.: +34-985182597
Fax: +34-985181986
E-mail: villarjose@uniovi.es

E. de la Cal
Computer Science Department
University of Oviedo (SPAIN)

M. Fañez
Computer Science Department
University of Oviedo (SPAIN)

V. M. Gonález
Automatica Department
University of Oviedo (SPAIN)

J. Sedano
Instituto Tecnológico de Castilla y León (SPAIN)

evaluated using two publicly available data sets that include simulated falls with the sensory system placed on a wrist. Instead of focusing on learning a generalized model, in this study we focus on learning specific models for each user. To overcome the lack of data from the Fall class, an online-learning method to group the events arising from the activities of daily life and the consequent analysis of how fall events can then be detected is included. In addition, a transfer learning stage produces a priori knowledge from previous experiences that can be introduced into the on-line learning classifier in order to enhance its performance. Complete experimentation is carried out on two publicly available data sets, analyzing the performance of the supervised classifiers and the performance of the whole detection system composed of the fall detection plus the classifier. In the case of on-line learning, only the results for the classifier, either with or without transfer learning, are included. Results suggest that FD user-centered solutions have the capacity to adapt to the information of a specific user, mostly when using both on-line learning and transfer learning. Nevertheless, the transfer learning stage needs refinements in order to avoid the complexity that this might introduce.

Keywords Fall Detection · Smart Wearables · Applied Machine Learning

1 Introduction

Fall Detection is a very challenging research topic that has attracted attention for several years. Different solutions have been proposed in the literature, each of them focused on a specific population. Perhaps one of the most difficult populations is that of the elderly: in this case, the amount of movement is smaller and the activities are weaker in general (for instance, elderly people walk slowly, arm movement range is shorter) [13,41,40]. And most importantly, the impact and consequences of delayed assistance on the health of an elderly person after a fall are much more severe. Besides, there is no clear solution and even commercial devices for the elderly perform with a relatively high false positive rate [33].

A wide range of FD solutions have been published in the literature; the technologies involved vary from video analysis to wearable sensors, including smart sensory systems in smart homes [10,11,19,25]. One of these technologies is the use of wearable devices in FD, which represents the starting point of this research. The focus of this research is to develop an on-wrist wearable FD solution that helps senior citizens to continue living autonomously.

The majority of published studies on wearable FD focus on either the intelligent technique (a combination of pre-processing and modelling techniques) or on the design of a sensor or sensor network, or on both of these. Basically, these solutions include an event detection stage (responsible for extracting the corresponding set of features) followed by a Machine Learning (ML) stage; this ML stage is responsible for obtaining the classification model. Alternatively, some methods apply a ML method directly to each sliding window without any event detection.

The most common sensor used in FD is the tri-axial accelerometer (3DACC), used independently or combined with other sensors (like gyroscopes or inertial sensors), or the combination of several sensors. The sensory system is located mainly on the waist or on a wrist, and in some cases on a thigh.

Besides, there are two confronted approaches: the first one focuses on the very specific user, while the second focuses on learning generalized models. We refer as *generalized* models those obtained with all the available data, independently from which participant the data comes from. In this case, data from the different sources are introduced in an unique dataset containing the TS for both the ADLs and the fall events. The participant for which the TS was gathered is irrelevant. The data set is organized in instances that includes a TS for an ADL or Fall and the corresponding label; cross-validation is performed on this data set by shuffling the instances among the folds.

On the other hand, we refer as *user-centered* those approaches that learn a specific model for each participant. For this purposes, only data gathered for the current user is used to learn the model; the data gathered from other different users are either ignored or, as in this study, used to enhance the obtained user specific models.

From now on, when we refer a solution as generalized, we are saying that the approach obtains a single generlized model for all the users. Alternitevely, user-centered models are those that has been learned with data from the current user and are not used for FD with any other user.

We have summarized the related work on Table **??**, specifying the sensory system and its location, the event detection method proposed in the study and the ML method. The following acronyms apply: Finite State Machine (FSM), threshold (TH), Neural Network (NN), Rule set (RS), K-Nearest Neighbour (kNN), Decision Trees (DT), Discriminant Analysis (DA), Support Vector Machine (SVM), One-class SVM (OSVM), Classification and Regression Trees (CART), Logistic Regression (LR). I-P refers to the proposal shown in [24].

This study explores both supervised and on-line learning user-centered solutions for FD. On the one hand, in the supervised solution, two different ML methods are used: i) K-Nearest Neighbour classifier and ii) Symbolic Aggregation approXimation (SAX) for Time Series (TS) representation together with a TF-IDF bag of words classifier. On the other hand, the on-line learning solution makes use of the SAX TS representation and the TF-IDF bag of words that is learned from the data whenever a peak is detected and using feedback from the user. The different approaches have been evaluated using two well-known publicly available data sets that include 3DACC data from either simulated falls or Activities of Daily Living (ADL).

This study is organized as follows. This introductory section is devoted to describing the problem and related work. The next Section focuses on the proposal, detailing the different stages included. Section 3 gives details of the experimentation performed in this research, while Section 4 shows the obtained results and the discussion. The study ends with conclusions drawn.

Table 1 Summary of fall detection approaches in the literature. Ref: the reference, Sensor: type of the sensor, SP: sensor placement, ED: event detection method, MM: modelling method, G/U: generalized versus user-centered modelling (I).

Ref	**Sensor**	**SP**	**ED**	**MM**	**G/U**
[1]	3DACC	Waist	FSM	NN	Generalized
[3]	3DACC + Air pressure	Wrist	TH + time	If-Then rules	Generalized
[4]	3DACC	Wrist + Thigh	TH	If-Then rules	Generalized
[6]	3DACC + Gyroscope	Waist	TH	C4.5	Generalized
[7]	3DACC + Gyroscope	Wrist + Pocket	TH	RS	User-Centered
[12]	3DACC	Wrist	TH	NN	Generalized
[16]	3DACC	Wrist	TH	RS	Generalized
[17]	3DACC	Wrist	TH	kNN, DT SVM, DA	Generalized
[18]	3DACC + Gyroscope	Wrist	TH	If-Then rules	Generalized
[20]	3DACC	Waist or Pocket		NN or SVM	Generalized
[22]	3DACC + Gyroscope + Barometer	Waist	TH	SVM	Generalized
[23]	3DACC	Waist	TH	RS	Generalized
[24]	3DACC	Waist	TH	RS	Generalized
[26]	3DACC	Wrist	TH	NN, SVM DT, RBS	Generalized
[27]	3DACC	Wrist	TH	RS	Generalized
[29]	3DACC	Pocket	-	NN, SVM kNN	Generalized + model fitting
[30]	3DACC	Wrist	-	SVM kNN	Generalized
[32]	3DACC	Chest + Thigh	FSM	SVM, LR, kNN, CART, I+P	Generalized
[34]	3DACC + Gyroscope + Barometer	Wrist	-	RS	Generalized
[36]	3DACC + Gyroscope	Waist + Ankle	-	RS	Generalized
[38]	3DACC	Wrist Waist	FSM + TH	kNN	Generalized
[42]	3DACC	Wrist	-	RS	Generalized
[43]	3DACC	Waist	TH	OSVM	Generalized

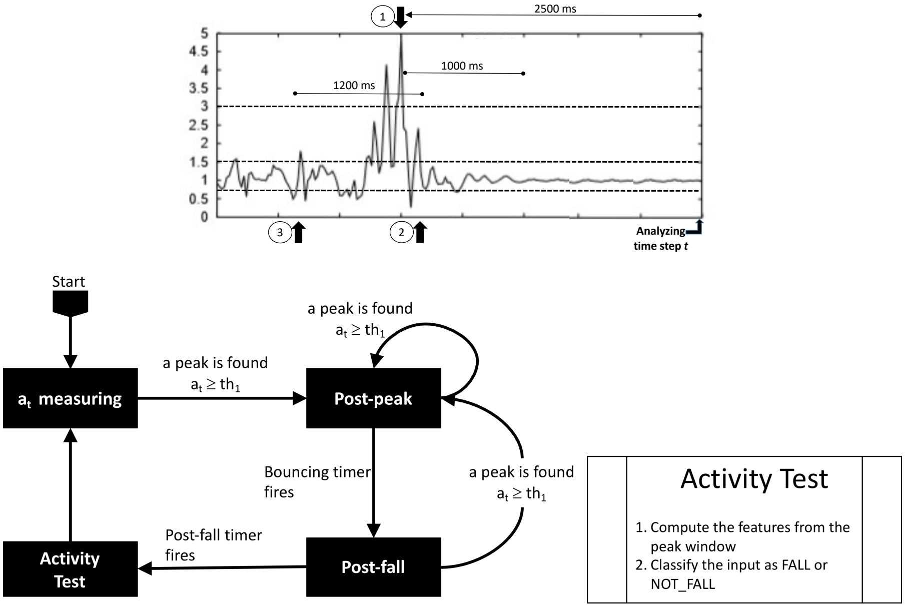

Fig. 1 Fall event detection. The upper part shows the dynamics of a fall event measured with the magnitude of the acceleration (see the text for a description). The lower left section depicts the finite state machine, where the bouncing and post-fall timers are set to 1000 and 1500 ms, respectively. The lower right section includes the block diagram; the features are computed and the extracted sample is classified whenever a peak is detected.

2 Supervised and On-line learning Fall Detection

As mentioned before, the most common approach to FD includes an Event Detection stage. In this study, we continue with our previous research [26], so the Event Detection proposed in [1], with some minor modifications, applies. This section firstly outlines this Event Detection stage; then, it gives details of the two supervised solutions. Finally, the section ends by introducing the on-line learning solution and the transfer learning stage.

2.1 Fall event detection

As proposed in [1,26], peaks in acceleration magnitude are the basis for fall detection. Figure 1 depicts the very simple way the finite state machine detects the possible peaks. The data gathered from a 3DACC located on the wrist is processed using a sliding window. Once a peak is found, several features are extracted from the data within the sliding window for which a peak was detected. The second stage includes a classifier that labels each sample as FALL or NOT_FALL. In [26] it was claimed that the lower the computational cost of the classifier the better, as it must be run on the wearable device.

