# Discovering self-quantified patterns using multi-time window models

Luke McCully and Hung Cao
*University of New Brunswick, Fredericton, Canada*
Monica Wachowicz
*RMIT University, Melbourne, Australia*
Stephanie Champion
*College of Nursing and Health Sciences, Flinders University, Adelaide, Australia, and*
Patricia A.H. Williams
*College of Science and Engineering, Flinders University, Adelaide, Australia*

Received 17 December 2021
Revised 25 January 2022
19 February 2022
Accepted 21 February 2022

## Abstract

**Purpose** – A new research domain known as the Quantified Self has recently emerged and is described as gaining self-knowledge through using wearable technology to acquire information on self-monitoring activities and physical health related problems. However, very little is known about the impact of time window models on discovering self-quantified patterns that can yield new self-knowledge insights. This paper aims to discover the self-quantified patterns using multi-time window models.

**Design/methodology/approach** – This paper proposes a multi-time window analytical workflow developed to support the streaming *k*-means clustering algorithm, based on an online/offline approach that combines both sliding and damped time window models. An intervention experiment with 15 participants is used to gather Fitbit data logs and implement the proposed analytical workflow.

**Findings** – The clustering results reveal the impact of a time window model has on exploring the evolution of micro-clusters and the labelling of macro-clusters to accurately explain regular and irregular individual physical behaviour.

**Originality/value** – The preliminary results demonstrate the impact they have on finding meaningful patterns.

**Keywords** Wearable devices, Fitbit, Multi-time window analytical workflow, Physical activity behaviour, Self-quantified patterns, Streaming *k*-means clustering

**Paper type** Research paper

## 1. Introduction

Continuously-worn wearable devices are becoming more prevalent in society for quantifying yourself through collecting data for the monitoring of physical health related problems such as blood pressure, sugar level and obesity, which are usually associated to chronic diseases like cardiovascular disease and diabetes [1–3], and early detection of neurodegenerative disorders [4]. There has also been considerable research using clustering algorithms for analysing wearable device logs since a variety of information about the individuals' activity



This research was supported by the NSERC/Cisco Industrial Research Chair [Grant IRCPJ 488403-14]. The authors would also like to thank the Cisco-Flinders Digital Health Design Lab for providing us with the Fitbit data logs.

Applied Computing and Informatics
Emerald Publishing Limited
e-ISSN: 2210-8327
p-ISSN: 2634-1964
DOI 10.1108/ACI-12-2021-0331

is reported, including the calories consumed, sleep patterns, steps walked, distance moved and stairs climbed; despite the fact that there might be a specificity issue with this information related to low accuracy.

Previous research has shown that the *k*-means algorithm is the most commonly partitioning based approach using historical wearable device logs [5–7]. However, very few attempts can be found in the literature in clustering wearable data streams [8, 9]. Mainly because analysing them brings along a research challenge in which the rate of the data compiled and stored is not being optimised to each use case, which is where time window models come into effect. When the wearable data streams are continuously brought into the *k*-means algorithm, it is challenging to retrieve any insight since previous and future data streams are needed to provide context. For example, a user may have a one minute peak in heart rate while not having any steps taken during the one minute interval, but looking at the previous minute timestamps may provide important context such as the user may have just sprinted during previous minute. This detracts from the possibility that this sudden heart-rate spike could be due to a health issue. This contextual information is what makes the time window models an important factor to take into account in the streaming *k*-means clustering algorithm when analysing wearable data streams.

Different time window models can be coupled within the streaming *k*-means clustering algorithm, including sliding, landmark, damped and pyramidal [10]. Each of these models aims to handle the evolution of the distribution of the data streams over time, and as a result, they determine at which time frame the streams are stored and analysed, and when the previous historical streams are discarded [11]. With regards to wearable devices, this can become an issue since historical data streams might be as important as new incoming data streams. It is paramount to understand what the impact these windows have on generating self-quantified patterns over time.

This paper proposes an analytical workflow to reveal self-quantified patterns by using a streaming *k*-means clustering algorithm based on finding online micro-clusters from the wearable data streams and offline macro-clusters from re-clustering these micro-clusters. The sliding time window model is used to understand micro-cluster evolution, which plays an important role in distinguishing actual novel self-quantified patterns from possible existing outliers. Meanwhile, the damped time window model draws on micro-cluster scalability, defined here as the maximum number of current and historical data streams which guarantees context consistency that is needed to compute micro-clusters. Consequently, self-quantified patterns are inferred from the $k$ macro-clusters that are computed by re-clustering the set of $k'$ micro-clusters using a particular time window model. The labelling of these $k$ macro-clusters is a process aimed to reveal changes in physical activity behaviour, targeting on individuals rather than their physical and social environments.

The scientific contributions of this paper can be described as follows:

(1) A new multi-window analytical workflow for streaming *k*-means clustering since previous research work has neglected the role of time window models in cluster evolution and cluster scalability.

(2) The multi-window analytical workflow achieves the optimal storage capacity in the online component, so the wearable data streams never overwhelm the streaming *k*-means clustering algorithm, as well as immediately storing the processed micro-clusters for further re-clustering. This is achieved despite the complexity of integrating two different time window models.

(3) This proposed new workflow has not been described previously, and is a ground-breaking research in applying the damped time window model for clustering wearable data streams.

(4) Unique empirical results are provided that advance understanding of the impact of a time window modelling for finding self-quantified patterns from wearable stream data.

## 2. Related work

Although seemingly a modern invention, wearable technology has been around since the early 1960's when Claude Shannon and Edward Thorp invented what is known as the first wearable computer in order to beat a casino game of roulette [12]. One of the main use cases comes from the health sector. Devices such as Fitbit make use of sensors through a wearable watch to track the health and fitness activities of the user. Invented in 2007, Fitbit has become the current market share leader for wearables across the world [13]. Currently, Fitbit has seven different models in the market, all varying in price, health features, exercise features, smart features and design style. All the current models feature an accelerometer to measure acceleration and determine orientation used to compute step count, where five types (i.e. Versa 2, Versa, Ionic, Charge 3 and Inspire HR) utilise a heart rate sensor to measure a user's heart beats per minute [14].

Recent research has tested different Fitbit models to determine whether they are effective at monitoring physical activity, and whether they can potentially be used by healthcare professionals to guide decision making and treatment plans. Feehan *et al.* [15] evaluated 67 studies and experiments carried out by other researchers in the field to evaluate the data reliability of Fitbit devices. They found consistent evidence indicating that these devices would meet an acceptable accuracy for step count only half the time, with a tendency to underestimate steps in a controlled setting, while overestimating in a real-world setting. They further describe the accuracy rates for different activities such as jogging, sleeping and slow walking in comparison to research grade accelerometers. When measuring a user's sleep activity, such as sleep time and time in bed; the Fitbit devices provided similar measurements in comparison to the research grade accelerometers such as an Actigraph. They recommend using discretion when considering using Fitbit devices as an outcome measurement tool in research and making health care decisions, bearing in mind this is less so in adults with no mobility issues.

