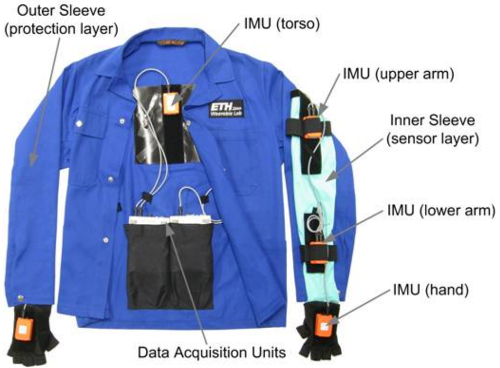


Figure 1: *Wearable motion jacket on which sensors are attached. Figure taken from [1].*

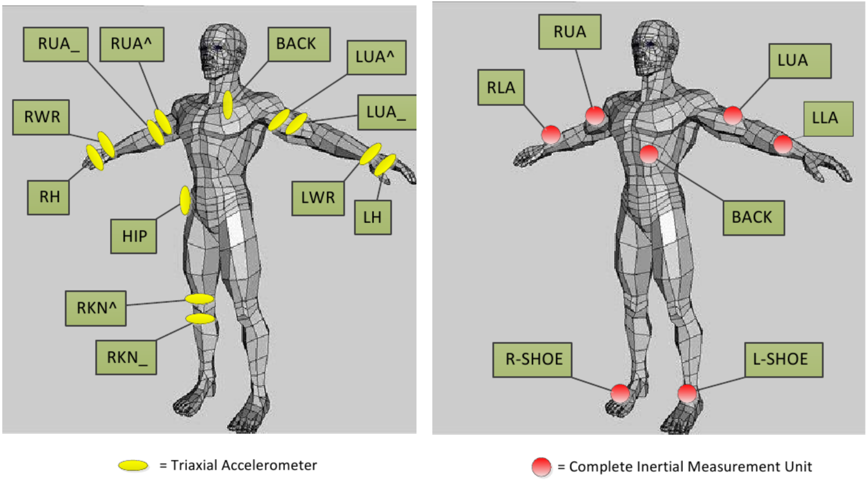


Figure 2: *Body sensor placement over the subject, for what concerns Inertial Measurement Unit on the right and Triaxial accelerometers on the left. Figure taken from [1].*

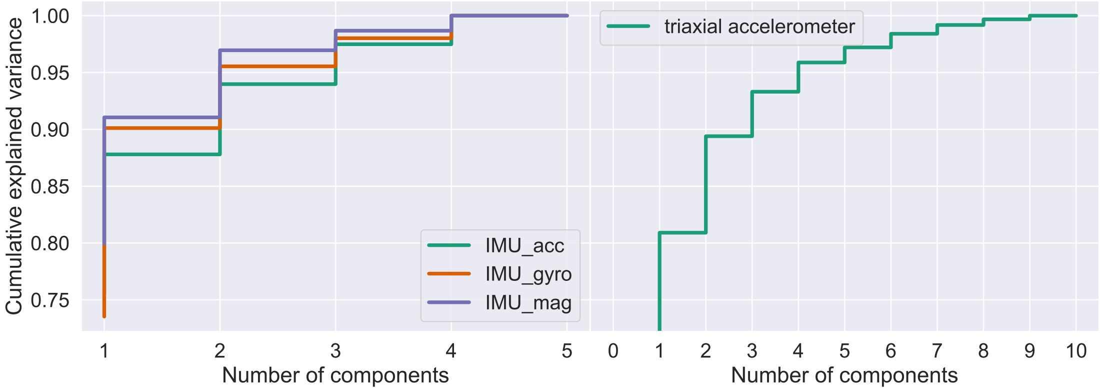


Figure 3: *Cumulative explained variance of each component for subject 1, run 1. Left panel shows the explained variance referring to different IMU sensors, while right panel refers to triaxial accelerometers.*

variance refers to PCs, not to the original feature data. This is true, but since PCs are a linear combination of the physical signals, the first reflect the behaviour the latter. Moreover, one could also directly work with PCs, applying the natural dimensionality reduction induced by PCA. Nevertheless, it is relevant to highlight that PCs do not have a clear physical meaning, since these are obtained by means of a geometric transformation of the original data and live in a geometric space with a different base.

