sustainability

Article

AI-Based Quantification of Fitness Activities Using Smartphones

Junhui Huang, Sakdirat Kaewunruen * and Jingzhiyuan Ning

Department of Civil Engineering, School of Engineering, University of Birmingham, Birmingham B15 2TT, UK; jxh596@student.bham.ac.uk (J.H.); jxn948@alumni.bham.ac.uk (J.N.)
* Correspondence: s.kaewunruen@bham.ac.uk

Abstract: To encourage more active activities that have the potential to significantly reduce the risk of people's health, we aim to develop an AI-based mobile app to identify four gym activities accurately: ascending, cycling, elliptical, and running. To save computational cost, the present study deals with the dilemma of the performance provided by only a phone-based accelerometer since a wide range of activity recognition projects used more than one sensor. To attain this goal, we derived 1200 min of on-body data from 10 subjects using their phone-based accelerometers. Subsequently, three subtasks have been performed to optimize the performances of the K-nearest neighbors (KNN), Support Vector Machine (SVM), Shallow Neural Network (SNN), and Deep Neural Network (DNN): (1) During the process of the raw data converted to a 38-handcrafted feature dataset, different window sizes are used, and a comparative analysis is conducted to identify the optimal one; (2) principal component analysis (PCA) is adopted to extract the most dominant information from the 38-feature dataset described to a simpler and smaller size representation providing the benefit of ease of interpreting leading to high accuracy for the models; (3) with the optimal window size and the transformed dataset, the hyper-parameters of each model are tuned to optimal inferring that DNN outperforms the rest three with a testing accuracy of 0.974. This development can be further implemented in Apps Store to enhance public usage so that active physical human activities can be promoted to enhance good health and wellbeing in accordance with United Nation's sustainable development goals.

Keywords: gym activity recognition; fitness; health; wellbeing; artificial intelligence; machine learning; feature extraction

check for updates

Citation: Huang, J.; Kaewunruen, S.; Ning, J. AI-Based Quantification of Fitness Activities Using Smartphones. *Sustainability* **2022**, *14*, 690. https://doi.org/10.3390/su14020690

Academic Editors: Rezzy Eko Caraka, Anjar Dimara Sakti, Youngjo Lee, Teguh Dartanto and Toni Toharudin

Received: 16 December 2021
Accepted: 5 January 2022
Published: 9 January 2022

1. Introduction

One of the most important United Nations sustainable development goals (SDGs) is 'Good health and wellbeing'. To improve healthy lifestyle and ascertain positive wellbeing, physical, and mental activities have been promoted for all at all ages. In recent years, the growth of the fitness industry is exponential in many countries in order to enhance healthy lives and positive wellbeing. With the vigorous development of the fitness industry, a gym has become one of the common facilities within a neighborhood or on a campus in order to promote more and more people to participate in fitness activities. Physical inactivity is the most significant public health issue in the 21st century, which leads to a growing risk of chronic diseases such as type 2 diabetes and heart disease [1]. More importantly, physical inactivity has been claimed as the fourth leading cause of death worldwide [2]. Therefore, regular physical activity is fast becoming an important way to reduce the risk of chronic diseases. An artificial intelligence (AI)-based mobile phone application monitoring gym activity is developed in this study to promote a healthier lifestyle with the potential of remarkably reducing the investment in healthcare. The application is used to detect the four most common aerobic exercises ascending on a treadmill, running on a treadmill, using an elliptical machine, and cycling in gyms since we consider that gym users tend to place their phones not on their bodies when they are weightlifting. The main obstacle is the performance of the proposed application with only a mobile phone employed, as the other similar studies used a mobile phone with multiple wearable sensors to sense the subjects' movements.

Among the activity recognition projects, most of the interests fell into the scope of multiple sensors used. For example, Jun et al. [3] employed 2 wearable sensors; in [4], the authors used 11 dedicated gesture recognition sensors; and in reference [5], they employed 2 accelerometers. However, ref. [6] has surveyed a single phone-based accelerometer to classify 26 gym excises that were all weight training. Our interest falls into aerobic activity which remains unknown.

The handcrafted feature is widely used in activity recognition works [3–6] since raw data collected from an accelerometer is highly variable, oscillatory, and noisy [7]. However, there is no evidence that the handcrafted features produced manually match the aims for different strategies. Some of the features might be irrelevant and need to be discarded. Feature extraction techniques can be introduced to extract the most crucial information from the original handcrafted feature dataset. The information extracted can be cast to some fewer complexity features [8]. PCA is a representative of feature extraction, which is extensively exploited in different domains [9–11].