The feature extraction is executed whenever a peak is detected and follows the dynamics within a fall -refer to the upper part in Fig. 1-. Let a_t be the acceleration magnitude at time t calculated following Equation 1, where a_{tx}, a_{ty} and a_{tz} are the components of the acceleration in each of the three axes. Let us assume that gravity be $g = 9.8 m/s^2$. Given the current time stamp t, we find a peak at **peak time** $pt = t - 2500$ ms (point 1) if at time pt the magnitude of the acceleration a_t is higher than $th_1 = 3 \times g$ and there is no other peak in the period $(t - 2500, t]$ ms (no other a value higher than th_1). If this condition holds, then it is stated that a peak occurred at pt.

$$a_t = \sqrt{a_{tx}^2 + a_{ty}^2 + a_{tz}^2} \tag{1}$$

Once pt has been determined, we have to define the start and the end of the fall event, called **impact-start** and **impact-end** respectively. The **impact-end** (ie, corresponding to point 2 in the upper part in Fig. 1) is the time from pt for which $a_t \geq th_2$ (with $th_2 = 1.5 \times g$), with a maximum of 1000 ms. If no impact-end is found, then it is fixed to $pt + 1000$ ms. Finally, the **impact-start** (is) (point 3) denotes the starting time of the fall event, just before pt; is is computed as the time of the first sequence of an $a_t <= th_3$ ($th_3 = 0.8 \times g$) followed by a value of $a_t >= th_2$. The impact-start must belong to the interval $[ie - 1200, pt]$ in ms. If no impact-start is found, it is fixed to pt. From now on, without losing generalization, as long as we know the sampling frequency, we can refer to time stamp or to positions within a sliding window that include the samples in $[is, ie]$. We call impact window the sequence of values from is to ie. All these time stamps can be absolute, but in practice they are relative to the current time stamp.

When using subindex i we refer to the sample position within the sliding window, and when using subindex t we refer to a time stamp; however, they are interchangeable. When a peak is detected the feature extraction is performed, computing several parameters and features for this peak time.

After these three times -is, pt, ie- have been calculated, the following transformations should be computed:

- Average Absolute Acceleration Magnitude Variation (AAMV), calculated as $AAMV = \sum_{t=is}^{ie} \frac{|a_{t+1} - a_t|}{N}$, with N the number of samples in the interval.
- Impact Duration Index, $IDI = ie - is$.
- Maximum Peak Index, $MPI = max_{t \in [is, ie]}(a_t)$.
- Minimum Valley Index, $MVI = min_{t \in [is-500, ie]}(a_t)$.
- Peak Duration Index, $PDI = pe - ps$, with ps the peak start defined as the time of the last magnitude sample below $th_{PDI} = 1.8 \times g$ occurred before pt, and pe, the peak end defined as the time of the first magnitude sample below $th_{PDI} = 1.8 \times g$ occurred after pt.
- Activity Ratio Index, ARI, calculated as the ratio between the number of samples not in $[th_{ARIlow} = 0.85 \times g, th_{ARIhigh} = 1.3 \times g]$ and the total number of samples in the 700 ms interval centered in $(is + ie)/2$.
- Free Fall Index, FFI, the average of the acceleration magnitude in the interval $[t_{FFI}, pt]$. The value of t_{FFI} is the time between the first acceler-

ation magnitude below $th_{FFI} = 0.8 \times g$ occurring up to 200 ms before pt; if not found, it is set to $pt - 200\ ms$.
- Step Count Index, SCI, measured as the number of peaks in the interval $[pt - 2200, pt]$. SCI is the step count evaluated 2200 ms before peak time. The number of valleys are counted, defining a valley as a region with acceleration magnitude below $th_{SCIlow} = 1 \times g$ for at least 80 ms, followed by a magnitude higher than $th_{SCIhigh} = 1.6 \times g$ during the next 200 ms. Some ideas on computing the time between peaks [39] were used when implementing this feature.

Moreover, when working with normalized TS, the SAX representation [28] of the acceleration magnitude TS for the period from the impart-start to the impart-end is also determined for each detected peak. The normalization of the TS window is performed using the mean and standard deviation of the data within the window.

As stated in the Introduction section of this study, several solutions for FD detect peaks using thresholds and extra processing [2,4,5,6]. The solution proposed in [1] is not different. Furthermore, the solution in [1] makes use of a rather long list of thresholds, some of them defined in the domain of the acceleration and others in the time domain. All of them have been set up after a study of the dynamics of a fall event, thus, we continue using them. However, as will be shown later in this study, the thresholds for the normalized time series that will be employed in peak detection for the approaches using SAX should be different.

In this study, two TS representations are used: raw data when developing the solution based on previous research [1,26], and the SAX TS representation. In each case, a different set of thresholds should be employed for peak detection. From now on, the following notation is used to refer to the TS representation and the threshold set:

- **origTSorigTH**: using both raw TS and the original thresholds proposed in [1]. When modelling, the mean and standard deviation are calculated over all of the ADL TS; these statistics are used to normalize both the TS and the acceleration-based thresholds.
- **normTSnormTH**: using both normalized TS and normalized thresholds. The mean and standard deviation are calculated over all of the ADL TS; then, they are used to normalize both the TS and the acceleration-based thresholds.
- **normTSscldTH**: using normalized TS and scaled thresholds. In this case, the peak threshold th_p^{norm} is determined as a percentage of the maximum peak value for any fall in the data set (more specifically, $0.9 \times \min(\max_{i \in FALL}(TS_i)))$; the remaining acceleration-based thresholds are obtained by scaling the thresholds in [1] using the peak threshold as reference: $th_i^{norm} = th_i^{Abbate} * th_p^{norm} / th_p^{Abbate}$.

2.2 Fall detection by means of supervised methods

This part of the study assumes a two-class problem, labelling each TS as FALL or as NOT FALL. Each participant from a data set includes a set of labelled TS, though the number of TS per participant is assumed to be small.

As stated before, each TS is analyzed to detect peaks; if a peak is detected then the set of features and the SAX representation of the impact window are determined. By gathering the information from these peak events a new extracted data set is obtained, where each sample comes from each detected peak and is labelled with the corresponding TS label. A model can be learned using this newly created data set.

In this study, we propose two different alternatives. On the one hand, we propose the K-Nearest Neighbour (KNN) because of its simplicity and also because the number of samples is small. Because the dimension of the participant's data set is small, only short values of K can be used (that is, values 1 or 3). In a preliminary study, we found that the best results were obtained when K is set to 1, despite the well-known noisy behaviour this configuration might suffer.

On the other hand, we propose the use of SAX [28] TS representation together with the Term Frequency - Inverse Document Frequency (TF-IDF) statistic to classify each peak. It is worth noticing that the normalization of the training and testing (or deploying) TS is performed using the mean and standard deviation from the ADLs TS.

Once a peak is detected in a TS, then the SAX word is determined from the impact window. Considering a TS belonging to the training data set, each detected peak generates a SAX word; this SAX word is used to create a bag of words for each class FALL and NOT FALL, including the corresponding counters. This bag of words is used to compute the TF-IDF which is used to classify an incoming sample. We follow the TF-IDF calculations included in [35]. The term frequency is measured as the word frequency for the label, while the inverse document frequency is calculated as $log_{10}(word, LABELS) = \frac{|LABELS|}{|d \in LABELS : word \in d|}$, where $LABELS = \{FALL, NOTFALL\}$.

To classify an incoming sample we use two stages. If the SAX word w_i for this incoming sample is in the dictionary, then it is labelled with the label with a higher value of TF-IDF. Otherwise, the similarity of the w_i with each of the words in the dictionary is calculated using the SAX min_dist. The min_dist function is defined together with SAX in [28]; it is implemented as a look-up table, the values of each cell in this table are either 0 for adjacent symbols or a value that is determined using the number of symbols and the equi-probability of all the symbols using a Gaussian distribution. The TF-IDF values for the words with maximum similarity are aggregated using the max function. Finally, the label with a higher TF-IDF value is proposed as the label for the incoming sample.

2.3 On-line learning and transfer learning fall detection

The user-centered supervised methods, including those presented above, suffer from a very important drawback: the data set rarely includes a real fall TS. Using publicly available data sets can help with this issue, but the main part of these includes simulated falls (that is, the participants fall down from a static standing-still position). Moreover, the differences between ADLs and falls decrease with age: the movements of the elderly are weaker and the falls are mainly due to fainting or loss of balance [13,41,40]. In many cases, the subject tries to grab a handrail or even a wall so as to avoid falling; this staggering may mask the drastic changes in acceleration that a normal fall shows. Obviously, it is clear that several different event detection solutions would be needed in order to solve the FD problem, at least for the elderly. Furthermore, there is a problem in simulating falls as these simulations might be biased or may represent a risk to the integrity of the user even when mattresses are used -for instance, it is not plausible to simulate falls with senior citizens. This fact might be one of the main reasons why user-centered supervised learning approaches have not been addressed in the literature.

In order to cope with this lack of positive class data, we propose using the SAX TS representation together with the TF-IDF statistic in a learning scheme that makes use of transfer learning and an on-line learning scheme. On the one hand, transfer learning allows deployment of the general knowledge that might have been extracted from real fall data gathered from the focused population. On the other hand, the on-line learning scheme allows us to obtain immediate feedback about any false positive alarm (and for true positive alarms as well when an alarm is raised and there is no feedback), so the bag of words information and the TF-IDF matrix can be updated.

The SAX representation and the TF-IDF solution is similar to that described in the previous stage. Let us assume there is a period of initialization, where data from ADLs is gathered. The mean and the standard deviation of the data are calculated with the first subset of these data, which would allow normalizing the TS. From this point, the normalized TS from ADLs are analyzed; whenever a peak is detected, the SAX word is computed and introduced into the bag of words as belonging to the negative class (NOT FALL). Once the initialization stage ends, the bag of words is used to generate the TF-IDF matrix, storing both matrices for later use.

In the deployment stage, whenever a peak is detected the TF-IDF statistics are used to determine if the word belongs to one of the labels. In this on-line learning scheme, a word $w_{current}$ from the currently detected peak should be contained in the bag of words in order to determine the most plausible class. To measure the similarity we propose using the min_dist distance proposed for SAX; those words in the bag that have a distance smaller than a threshold th_{sim} are used to determine the label for the current peak. The sum has been used to aggregate the TF-IDF statistics among the similar words in the dictionary; nevertheless, the best operator for this aggregation needs further study. Besides, if the word is not in the dictionary, it is assumed to be a possible FALL event. In any case, there is log information about the words and the proposed labels. This log allows the user to mark any of the past peaks with the real label between FALL or NOT FALL. With every detected peak or whenever feedback is given, the bag of words and the TF-IDF statistics are recalculated. In this research, the value of th_{sim} was set to 5.0, although more research is needed to optimize this threshold.