4. Dimensionality reduction

4.1. KMeans Clustering

In the previous section we asserted that it is possible to use data coming from a lower number of sensor for each type still preserving the variety of original signals. In this section we drive deeper into this possibility analyzing dimensionality reduction from a different point of view: KMeans clustering [4] [5]. The basic idea is that, for a given number of centers, KMeans identifies the optimal centers with reference to a given metric. These centers are addressed as centroids, and they are computed at each iteration of KMeans algorithm, which behaviour is based on optimizing the following loss function:

$$\Phi(P, S) = \sum_{x \in P} d^2(x, S), \tag{3}$$

where P is the set of points which has to be clustered, d is the metric of the metric space and S is the set of centers. We consider both euclidean and dynamic time wrapping distance [6].

4.1.1 Same sensor type analysis

Here we consider data referring to two locomotion activities, walking and laying, and for each millisecond we apply KMeans clustering with 1 to 4 centers. Figures 5, 6 and 7 shows the signals of the centers for the three types of sensors considered, i.e. accelerometers, gyroscopes and triaxial accelerometers. For what concerns IMU accelerometers, Figure 5 shows that considering more centers does not add significant information to the single center case, since trends are reasonably overlapping. Moreover, it is possible to highlight that even considering the signals of the centers instead of the original time-series clearly allows to distinguish between the two locomotion activities. Indeed, one could just look at the amplitudes to visually discriminare if the subject is walking or laying. A similar behaviour can be identified also in Figure 6 with reference to gyroscopes and in Figure 7 for triaxial accelerometers, nevertheless, in this last case the addiction of the second center seems to add information to the single center case. Such observation is coherent with what we observed in the previous section applying PCA: triaxial sensors cannot be reduced to a single signal, thus we need to consider more than one time-serie to preserve the information content. Similar results can be obtained for other subjects and runs.

To summarize, two main results can be highlighted: first, KMeans clustering allows to reduce the number of signals for each sensor type, and the time-series of the centroids still allows to distinguish the locomotion activities performed by the subject. Indeed, it is worth to specify that the signals of each center are not physical, in the sense that these do not come directly from measurements, but they are computed as KMeans centroids.

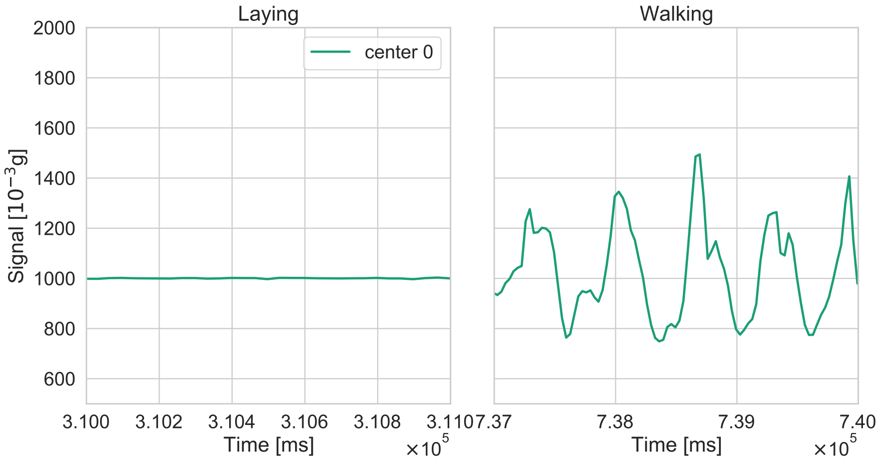
4.2. Homogeneous sensor type classification

Clustering allows to visually explore the effects of a dimensionality reduction, nevertheless, we are interested in providing a quantitative estimation of the information leak. To this aim, in this section we try to give an answer to the following question: how well can we still distinguish high level activity after clustering? The approach is straightforward: we train a binary classifier, binary for sake of simplicity, on part of the original features and validate its performances on a test subset. Finally, we test the accuracy of our model on the data obtained from the signals of the centroids obtained through KMeans.