Due to the data reliability, the use of Fitbit devices has been limited to physical activity monitoring to produce acceptable accurate results. Koolean *et al.* [16] proposed a method to relate physical activity to physical capacity. This was done by using a quadrant method to place individuals into different categories based on one variable (i.e. step count) to represent physical activity, and one variable (i.e. 6 minute walk distance (MWD)) to represent physical capacity. If an individual had a high step count but low MWD, then he would be categorised in "Can't do, do do" which represents that he does not have capacity to do what he is doing, and vice versa for the other categories. The notion that physical activity can be represented by step count is currently accepted in the *Quantified Self* domain, however, it is still an issue of debate coming back to how reliable the wearable devices are, and also how just one variable can accurately represent a user's capacity level.

From an analytical perspective, Carnein *et al.* [11] provide an extensive survey on stream clustering algorithms, outlining how each algorithm performs during a streaming process by delineating their advantages and limitations. The overall strategy is based on a two-phase clustering approach, having an online phase which uses a time window model to capture the data streams and then computing micro-clusters (i.e. preliminary clusters within each time window). The second phase is carried out offline as the micro-clusters are re-clustered to generate the macro-clusters after the entire stream data is processed. The use cases revolve largely around clustering sensor based data streams due to the need of supporting real-time communication between the sensors themselves and the resultant output. More in-depth investigation is needed for supporting multi-window analytical workflows, identifying which

stream clustering algorithm should be used, and which time window actually reveals interesting self-quantified patterns from a vast amount of wearable data streams.

The streaming *k*-means clustering was chosen for finding self-quantified patterns due to its ease of use and overall popularity amongst clustering algorithms. Originally published in 1955, *k*-means has stood the test of time due to its simplicity and overall insightful results associated with many use cases [17]. Research revolving around the usage of *k*-means can be found ranging from the 1960's to today in 2020 [18]. The idea behind *k*-means can also be traced back to 1957 from polish mathematician Hugo Steinhaus [19]. With the nature of wearable stream data, an unsupervised learning method is needed to further gain new insights, and using *k*-means meets this requirement as well as provides easy comparisons to the numerous amounts of use cases that have as well applied *k*-means in the past.

From a temporal perspective, time windows have been used to extract small, quasi-static subsets from the data streams [20]. The main time window models proposed in the literature are damped, sliding, landmark and pyramidal [11]. The sliding time window model has been previously proposed to improve clustering results from wearable data streams. Park *et al.* [8] provide empirical results showing the main limitations of considering a wearable device log as one whole snapshot rather than considering accumulated wearable data streams using a time window. They were able to find insightful consecutive insomnia-activity clusters of individuals with similar sleep-related dysfunctions by coupling sliding time windows of an 8-day period of daily intervals with a neural-net based unsupervised method, using various information modalities from smart bands.

Interestingly, Keogh and Lin [21] demonstrate that clustering time series sub-sequences is meaningless while using a sliding window. They state that since the output is independent of the input that a time window is *meaningless*. Their research can be considered as the first disclosure of the importance of investigating the use of multiple types of time window models to ensure that clustering results gathered have meaning.

To the best of our knowledge, no previous research work has been focussed on exploring wearable stream data by coupling different time window models with streaming *k*-means clustering.

## 3. Multi-window analytical workflow

This section describes our proposed analytical workflow that is developed to cluster wearable stream data using the sliding and damped time window models. The workflow consists of six phases as shown in Figure 1, which are described as follows:

(1) Data collection phase: The process of gathering wearable device logs from individual users.

(2) Data pre-processing phase: Encompasses the cleaning, transforming, and encoding tasks.

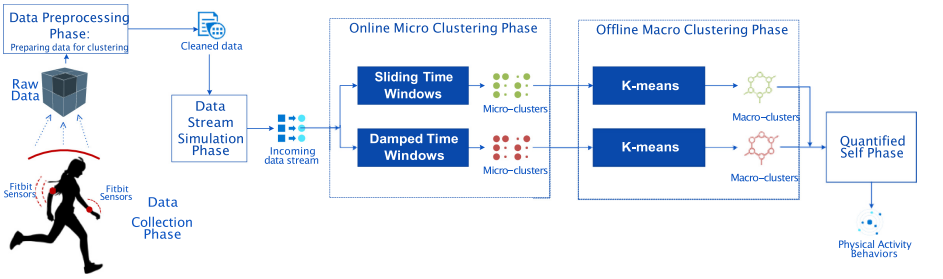


**Figure 1.**
Main phases of the multi-window analytical workflow

(3) Data stream simulation phase: The process of generating streams from a user's wearable data log.

(4) Online micro-clustering phase: The tasks that are necessary to compute the micro-clusters using the sliding and damped time window models.

(5) Offline macro-clustering phase: Generates the macro-clusters by re-clustering the micro-clusters found in each time window model.

(6) Quantified Self phase: The process of visualising the clustering results to outline the self-quantified patterns of a user.

### *3.1 Data collection phase*

This phase consists of retrieving Fitbit device logs, which are usually stored within the device and can only allow for retrieval once connected to a computer or synced to a third party cloud platform. Initial pre-sets are needed to account for individual tracking, such as personal information such as age, weight, height and sex. Other sensors connected to the Fitbit devices also generate data such as heart rate, steps, temperature and location, which contribute to a different range of physical outputs.

Fitbit device logs are usually fetched as an offline data package. Third party platforms such as Fitabase can allow wearable device logs to be retrieved in its raw format in comparison to usual summarised data from device manufacturers. In conjunction with this, if continuously synced to the platform the offline data can be monitored in real-time, acting as a data stream on itself. Once the offline logs are fetched, the data pre-processing phase is initiated as described in the next section.

However, retrieving raw data streams directly from the devices online software brings many technical issues since the information is currently generated to be as simplified as possible for the end user. The implementations of new capabilities need to be developed by the manufactures to allow access to raw wearable data streams that are essential for the next generation of multi-window analytical workflows.

### *3.2 Data pre-processing phase*

Data pre-processing is an important phase in the proposed analytical workflow. The ultimate goal of this phase is to clean, encode and transform the Fitbit data logs (i.e. raw data) into a revised format that is easily readable by a machine in such a way that data points can be easily processed by the streaming $k$-means clustering algorithm. Guaranteeing data quality and providing accurate data points is key to the success of the subsequent analytical phases in the proposed workflow. However, due to usage deviation, limitations of Fitbit devices or flaws in the data collection phase, it is not realistic to expect that the raw data will always be ready to be analysed. Therefore, five main data pre-processing tasks are designed to deal with the common issues including missing data points, duplicated data points, missing variables, redundant variables and variables selection. Once the data pre-processing phase is completed, a target data set is ready to be used by the data stream simulation phase.