The handcrafted feature is calculated based on each window generated by splitting the continuous temporal signals to time windows. The size of the time window has long been a question of great interest in the human activity recognition (HAR) fields. It is now well established from [12] that thoroughly appraised the influence of the window size in 33 HAR studies and concluded that the 2 s window offers the most accurate scenario. Although most of the projects supported the finding of 2 s window size, there were still some rivals as most of projects studied were not gym-based activities. Accordingly, the evaluation of the window size is also introduced to our study for identifying and characterizing the optimal window size fit to our aim.

Using ML algorithms, researchers have been able to classify the different patterns among the handcrafted features. Gary et al. [13] exemplified five ML algorithms Naïve Bayes, Random Forest, IB3 instance-based learning algorithm, Decision Tree, and the multi-layer perceptron to process the 43 high-level features computed from the raw time series sensory data. The 5 algorithms have delivered an average accuracy of 91.9%, discriminating 18 daily activities. In [14], the analysts insinuated SVM, Random Forest, and Binary Decision Tree to detect adolescents with cerebral palsy using features derived from the vector magnitude of the raw acceleration information. The optimal results were 89.0%, 88.8%, and 86.2%, corresponding to SVM, Random Forest, and Binary Decision Tree. It can be inferred that the machine learning models cooperate well with the handcrafted features.

This study builds four candidate models KNN, SVM, SNN, and DNN to quantify four gym activities. Before the selection of the final choice of the model, we carry out an in-depth evaluation to optimize the window size for handcrafted feature generation, the dimensionality reduction by PCA, and the fine-tuned hyper-parameters of the models. We benchmark the performance of the best candidate with that of the state-of-the-art model and the proposed model surpasses the baseline model. The main contributions of this study are presented in the following:

1. The study investigates the impact of the window sizes, showing that the optimal window size depends on the different datasets used in various models. For example, KNN and SVM perform better using a 2 s window size, but SNN and DNN are more favorable to a 40 s window size.
2. Different machine models act similarly well if the handcrafted dataset is evaluated thoroughly in terms of the optimal window size and sophisticated feature extraction.
3. The DNN presents superior performance classifying four gym aerobic excises with only an accelerometer in terms of the overall performance and the performance associated with each activity provided in Appendix C.

2. Data and Methods

Figure 1 presents the whole picture of how we deliver the aim claimed. Following the main steps (see the top of Figure 1) result in a fine-tuned model at the end. Each step is decomposed, and the corresponding detail is shown below the step. The primary focus is evaluating the window size, the number of most informative features, and the hyper-parameters of each algorithm.

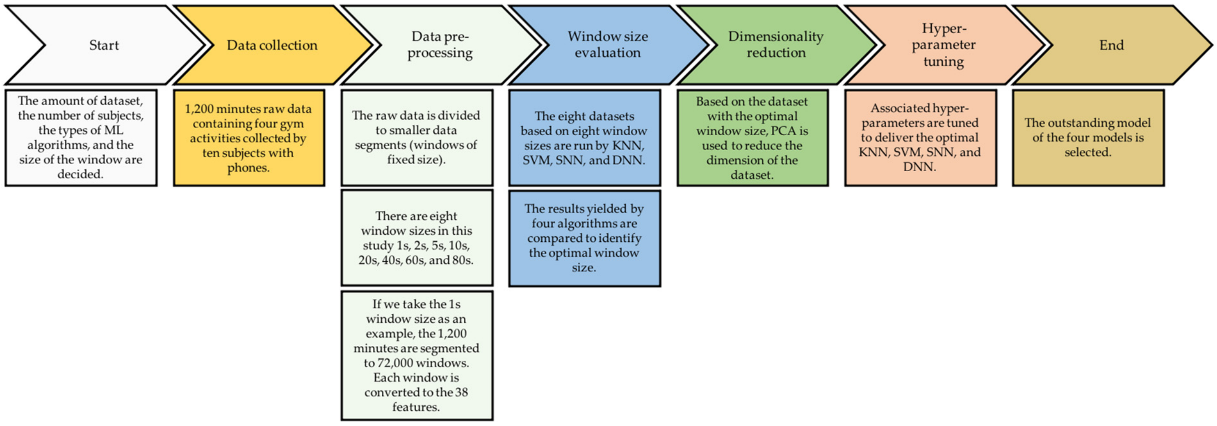
Start
Data collection
Data pre-processing
Window size evaluation
Dimensionality reduction
Hyper-parameter tuning
End
The amount of dataset, the number of subjects, the types of ML algorithms, and the size of the window are decided.
1,200 minutes raw data containing four gym activities collected by ten subjects with phones.
The raw data is divided to smaller data segments (windows of fixed size).
There are eight window sizes in this study 1s, 2s, 5s, 10s, 20s, 40s, 60s, and 80s.
If we take the 1s window size as an example, the 1,200 minutes are segmented to 72,000 windows. Each window is converted to the 38 features.
The eight datasets based on eight window sizes are run by KNN, SVM, SNN, and DNN.
The results yielded by four algorithms are compared to identify the optimal window size.
Based on the dataset with the optimal window size, PCA is used to reduce the dimension of the dataset.
Associated hyper-parameters are tuned to deliver the optimal KNN, SVM, SNN, and DNN.
The outstanding model of the four models is selected.