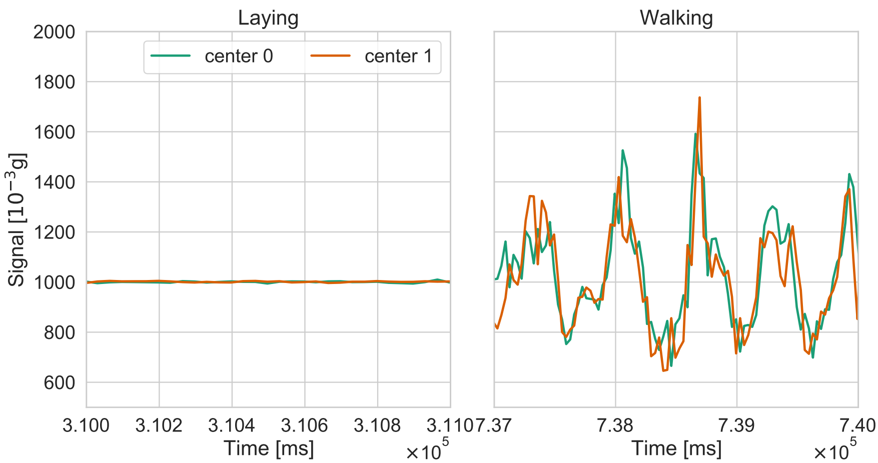
Two strategies are exploited: linear classifier on the amplitude of the signals and a neural model on the entire time-series.

4.2.1 Linear model: logistic regression

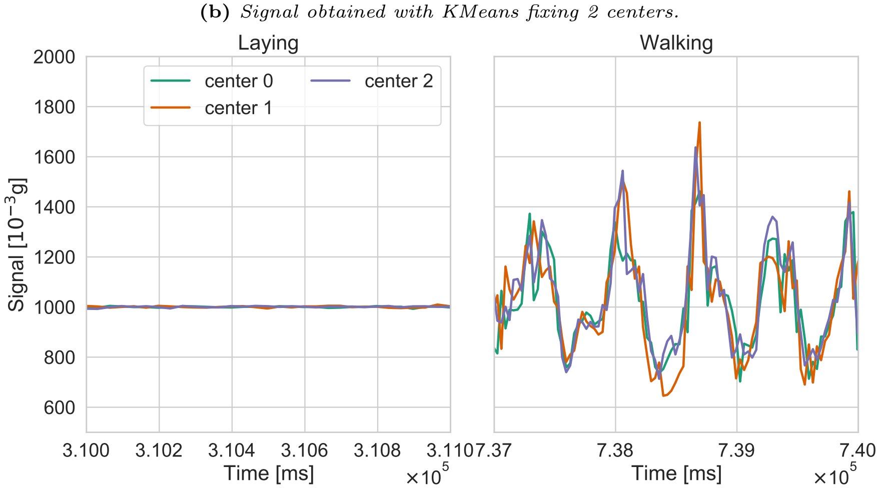
Let us first explore an approach based on performing binary classification on amplitudes. The main idea is to create a dataset of ampli-