### *3.3 Data stream simulation phase*

Data streams are a countable infinite sequence of data points that can be formalised as follows [22]:

$$T = [t_1, t_2, \ldots, t_n] \tag{1}$$

Where each data point contains many sets of variables as follows:

$$[t_1 = (P_1, S_1, Q_1, X_1, U_1)], [t_2 = (P_2, S_2, Q_2, X_2, U_2)], \ldots, [t_n = (P_n, S_n, Q_n, X_n, U_n)]$$

Where

(1) $P_n$: is a set of categorical variables related to personal information (e.g. age, weight, height and sex);

(2) $S_n$: is a set of numerical variables related to sensor measurements (e.g. temperature, vibration and location);

(3) $Q_n$: is a set of ordinal variables related to ratio scales (e.g. sleep quality and activity intensity);

(4) $X_n$: is a set of numerical variables related to physical measurements (e.g. step count and heart rate);

(5) $U_n$: the identifier of a wearable device.

This research replicated the stream process using the target data set created in the previous phase, since the current manufacturers do not support this capability. To achieve this, an assortment of frameworks is available to simulate or connect to a data stream, including MOA (massive online analysis) [23], MLFlow [24] and Stream *R* [20]. The Stream *R* framework to simulate the data streams, compute the clusters and visualise the results. This framework is further explained in Section 4.

### *3.4 Online micro-clustering phase*

For this phase, the simulated data streams of each Fitbit device arrive as a continuous sequence of data points that are accumulated using a time window model. Each time window has the same time frame (e.g. 2-h). The streaming *k*-means clustering algorithm requires incrementally updating the computation of the micro-clusters, which are represented by their respective $k'$ centroids.

A micro-cluster represents a set of similar data points, created using a single pass over the data currently available within a time window. The algorithm selects $k'$ random data points as seeds until clustering converges in such a way that for each time window, any new data point $t_i$ is always assigned to one unique micro-cluster $mc_j$ by minimising the sum of square distances [25]. Therefore, a centroid is the centre (i.e. the mean point) of a micro-cluster belonging to a specific time window.

There are two approaches for selecting the partitions for computing the micro-clusters. The first approach is based on applying the elbow method for computing the optimal number of $k'$ partitions for each time window. In this case, the number of centroids will vary from one time window to another. A second approach consists of applying a fixed number of $k'$ partitions for all time windows. In other words, it is assumed that the optimal number of micro-clusters should be the same across the time windows. The choice between these two approaches will depend on the selected variables for performing the clustering.

This phase is the most important in the proposed analytical workflow, as the different time windows used have a direct impact on the computation of micro-clusters, and play an important role in finding the macro-clusters in the next phase.

*3.4.1 Sliding time window model.* The sliding time window model only considers the most recent data point for computing the micro-clusters, since the older data point is removed once a new data point is available. A start window is initiated having a-priori defined time frame (e.g. 2-h) and containing the accumulated data points that were streamed during this time frame. As soon as the new data point arrives, the algorithm incrementally updates the micro-clusters. The next window utilises all the new data points and clusters as the data points enter

and exit the stream, storing the micro-cluster and its representative centroid after each completion.

It is important to point out that using a sliding time window model, the minimisation of the sum of square distances will often terminate at a local optimum, as expected when using $k$-means clustering. Therefore, the analytical workflow aims to gather insights on the evolution of micro-clusters rather than a full explanation as to why the data points were grouped under them. The main focus is on exploring the evolution of micro-clusters to distinguish new clusters from outliers, indicating the actual *cluster evolution* from the wearable data streams.

*3.4.2 Damped time window model.* The damped time window model continuously adds new data points into the feature space with each iteration lessening the weight of each point, the less weight it has the less it contributes to generating a micro-cluster. This is done to give the highest weight to the most recently captured instances. The streaming $k$-means algorithm computes the $k'$ micro-clusters after a set of data points are damped due to the decay function. As defined in Ref. [26], the weight of each data point within a damped time window decreases exponentially with time $t$ using the decay function

$$f(t) = 2^{-\lambda t} \tag{2}$$

Where, $\lambda$ should be always greater than 0.

The smaller the value of $\lambda$, the most important the historical data points are in comparison to the current data points. This makes the damped time window model effective to indicate the *cluster scalability*, In this research, the cluster scalability is defined as the maximum number of data points which guarantees context consistency in the data streams that are needed to compute micro-clusters. This time window model supports the ability of a micro-cluster to grow while conforming within the a-priori $k'$ partitioning.

The outcomes from this phase are two sets of $k'$ micro-clusters, one for each type of time window model being used for the computation. They will be re-clustered as macro-clusters in the next phase.

### *3.5 Offline macro-clustering phase*

After the streaming has ended and all the centroids of the micro-clusters have been computed using both time window model; macro-clusters are generated by re-clustering these centroids. The $k$-means algorithm is again used to compute the final $k$ macro-clusters. The $k$ centroids will be generated from re-clustering the $k'$ centroids of micro-clusters found using the sliding and damped time window, respectively. The benefit of using the proposed offline clustering is to gain further insight from the entirety of the $k'$ centroids after it has finished streaming while not adding stress to the stream flow itself due to the half the process being performed on a solid state [27].

### *3.6 Quantified-self phase*

The Quantified-Self phase begins with plotting each time window $k'$ micro-clusters and comparing them with the final $k'$ macro-clusters. Moreover, external variables obtained in the data collection phase are also used to label the final macro-clusters. Each time window will consist of different micro-clusters, which will directly impact the meaning behind their macro-clusters.

The macro-clusters ultimately represent self-quantifying patterns that can be interpreted as regular and irregular physical activity behaviour. They may facilitate our understanding of the reasons leading to individual changes in lifestyles and health care settings. Therefore, the outcomes of the proposed multi-window analytical workflow provide empirical evidence that captures the range of self-quantifying patterns on behaviour; these outcomes also offer

an analytical perspective that may help identifying a range of variables involved in monitoring behavioural changes in physical activities.

## 4. Implementation

### *4.1 Data collection phase*

The wearable device logs used in this research were collected from an intervention experiment performed at Flinders University, Australia, where 15 participants continuously wore a Fitbit Charge 2 device for approximately a 2-month period. The raw Fitbit data consisted of approximately 87,600 data points per participant with a data rate of one data point per minute containing 31 variables as shown in Table A1.

Typically, extracting the data directly from the Fitbit device is possible but limited due to the majority of the data being summarised, rather than providing a minute by minute description of the collected data.

To address this issue, *Fitabase*, a third party research cloud platform designed to collect data from Fitbit devices with more diverse options, was used. The major benefit being the raw data can be extracted in a range of formats. Using the third party cloud platform allows the data to be retrieved on a per participant basis or in a batch with all participants on one spreadsheet. To keep the participants separate, the raw data was retrieved on a per participant basis, and transformed to multiple. CSV files in order to be used in the next phase.

### *4.2 Data processing phase*

During this phase, a variety of pre-processing tasks were performed to determine the quality of collected data points. A number of issues were detected that allowed for insight on data quality. One instance was a participant who did not wear the device to bed, showing a defined gap in the data. Another instance was a failure to sync to the platform, which lead to missing minute-to-minute values although the daily step count and heart rate were still collected. This also resulted in a complete loss of sleep data during these intervals. Due to lack of connectivity, devices not worn and sensor problems, missing data points occurred during participant data streams.