The transfer learning stage is performed as follows. Let's assume that there are TS available from participants from the same population. This data can be used to enhance the user-centered solution. These TS can be labelled with one of the possible classes {FALL, NOT FALL}. Firstly, the bag of words is determined for each participant using the SAX representation of his/her TS. Secondly, the words that are common to a high percentage of the participants are chosen as the bag of words to be transferred to each new user as the consequence of the transfer learning stage. The counters of a word for each label are set as the average of the counters for this word among all the participants that used that word.

Finally, it is necessary to determine whether a word w is common or not. Let n_w be the number of participants for which w appeared as the SAX representation of an impact window, and let $max_n = \max_{\forall w}(n_w)$. Then a word is common if the ratio $n_w/max_n \geq th_{common}$ and, thus, the word w is included in the bag of words to be transferred. In this study the value of this threshold has been set to $th_{common} = 0.7$ but, again, further experiments are needed to optimize this value.

3 Material and methods

3.1 Data sets

In a recent study [9] up to twelve publicly available data sets related to FD and ADL were compared; these data sets have the common characteristic of using 3DACC sensors located on different body parts. Recently, a new data set has also been published in [37]. In this study we have chosen two of these TS data sets, the UMA Fall and the TST (details are given in Table 2; these data sets were chosen because they use a 3DACC sensor placed on a wrist and also because their design was different. In the case of UMA Fall, there is neither a common number of activities and fall simulations nor the same number of repetitions, while in TST all the participants performed the same activities, more or less the same number of repetitions. Furthermore, the performance of the ADLs for the UMA Fall is stronger than for TST.

Because of the variability in the number of TS associated with the participants in each of the data sets, we decided to only include in the experiments those participants for whom there are enough instances from both classes. Therefore, we ignore any participant from a data set with less than 20 TS (labelled with one of the different activities or fall types) or with less than 9 TS labelled as a FALL.

Table 2 Descriptions of the different data sets used in this research. Columns NP, NTS and NF stand for the number of participants, the number of available TS, and the number of falls in the data set, respectively. The sampling frequency in Hz is shown in the corresponding Fqcy column.

Dataset	**NP**	**NTS**	**NF**	**Fqcy**	**Description**
UMA Fall [8]	17	531	208	20	Includes forward, backward and lateral falls, running, hopping, walking and sitting. Neither do all the participants have every type of activity nor the same number of repetitions. Sensors on the wrist, waist, ankle, chest and in the trouser pocket.
TST FD [15]	11	264	132	100	Includes forward, backward and lateral falls, with two sensors, one on the waist and one on the right wrist.

As mentioned in Section 2.1, three different combinations of scaling of both the TS and thresholds have been defined for this study: using the original Abbate TS representation and thresholds (origTSorigTH), using normalized TS and normalized thresholds (normTSnormTH) and using normalized TS and proportional thresholds (normTSscldTH).

For this study, one repetition of 5x2 cross-validation on the TS data set was used as the cross-validation scheme. This means that the TS data set is shifted and split in two (one for training and one for testing) ten times, trying to keep both splits balanced.

When analyzing the on-line learning approaches, the TF-IDF statistic for each participant and fold was generated using the TS from ADLs included in the training fold. When transfer learning was used, the transferred bag of words was extracted with the TS from all the individuals but the one that is being trained. All the code has been developed with the R project [14] using the RStudio [21].

3.2 Performance measurements

A peak can be detected in a TS; each detected peak generates a sample that should be classified. Besides, a TS can generate more than one peak; in the case of a fall TS, only one peak corresponds to the fall event, while the others are false alarms. With all of these in mind, we propose measuring the performance at the peak detection level and at the TS level.

To measure the peak detection's performance, each detected peak is given its real label. For instance, if a fall TS includes several peaks in the acceleration magnitude, only the peak that generated the fall is given the FALL label. Nevertheless, this problem did not appear for the publicly available data sets used in this study, and the fall TS only generated the peak that corresponded to the fall. Therefore, the performance was measured using only the detected peaks and how well each of the ML methods classified the peak.

On the other hand, to measure the performance at the TS level we did as follows. For each TS, if any of its detected peaks is classified as a FALL, then the TS is classified as a FALL; otherwise, the TS is labelled as NOT FALL. Whenever no peak is detected for a TS, then this TS is classified as NOT FALL.

We used classical Accuracy, Kappa factor, Sensitivity, Specificity, and F1 measurements. However, only the Accuracy and Sensitivity are shown in the result tables for the sake of simplicity. Furthermore, the aggregate results will be shown per data set, that is, the mean performance for each participant and data set.

4 Results and discussion

4.1 Results for the supervised experimentation

The results for the supervised experiments are shown in several tables and boxplots. From now on, we use TP, FN, FP and TN as the true positive, false negative, false positive and true negative counters, respectively. On the one hand, the performance of the classifiers is presented. For these results, only those TS belonging to the test folds for which a peak was detected are considered. The results of the performance of the models includes:

- the aggregated results of the confusion matrix of each participant and data set (Table 3),
- the mean and standard deviation of Accuracy and Sensitivity among the fold for each participant from the TST data set (Table 5),
- the mean and standard deviation of Accuracy and Sensitivity among the fold for each participant from the UMA Fall data set (Table 6),
- and two boxplots for participant 1 from the UMA Fall data set, showing Accuracy (Fig. 2) and Sensitivity (Fig. 3). This participant was neither the best nor the worst, but representative of the mean performance; this is the reason why it was chosen.

On the other hand, the performance of the fall detection system (including peak detection and the classifier) is shown in several tables. For these results, all the TS are considered, if no peak is detected then the TS is labelled as NOT FALL. The results are shown with:

- the aggregated results of the confusion matrix of each participant and data set (Table 4),
- the mean and standard deviation of the Accuracy and Sensitivity among the fold for each participant from the TST data set (Table 7),
- and the mean and standard deviation of the Accuracy and Sensitivity among the fold for each participant from the UMA Fall data set (Table 8).

From these results it can be seen that none of the classifiers outperform the others. This is the reason why no remark was introduced in the tables: for some participants, the KNN is slightly better than the TF-IDF; for others, just the opposite happened. Besides, there is a clear correlation in the results of both classifiers: if classifier A performance improves/worsens for a participant, the same happens with classifier B, independently of which one is A or B between KNN and TF-IDF.

Nevertheless, the variability in the results from one participant to another may be due to several reasons. Firstly, the data sets include data from participants, each participant simulated the fall event as fair as possible. However, falling is not easy: some participants might have been afraid or did not know how to fall. This means that the different fall events are not always similar, and thus, the erratic behaviour of the classifiers.

Secondly, this study proposes user-centered training; the consequences are that the amount of data for training is reduced. The obtained results suggest this or that the number of training TS be incremented or that more techniques are needed so the performance can be enhanced.

In spite of the variability in the results from one participant to another, the performance of both methods are remarkably good given the reduced amount of training data. The results were, for most of the individuals, better than reported in commercial devices. Although the results can not be compared, the accuracy and sensitivity obtained for the main part of the population are better than those obtained with generalized solutions [26].

Normalizing both the TS and the thresholds (scenario normTSnormTH) showed as a more robust solution (the best results were obtained for this scenario with the KNN and the TF-IDF), especially for the UMA FAll dataset. The normTSscldTH scenario (with normalized TS and scaled thresholds), though in some cases performed rather well, was undoubtedly the worse if we consider the UMA Fall dataset. The original approach (origTSorigTH, the raw TS and the original thresholds) is close to the normTSnormTH, but in more cases with slightly worse performance.

Besides, it is to be noted how clearly the nature of the data sets has been shown. On the one hand, we have the structure of the samples; on the other, the ADLs included in the data set. Concerning the structure of the samples, the TST data set includes the same number of repetitions of the same type of ADLs and falls for all the participants. This fact is interesting as long as all the participants have enough data to be included in the experimentation. In turn, the UMA Fall data set does not include the same number of activities and repetitions -even the fall simulations- for all the individuals. This fact forced us to dismiss the data from those participants with less than 20 TS and that included less than 9 simulated falls.

It is worth noticing the differences in the strength of the acceleration signals recorded for the ADLs in both data sets. In the case of the TST data set, the activities show small changes in acceleration, meaning the ADLs were weaker than in UMA Fall. For some participants in the TST data set, using peak detection was almost enough to classify all the TS! However, the UMA Fall included high-intensity ADLs; this data set is much more challenging than the TST in terms of FD. Consequently, the results for the UMA Fall differ from those obtained with the TST.

4.2 Results from the on-line learning experiment

For the sake of simplicity, only the performance of the classifiers is included in this section as long as the whole system (peak detection and classifier) only leads to better results (all the TS that do not generate a peak are classified as NOT FALL, which is mainly True for the types of falls considered in this study). The classifier results consider only those TS belonging to the test folds for which a peak was detected, as these are the samples evaluated by the classifier. The results of the performance of the models includes:

- the aggregated results of the confusion matrix of each participant and data set (Table 9),
- the mean and standard deviation of Accuracy and Sensitivity among the folds for each participant from the TST data set (Table 10),
- and the mean and standard deviation of Accuracy and Sensitivity among the fold for each participant from the UMA Fall data set (Table 11).

The results clearly show that i) the setting of the classifier to label as FALL by default introduces benefits and errors, and ii) the transfer learning partially enhances the performance of the classifier. Let us develop these points.

As explained above, the on-line learning method classifies an incoming impact window as FALL whenever no similar SAX word is found in the bag of words. Labelling in this way allows the setting of a signal to the user so he/she can give feedback to the system. This setting favors the classifier, as the greater part of the detected peaks belong to the label FALL. To diminish this effect the th_{common} was introduced, allowing a more relaxed matching between the current SAX word and the words in the bag of words. The consequences for the TF-IDF without transfer learning is that there are too many False Negatives due to the similarity with NOT FALL words. However, the consequences for the TF-IDF is the higher ratio of False Positives. On the other hand, transfer learning seems to improve the performance of the classifier as it proposes the alarm for 100% of cases. Unfortunately, this good performance in fall detection is penalized with a high number of False Positives.