(a) *Signal obtained with KMeans fixing 1 center.*

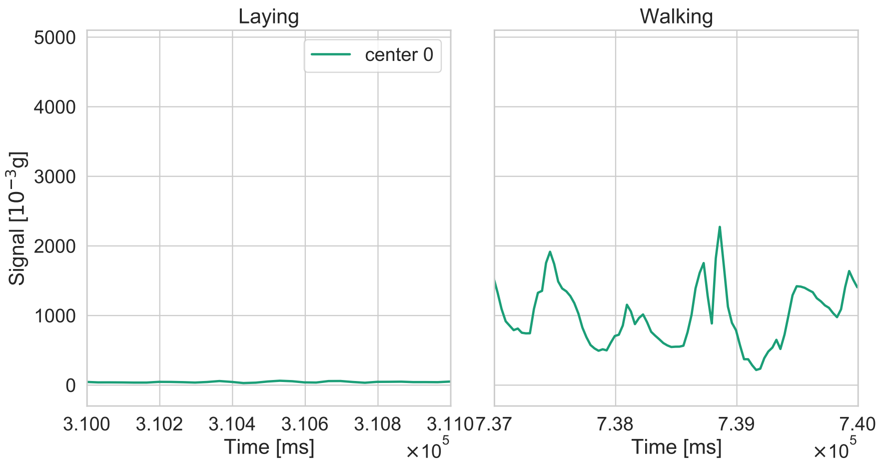


(b) *Signal obtained with KMeans fixing 2 centers.*

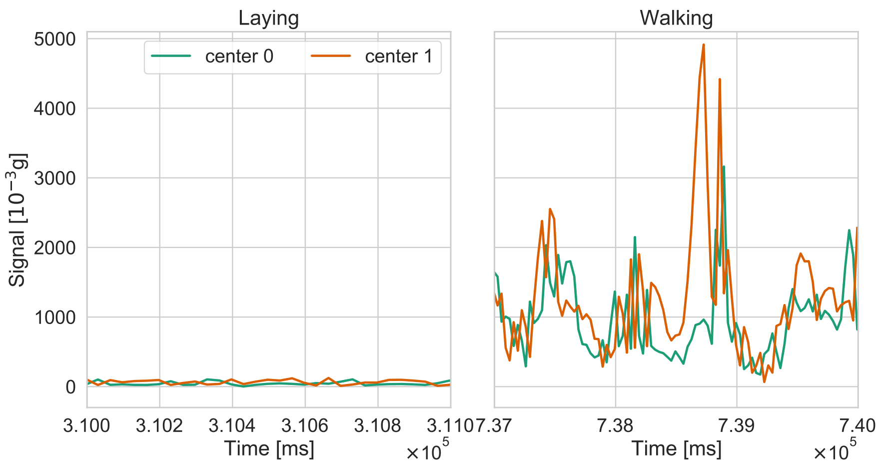


(c) *Signal obtained with KMeans fixing 3 centers.*

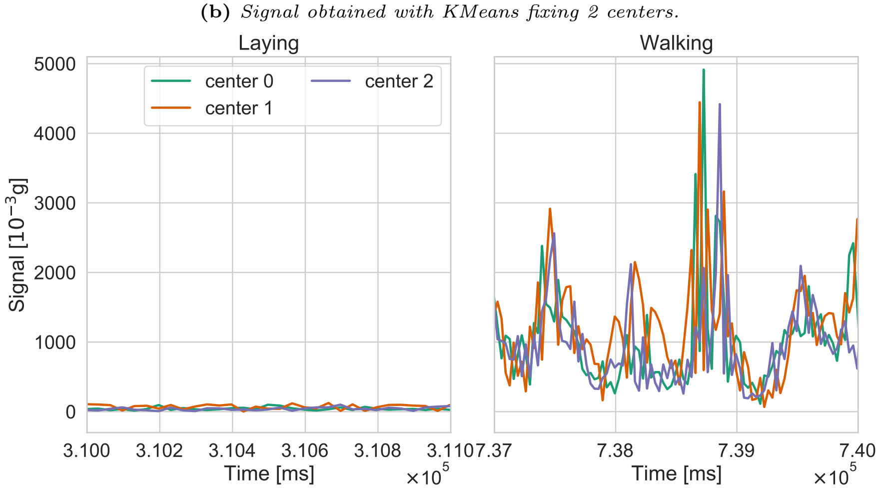
Figure 5: *Signals of the centers obtained applying KMeans on IMU accelerometers for different numbers of clusters. Plots obtained for subject 1, run 1*