Another example being a participant having mismatched variables that should be the same such as sleeping and quality of sleep, it may show in one category that a participant was asleep, but awake in the other. Finally there were instances of times when it was proven a participant was not moving (via video recording or a time use diary) where step counts were recorded.

Temperature was being captured from a sensor located at the office of each participant. Considering that the participants were not always in their offices, this variable provided an unrealistic context on the participants' behaviour. Adding outdoor weather information may have provided additional context to explaining some self-quantified patterns. However, there was little to no rain during the duration of the 2-month experiment.

Finally, the variable selection task was performed to prepare a target data set ready to be used by the two phase clustering algorithm. For this research, three numerical variables are summarised in Table A2.

### *4.3* R *stream framework*

The *R* Stream framework was used to simulate the data streams, generate the time windows and run the online and offline clustering. Therefore, the data stream simulation phase, the online micro-clustering and offline macro-clustering phases were implemented using the stream *R* framework, seamlessly integrating the extensive existing *R* packages, including stream MOA [28], cluster [29], clusterGeneration [30], and fpc [31].

The overall architecture is shown in Figure 2. Initially, the data stream data was used due to its ease of use and ability to simulate a live stream from any CSV file. Table A3 illustrates the CSV format of our target input data set used for generating the stream simulation.

The Stream *R* framework was also developed to focus in the domain of data stream clustering (DSC) which fits well to our need. The DSC was used to compute the online micro-clusters, with the option of either a sliding window or a damped window, and then passing it on to the offline phase which consisted of generating the macro-clusters.

In conjunction with implementing the time window models using the Stream *R* framework, we have also explored different time frames that could generate the most meaningful self-quantified patterns whilst not being computationally expensive were also explored. The initial selected time frame had 1-h time intervals (60 data points), and was expected to be an acceptable time frame due to the ability to visually recognise clusters.

However after adding more time for the time frame, it was discovered that a 2-h interval was more appropriate. Figure 3 demonstrates that the micro-clusters are more diversified with the addition of an extra hour of data. Furthermore, there was more distinction between areas of no steps being taken, low amount of steps taken (<15 steps) and medium to high amount of steps (>15 steps). The 2-h time frame was selected for implementing both time window data models since it provided a richer context for generating the micro-clusters that have optimised and further investigated to infer the self-quantified patterns for each participant.

For the sliding time window model, the online clustering initialises with 120 data points within the initial window where gradually over time new data points are introduced and old data points are disposed. Micro-clusters were generated in each of these windows using the DSC function generation, and stored once the sliding window passed over the $120^{th}$ data point relative to the starting data point.

For the damped time window model, micro-clusters were continuously generated after a set of input data points were damped due to the decay function of $\lambda = 0.033$. This model like the sliding time window model was also implemented using the DSC function generation.

With a fixed $k' = 4$ for each time window (i.e. sliding and damped), once the data stream was finished and the micro-clusters stored, the online clustering was finished. The DSC

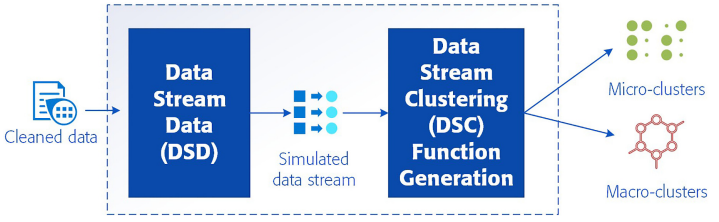


**Figure 2.**
Overview of the stream *R* architecture

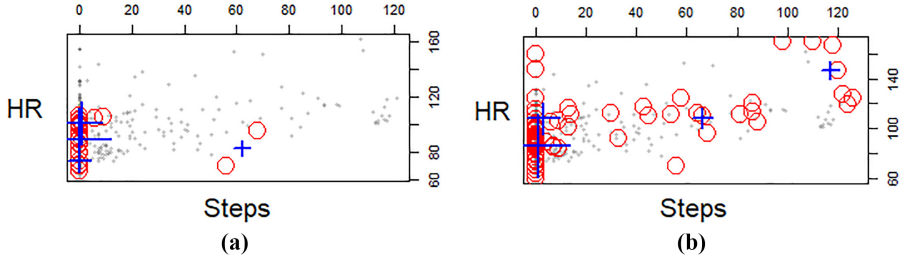


**Figure 3.**
Macro-cluster centroids (blue crosses) and micro-cluster centroids (red circles) results using the sliding time window model: (a) with 1-h time frame and (b) with 2-h time frame

function generation was again used for the computation of the macro-clusters in the offline macro-clustering phase. The current version of the DSC does not support the streaming elbow method yet, therefore, the optimal $k$ value was computed separately using $R$, and later used as the input parameter for the DSC function.

## 5. Discussion of the results

### *5.1 The evolution of micro-cluster patterns*

It was determined that the most relevant variables in our analytical workflow to be steps and heart beat/minute (HR) due to these variables being the most accurate numerical values in the collected data streams. Heart rate variability (HRV) was also used as in an input for the streaming $k$-means algorithm. The initial $k' = 4$ micro-clusters results from clustering data points can be seen in Figure 4, using the first four sliding time windows of stream data, and the steps and HR variables for comparison.

The evolution of the micro-clusters can be observed between these sliding time windows, in particular the linear and constant micro-cluster patterns when steps = 0 across a distributed range of HR values during a period of 4-h. It is also possible to distinguish new random turning shape patterns that have occurred when steps >0, within more specific ranges of HR values (e.g. from HR > 90 to HR > 100). These types of patterns have emerged throughout the time windows of several participants' data streams. For example, in the first and last sliding time windows, the new random turning shape patterns indicate movement patterns of a participant after a period of 4-h staying still, rather than being outliers. These results provide empirical evidence that the variables should be targeted for influencing behaviour change when devising interventions, and that the evolving patterns of actual novel micro-clusters represent a different context.

### *5.2 Macro-cluster results and the self-quantified patterns*

A selection of four prominent participants out of 15 participants are used here to illustrate the macro-clusters results and their respective self-quantified patterns that were found using both sliding and damped time window models. Table A4 provides an overview of their personal information.

The macro-clusters results of participant 12 can be seen in Figure 5(a), where the red circles represent the centroids of the micro-clusters, and the blue crosses being the centroids of the macro-clusters, which were found using the sliding time window model. The centroids of the macro-clusters represent the self-quantified patterns, which reveal a balanced relationship between strong regular physical activity behaviour that consists of no movement (i.e. steps = 0) with moderate regular physical activity behaviour due to mobility (i.e. steps between 10 and 40). Finally, it is also possible to visually identify the outliers by looking at the off-set centroids of the micro-clusters that have emerged from the data points.

The moderate regular physical activity behaviour of participant 12 is more prominent in the macro-clusters results using the damped time window model (Figure 5(b)). It is also interesting to point out that this participant has shown very few outliers: only one outlier micro-cluster in both time window models.

