

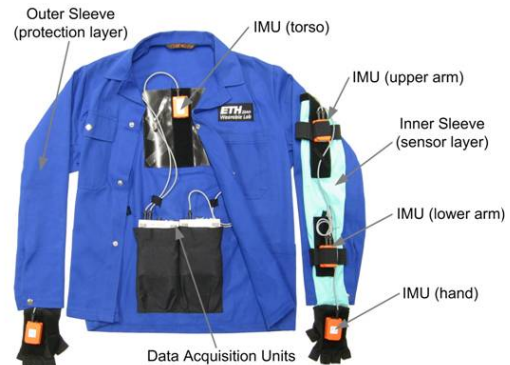
# Efficient information distribution in Internet of Medical Things (IoMT) scenarios

Tullia Fontana, Nicolás Ortiz De Zarate, Nicole Zattarin

## ABSTRACT

The Internet of Medical Things (IoMT) is playing a central role in the healthcare industry to improve the living conditions of individuals through suitable technological solutions. Clearly, such advanced systems work by processing complex data continuously produced across a variety of different scenarios, such as physical and environmental signals.

Nevertheless, data coming from all of these sensors can easily saturate the capacity of communication networks, which makes it necessary to design a proper transmission process capable of preserving the reliability of the network at the cost of lowest possible loss of information.



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

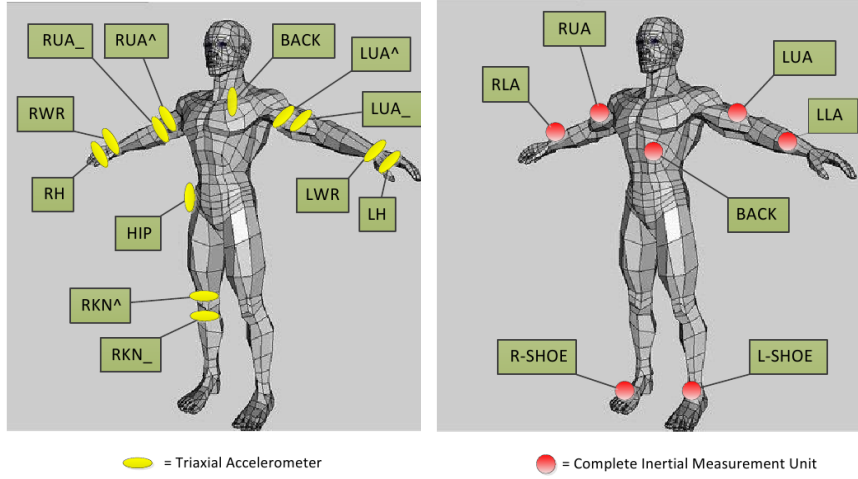
## 2. DATASET

### 1. INTRODUCTION

The purpose of this work is to use Machine Learning (ML) tools to assert in a quantitative way if it is possible to reduce the amount of data transmitted with a reasonable loss of information. First, we show that many of the signals produced are correlated, thus similar sensors may share the same information. As a consequence, in a practical scenario, this observation suggests that it is possible to transmit only a few signals with a reasonable loss of information content. Second, we aim to quantify how much information can still be extracted by reducing the dimensionality of the dataset. To this end, a possible approach is represented by testing how well we are still able to perform a classification task on a reduced version of the original dataset. This allows us to quantify the loss of information we are ready to pay in order to limit the amount of data shared among a channel.

### 2.1. Dataset description

We refer to the OPPORTUNITY Activity Recognition Dataset [1], which is designed to benchmark human activity recognition algorithms such as classification, automatic data segmentation, sensor fusion, feature extraction, etc. The dataset combines a collection of signals coming from different motion sensors recorded while users execute typical daily activities. In particular we focus on a subset of the recorded runs, to which we refer to as Activity of Daily Living (ADL), in which the subjects are asked to do the following activities: sit, move around the room, go out for a walk, prepare and have a coffee, prepare and have a sandwich, clean up and lay down. Data are collected by means of body worn sensors, see Figure 1, which provide 3D measurements. For our purposes, we consider Inertial Measurement Units (IMU) accelerators and gyroscopes, and triaxial accelerometers, see Figure 2, with particular focus on different locomotion activities, e.g. walking, laying.



**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].

## 2.2. Data preprocessing

Since each of the sensor provides 3D measurements, we introduce a first approximation by taking the euclidean modulus of the three components, as follows:

$$M = \sqrt{x^2 + y^2 + z^2}, \quad (1)$$

for each of the considered sensors. Therefore, from now on we will always implicitly refer to the modulus instead of the single component. Moreover, let us point out that the dataset contains a non-negligible amount of missing values, nevertheless, since our analyses take into account different subsets of the entire dataset, the NaN values problem is addressed differently each time.

## 3. CORRELATIONS: PRINCIPAL COMPONENT ANALYSIS (PCA)

Let us recall that the purpose of our analysis is to provide a strategy to reduce the amount of data to share among a communication channel, with the lowest possible loss of information. To this end, we first need to show that there is indeed shared information among different signals, thus it is reasonable to approach the problem of data reduction. A possible strategy is presented by Principal Component Analysis (PCA) [2], a dimensionality reduction technique which can be applied to transform data in a space of lower dimension, but also as a

mathematical tool to compute how much variance of the data can be explained by means of the so-called Principal Components (PC).