(a) *Signal obtained with KMeans fixing 1 center.*

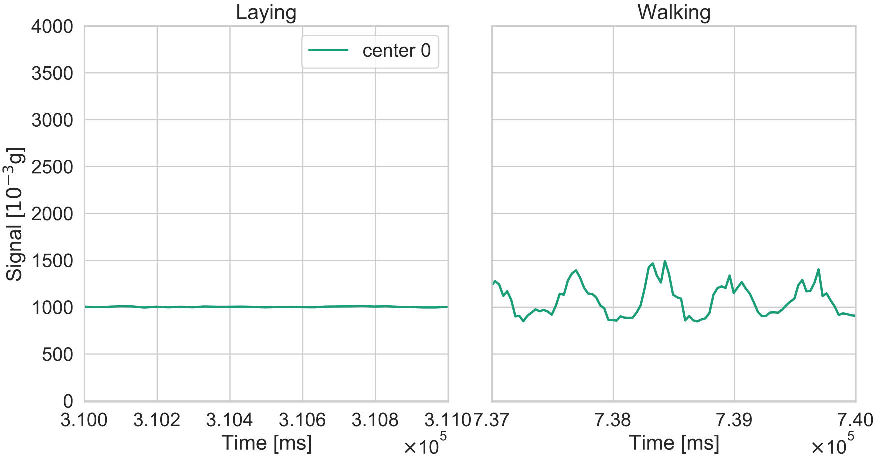


(b) *Signal obtained with KMeans fixing 2 centers.*

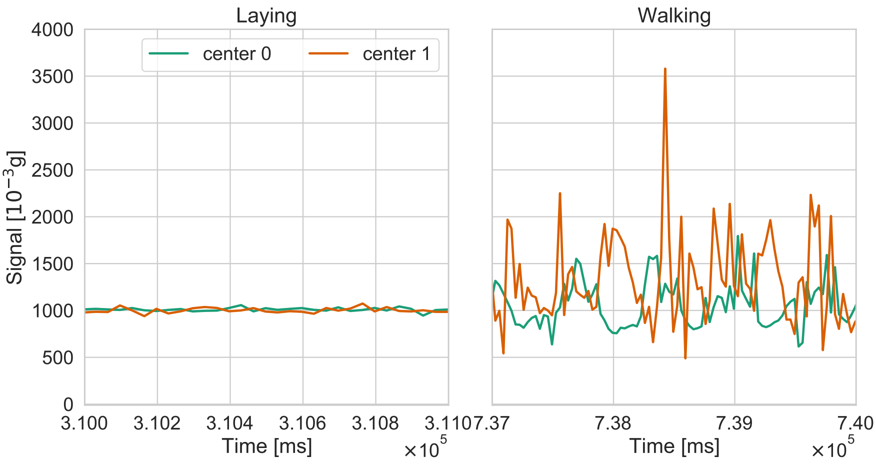


(c) *Signal obtained with KMeans fixing 3 centers.*

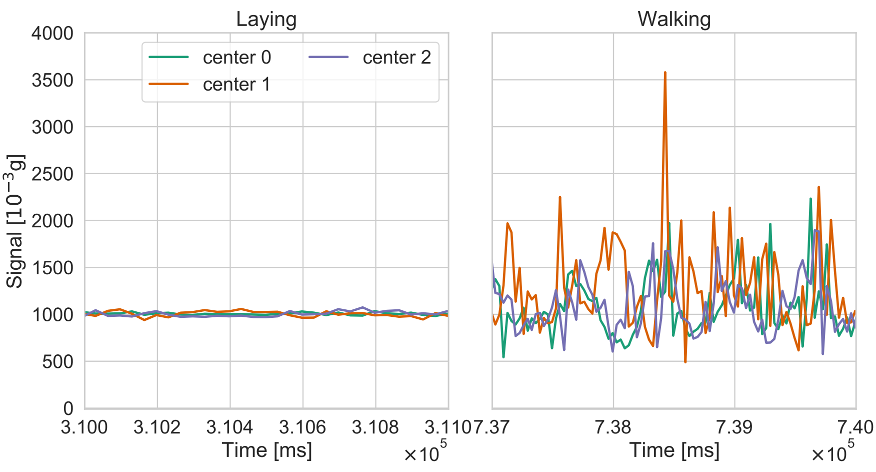
Figure 6: *Signals of the centers obtained applying KMeans on IMU gyroscopes for different numbers of clusters. Plots obtained for subject 1, run 1*



(a) *Signal obtained with KMeans fixing 1 center.*



(b) *Signal obtained with KMeans fixing 2 centers.*



(c) *Signal obtained with KMeans fixing 3 centers.*

Figure 7: *Signals of the centers obtained applying KMeans on triaxial accelerometers for different numbers of clusters. Plots obtained for subject 1, run 1*

tudes labeled with the corresponding locomotion activity, walking or laying, train a logistic regression on the classification task and test in on the cluster data.