With these facts, we think the transfer learning TF-IDF is an interesting solution but needs refinement. One of these possible refinements is to introduce

Table 3 Supervised Results. Evaluation of the classifiers. Aggregation of the results from the confusion matrix for all the scenarios and methods. partID stands for participant Identification number within each data set.

		normTSnormTH: normalized TS and normalized thresholds							
		KNN				TF-IDF			
source	parID	TN	TP	FP	FN	TN	TP	FP	FN
TST	1	0	47	3	6	0	53	3	0
TST	2	0	57	0	3	0	60	0	0
TST	3	9	50	10	10	6	57	13	3
TST	4	0	60	4	0	0	43	4	17
TST	5	7	41	20	9	16	43	11	7
TST	6	0	55	0	0	0	55	0	0
TST	7	0	60	0	0	0	60	0	0
TST	8	0	63	3	4	0	66	3	1
TST	9	0	52	0	0	0	52	0	0
TST	10	0	57	7	3	0	52	7	8
TST	11	0	60	0	0	0	60	0	0
UMA Fall	1	14	85	16	10	16	93	14	2
UMA Fall	2	0	58	0	2	0	60	0	0
UMA Fall	3	20	76	23	14	33	73	10	17
UMA Fall	4	3	71	4	3	0	70	7	4
UMA Fall	9	0	82	3	11	0	93	3	0
UMA Fall	12	5	23	10	8	5	29	10	2
UMA Fall	15	2	41	12	4	2	45	12	0
UMA Fall	16	26	270	13	2	7	263	32	9
UMA Fall	17	0	57	16	6	3	54	13	9

		normTSscldTH: normalized TS and scaled thresholds							
		KNN				TF-IDF			
source	parID	TN	TP	FP	FN	TN	TP	FP	FN
TST	1	4	44	2	16	4	60	2	0
TST	2	0	56	0	4	0	60	0	0
TST	3	0	57	7	3	0	58	7	2
TST	4	0	54	0	6	0	60	0	0
TST	5	6	50	26	10	23	54	9	6
TST	6	4	55	3	5	0	54	7	6
TST	7	0	60	0	0	0	60	0	0
TST	8	0	67	0	0	0	67	0	0
TST	9	0	56	0	4	0	60	0	0
TST	10	0	53	7	7	0	57	7	3
TST	11	0	60	0	0	0	60	0	0
UMA Fall	1	20	93	10	7	8	91	22	9
UMA Fall	2	0	56	0	4	0	60	0	0
UMA Fall	3	1	85	8	5	1	88	8	2
UMA Fall	4	3	74	4	0	0	72	7	2
UMA Fall	9	6	91	16	2	5	90	17	3
UMA Fall	12	5	31	15	9	10	29	14	11
UMA Fall	15	3	45	11	5	0	50	14	0
UMA Fall	16	35	277	13	9	16	258	32	28
UMA Fall	9	12	74	18	7	8	56	22	25

		origTSorigTH: raw TS and original thresholds							
		KNN				TF-IDF			
source	parID	TN	TP	FP	FN	TN	TP	FP	FN
TST	1	2	45	1	8	0	53	3	0
TST	2	0	60	0	0	0	60	0	0
TST	3	6	57	13	3	2	59	17	1
TST	4	0	58	4	2	0	45	4	15
TST	5	8	37	19	13	9	43	18	7
TST	6	0	55	0	0	0	55	0	0
TST	7	0	54	0	6	0	60	0	0
TST	8	0	66	3	1	0	65	3	2
TST	9	0	47	0	5	0	52	0	0
TST	10	0	56	7	4	0	59	7	1
TST	11	0	60	0	0	0	60	0	0
UMA Fall	1	18	88	12	7	8	93	22	2
UMA Fall	2	0	57	0	3	0	60	0	0
UMA Fall	3	19	82	24	8	20	82	23	8
UMA Fall	4	5	71	2	3	0	74	7	0
UMA Fall	9	0	83	3	10	0	88	3	5
UMA Fall	12	5	29	10	2	5	31	10	0
UMA Fall	15	6	39	8	6	1	45	13	0
UMA Fall	16	24	269	15	3	4	269	35	3
UMA Fall	17	0	58	16	5	0	63	16	0

Table 4 Supervised Results. Evaluation of peak detection and each of the classifiers together. Figures come from the sum of the results from the confusion matrix for all the scenarios and methods. Each row shows the results for a participant from one of the data sets. partID stands for participant Identification number within each data set.

		normTSnormTH: normalized TS and normalized thresholds							
		KNN				TF-IDF			
source	parID	TN	TP	FP	FN	TN	TP	FP	FN
TST	1	57	47	3	13	57	53	3	7
TST	2	60	57	0	3	60	60	0	0
TST	3	50	50	10	10	47	57	13	3
TST	4	56	60	4	0	56	43	4	17
TST	5	40	41	20	19	49	43	11	17
TST	6	60	55	0	5	60	55	0	5
TST	7	60	60	0	0	60	60	0	0
TST	8	57	56	3	4	57	59	3	1
TST	9	60	52	0	8	60	52	0	8
TST	10	53	57	7	3	53	52	7	8
TST	11	60	60	0	0	60	60	0	0
UMA Fall	1	74	85	16	15	76	93	14	7
UMA Fall	2	90	58	0	2	90	60	0	0
UMA Fall	3	67	76	23	14	80	73	10	17
UMA Fall	4	96	71	4	9	93	70	7	10
UMA Fall	9	87	81	3	9	87	90	3	0
UMA Fall	12	100	23	10	17	100	29	10	11
UMA Fall	15	38	41	12	9	38	45	12	5
UMA Fall	16	307	270	13	10	293	263	27	17
UMA Fall	17	74	57	16	33	77	54	13	36

		normTSscldTH: normalized TS and scaled thresholds							
		KNN				TF-IDF			
source	parID	TN	TP	FP	FN	TN	TP	FP	FN
TST	1	58	44	2	16	58	60	2	0
TST	2	60	56	0	4	60	60	0	0
TST	3	53	57	7	3	53	58	7	2
TST	4	60	54	0	6	60	60	0	0
TST	5	38	50	22	10	52	54	8	6
TST	6	57	55	3	5	53	54	7	6
TST	7	60	60	0	0	60	60	0	0
TST	8	60	60	0	0	60	60	0	0
TST	9	60	56	0	4	60	60	0	0
TST	10	53	53	7	7	53	57	7	3
TST	11	60	60	0	0	60	60	0	0
UMA Fall	1	80	93	10	7	68	91	22	9
UMA Fall	2	90	56	0	4	90	60	0	0
UMA Fall	3	82	85	8	5	82	88	8	2
UMA Fall	4	96	74	4	6	93	72	7	8
UMA Fall	9	78	88	12	2	78	88	12	2
UMA Fall	12	99	31	11	9	99	29	11	11
UMA Fall	15	39	45	11	5	36	50	14	0
UMA Fall	16	307	277	13	3	289	255	31	25
UMA Fall	17	72	74	18	16	68	56	22	34

		origTSorigTH: raw TS and original thresholds							
		KNN				TF-IDF			
source	parID	TN	TP	FP	FN	TN	TP	FP	FN
TST	1	59	45	1	15	57	53	3	7
TST	2	60	60	0	0	60	60	0	0
TST	3	47	57	13	3	43	59	17	1
TST	4	56	58	4	2	56	45	4	15
TST	5	41	37	19	23	42	43	18	17
TST	6	60	55	0	5	60	55	0	5
TST	7	60	54	0	6	60	60	0	0
TST	8	57	59	3	1	57	58	3	2
TST	9	60	47	0	13	60	52	0	8
TST	10	53	56	7	4	53	59	7	1
TST	11	60	60	0	0	60	60	0	0
UMA Fall	1	78	88	12	12	68	93	22	7
UMA Fall	2	90	57	0	3	90	60	0	0
UMA Fall	3	66	82	24	8	68	82	22	8
UMA Fall	4	98	71	2	9	93	74	7	6
UMA Fall	9	87	82	3	8	87	85	3	5
UMA Fall	12	100	29	10	11	100	31	10	9
UMA Fall	15	42	39	8	11	37	45	13	5
UMA Fall	16	307	269	13	11	289	269	31	11
UMA Fall	17	74	58	16	32	74	63	16	27

Table 5 Supervised Results. Evaluation of the classifiers. Participants from the TST data set. Mean and standard deviation values of Accuracy and Sensitivity for all the scenarios and methods. The scenarios are normTSnormTH (normalized TS and normalized thresholds), normTSscldTH (normalized TS and scaled thresholds) and origTSorigTH (raw TS and original thresholds). partID stands for participant Identification number within each data set.

			normTSnormTH				normTSscldTH				origTSorigTH			
			KNN		TF-IDF		KNN		TF-IDF		KNN		TF-IDF	
source	parID	Stat	Acc	Sen	Acc	Sen	Acc	Sen	Acc	Sen	Acc	Sen	Acc	Sen
TST	1	mean	0.8348	0.8800	0.9548	1.0000	0.7238	0.7333	0.9714	1.0000	0.8348	0.8433	0.9548	1.0000
		std	0.1750	0.1932	0.0731	0.0000	0.1438	0.1956	0.0602	0.0000	0.1875	0.1833	0.0731	0.0000
TST	2	mean	0.9500	0.9500	1.0000	1.0000	0.9333	0.9333	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
		std	0.1581	0.1581	0.0000	0.0000	0.0861	0.0861	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
TST	3	mean	0.7548	0.8333	0.8024	0.9500	0.8571	0.9500	0.8667	0.9667	0.8065	0.9500	0.7815	0.9833
		std	0.1379	0.1757	0.0865	0.0805	0.1166	0.0805	0.0479	0.0703	0.1591	0.1581	0.1178	0.0527
TST	4	mean	0.9429	1.0000	0.6595	0.7167	0.9000	0.9000	1.0000	1.0000	0.9095	0.9667	0.6929	0.7500
		std	0.0738	0.0000	0.2094	0.2727	0.1165	0.1165	0.0000	0.0000	0.0784	0.0703	0.1923	0.2515
TST	5	mean	0.6248	0.8333	0.7681	0.8517	0.6075	0.8333	0.8419	0.9000	0.5808	0.7450	0.6780	0.8617
		std	0.0710	0.1912	0.1177	0.1850	0.0690	0.1571	0.0733	0.0861	0.1126	0.1799	0.1171	0.1905
TST	6	mean	1.0000	1.0000	1.0000	1.0000	0.8958	0.9167	0.8095	0.9000	1.0000	1.0000	1.0000	1.0000
		std	0.0000	0.0000	0.0000	0.0000	0.1856	0.1620	0.0937	0.1165	0.0000	0.0000	0.0000	0.0000
TST	7	mean	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	0.9000	0.9000	1.0000	1.0000
		std	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2108	0.2108	0.0000	0.0000
TST	8	mean	0.8982	0.9357	0.9482	0.9857	1.0000	1.0000	1.0000	1.0000	0.9500	0.9857	0.9292	0.9667
		std	0.1068	0.1145	0.0671	0.0452	0.0000	0.0000	0.0000	0.0000	0.0874	0.0452	0.1094	0.1054
TST	9	mean	1.0000	1.0000	1.0000	1.0000	0.9333	0.9333	1.0000	1.0000	0.9033	0.9033	1.0000	1.0000
		std	0.0000	0.0000	0.0000	0.0000	0.0861	0.0861	0.0000	0.0000	0.1024	0.1024	0.0000	0.0000
TST	10	mean	0.8500	0.9500	0.7667	0.8667	0.7929	0.8833	0.8500	0.9500	0.8333	0.9333	0.8833	0.9833
		std	0.0794	0.1125	0.1526	0.2194	0.1041	0.1125	0.0115	0.0805	0.0594	0.1165	0.0619	0.0527
TST	11	mean	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
		std	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