Let us then recall a well-known quantity in the PCA scenario: the explained variance. The explained variance is a statistical measure of how much variation in a dataset can be attributed to each of the principal components generated by PCA, in the formula:

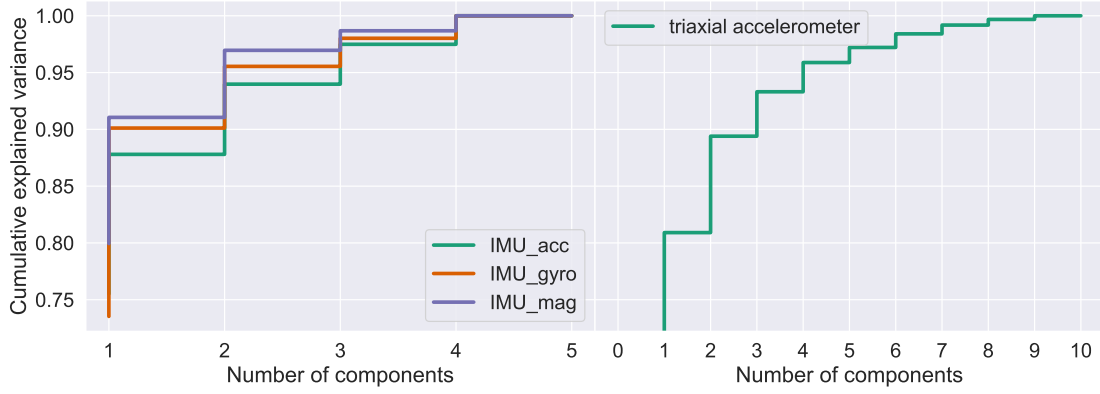
$$EV_i = \frac{\lambda_i}{\sum_{k=0}^{d-1} \lambda_k}, \quad (2)$$

where,  $\lambda_k$  is the eigenvalue of the  $k$ -th component and  $d$  is the dimension of the new feature space.

### 3.1. Homogeneous sensor type analysis

For each of the sensor type, we perform a PCA [3] from a space of dimension  $n$ , without lowering the dimensionality, and we compute the explained variance. Such quantity gives indeed an estimation of how much variance of the original dataset is encoded in each component, thus, indirectly, it refers to the amount of correlation of the original features.

Figure 3 shows an example of cumulative explained variance vs number of components considered. For what concerns the IMU signals we can highlight that, for all of them, a single component explains around 90% of the variance, while by considering two components we are able to describe approximately 95% of data variety. On the other hand, in order to achieve



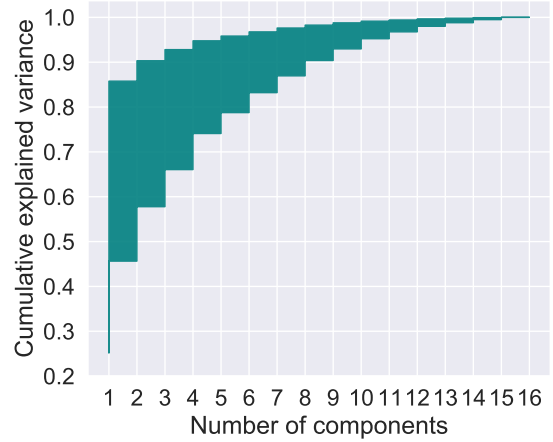
**Figure 3:** Cumulative explained variance of each component for subject 1, run 1. Left panel shows the explained variance referring to different IMU sensors, while right panel refers to triaxial accelerometers.

a 90% of explained variance in the triaxial accelerometers scenario we need to consider at least two components. Therefore, since PCs are obtained by means of a linear transformation of the original features, we can assert that in the case of IMU signals a single component carries almost 90% of the information, while for what concerns the triaxial accelerators we have to consider more components. PCA applied to different runs and subjects leads to comparable results.

From a physical point of view such observations suggest that there is high correlation among same sensor data, thus that it is possible to consider only a few sensors for each type. Clearly, more sensors are needed in the triaxial accelerometers case, since there are originally more signals to combine. On the other hand, we could use only data coming from one or two IMU sensors of each type, with a reasonable loss in terms of information.

### 3.2. Heterogeneous sensor type analysis

As a further analysis, we consider *RUA*, *RLA*, and *BACK* sensors for the IMU measurements and hip, back, *RUA*, *RUA*, *RWR*, *RKN* for the triaxial accelerators, and perform PCA on all these signals gathered together. Figure 4 shows results for all the different combinations of runs and subjects. The main outcome is that, in order to explain at least 90% of variance in every condition, 8 components are needed in the worst case scenario.



**Figure 4:** Cumulative explained variance of each component, computation performed with all the different sensor types. The filled area represents the area between the minimum and the maximum for each component, among different combination of subject and run.

Finally, one could argue that the explained variance refers to PCs, not to the original feature data. This is true, but since PCs are a linear combination of the physical signals, they 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 while still preserving the variety of original signals. In this section we dive 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 referred to as centroids, and they are computed at each iteration of the KMeans algorithm, whose behaviour is based on optimizing the following loss function:

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

where  $P$  is the set of points 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 discriminate 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 addition of the second center seems to add information to the single center case. Such an observation is coherent with what we observed in the previous section when applying PCA: triaxial sensors cannot be reduced to a single signal, thus we need to consider more than one time-series 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 us to reduce the number of signals for each sensor type, and the time-series of the centroids still allows us to distinguish the locomotion activities performed by the subject. Indeed, it is worthwhile 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.

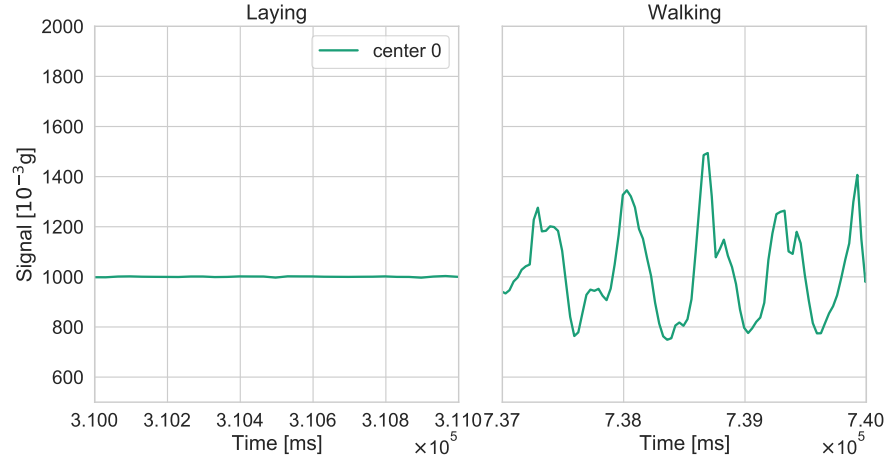
### 4.2. Homogeneous sensor type classification

Clustering allows us to visually explore the effects of a dimensionality reduction. Nevertheless, we are interested in providing a quantitative estimation of the information loss. 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.