Dataset extraction For each sensor type, let us collect all the time series referring to walking, class 1, and laying subjects, class 0. We set a window of 80 ms, and for each interval we compute the amplitude of the signal as follows:

$$\begin{gathered} A_W = |X_{max} - X_{min}|, \quad \text{with} \\ X_{max} = \max_{x \in W} x, \quad X_{min} = \min_{x \in W} x, \end{gathered} \tag{4}$$

where W is a fixed window.

Same process is applied to the signals obtained from clustering. At the end we have three datasets: train data, i.e. 80% of amplitudes extracted from original data, test data, the other 20%, and the clustering dataset.

Train and testing on original data The model is trained and tested on the original dataset, results in terms of accuracy are shown in Tab. 1, while confusion matrices are shown in Figure 8. Note that, if we refer to 'walking' as positive and 'laying' as negative examples, the model shows a non negligible false positive rate. This is probably due to the fact that the amplitude is still large when there is a transition from a certain locomotion activity to laying down.

Test on centroids amplitudes Finally, in order to quantify how good be a classification could be after applying KMeans clustering, we compute accuracy on the centroids dataset, for each sensor and for each number of centers considered. Results are shown in Figure 9. The main outcome is that for IMU sensors a single center allows to distinguish the two locomotion activities with extremely high probability, while more centers are needed to reach the same accuracy in the case of triaxial accelerometers. Note that these observations are perfectly coherent with what we observed in PCA and heuristically by simply plotting the signals of the centers in Figure 7.

4.2.2 Neural model: InceptionTime

Let us now explore a second approach based on binary classification of the entire time-series, to this end, more delicate and sophisticated tools are needed. We implement a binary classifier of time-series by means of a specific Python module [7] with InceptionTime architecture [8].

Dataset and training In Figure 10 we show the original measurements used to train the model, training is performed with 4 epochs and learning rate 10^{-4}.

Test on centroids time-series Testing the neural model on the signals obtained from clustering returns always a 100% accuracy and a diagonal confusion matrix. Thus, we may conclude that by means of neural models it is possible to distinguish with perfect precision the locomotion activity also on the clustered data.

Nevertheless, neural architectures are more delicate than logistic regressors, since they need to be fine tuned and usually require more computational time. As a consequence, since in a IoMT scenario we are interested in transmitting and processing data quickly and with the fewer number of assumption possible, the logistic regression turns out to be better suited for the problem.

4.3. Heterogeneous sensor type classification

Let us now consider an heterogeneous scenario, gathering data from different sensors, in particular we consider RUA, RLA, and $BACK$ sensors for the IMU measurements and hip, back, RUA, RUA, RWR, RKN for the triaxial accelerators. We apply the same process already discussed in Section 4.2. Figure 11 shows accuracy obtained by the linear regressor on the clustering dataset for different number of clusters used in KMeans.

References

[1] Mirco Rossi Thomas Holleczek Gerhard Tröster Paul Lukowicz Gerald Pirkl David Bannach Alois Ferscha Jakob Doppler Clemens Holzmann Marc Kurz Gerald Holl Ricardo Chavarriaga Hesam Sagha Hamidreza Bayati Daniel Roggen, Alberto Calatroni and José del R. Millàn. Collecting complex activity data sets in highly rich networked sensor environments.