Table 6 Supervised Results. Evaluation of the classifiers. Participants from UMA Fall. Mean and standard deviation values of the Accuracy and Sensitivity for all the scenarios and methods. The scenarios are normTSnormTH (normalized TS and normalized thresholds), normTSscldTH (normalized TS and scaled thresholds) and origTSorigTH (raw TS and original thresholds). partID stands for participant Identification number within each data set.

			normTSnormTH				normTSscldTH				origTSorigTH			
			KNN		TF-IDF		KNN		TF-IDF		KNN		TF-IDF	
source	parID	Stat	Acc	Sen	Acc	Sen	Acc	Sen	Acc	Sen	Acc	Sen	Acc	Sen
UMA Fall	1	mean	0.7933	0.8944	0.8717	0.9800	0.8689	0.9300	0.7627	0.9100	0.8503	0.9256	0.8105	0.9789
		std	0.1137	0.1243	0.0647	0.0422	0.0739	0.1059	0.0612	0.0994	0.0689	0.0516	0.0564	0.0446
UMA Fall	2	mean	0.9667	0.9667	1.0000	1.0000	0.9333	0.9333	1.0000	1.0000	0.9500	0.9500	1.0000	1.0000
		std	0.1054	0.1054	0.0000	0.0000	0.1405	0.1405	0.0000	0.0000	0.1581	0.1581	0.0000	0.0000
UMA Fall	3	mean	0.7212	0.8444	0.8000	0.8111	0.8714	0.9444	0.9014	0.9778	0.7621	0.9111	0.7699	0.9111
		std	0.0782	0.0937	0.1000	0.0915	0.0785	0.1080	0.0611	0.0468	0.0818	0.0876	0.0842	0.1021
UMA Fall	4	mean	0.9149	0.9571	0.8727	0.9464	0.9578	1.0000	0.8945	0.9714	0.9349	0.9571	0.9231	1.0000
		std	0.0932	0.0690	0.0986	0.0694	0.0892	0.0000	0.1054	0.0904	0.0871	0.0690	0.0885	0.0000
UMA Fall	9	mean	0.8578	0.8878	0.9700	1.0000	0.8462	0.9778	0.8310	0.9689	0.8700	0.9000	0.9156	0.9456
		std	0.1833	0.1969	0.0483	0.0000	0.0920	0.0703	0.0969	0.0502	0.2058	0.2160	0.0689	0.0778
UMA Fall	12	mean	0.6283	0.7583	0.7567	0.9417	0.6487	0.7750	0.6287	0.7250	0.7583	0.9500	0.7983	1.0000
		std	0.1644	0.2436	0.2079	0.1245	0.1898	0.1845	0.1350	0.2486	0.1601	0.1581	0.1660	0.0000
UMA Fall	15	mean	0.7295	0.9100	0.7995	1.0000	0.7571	0.9000	0.7857	1.0000	0.7605	0.8650	0.7829	1.0000
		std	0.0820	0.1174	0.1208	0.0000	0.1346	0.1054	0.0615	0.0000	0.0906	0.1180	0.0997	0.0000
UMA Fall	16	mean	0.9523	0.9926	0.8697	0.9668	0.9360	0.9690	0.8196	0.9012	0.9430	0.9890	0.8788	0.9889
		std	0.0302	0.0234	0.0579	0.0476	0.0491	0.0302	0.0938	0.1279	0.0370	0.0177	0.0391	0.0179
UMA Fall	17	mean	0.7378	0.9071	0.7114	0.8481	0.7829	0.9125	0.5745	0.6889	0.7489	0.9214	0.8128	1.0000
		std	0.1669	0.1620	0.1137	0.1801	0.1155	0.1324	0.1596	0.1991	0.1478	0.1328	0.1122	0.0000

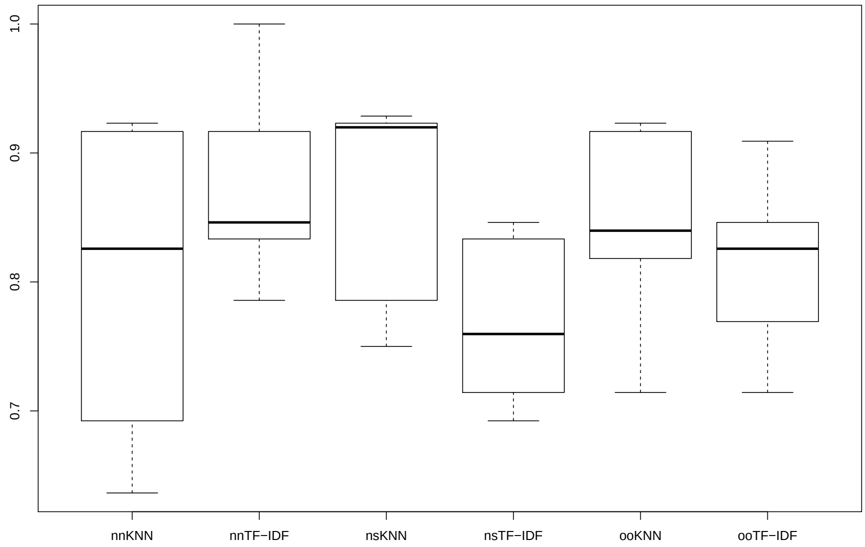

Fig. 2 Boxplot of the Accuracy for participant 1 from UMA Fall. From left to right, Accuracy measured for i) the KNN and TF-IDF with the normTSnormTH; ii) the KNN and TF-IDF with the normTSscldTH; finally, iii) the KNN and TF-IDF with the origTSorigTH.

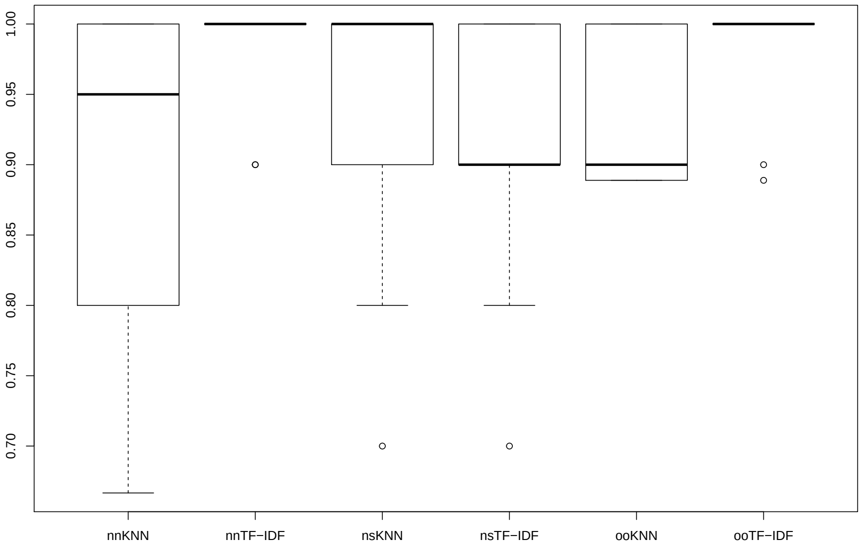

Fig. 3 Boxplot of the Sensitivity for participant 1 from UMA Fall. From left to right, Accuracy measured for i) the KNN and TF-IDF with the normTSnormTH; ii) the KNN and TF-IDF with the normTSscldTH; finally, iii) the KNN and TF-IDF with the origTSorigTH.

Table 7 Supervised Results. Evaluation of peak detection and each of the classifiers together. Participants from the TST data set. Mean and standard deviation values of Accuracy and Sensitivity for all the scenarios and methods. The scenarios are normTSnormTH (normalized TS and normalized thresholds), normTSscldTH (normalized TS and scaled thresholds) and origTSorigTH (raw TS and original thresholds). partID stands for participant Identification number within each data set.

			normTSnormTH				normTSscldTH				origTSorigTH			
			KNN		TF-IDF		KNN		TF-IDF		KNN		TF-IDF	
source	parID	Stat	Acc	Sen	Acc	Sen	Acc	Sen	Acc	Sen	Acc	Sen	Acc	Sen
TST	1	mean	0.8667	0.7833	0.9167	0.8833	0.8500	0.7333	0.9833	1.0000	0.8667	0.7500	0.9167	0.8833
		std	0.0896	0.2086	0.0393	0.0805	0.0766	0.1956	0.0351	0.0000	0.0978	0.1964	0.0393	0.0805
TST	2	mean	0.9750	0.9500	1.0000	1.0000	0.9667	0.9333	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
		std	0.0791	0.1581	0.0000	0.0000	0.0430	0.0861	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
TST	3	mean	0.8333	0.8333	0.8667	0.9500	0.9167	0.9500	0.9250	0.9667	0.8667	0.9500	0.8500	0.9833
		std	0.0962	0.1757	0.0703	0.0805	0.0680	0.0805	0.0264	0.0703	0.1125	0.1581	0.0861	0.0527
TST	4	mean	0.9667	1.0000	0.8250	0.7167	0.9500	0.9000	1.0000	1.0000	0.9500	0.9667	0.8417	0.7500
		std	0.0430	0.0000	0.0998	0.2727	0.0583	0.1165	0.0000	0.0000	0.0430	0.0703	0.0917	0.2515
TST	5	mean	0.6750	0.6833	0.7667	0.7167	0.7333	0.8333	0.8833	0.9000	0.6500	0.6167	0.7083	0.7167
		std	0.0615	0.1459	0.1097	0.2086	0.0766	0.1571	0.0583	0.0861	0.0657	0.1581	0.0900	0.1933
TST	6	mean	0.9583	0.9167	0.9583	0.9167	0.9333	0.9167	0.8917	0.9000	0.9583	0.9167	0.9583	0.9167
		std	0.0589	0.1179	0.0589	0.1179	0.1230	0.1620	0.0562	0.1165	0.0589	0.1179	0.0589	0.1179
TST	7	mean	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	0.9500	0.9000	1.0000	1.0000
		std	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1054	0.2108	0.0000	0.0000
TST	8	mean	0.9417	0.9333	0.9667	0.9833	1.0000	1.0000	1.0000	1.0000	0.9667	0.9833	0.9583	0.9667
		std	0.0562	0.1165	0.0430	0.0527	0.0000	0.0000	0.0000	0.0000	0.0583	0.0527	0.0589	0.1054
TST	9	mean	0.9333	0.8667	0.9333	0.8667	0.9667	0.9333	1.0000	1.0000	0.8917	0.7833	0.9333	0.8667
		std	0.0351	0.0703	0.0351	0.0703	0.0430	0.0861	0.0000	0.0000	0.0562	0.1125	0.0351	0.0703
TST	10	mean	0.9167	0.9500	0.8750	0.8667	0.8833	0.8833	0.9167	0.9500	0.9083	0.9333	0.9333	0.9833
		std	0.0393	0.1125	0.0708	0.2194	0.0583	0.1125	0.0000	0.0805	0.0264	0.1165	0.0351	0.0527
TST	11	mean	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
		std	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