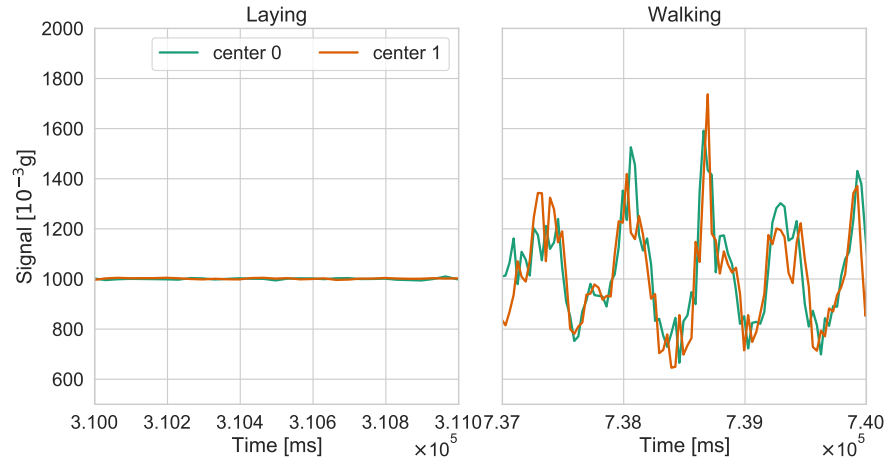
Two strategies are exploited: a linear classifier on the amplitude of the signals and a neural model on the entire time-series.

#### 4.2.1 Linear model: logistic regression

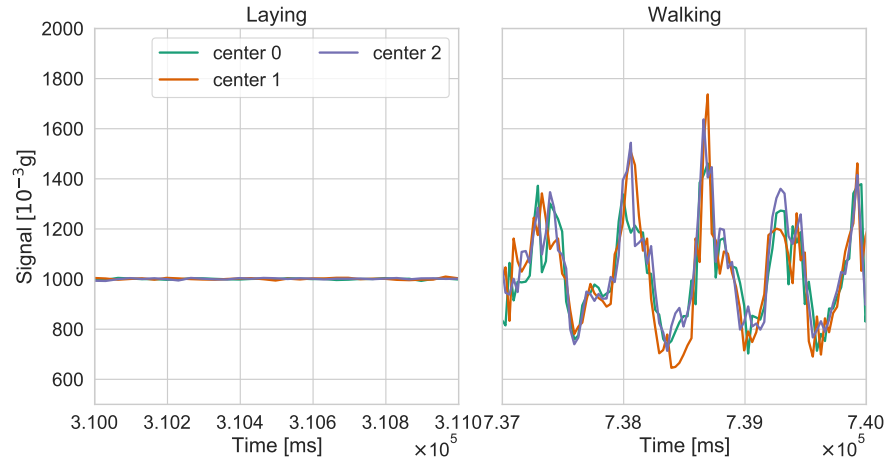
Let us first explore an approach based on performing binary classification on the amplitudes. The main idea is to create a dataset of amplitudes labeled with the corresponding locomotion



(a) Signal obtained with KMeans fixing 1 center.

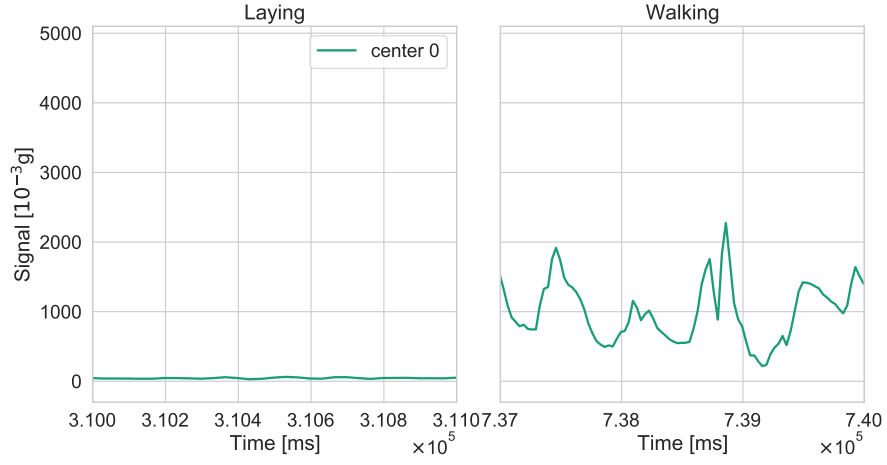


(b) Signal obtained with KMeans fixing 2 centers.

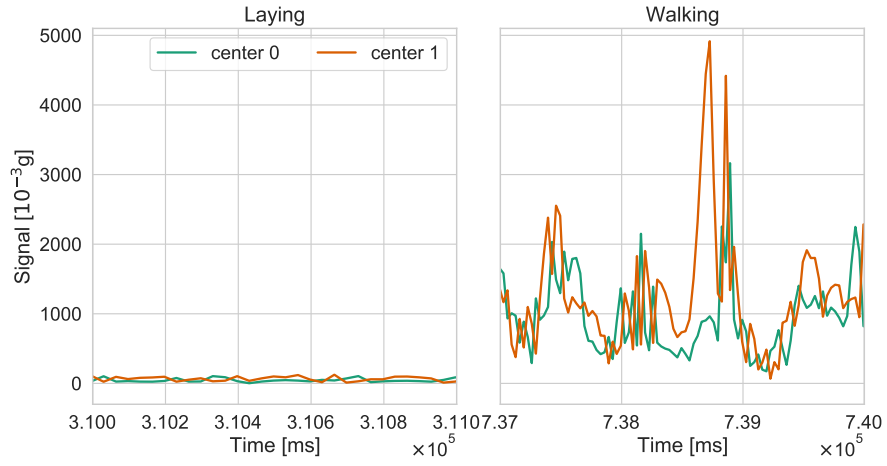


(c) Signal obtained with KMeans fixing 3 centers.