Table 1 provides statistics of the macro-clusters using some explanatory variables. Cluster 3 for example, makes up 79% of leisure time with an average step count of 11.44 steps per minute. This cluster represents time periods when the participant is active. Any new values entering this cluster will fall into the same class. We can also see that the data points belonging to this cluster fall largely on days 3 and 4 of the week (i.e. Tuesday and Wednesday) leading further insight into the participant's lifestyle, such as the participant become more active during these days.

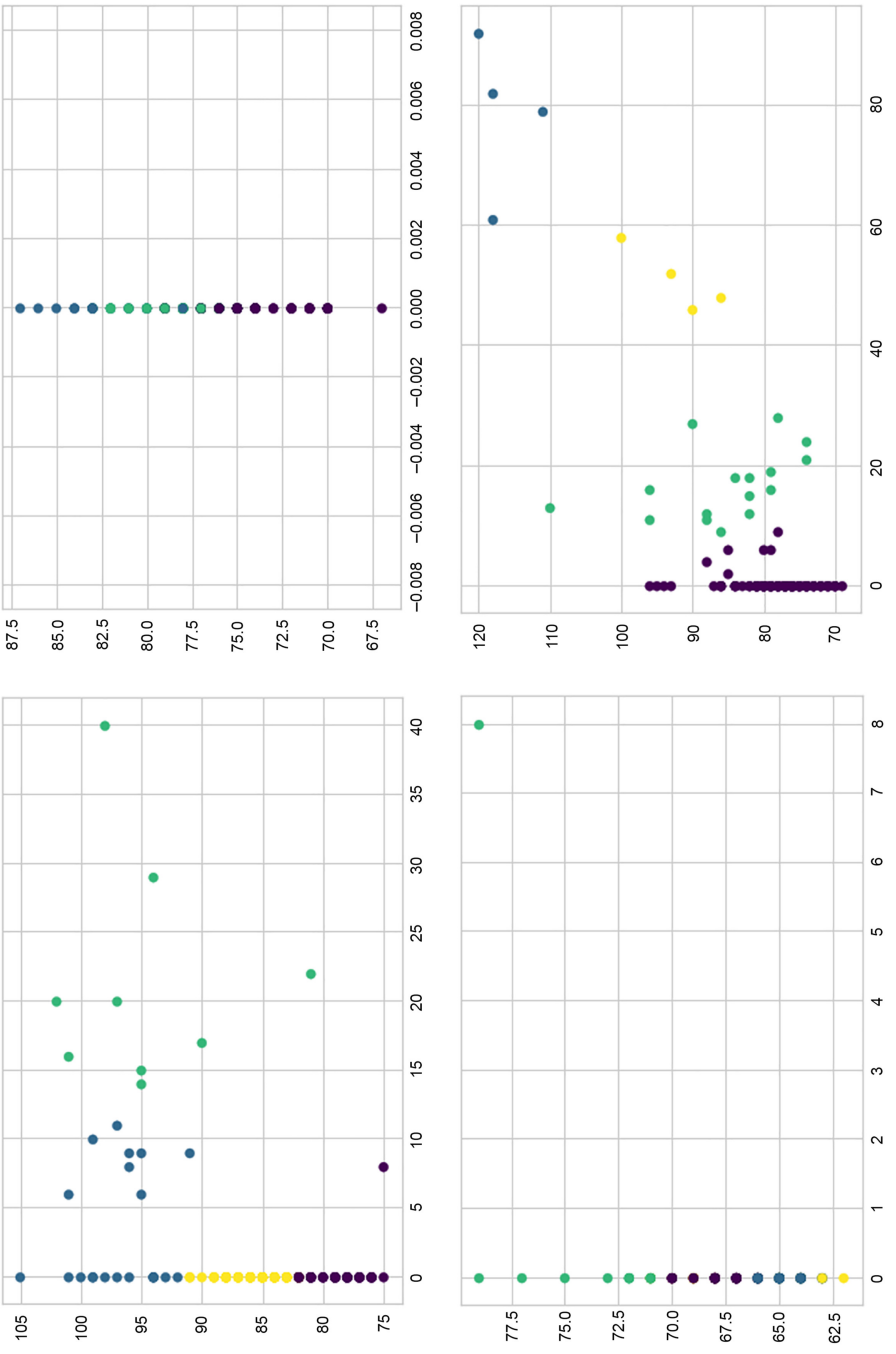


**Figure 4.**
Initial micro-cluster results for the first four consecutive sliding time windows ($x$ = steps, $y$ = heart rate/min)

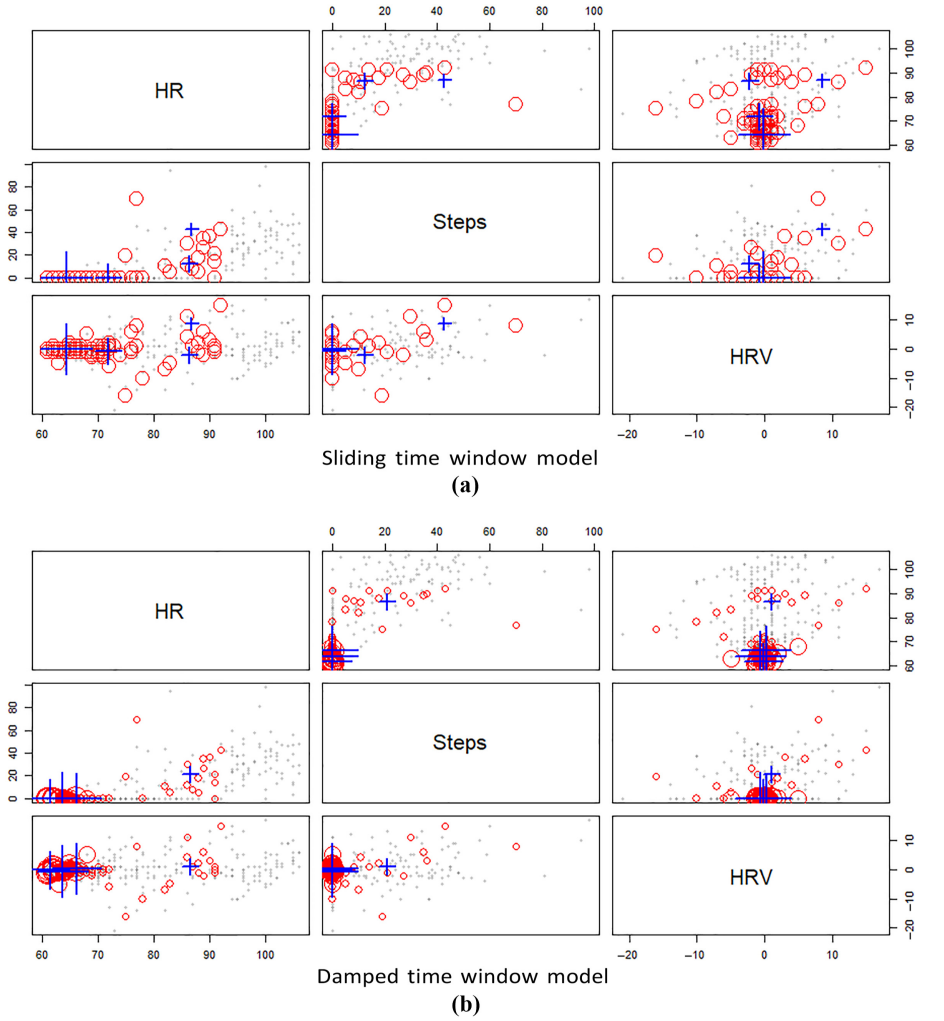


**Figure 5.**
Macro-cluster (blue crosses) and micro-clusters (red circles) results for participant 12 using the sliding time window model and damped time window model