Table 8 Supervised Results. Evaluation of peak detection and each of the classifiers together. Participants from the UMA Fall data set. Mean and standard deviation values of Accuracy and Sensitivity for all the scenarios and methods. The scenarios are normTSnormTH (normalized TS and normalized thresholds), normTSscldTH (normalized TS and scaled thresholds) and origTSorigTH (raw TS and original thresholds). partID stands for participant Identification number within each data set.

			normTSnormTH				normTSscldTH				origTSorigTH			
			KNN		TF-IDF		KNN		TF-IDF		KNN		TF-IDF	
source	parID	Stat	Acc	Sen	Acc	Sen	Acc	Sen	Acc	Sen	Acc	Sen	Acc	Sen
UMA Fall	1	mean	0.8368	0.8500	0.8895	0.9300	0.9105	0.9300	0.8368	0.9100	0.8737	0.8800	0.8474	0.9300
		std	0.0763	0.1269	0.0523	0.0483	0.0499	0.1059	0.0461	0.0994	0.0508	0.0789	0.0388	0.0675
UMA Fall	2	mean	0.9867	0.9667	1.0000	1.0000	0.9733	0.9333	1.0000	1.0000	0.9800	0.9500	1.0000	1.0000
		std	0.0422	0.1054	0.0000	0.0000	0.0562	0.1405	0.0000	0.0000	0.0632	0.1581	0.0000	0.0000
UMA Fall	3	mean	0.7944	0.8444	0.8500	0.8111	0.9278	0.9444	0.9444	0.9778	0.8222	0.9111	0.8333	0.9111
		std	0.0589	0.0937	0.0830	0.0915	0.0457	0.1080	0.0370	0.0468	0.0683	0.0876	0.0586	0.1021
UMA Fall	4	mean	0.9278	0.8875	0.9056	0.8750	0.9444	0.9250	0.9167	0.9000	0.9389	0.8875	0.9278	0.9250
		std	0.0589	0.1094	0.0457	0.0833	0.0524	0.0645	0.0540	0.1149	0.0611	0.1094	0.0457	0.0645
UMA Fall	9	mean	0.9333	0.9000	0.9833	1.0000	0.9222	0.9778	0.9222	0.9778	0.9389	0.9111	0.9556	0.9444
		std	0.0820	0.1772	0.0268	0.0000	0.0388	0.0703	0.0468	0.0468	0.0924	0.1946	0.0351	0.0786
UMA Fall	12	mean	0.8200	0.5750	0.8600	0.7250	0.8667	0.7750	0.8533	0.7250	0.8600	0.7250	0.8733	0.7750
		std	0.0549	0.2058	0.0798	0.1419	0.0703	0.1845	0.0689	0.2486	0.0584	0.1419	0.0734	0.1419
UMA Fall	15	mean	0.7900	0.8200	0.8300	0.9000	0.8400	0.9000	0.8600	1.0000	0.8100	0.7800	0.8200	0.9000
		std	0.0738	0.1476	0.1059	0.1054	0.0966	0.1054	0.0516	0.0000	0.0738	0.1476	0.0919	0.1054
UMA Fall	16	mean	0.9617	0.9643	0.9267	0.9393	0.9733	0.9893	0.9067	0.9107	0.9600	0.9607	0.9300	0.9607
		std	0.0177	0.0337	0.0296	0.0534	0.0251	0.0173	0.0492	0.1296	0.0161	0.0264	0.0205	0.0313
UMA Fall	17	mean	0.7278	0.6333	0.7278	0.6000	0.8111	0.8222	0.6889	0.6222	0.7333	0.6444	0.7611	0.7000
		std	0.0805	0.1486	0.0552	0.1829	0.0794	0.1304	0.0951	0.1903	0.0731	0.1366	0.0589	0.1054

different distance and similarity measurements. The Manhattan distance, the cosine distance or the pattern based distance proposed in [31] could help in this matter. Balancing the bag of words might also enhance the performance of the solution; for instance, this can be done by favouring the selection of ADL's words to be considerd as part of the transfer learning. Finally, further study to set the different thresholds (th_{sim} and th_{common}) is needed.

Apart from this, some more issues have been found. Obviously, the higher the number of participants to create the bag of words for transfer learning the bigger the size of the bag of words, and the higher the computational cost. This is an important parameter if this should be implemented in a smartwatch or similar device. Furthermore, as long as the bag of words keeps increasing with every new peak whose SAX word is not in the database, the conclusion is that the bag of words might become as big as the whole set of possible combinations, which is unfeasible. Therefore, pruning is required and also a forgetting tool, something like the four Rs in Case-Based Reasoning.

Moreover, it has been found that words are not so often repeated, which also leads to a data base of all the possible words. One of the reasons for this variability is that the data sets were not gathered with participants from the focused population; in all cases the participants were relatively young people or middle-aged adults. With age, the amount of movement becomes reduced, so it can be expected that the variability in the movements can also become reduced. This issue requires further study of the elderly population, with ADL data and, if possible, by gathering information from real fall events suffered from senior citizens.

Furthermore, two more ideas can enhance the performance of the TF-IDF with transfer learning. Firstly, a clustering stage of the words extracted from the training with ADLs, and a simple on-line clustering of the words that are being learned on the fly would eventually produce an efficient bag of words, which is, in the light of the results, essential for obtaining better classifier performances. Secondly, different classification schemes, such as the SAX-VSM [35], would represent interesting alternatives to reduce the number of False Positives.

Finally, the idea of transfer learning can also be applied to a supervised case. Then, the point of training would be learning to discriminate the ADLs from the already transferred knowledge related to fall detection. Besides, the fusion of information deserves more study. In other words, the TS is normalized for the current user, but the a priori knowledge is extracted for a different participant. Introducing a fusion of the different types of measurements, the current one and the stored one, may also enhance the capability of the SAX words to adapt to the current user.

Table 9 On-line learning Results. Evaluation of the classifiers. Aggregation of the results from the confusion matrix for all the scenarios and methods. TL stands for transfer learning. partID stands for participant Identification number within each data set.

		normTSnormTH: normalized TS and normalized thresholds							
		TF-IDF				TF-IDF + TL			
source	parID	TN	TP	FP	FN	TN	TP	FP	FN
TST	1	0	53	3	0	0	53	3	0
TST	2	0	60	0	0	0	56	0	4
TST	3	9	55	10	5	0	60	19	0
TST	4	0	35	4	25	0	60	4	0
TST	5	17	35	10	15	0	50	27	0
TST	6	0	55	0	0	0	55	0	0
TST	7	0	60	0	0	0	60	0	0
TST	8	0	61	3	6	0	67	3	0
TST	9	0	52	0	0	0	52	0	0
TST	10	0	41	7	19	0	60	7	0
TST	11	0	60	0	0	0	60	0	0
UMA Fall	1	24	52	6	43	0	95	30	0
UMA Fall	2	0	60	0	0	0	60	0	0
UMA Fall	3	41	35	2	55	0	90	43	0
UMA Fall	4	2	64	5	10	0	74	7	0
UMA Fall	9	0	81	3	12	0	93	3	0
UMA Fall	12	7	13	8	18	0	31	15	0
UMA Fall	15	8	28	6	17	2	45	12	0
UMA Fall	16	35	217	4	55	0	272	39	0
UMA Fall	17	12	37	4	26	0	63	16	0
		normTSscldTH: normalized TS and scaled thresholds							
		KNN				TF-IDF			
source	parID	TN	TP	FP	FN	TN	TP	FP	FN
TST	1	6	49	0	11	0	60	6	0
TST	2	0	60	0	0	0	60	0	0
TST	3	0	48	7	12	0	59	7	1
TST	4	0	60	0	0	0	60	0	0
TST	5	25	48	7	12	0	60	32	0
TST	6	0	45	7	15	0	60	7	0
TST	7	0	60	0	0	0	60	0	0
TST	8	0	67	0	0	0	67	0	0
TST	9	0	60	0	0	0	60	0	0
TST	10	0	41	7	19	0	60	7	0
TST	11	0	60	0	0	0	60	0	0
UMA Fall	1	28	44	2	56	1	100	29	0
UMA Fall	2	0	60	0	0	0	60	0	0
UMA Fall	3	7	76	2	14	0	90	9	0
UMA Fall	4	1	54	6	20	1	74	6	0
UMA Fall	9	18	63	4	30	0	93	22	0
UMA Fall	12	24	3	0	37	0	40	24	0
UMA Fall	15	0	33	14	17	2	50	12	0
UMA Fall	16	48	181	0	105	0	286	48	0
UMA Fall	17	30	18	0	63	0	81	30	0
		origTSorigTH: raw TS and original thresholds							
		KNN				TF-IDF			
source	parID	TN	TP	FP	FN	TN	TP	FP	FN
TST	1	0	43	3	10	0	53	3	0
TST	2	0	60	0	0	0	60	0	0
TST	3	17	30	2	30	0	60	19	0
TST	4	0	50	4	10	0	60	4	0
TST	5	27	20	0	30	0	50	27	0
TST	6	0	55	0	0	0	55	0	0
TST	7	0	60	0	0	0	60	0	0
TST	8	0	57	3	10	0	67	3	0
TST	9	0	52	0	0	0	52	0	0
TST	10	0	50	7	10	0	60	7	0
TST	11	0	60	0	0	0	60	0	0
UMA Fall	1	30	29	0	66	0	95	30	0
UMA Fall	2	0	60	0	0	0	60	0	0
UMA Fall	3	43	0	0	90	0	90	43	0
UMA Fall	4	5	44	2	30	0	74	7	0
UMA Fall	9	0	83	3	10	0	93	3	0
UMA Fall	12	14	2	1	29	0	31	15	0
UMA Fall	15	10	15	4	30	0	45	14	0
UMA Fall	16	39	192	0	80	0	272	39	0
UMA Fall	17	14	33	2	30	0	63	16	0