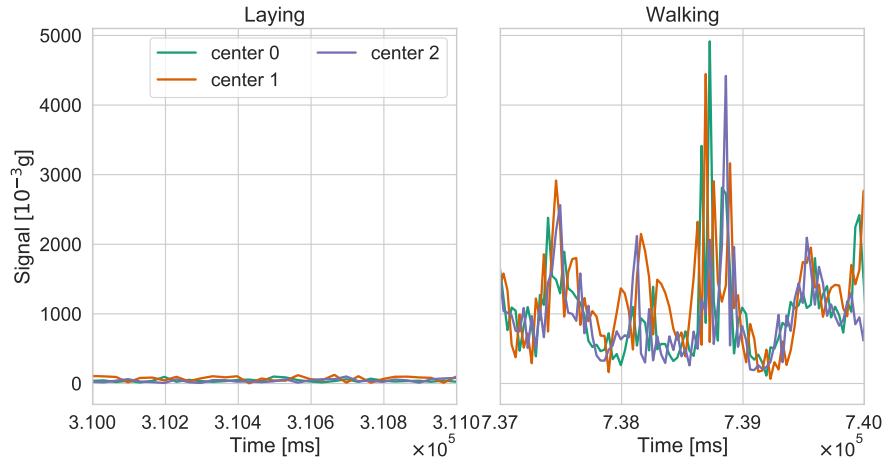
**Figure 5:** Signals of the centers obtained applying KMeans on IMU accelerometers for different numbers of clusters. Plots obtained for subject 1, run 1



(a) Signal obtained with KMeans fixing 1 center.

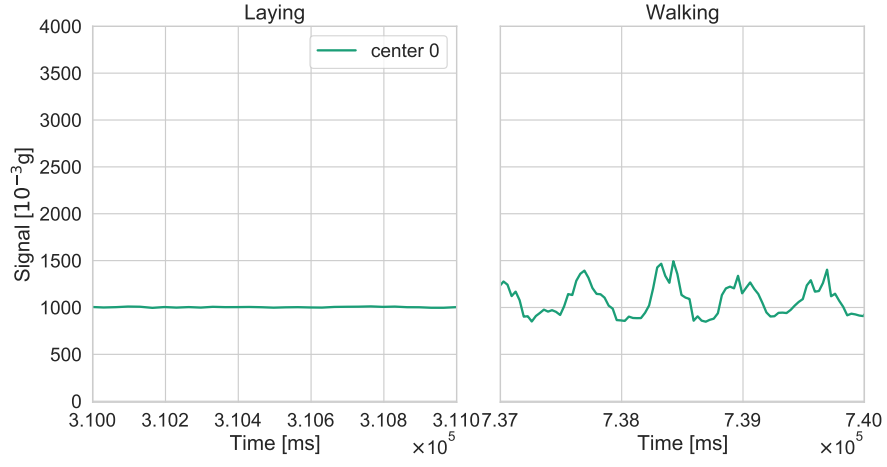


(b) Signal obtained with KMeans fixing 2 centers.

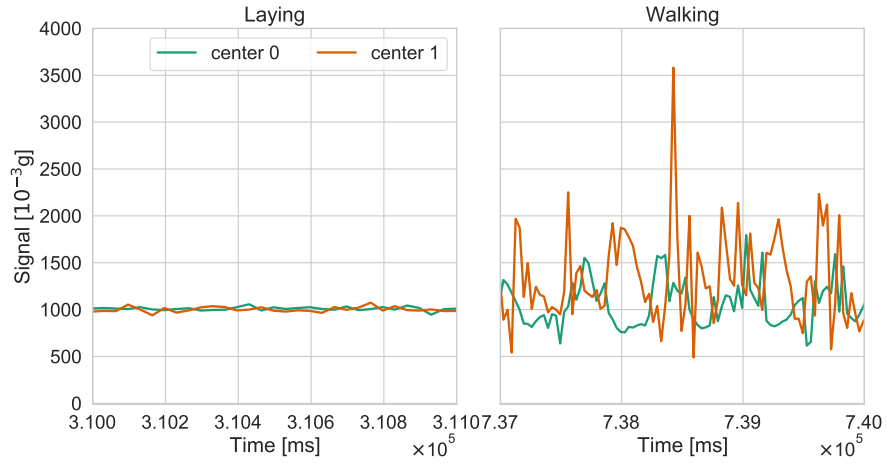


(c) Signal obtained with KMeans fixing 3 centers.