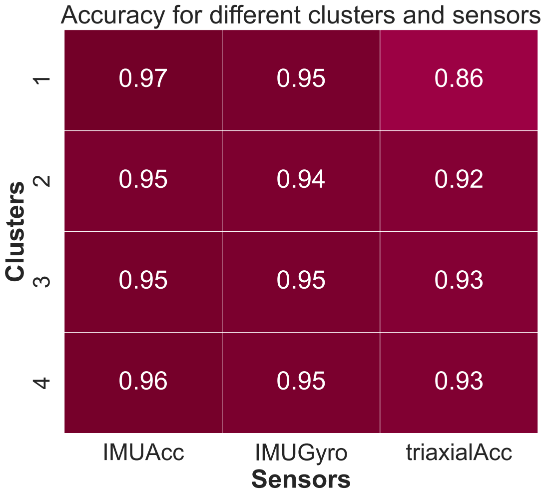
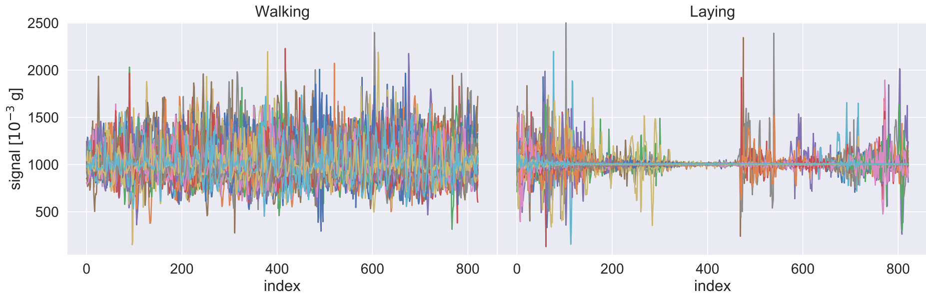
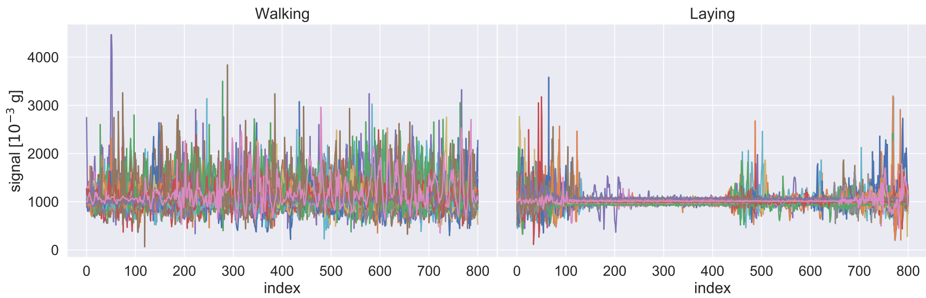


Figure 9: *Accuracy of the logistic model for each sensor type and number of centers considered for clustering.*



(a) *Signals of IMU accelerometers*

(b) *Signals of IMU gyroscopes.*



(c) *Signals of triaxial accelerometers*

Figure 10: *Dataset of original signals used to train InceptionTime module for binary classification, signals are divided according to the sensor type.*

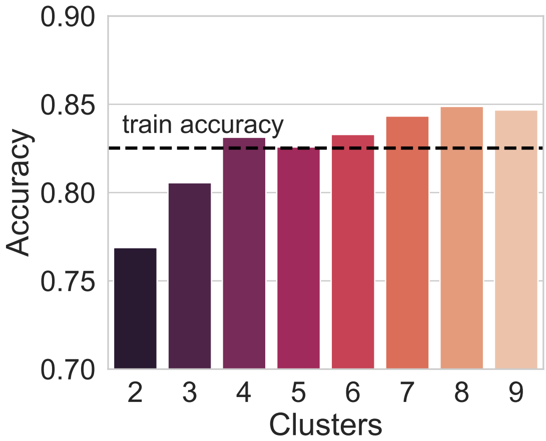


Figure 11: *Linear regression accuracy.*

for time series classification. *Data Mining and Knowledge Discovery*, 34(6): 1936–1962, sep 2020. doi: 10.1007/s10618-020-00710-y. URL `https://doi.org/10.1007%2Fs10618-020-00710-y`.