Table 10 On-line learning Results. Evaluation of the classifiers. Participants from the TST data set. Mean and standard deviation values of Accuracy and Sensitivity for all the scenarios and methods. TL stands for transfer learning. The scenarios are normTSnormTH (normalized TS and normalized thresholds), normTSscldTH (normalized TS and scaled thresholds) and origTSorigTH (raw TS and original thresholds). partID stands for participant Identification number within each data set.

			normTSnormTH				normTSscldTH				origTSorigTH			
			TF-IDF		TF-IDF+TL		TF-IDF		TF-IDF+TL		TF-IDF		TF-IDF+TL	
source	parID	Stat	Acc	Sen	Acc	Sen	Acc	Sen	Acc	Sen	Acc	Sen	Acc	Sen
TST	1	mean	0.9548	1.0000	0.9548	1.0000	0.8310	0.8167	0.9143	1.0000	0.7729	0.8100	0.9548	1.0000
		std	0.0731	0.0000	0.0731	0.0000	0.0894	0.0946	0.0738	0.0000	0.0540	0.0161	0.0731	0.0000
TST	2	mean	1.0000	1.0000	0.9333	0.9333	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
		std	0.0000	0.0000	0.0861	0.0861	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
TST	3	mean	0.8135	0.9167	0.7690	1.0000	0.7214	0.8000	0.8857	0.9833	0.5933	0.5000	0.7690	1.0000
		std	0.0733	0.0878	0.0967	0.0000	0.0959	0.0703	0.0904	0.0527	0.0448	0.0000	0.0967	0.0000
TST	4	mean	0.5500	0.5833	0.9429	1.0000	1.0000	1.0000	1.0000	1.0000	0.7857	0.8333	0.9429	1.0000
		std	0.1395	0.1416	0.0738	0.0000	0.0000	0.0000	0.0000	0.0000	0.0615	0.0000	0.0738	0.0000
TST	5	mean	0.6794	0.6917	0.6502	1.0000	0.7995	0.8000	0.6595	1.0000	0.6036	0.3850	0.6502	1.0000
		std	0.1930	0.2315	0.0675	0.0000	0.1035	0.1054	0.0733	0.0000	0.0555	0.1029	0.0675	0.0000
TST	6	mean	1.0000	1.0000	1.0000	1.0000	0.6857	0.7500	0.9071	1.0000	1.0000	1.0000	1.0000	1.0000
		std	0.0000	0.0000	0.0000	0.0000	0.1428	0.0878	0.1054	0.0000	0.0000	0.0000	0.0000	0.0000
TST	7	mean	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
		std	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
TST	8	mean	0.8798	0.9119	0.9625	1.0000	1.0000	1.0000	1.0000	1.0000	0.8179	0.8500	0.9625	1.0000
		std	0.1113	0.0762	0.0604	0.0000	0.0000	0.0000	0.0000	0.0000	0.0480	0.0115	0.0604	0.0000
TST	9	mean	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
		std	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
TST	10	mean	0.6143	0.6833	0.9000	1.0000	0.6143	0.6833	0.9000	1.0000	0.7500	0.8333	0.9000	1.0000
		std	0.1792	0.1995	0.0690	0.0000	0.1360	0.1459	0.0690	0.0000	0.0575	0.0000	0.0690	0.0000
TST	11	mean	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
		std	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

Table 11 On-line learning Results. Evaluation of the classifiers. Participants from UMA Fall. Mean and standard deviation values of Accuracy and Sensitivity for all the scenarios and methods. TL stands for transfer learning. The scenarios are normTSnormTH (normalized TS and normalized thresholds), normTSscldTH (normalized TS and scaled thresholds) and origTSorigTH (raw TS and original thresholds). partID stands for participant Identification number within each data set.

			normTSnormTH				normTSscldTH				origTSorigTH			
			TF-IDF		TF-IDF+TL		TF-IDF		TF-IDF+TL		TF-IDF		TF-IDF+TL	
source	parID	Stat	Acc	Sen	Acc	Sen	Acc	Sen	Acc	Sen	Acc	Sen	Acc	Sen
UMA Fall	1	mean	0.6032	0.5433	0.7623	1.0000	0.5497	0.4400	0.7788	1.0000	0.4645	0.3000	0.7623	1.0000
		std	0.1670	0.2142	0.0357	0.0000	0.1637	0.1838	0.0463	0.0000	0.1366	0.1610	0.0357	0.0000
UMA Fall	2	mean	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
		std	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
UMA Fall	3	mean	0.5728	0.3889	0.6804	1.0000	0.8390	0.8444	0.9136	1.0000	0.3196	0.0000	0.6804	1.0000
		std	0.0697	0.1080	0.0524	0.0000	0.0993	0.0937	0.0679	0.0000	0.0524	0.0000	0.0524	0.0000
UMA Fall	4	mean	0.8180	0.8625	0.9231	1.0000	0.6864	0.7304	0.9331	1.0000	0.6025	0.5929	0.9231	1.0000
		std	0.1103	0.1170	0.0885	0.0000	0.0848	0.0574	0.0781	0.0000	0.0413	0.0277	0.0885	0.0000
UMA Fall	9	mean	0.8422	0.8700	0.9700	1.0000	0.7039	0.6767	0.8130	1.0000	0.8656	0.8922	0.9700	1.0000
		std	0.0791	0.0887	0.0483	0.0000	0.0882	0.1182	0.0688	0.0000	0.0455	0.0054	0.0483	0.0000
UMA Fall	12	mean	0.4383	0.4167	0.6933	1.0000	0.3886	0.0750	0.6592	1.0000	0.3267	0.0500	0.6933	1.0000
		std	0.1591	0.1964	0.1401	0.0000	0.1816	0.1208	0.1668	0.0000	0.1375	0.1054	0.1401	0.0000
UMA Fall	15	mean	0.6176	0.6250	0.8029	1.0000	0.5262	0.6600	0.8190	1.0000	0.4224	0.3250	0.7662	1.0000
		std	0.2184	0.2617	0.1213	0.0000	0.1794	0.1897	0.1107	0.0000	0.0628	0.0791	0.0684	0.0000
UMA Fall	16	mean	0.8103	0.7978	0.8763	1.0000	0.6848	0.6325	0.8575	1.0000	0.7422	0.7057	0.8763	1.0000
		std	0.0293	0.0389	0.0409	0.0000	0.0494	0.0557	0.0300	0.0000	0.0132	0.0069	0.0409	0.0000
UMA Fall	17	mean	0.6109	0.5782	0.8128	1.0000	0.4248	0.2208	0.7383	1.0000	0.5849	0.5139	0.8128	1.0000
		std	0.1102	0.1665	0.1122	0.0000	0.1026	0.1094	0.0783	0.0000	0.0634	0.0738	0.1122	0.0000

5 Conclusion

This study proposes user-centered solutions to FD. On the one hand, supervised learning has been applied to obtain a KNN and a TF-IDF classifier using training data that includes simulated falls and ADLs. On the other hand, online learning has been applied to obtain a TF-IDF classifier that is enhanced with a transfer learning stage using the same data set collections.

The results show that modelling the FD for each user instead of obtaining generalized models can produce better performance. On-line learning methods have shown validity in order to avoid the need for fall events to train the model for the user. Nevertheless, transfer learning has improved the on-line learning classifier, suggesting this might be the path to follow in future research.

This future research should also focus on several issues, including i) determining the best thresholds for the on-line learning TF-IDF solution, ii) enhancing transfer learning to focus on notably important words, iii) integrating transfer learning with the supervised scheme, so while one learns the main references for the positive class the training focuses on the negative class, iv) introducing clustering among the words to reduce the size of the bag of words, v) studying how to integrate the R's stages from Case-Based Reasoning to the online learning and tuning of the models.

References

1. Abbate, S., Avvenuti, M., Bonatesta, F., Cola, G., Corsini, P., AlessioVecchio: A smartphone-based fall detection system. Pervasive and Mobile Computing **8**(6), 883–899 (2012)
2. Abbate, S., Avvenuti, M., Corsini, P., Light, J., Vecchio, A.: Wireless Sensor Networks: Application - Centric Design, chap. Monitoring of human movements for fall detection and activities recognition in elderly care using wireless sensor network: a survey, p. 22. Intech (2010). DOI 10.5772/13802
3. Bianchi, F., Redmond, S.J., Narayanan, M.R., Cerutti, S., Lovell, N.H.: Barometric pressure and triaxial accelerometry-based falls event detection. IEEE Transactions on Neural Systems and Rehabilitation Engineering **18**(6), 619–627 (2010). DOI 10.1109/TNSRE.2010.2070807
4. Bourke, A., O'Brien, J., Lyons, G.: Evaluation of a threshold-based triaxial accelerometer fall detection algorithm. Gait and Posture **26**, 194–199 (2007)
5. Bourke, A., van de Ven, P., Gamble, M., O'Connor, R., Murphy, K., Bogan, E., McQuade, E., Finucane, P., Olaighin, G., Nelson, J.: Evaluation of waist-mounted tri-axial accelerometer based fall-detection algorithms during scripted and continuous unscripted activities. Journal of Biomechanics **43**, 3051–3057 (2010)
6. Bourke, A.K., Klenk, J., Schwickert, L., Aminian, K., Ihlen, E.A.F., Mellone, S., Helbostad, J.L., Chiari, L., Becker, C.: Fall detection algorithms for real-world falls harvested from lumbar sensors in the elderly population: A machine learning approach. In: 2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), pp. 3712–3715 (2016). DOI 10.1109/EMBC.2016.7591534
7. Casilari, E., Oviedo-Jimnez, M.A.: Automatic fall detection system based on the combined use of a smartphone and a smartwatch. PLOS ONE **10**(11), 1–11 (2015). DOI 10.1371/journal.pone.0140929. URL `https://doi.org/10.1371/journal.pone.0140929`
8. Casilari, E., Santoyo-Ramn, J.A., Cano-Garca, J.M.: Umafall: A multisensor dataset for the research on automatic fall detection. Procedia Computer Science **110**(Supplement C), 32 – 39 (2017). DOI https://doi.org/10.1016/j.procs.2017.06.110. URL `http:`