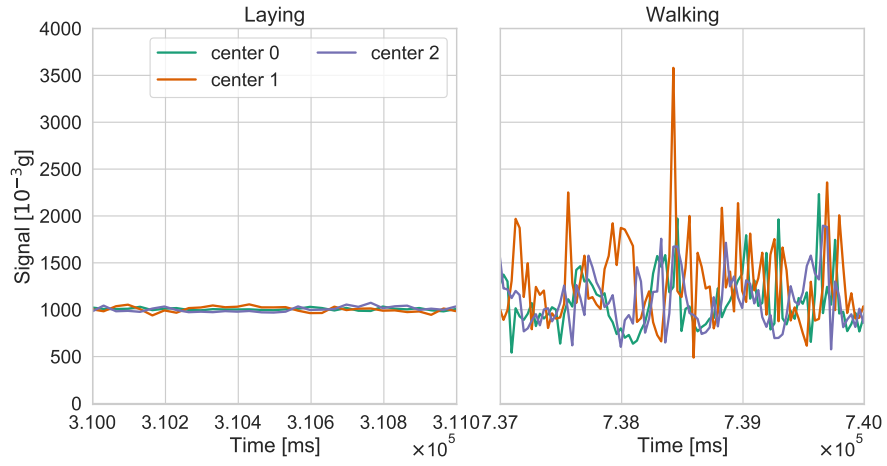
**Figure 6:** Signals of the centers obtained applying KMeans on IMU gyroscopes for different numbers of clusters. Plots obtained for subject 1, run 1



(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

tion activity, walking or laying, train a logistic regression on the classification task and test it on the cluster data.

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

$$A_W = |X_{max} - X_{min}|, \quad \text{with} \quad (4)$$

$$X_{max} = \max_{x \in W} x, \quad X_{min} = \min_{x \in W} x,$$

where  $W$  is a fixed window.

The 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, with the 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 a classification could be after applying KMeans clustering, we compute the accuracy on the centroids dataset, for each sensor and for each number of centers considered. The results are shown in Figure 9. The main outcome is that for IMU sensors a single center allows us 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,

and 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 a learning rate of  $10^{-4}$ .

**Test on centroids time-series** Testing the neural model on the signals obtained from clustering always returns 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 an IoMT scenario we are interested in transmitting and processing data quickly and with the fewest number of assumptions 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 the accuracy obtained by the linear regressor on the clustering dataset for different number of clusters used in KMeans.

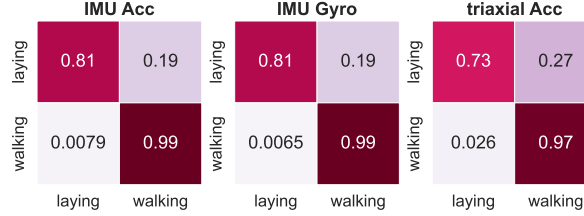
### 5. COMMUNICATION TECHNOLOGIES

After the ML analysis, we aim to project an IoMT platform to connect our sensors to the Internet. In the last few years, IoMT has faced new challenges, both in terms of network design and dimensioning. In particular, a path of research is focusing on the development of communication technologies optimized for low-power operations, i.e. long battery life, and reduced device complexity [? ].

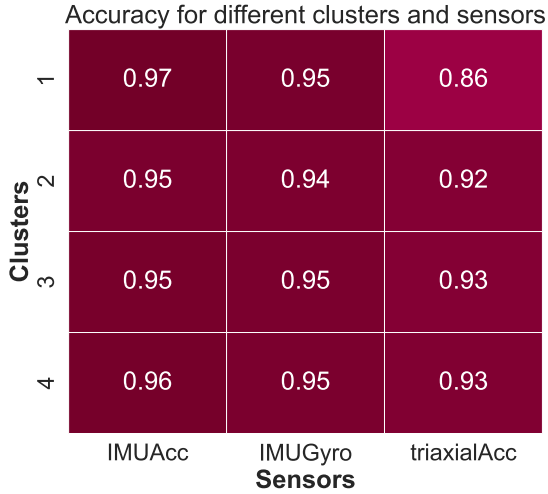


	test accuracy	train accuracy
IMU Accelerometer	0.96	0.96
IMU Gyroscope	0.95	0.95
triaxial Accelerometer	0.93	0.92

**Table 1:** Test and train accuracy of a linear regressor for each sensor type.



**Figure 8:** Confusion matrices on test set for each sensor type logistic classifier.



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

In order to achieve such purposes, various supported connectivity solutions have been developed. Each proposed technology relies on the usage of different range, data rate, power restrictions, encryption level, coding and transmission schemes, modulation and several other aspects. Therefore, in the following sections, we explore one of the most promising technologies in the field. In particular, we decide to focus on long-range communication technologies with low energy consumption requirement, the LPWA (Low-Power Wide Area) technologies [? ]. We perform a simulation and set realistic parameters for our problem. The main analysis focuses on

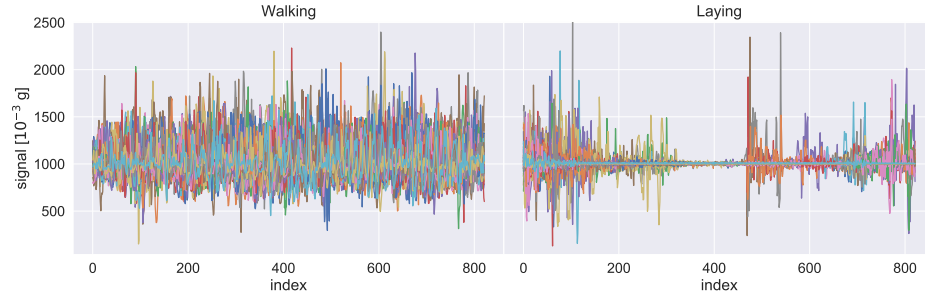
studying network metrics, e.g. throughput and packet error rate (PER), for different numbers of sensors that transmit information. Eventually, we compare these values with the accuracy obtained with the ML analysis.