Macro-clusters 3 and 1 are more generic in nature but still have unique labelling classes. For example, macro-cluster 3 represents the largest portion of work time compared to the other macro-clusters and macro-cluster 1 contains a mixture of values pertaining to the normal. Comparing these macro-clusters to macro-cluster 4, it is noted that this macro-cluster mainly represents 54% and 15% times that the participant is inactive with an average step count of 5.9 per minute, while maintaining a low average heart rate. The data points belonging to this macro-cluster fall largely on days 1 and 7 (i.e. Saturday and Sunday), which shows that the participant may be less active or sleeping more than on other days.

In comparison to the sliding window clusters, the damped window clusters offer a similar but different perspective on the participants' activity level as summarised in Table 1. We can see that the main difference lies in there now being two macro-clusters (1 and 3) that captured high physical activity instances, whereas sliding time windows stored these into just one macro-cluster. Although the count of data points per macro-clusters 1 and 3 are lower, they

| Time window | Cluster | Count | Average heart rate | Average step count | Average HRV | Majority day | Least day | Leisure time | Sleep time | Work time |
|---|---|---|---|---|---|---|---|---|---|---|
| Sliding | 1 | 28,009 | 77.13 | 8 | 0.01 | 5,6 | 7,1 | 0.43 | 0.41 | 0.16 |
| Time | 2 | 25,755 | 83.07 | 11.44 | −0.02 | 3,4 | 2,7 | 0.79 | 0.1 | 0.11 |
| Window | 3 | 18,922 | 74.73 | 6.51 | 0.06 | 6,5 | 3,4 | 0.35 | 0.46 | 0.19 |
| | 4 | 9,885 | 73.18 | 5.9 | 0 | 1,7 | 3,4 | 0.32 | 0.54 | 0.15 |
| Damped | 1 | 6,743 | 82.87 | 11.26 | 0.09 | 4,6 | 5,7 | 0.77 | 0.11 | 0.12 |
| Time | 2 | 40,136 | 78.57 | 8.94 | 0 | 5,6 | 1,4 | 0.5 | 0.34 | 0.16 |
| Window | 3 | 9,109 | 82.96 | 11.05 | −0.08 | 3,1 | 7,2 | 0.86 | 0.06 | 0.08 |
| | 4 | 26,583 | 74.1 | 6.21 | 0.06 | 1,5 | 3,4 | 0.33 | 0.49 | 0.18 |

**Table 1.**
Explanatory variables for the macro-clusters found for participant 12 using the sliding time window model vs. damped time window model

offer a diverse range of activity with high step counts and heart rate. Macro-cluster 4 offers a higher sleep classification rate, and again, day 7 is recorded as a day with a low amount of leisure time, populated mostly by sleeping time.

The damped window model was particularly effective in revealing macro-clusters that could be associated to different physical activity intensity levels. One example was participant 12 who exhibited the whole spectrum of physical activity intensity levels, ranging from very low (macro-cluster 1) and low (macro-cluster 2) intensity levels; up to high (macro-cluster 3) and very high (macro-cluster 4) intensity levels. Figure 6(a) illustrates the evolution of these intensity levels during the whole duration of the experiment. It is important to point out that the highest intensity peaks have randomly occurred at any intensity level, showing a volitional regulatory behaviour on different days of the week.

A different evolution was observed for the intensity activity patterns of participant 18 who exhibit few peaks of very low intensity activities in macro-cluster 1, as opposed to a wave

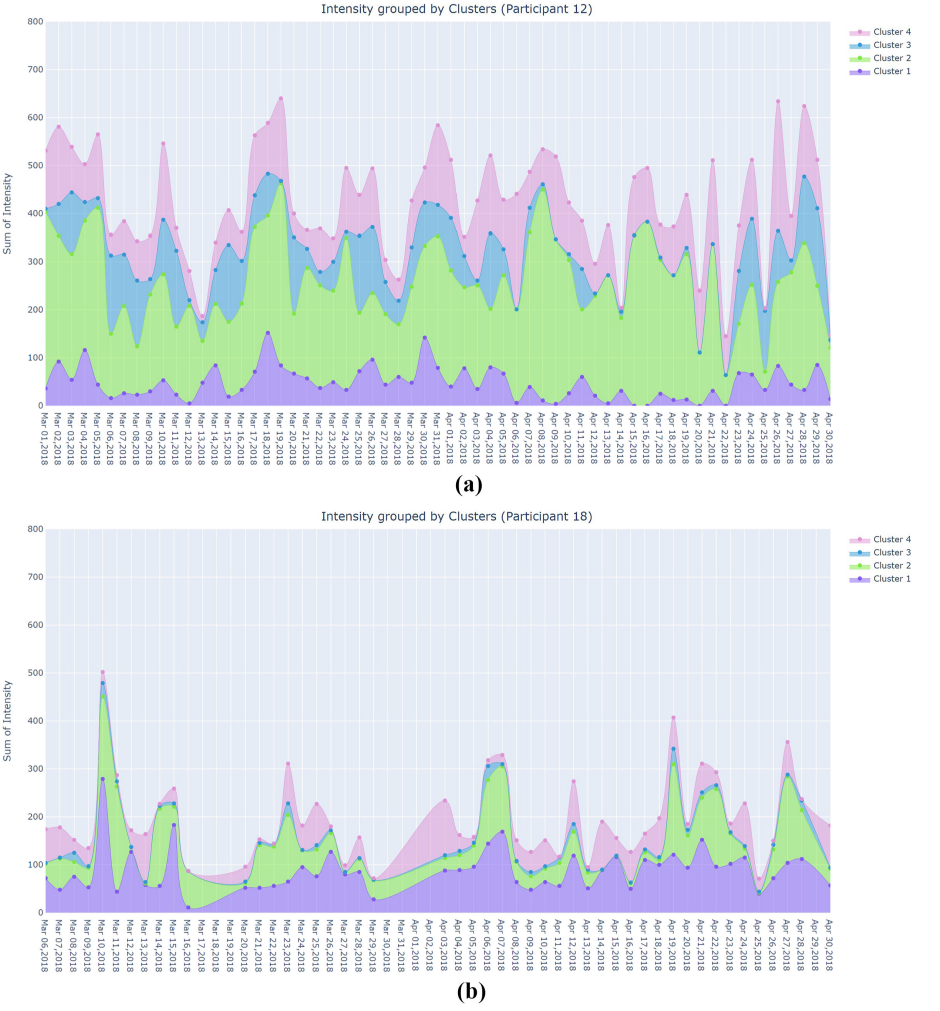


**Figure 6.**
The evolution of intensity activity patterns of participant 12 and 18 using the damped time window model

pattern for macro-clusters 2, 3 and 4 that reveal intensity peaks that occurred after a wave has passed (Figure 6(b)).