//www.sciencedirect.com/science/article/pii/S1877050917312899. 14th International Conference on Mobile Systems and Pervasive Computing (MobiSPC 2017) / 12th International Conference on Future Networks and Communications (FNC 2017) / Affiliated Workshops

9. Casilari, E., Santoyo-Ramón, J.A., Cano-García, J.M.: Analysis of public datasets for wearable fall detection systems. Sensors **17**(1513), 4324 – 4338 (2017). DOI https://10.3390/s17071513. URL http://www.mdpi.com/1424-8220/17/7/1513
10. Chaudhuri, S., Thompson, H., Demiris, G.: Fall detection devices and their use with older adults: A systematic review. J Geriatr Phys Ther **37**, 178–196 (2014)
11. Delahoz, Y.S., Labrador, M.A.: Survey on fall detection and fall prevention using wearable and external sensors. Sensors **14**(10), 19806–19842 (2014). DOI doi:10.3390/s141019806. URL http://www.mdpi.com/1424-8220/14/10/19806/htm
12. Deutsch, M., Burgsteiner, H.: Health informatics meets ehealth. chap. A Smartwatch-Based Assistance System for the Elderly Performing Fall Detection, Unusual Inactivity Recognition and Medication Reminding, pp. 259–266. IOS Press (2016). DOI 10.3233/978-1-61499-645-3-259
13. Dugan, S.A., Gabriel, K.P., Lange-Maia, B.S., Karvonen-Gutierrez, C.: Physical activity and physical function: Moving and aging. Obstetrics and Gynecology Clinics of North America **45**(4), 723 – 736 (2018). DOI https://doi.org/10.1016/j.ogc.2018.07.009. URL http://www.sciencedirect.com/science/article/pii/S088985451830069X. Reproductive Aging
14. Foundation, T.R.: The r project website. https://www.r-project.org (2019)
15. Gasparrini, S., Cippitelli, E., Gambi, E., Spinsante, S., Wahslen, J., Orhan, I., Lindh, T.: Proposal and experimental evaluation of fall detection solution based on wearable and depth data fusion. In: ICT Innovations 2015, Advances in Intelligent Systems and Computing, vol. 399, pp. 99–108. Springer (2016). DOI 10.1007/978-3-319-25733-4_11. URL http://www.tlc.dii.univpm.it/blog/databases4kinectandhttps://ieee-dataport.org/documents/tst-fall-detection-dataset-v2
16. Gjoreski, H., Bizjak, J., Gams, M.: Using smartwatch as telecare and fall detection device. In: 2016 12th International Conference on Intelligent Environments (IE), pp. 242–245 (2016). DOI 10.1109/IE.2016.55
17. Hakim, A., Huq, M.S., Shanta, S., Ibrahim, B.: Smartphone based data mining for fall detection: Analysis and design. Procedia Computer Science **105**, 46–51 (2017). DOI https://doi.org/10.1016/j.procs.2017.01.188. URL http://www.sciencedirect.com/science/article/pii/S1877050917302065
18. Huynh, Q.T., Nguyen, U.D., Irazabal, L.B., Ghassemian, N., Tran, B.Q.: Optimization of an accelerometer and gyroscope-based fall detection algorithm. Journal of Sensors **2015, Article ID 452078**(452078), 8 pages (2015). DOI http://dx.doi.org/10.1155/2015/452078
19. Igual, R., Medrano, C., Plaza, I.: Challenges, issues and trends in fall detection systems. BioMedical Engineering OnLine **12**(66) (2013). URL http://www.biomedical-engineering-online.com/content/12/1/66
20. Igual, R., Medrano, C., Plaza, I.: A comparison of public datasets for acceleration-based fall detection. Medical Engineering and Physics **37**(9), 870–878 (2015). DOI https://doi.org/10.1016/j.medengphy.2015.06.009. URL http://www.sciencedirect.com/science/article/pii/S1350453315001575
21. inc, R.: The rstudio website. https://www.rstudio.com/ (2019)
22. Jatesiktat, P., Ang, W.T.: An elderly fall detection using a wrist-worn accelerometer and barometer. In: 2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), pp. 125–130 (2017). DOI 10.1109/EMBC.2017.8036778
23. Kangas, M., Konttila, A., Lindgren, P., Winblad, I., Jämsaä, T.: Comparison of low-complexity fall detection algorithms for body attached accelerometers. Gait and Posture **28**, 285–291 (2008)
24. Kangas, M., Vikman, I., Nyberg, L., Korpelainen, R., Lindblom, J., Jamsa, T.: Comparison of real-life accidental falls in older people with experimental falls in middle-aged test subjects. Gait and Posture **35**, 500–505 (2012)

25. Khan, S.S., JesseHoey: Review of fall detection techniques: A data availability perspective. Medical Engineering and Physics **39**, 12–22 (2017). DOI https://doi.org/10.1016/j.medengphy.2016.10.014. URL http://www.sciencedirect.com/science/article/pii/S1350453316302600
26. Khojasteh, S.B., Villar, J.R., Chira, C., González, V.M., de la Cal, E.: Improving fall detection using an on-wrist wearable accelerometer. Sensors **18**, 1350 (2018). DOI https://doi.org/10.3390/s18051350
27. Kostopoulos, P., Nunes, T., Salvi, K., Deriaz, M., Torrent, J.: F2d: A fall detection system tested with real data from daily life of elderly people. In: 2015 17th International Conference on E-health Networking, Application Services (HealthCom), pp. 397–403 (2015)
28. Lin, J., Keogh, E., Lonardi, S., Chiu, B.: A symbolic representation of time series, with implications for streaming algorithms. In: Proceedings of the 8th ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery, DMKD'03, pp. 2–11. ACM (2003)
29. Medrano, C., Plaza, I., Igual, R., Sánchez, Á., Castr0, M.: The effect of personalization on smartphone-based fall detectors. Sensors **16, Article ID 117**(1), 117 (2016). DOI 10.3390/s16010117. URL http://www.mdpi.com/1424-8220/16/1/117/htm
30. Ngu, A., Wu, Y., Zare, H., Polican, A., Yarbrough, B., Yao, L.: Fall detection using smartwatch sensor data with accessor architecture. In: H. Chen, D. Zeng, E. Karahanna, B. I. (eds.) Proceedings of the International Conference on Smart Health ICSH 2017, Lecture Notes in Computer Science, vol. 10347, pp. 81–93. Springer (2017)
31. Phan, S., Famili, F., Tang, Z., Pan, Y., Liu, Z., Ouyang, J., Lenferink, A., Oconnor, M.M.C.: A novel pattern based clustering methodology for time-series microarray data. International Journal of Computer Mathematics **84**(5), 585–597 (2007). DOI 10.1080/00207160701203419. URL http://dx.doi.org/10.1080/00207160701203419
32. Putra, I.P.E.S., Brusey, J., Gaura, E., Vesilo, R.: An event-triggered machine learning approach for accelerometer-based fall detection. Sensors **18**, 2034 (2018). DOI https://doi:10.3390/s18010020
33. Roberts, C.: How to choose a medical alert system (2018). URL https://www.consumerreports.org/medical-alert-systems/how-to-choose-a-medical-alert-system/
34. Sabatini, A.M., Ligorio, G., Mannini, A., Genovese, V., Pinna, L.: Prior-to- and post-impact fall detection using inertial and barometric altimeter measurements. IEEE Transactions on Neural Systems and Rehabilitation Engineering **24**(7), 774–783 (2016). DOI 10.1109/TNSRE.2015.2460373. URL http://ieeexplore.ieee.org/abstract/document/7173441/
35. Senin, P., Malinchik, S.: Sax-vsm: Interpretable time series classification using sax and vector space model. In: In Proceedings of the 2013 IEEE 13th International Conference on Data Mining, pp. 1175–1180 (2013). DOI 10.1109/ICDM.2013.52
36. Sorvala, A., Alasaarela, E., Sorvoja, H., Myllyla, R.: A two-threshold fall detection algorithm for reducing false alarms. In: Proceedings of 2012 6th International Symposium on Medical Information and Communication Technology (ISMICT) (2012)
37. Sucerquia, A., López, J.D., Vargas-Bonilla, J.F.: Sisfall: A fall and movement dataset. Sensors **17**, 198 (2017). DOI http://doi.org/10.3390/s17010198
38. Tsinganos, P., Skodras, A.: A smartphone-based fall detection system for the elderly. In: Proceedings of the 10th International Symposium on Image and Signal Processing and Analysis (2017)
39. Villar, J.R., González, S., Sedano, J., Chira, C., Trejo-Gabriel-Galán, J.M.: Improving human activity recognition and its application in early stroke diagnosis. International Journal of Neural Systems **25**(1450036), 1450036–1450055 (2015)
40. Virmani, T., Gupta, H., Shah, J., Larson-Prior, L.: Objective measures of gait and balance in healthy non-falling adults as a function of age. Gait & Posture **65**, 100 – 105 (2018). DOI https://doi.org/10.1016/j.gaitpost.2018.07.167. URL http://www.sciencedirect.com/science/article/pii/S0966636218312761
41. de Walle, P.V., Meyns, P., Desloovere, K., Rijck, J.D., Kenis, J., Verbecque, E., Criekinge, T.V., Hallemans, A.: Age-related changes in arm motion during typical gait. Gait & Posture **66**, 51 – 57 (2018). DOI https://doi.org/10.1016/j.gaitpost.2018.07.176. URL http://www.sciencedirect.com/science/article/pii/S0966636218313006

42. Wu, F., Zhao, H., Zhao, Y., Zhong, H.: Development of a wearable-sensor-based fall detection system. International Journal of Telemedicine and Applications **2016, Article ID 576364**(1), 11 pages (2015). DOI http://dx.doi.org/10.1155/2015/576364. URL `https://www.hindawi.com/journals/ijta/2015/576364/`
43. Zhang, T., Wang, J., Xu, L., Liu, P.: Fall detection by wearable sensor and one-class svm algorithm. In: I.G. Huang DS. Li K. (ed.) Intelligent Computing in Signal Processing and Pattern Recognition, *Lecture Notes in Control and Information Systems*, vol. 345, pp. 858–863. Springer Berlin Heidelberg (2006). DOI https://doi.org/10.1007/978-3-540-37258-5_104. URL `https://link.springer.com/chapter/10.1007%2F978-3-540-37258-5_104?LI=true#citeas`