### 5.1. LPWA technologies

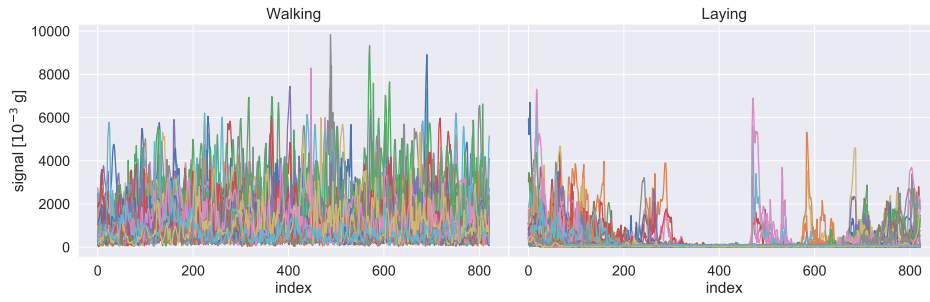
The IoT communication range can cover several tens of kilometers, since a typical IoT scenario is required to work in large operation area. Therefore, wireless devices need more power supply to communicate over large distance. Due to such issue, it is critical for IoT applications to employ low-complexity and low-cost end devices, in order to perform low power operations. A possible strategy to address these problems is offered by Low-Power Wide Area Network technologies (LPWANs), which have the capability to implement architectures with the characteristics required by an IoT scenario. [? ].

Moreover, in recent years, several technologies have been developed to achieve sustainable coverage for large scale IoT devices, both in the licensed and in non-licensed spectrum [? ]. In the licensed band, we have LTE-M and narrow-band IoT (NB-IoT). On the other hand, in the unlicensed bands, SigFox and LoRa are two prominent technologies, which are increasingly deployed in different IoT applications.

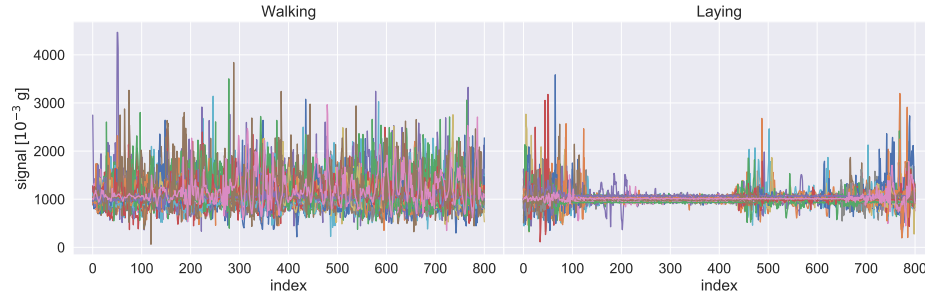
For instance, in 2018, Global Market Insight published a market research that showed the recent increasing of the LPWAN market. In particular, at the state of art LoRaWAN has the biggest portion of the market share, ap-



(a) Signals of IMU accelerometers

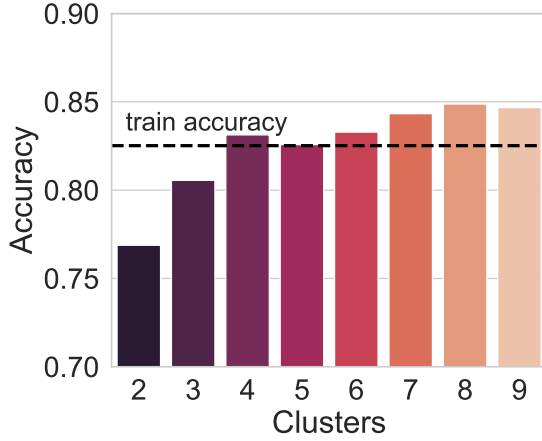


(b) Signals of IMU gyroscopes.



(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 in an heterogenous scenario. We consider RUA, RLA, and BACK sensors for the IMU measurements and hip, back, RUA, RUA, RWR, RKN for the triaxial accelerators.*

proximately 50 %. Thus we decide to focus on the LoRa technology for our analysis. Nevertheless, it is worth to highlight that NB-IoT is expected to get to the 30 % of the market share by 2025 [? ].

#### 5.1.1 LoRaWAN

LoRaWAN is a Low Power and Wide Area networking protocol based on LoRa technology, in order to connect battery operated device to the internet [? ].

It is designed to wirelessly connect battery operated ‘things’ to the internet in regional, national or global networks. The network architecture of LoRaWAN is usually a star-of-stars topology in which gateways deliver messages from end-devices to central network server and vice versa [? ]. Figure 13 shows the LoRaWAN architecture with End-Devices, Gateways, Network Server and Application Server.

To provide long-range communication, LoRaWAN uses LoRa modulation which is based on Chirp Spread Spectrum (CSS). Moreover, it provides a data rate varying between 300 bps and 50 kbps. The maximum payload is 243 bytes and the number of messages transmitted both in the Uplink and in the Downlink per day is unlimited. The battery life of end devices can go beyond 10 years [? ]. Now that we have introduced the theoretical scenario, in the following section we are going to simulate a simple LoRaWAN with NS-3, on

Simulation parameters	Values
Data Rate	1000 bps
Gateway	1
End devices	from 4 to 86
Radius	7000 m
Simulation time	15 s

**Table 2:** *Simulation parameter.*

the basis of the parameters of the OPPORTUNITY IoMT scenario [? ].

## 6. LoRa NS-3 SIMULATION

Let us consider the scenario offered in the OPPORTUNITY dataset, in which data is generated every 33 ms. If each sample is quantized at 32 bits, every sensor would approximately produce samples at a rate of 960 bps. This is just a possible approach, eventually one could explore other possibilities. In [? ], we found that a typical IoMT data rate has an order of magnitude of a few hundreds bps, as we can see in Figure 14. Therefore, we decided to assume that our sensor sends data at a rate with an order of magnitude of thousands of bits per second. This is compatible with the choice of the LoRaWAN technology, which makes available a data rate that goes from 300 bps to 50 kbps.

### 6.1. Implementation

To simulate the network with sensors, we use the LoRa simulator projectLoRaWAN [? ], based on the initial version of a code developed by the Signet Lab at University of Padova [? ]. The parameters are shown in Table 2.

Let us consider a smart city scenario with several buildings, in which we introduce a channel model with two non-idealities: path loss and shadowing. In such scenario the End Devices, i.e. sensors, are created. Since we assume that these are worn by a person, sensors are placed in the same starting position at a height of 1.2 m and connected to the channel. Moreover, the gateway node is set in the position (0 m, 0 m, 15 m) and connected to the channel. Finally, the application is installed on the End Devices in order to generate a packet of size 23 bytes every second.