We can see from Table 2 that participant 19 produced similar clustering results from both the sliding window and the damped window. Each window model produces one main physically active cluster, where the makeup is approximately 68% leisure, 14% sleep and 19% working time. This encompasses a large portion of high step count minutes and a higher than average heart rate. Similarly, another cluster results in most of the sleeping time with approximately 46% sleeping time, 44% leisure time and 10% working time. This cluster captures moments of lower step count, lower heart rate and during sleep. In certain instances, like in the case of participant 19 the time windows model will produce similar results due to low diversity in the movement and heart rate of the participant, which can be seen in Figure A1(a) and (b) (participant 19 damped and sliding plot from *R*).

There is also evidence that the participant had less data captured on days 3 and 4 of the week during the 2-month interval due to every cluster having day 3 and 4 as its least captured day. This could be due to user routine (example: taking off the Fitbit during weekly practices) or random errors (device/human). This case provides a good example of how low diversity data sets provide minimal changes to the results of the time window models.

### *5.3 The impact of time window models on discovering self-quantified patterns*

Moreover, the time window models play an important role on discovering self-quantified patterns labelled as regular physical mobility behaviour because the actual steps trends during the weeks have been different from each other during the experiment. After analysing all the results of participant 20, it was clear that the macro-clusters have exhibited regular physical mobility behaviour using the sliding time window as shown in Figure 7. In this case, regular physical mobility was associated to the macro-cluster 1 on Monday (DOW = 2); Tuesday (DOW = 3) and Saturday (DOW = 7).

In contrast, the same findings were not found when analysing the macro-clusters using the damped time window model (Figure 7). In this case, the macro-clusters results reveal irregular physical activity throughout the various days of the week and macro-clusters. This exposes how challenging is to differentiate regular from irregular physical activity behaviour in self-quantified patterns due to the impact of a time window model being used to compute the macro-clusters.

Finally, the clustering results were visualised using density heat maps in order to compare the global self-quantified patterns amongst the participants. Figure A2 provide an overview of the variation of the number data points belonging to different macro-clusters of each participants when taking into account the relationship between the HR and steps variables.

## 6. Conclusions and future research

A multi-window analytical workflow was proposed for improving the streaming *k*-means clustering algorithm by integrating complementary time window models such as the sliding and damped time window models. Our preliminary results demonstrate the impact they have on finding meaningful patterns. Time window models have not been researched exclusively, as they have been considered as a minor step in current research on stream clustering algorithms and therefore, they have not been explicitly understood in the required depth, until now.

Future research will explore other time window models (e.g. landmark and pyramidal time window models) coupled with the streaming *k*-means clustering algorithm in order to develop further our multi-window analytical workflow. For example, our landmark time window model will then be changed from time interval collection to “event” based collection, where

**Table 2.**
Explanatory variables for the macro-clusters found for participant 19 using the sliding time window model

| Time window | Cluster | Count | Average heart rate | Average step count | Average HRV | Majority day | Least day | Leisure time | Sleep time | Work time |
|---|---|---|---|---|---|---|---|---|---|---|
| Sliding | 1 | 20,337 | 69.8 | 3.47 | 0.012 | 6,7 | 3,4 | 0.44 | 0.46 | 0.1 |
| Time | 2 | 4,184 | 70.7 | 4.42 | 0.05 | 5,2 | 3,4 | 0.44 | 0.4 | 0.17 |
| Window | 3 | 37,457 | 75.4 | 6.11 | 0.02 | 7,5 | 3,4 | 0.67 | 0.14 | 0.19 |
| | 4 | 11,786 | 73.13 | 5.27 | −0.05 | 5,6 | 1,3 | 0.45 | 0.31 | 0.24 |
| Damped | 1 | 21,250 | 69.9 | 3.52 | 0.01 | 6,7 | 3,4 | 0.44 | 0.45 | 0.11 |
| Time | 2 | 36,613 | 75.36 | 6.1 | 0.03 | 7,5 | 3,4 | 0.68 | 0.14 | 0.18 |
| Window | 3 | 15,96 | 69.53 | 3.83 | −0.01 | 1,2 | 3,4 | 0.41 | 0.47 | 0.13 |
| | 4 | 14,305 | 73.2 | 5.3 | −0.04 | 5,2 | 3,1 | 0.45 | 0.3 | 0.24 |

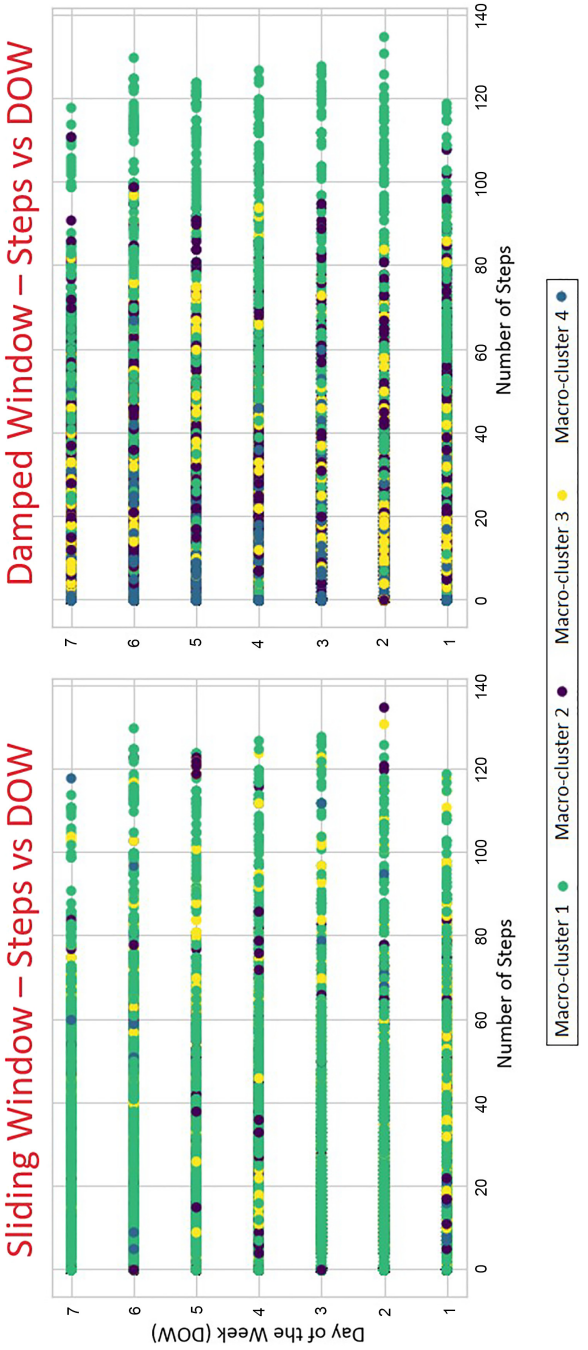


**Figure 7.**
The weekly evolution of the macro-clusters according to the steps taken by participant 20

data streams will be collected until a set "event" occurs. Initially we will start by setting the event as a drastic change in HR, this is dependent on the participant so will be adapted accordingly.