	LoRaWAN	Sigfox	NB-IoT	LTE Cat-M1
Coverage (MCL)	157 dB	162 dB	164 dB	155 dB
Technology	Proprietary	Proprietary	Open LTE	Open LTE
Frequency spectrum	Unlicensed	Unlicensed	Licensed (LTE/any)	Licensed (LTE/any)
Duty cycle limitation	Yes	Yes	No	No
Transmission power restrictions	Yes (14 dBm = 25 mW)	Yes (14 dBm = 25 mW)	No (23 dBm = 200 mW)	No (23 dBm = 200 mW)
Data rates in DL	0.3-50 kbps	< 1 kbps	0.5-27.2 kbps	< 300 kbps
Data rates UL	0.3-50 kbps	< 1 kbps	0.3-32.25 kbps	< 375 kbps
Max. message size UL	243 B	12 B	1280 B	1280 B
Max. message size DL	243 B	8 B	1280 B	1280 B
Battery life / Current consumption	8+ years <2 uA	10+ years <2 uA	10+ years <3 uA	10+ years <8 uA
Module cost	<\$ 10	<\$ 10	\$ 10 (2019)	<\$ 25 (2019)
Security	Medium (AES-128)	Low (AES-128)	Very high (LTE Security)	Very high (LTE Security)

Figure 12: LPWAN technologies.

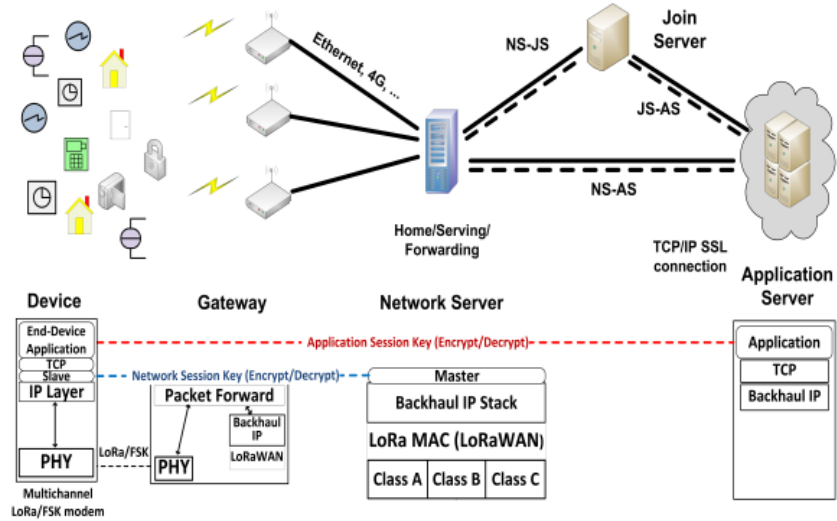


Figure 13: LoRaWAN Network Architecture.

WBAN Applications	Signals	Data Range	Frequency (Hz)	Accuracy (bits)	Data Rate
Medical/Health	Sweating	-	4	12	48 bps
	Stress	-	50	12	600 bps
	Respiratory Rate	2-50 breaths/min	20	12	240 bps
	Pulse Rate	0-150 BPM	4	12	48 bps
	Blood Pressure	10-400 mm Hg	100	12	1.2 Kbps
	Blood pH	6.8-7.8 pH units	4	12	48 bps
	Body Temperature	24-44C	0.2	12	2.4 bps
Non-Medical	High Quality Audio	-	-	-	1.4 Mbps
	Voice	-	-	-	100 Kbps
	Video	-	-	-	1-2 Mbps
	GPS positions	-	1	32	96 bps
	Motion Sensor	-	100	16	4.8 Kbps

**Figure 14:** Typical data rate in wearable wireless sensor networks in IoMT.

## 6.2. Simulation and results

The network is simulated 10 times, each time with different seeds, for each number of devices. After the run, for every fixed number of sensors, we take the average of the metric among all 10 simulations. In Figure 15, we can see the outcomes of the simulation, in terms of packet error rate and the throughput.

As expected, Figure 15a shows that the throughput increases with the number of end devices connected. Nevertheless, we need to highlight that the considered number of sensors can still be managed by the network without overloading its capacity. For what concerns the Packet Error Rate, we can notice that it grows as soon as the number of end devices increases, see Figure 15b.

At this point it is worthwhile to go back for a second to the original concern of this work: how do we find the proper trade-off between information loss and the capacity of a network?

Even though we consider a simple scenario, we can still perceive several benefits due to the reduction of the number of signals that are shared among the channel. In particular, recall that in the OPPORTUNITY dataset we considered 24 sensors and we realized that most of them are highly correlated, and thus it is possible to reduce the number of employed sensors. From the network point of view, selecting a smaller group of sensors means having a lower packet error rate and not overloading the network with traffic which does not bring much

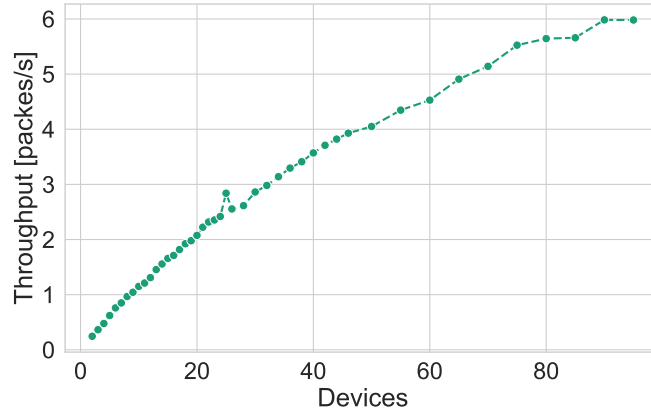
additional information. Therefore, here we aim at quantifying such gain in a LoRaWAN scenario.

In Figure 16, we show the percentage gain of the packet error probability that we observe considering a lower amount of sensors with respect to 24 devices. This is a possible way to quantify the benefits of employing less sensors in our simple scenario. Note in particular that considering 5 sensors instead of 24 we gain 15% probability of correctly delivering a packet.

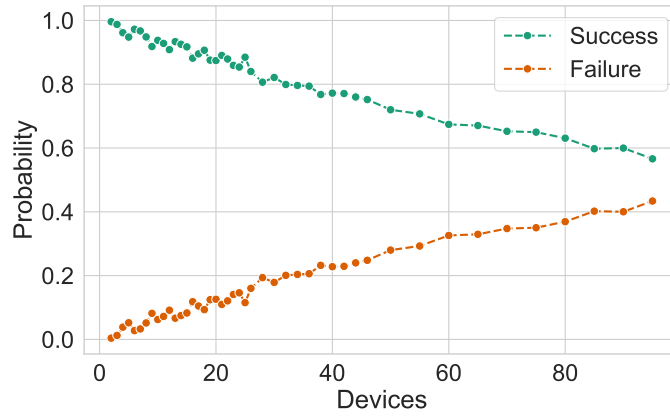
## 7. FURTHER STUDIES AND POSSIBLE IMPROVEMENTS

In addition to the simulation we performed, one could also consider to test NB-IoT technology. This possibility can be explored by means of the ongoing implementation of the new NB-IoT standard in ns-3 [? ]. Nevertheless, we choose LoRaWAN since the NB-IoT is more suitable to static sensors, or in general, to sensors that are not constantly moving.

Another possible, yet simpler, approach is represented by adopting a modified version of the LENA *lena-simple-epc.cc* example [? ] in NS-3, which can be adapted [? ] to LTE-M technology for IoT. Such strategy is interesting as a first exploration of the problem, but it leads to trivial results in terms of throughput and mean delay. Finally, further simulations could focus on making a performance comparison between LoRaWAN, NB-IoT technology

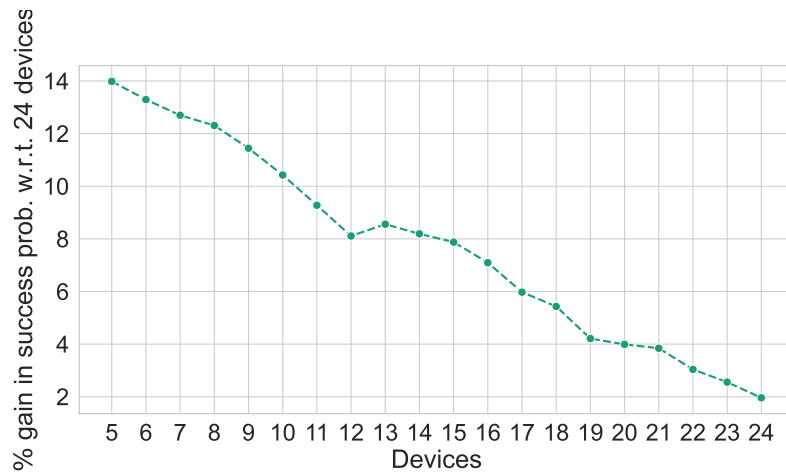


(a) *Throughput of LoRaWAN simulation.*



(b) *Probability of success and failure in LoRaWAN simulation.*

**Figure 15:** *LoRaWAN simulation outcomes: throughput and packet error rate (PER).*



**Figure 16:** *Percentage gain in success probability w.r.t. 24 devices.*

and LTE-M technology.

## REFERENCES

- [1] Mirco Rossi Thomas Holleczech 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. Millán. Collecting complex activity data sets in highly rich networked sensor environments. *Seventh International Conference on Networked Sensing Systems (INSS'10), Kassel, Germany*, 2010.
- [2] Ian Jolliffe. *Principal Component Analysis*, pages 1094–1096. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011. ISBN 978-3-642-04898-2. doi: 10.1007/978-3-642-04898-2\_455. URL [https://doi.org/10.1007/978-3-642-04898-2\\_455](https://doi.org/10.1007/978-3-642-04898-2_455).
- [3] Lars Buitinck, Gilles Louppe, Mathieu Blondel, Fabian Pedregosa, Andreas Mueller, Olivier Grisel, Vlad Niculae, Peter Prettenhofer, Alexandre Gramfort, Jaques Grobler, Robert Layton, Jake VanderPlas, Arnaud Joly, Brian Holt, and Gaël Varoquaux. API design for machine learning software: experiences from the scikit-learn project. In *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, pages 108–122, 2013.
- [4] S. P. Lloyd. Least squares quantization in pcm. In *Technical Report RR-5497, Bell Lab, September.*, 1957.
- [5] Romain Tavenard, Johann Faouzi, Gilles Vandewiele, Felix Divo, Guillaume Androz, Chester Holtz, Marie Payne, Roman Yurchak, Marc Rußwurm, Kushal Kolar, and Eli Woods. Tslern, a machine learning toolkit for time series data. *Journal of Machine Learning Research*, 21(118):1–6, 2020. URL <http://jmlr.org/papers/v21/20-091.html>.
- [6] S. Chiba H. Sakoe. Dynamic programming algorithm optimization for spoken word recognition. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 26: 43–49, 1978.
- [7] Ignacio Oguiza. tsai - a state-of-the-art deep learning library for time series and sequential data. Github, 2022. URL <https://github.com/timeseriesAI/tsai>.
- [8] Hassan Ismail Fawaz, Benjamin Lucas, Germain Forestier, Charlotte Pelletier, Daniel F. Schmidt, Jonathan Weber, Geoffrey I. Webb, Lhassane Idoumghar, Pierre-Alain Muller, and François Petitjean. InceptionTime: Finding AlexNet 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>.