There is no time window that should be considered the most optimal for determining whether *k*-means is an accurate algorithm to use for both the online and offline phases. It is anticipated that there is to be a point where if the time frame of any type of time window model is too large, noise will always overcome the results and clusters will not be recognisable as self-quantified patterns. After this point is found, we will be able to accurately explain any regular and irregular physical behaviour. It will also be possible that different time window models will use different time frames within the same analytical workflow.

## References


1. Jo A, Bryanl DC, Coakes CE, Mainous AG III. Is there a benefit to patients using wearable devices such as fitbit or health apps on mobiles? a systematic review. Am J Med. 2019; 132(12): 1394-400.
2. Waheed B. Utilization of wearable technology: A synthesis of literature review. EasyChair: Technical report; 2019.
3. Hu R, Helena van Velthoven M, Meinert E. Perspectives of people who are over- weight and obese on using wearable technology for weight management: systematic review. JMIR mHealth and uHealth. 2020; 8(1): e12651.
4. Frey A-L, Karran M, Jimenez RC, Baxter J, Adeogun M, Chan D, Crawford J, Paul D, Everson R, Hinds C, *et al.* Harnessing the potential of digital technologies for the early detection of neurodegenerative diseases; 2019.
5. Md Zuraini HH, Ismail W, Hendradi R, Justitia A. Students activity recognition by heart rate monitoring in classroom using k-means classification. J Inf Syst Eng Bus Intell. 2020; 6(1): 46-54.
6. Shah Y, Dunn J, Huebner E, Landry S. Wearables data integration: data- driven modeling to adjust for differences in jawbone and Fitbit estimations of steps, calories, and resting heart-rate. Comput Industry. 2017; 86: 72-81.
7. Bini SA, Shah RF, Bendich I, Patterson JT, Hwang KM, Zaid MB. Machine learning algorithms can use wearable sensor data to accurately predict six-week patient- reported outcome scores following joint replacement in a prospective trial. J Arthroplasty. 2019; 34(10): 2242-7.
8. Park S, Lee SW, Han S, Cha M. Clustering insomnia patterns by data from wearable devices: algorithm development and validation study. JMIR mHealth and uHealth. 2019; 7(12): e14473.
9. Jang J-Y, Hee-Seok O, Lim Y, Cheung YK. Ensemble clustering for step data via binning. Biometrics; 2020.
10. Mansalis S, Ntoutsi E, Pelekis N, Theodoridis Y. An evaluation of data stream clustering algorithms. Stat Anal Data Mining ASA Data Sci J. 2018; 11(4): 167-87.
11. Carnein M, Trautmann H. Optimizing data stream representation: an extensive survey on stream clustering algorithms. Bus Inform Syst Eng. 2019; 61(3): 277-97.
12. Thorp EO. The invention of the first wearable computer. In: Digest of Papers. Second international symposium on wearable computers (Cat. No. 98EX215). IEEE; 1998. p. 4-8.
13. Shanhong L. Fitbit - statistics & facts, 2019. Available from: https://www.statista.com/topics/2595/fitbit/(accessed 11 August 2020).
14. Fitbit. How do i track my heart rate with my fitbit device?; 2020. Available from: https://help.fitbit.com/articles/en_US/Help_article/1565.htm (accessed 11 August 2020).
15. Lynne MF, Geldman J, Sayre EC, Park C, Ezzat AM, Yoo JY, Hamilton CB, Li LC. Accuracy of fitbit devices: systematic review and narrative syntheses of quantitative data. JMIR mHealth and uHealth. 2018; 6(8): e10527.

16. Eleonore HK, W van Hees H, van Lummel RC, Dekhuijzen R, Remco SD, Spruit MA, Hul AJ. "Can do" versus "do do": a novel concept to better understand physical functioning in patients with chronic obstructive pulmonary disease. J Clin Med. 2019; 8(3): 340.
17. Jain AK. Data clustering: 50 years beyond k-means. Pattern Recogn Lett. 2010; 31(8): 651-66.
18. MacQueen J *et al.* Some methods for classification and analysis of multivariate observations. In Proceedings of the fifth Berkeley symposium on mathematical statistics and probability. Oakland, CA, USA. 1967; 1: 281-97.
19. Steinhaus H. Sur la division des corp materiels en parties. Bull Acad Polon Sci. 1956; 1(804): 801.
20. Hahsler M, Bolanos M, Forrest J, *et al.* Introduction to stream: an extensible frame- work for data stream clustering research with r. J Stat Softw. 2017; 76(14): 1-50.
21. Keogh E, Lin J. Clustering of time-series subsequences is meaningless: implications for previous and future research. Knowl Inf Syst. 2005; 8(2): 154-77.
22. Rabl T, Sakr S, Hirzel M. Big stream processing systems (dagstuhl seminar 17441). In: Dagstuhl reports, volume 7. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik; 2018.
23. Albert B, Holmes G, Pfahringer B, Kranen P, Kremer H, Jansen T, Seidl T. Moa: massive online analysis, a framework for stream classification and clustering. In Proceedings of the First Workshop on Applications of Pattern Analysis: 2010; 44-50.
24. Zaharia M, Chen A, Davidson A, Ali G, Hong SA, Konwinski A, Murching S, Nykodym T, Paul O, Parkhe M, *et al.* Accelerating the machine learning lifecycle with MLflow. IEEE Data Eng Bull. 2018; 41(4): 39-45.
25. Singh RV and Bhatia MPS. Data clustering with modified k-means algorithm. In 2011 International Conference on Recent Trends in Information Technology (ICRTIT), IEEE: 2011; 717-21.
26. Cao C, Estert M, Qian W, Zhou A. Density-based clustering over an evolving data stream with noise. In Proceedings of the 2006 SIAM international conference on data mining, SIAM, 2006; 328-39.
27. Ghesmoune M, Lebbah M, Azzag H. State-of-the-art on clustering data streams. Big Data Analytics. 2016; 1(1): 13.
28. Hahsler M, Bolanos M, Forrest J. streammoa: interface for moa stream clustering algorithms. R package version; 2015. p. 1-1.
29. Maechler M, Rousseeuw P, Struyf A, Hubert M, Hornik K. Cluster: Cluster analysis basics and extensions. r package v. 2.0. 5; 2016.
30. Qiu W, Joe H, Qiu MW. Package 'clustergeneration'; 2015.
31. Hennig C. fpc: flexible procedures for clustering. r package version 2; 2015. p. 1-10. Available from: https://cran.*R-project.org/package=fpc*.

## Appendix

Appendix is available online for this article.

**Corresponding author**
Hung Cao can be contacted at: hcao3@unb.ca