Figure 1. The overview of this study.

2.1. Overview of the Activities Analysed

To analyze the details of the four activities, a statistical study of the whole dataset including the maximum, mean, standard deviation, percentile of 20%, 50%, and 80% in three directions, X, Y, Z (defined in Figure 2), are carried out. It is noted that the assessment conducted in Table 1 is based on the whole dataset instead of a window presented in Table 2. It is noticeable that the activities performed in a treadmill experience a more dramatic vibration since the maximum of the ascending and running is approximately 10 m/s^2 greater than cycling and elliptical in all three directions. Regarding the mean value and the standard deviation, the cycling produces less fluctuation than the other three activities. Considering the three percentile values, most of the four activities' vibrations convergent to a relatively small value compared to the maximum value.

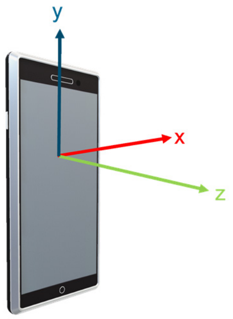

Figure 2. The coordinate system of the phone.

Table 1. Statistical analysis of the four activities.

Indicator (m/s^2)	Direction	Ascending	Cycling	Elliptical	Running
Maximum	X	55.6	22.9	32.6	56.1
	Y	56.9	20.6	37.0	47.7
	Z	42.7	33.0	47.1	45.7

Table 1. *Cont.*

Indicator (m/s^2)	Direction	Ascending	Cycling	Elliptical	Running
Mean	X	2.3	1.6	2.5	6.2
	Y	2.4	2.4	3.5	4.6
	Z	2.0	1.0	2.3	4.8
Standard deviation	X	1.9	1.4	2.2	4.9
	Y	2.2	1.9	2.6	3.3
	Z	2.1	0.9	1.8	4.1
P 20%	X	0.7	0.5	0.7	2.0
	Y	0.6	1.0	1.2	1.5
	Z	0.5	0.3	0.7	1.2
P 50%	X	1.9	1.3	1.9	5.3
	Y	1.8	2.4	3.0	4.1
	Z	1.4	0.7	1.9	3.7
P 80%	X	3.5	2.5	3.8	9.6
	Y	3.9	4.3	5.6	7.4
	Z	3.0	1.5	3.7	7.8

Table 2. Handcrafted features.

Category	Extracted Features
Frequency domain	Spectral energy
Time-domain	Max, min, average, standard deviation, percentile 20, percentile 50, percentile 80, interquartile, skewness, kurtosis, correlation, the standard deviation of SMV, an average of SMV, maximum of SMV, and a minimum of SMV.

2.2. Raw Data Collection

In this paper, we collect the dataset in a gym out of the lab using the phone-based accelerometer. Ten subjects are suggested to perform four activities ascending on a treadmill, cycling, elliptical, and running on a treadmill. To include the body location placement, the 10 participants place their phones (6 of them are iPhones, 2 are Samsung phones, and the remaining 2 are Huawei phones) in the pocket of the sweatpants, holding in one of their hands, or in their holsters attached on upper arm when they are performing the four activities. In particular, the phone is stored in the subject's pocket instead of a hand when the subjects are cycling as hands induce no movement. There are diverse models of phones used, which can improve the robustness and the generalization of the model if the model can cope with the data from different phones. An app "Vibsensor", available on the App Store for Apple phones and Google Play for Android phones, is used. The app casts the phone to a vibration meter with live displaying the data in real-time, collecting the vibration data, storing the data on the device, and exporting the data via email. All the volunteers are performing 30 min of each activity, leading to a total of 1200 min for 4 moves. The sample rate is 100 Hz, which is larger than 20 Hz as Maurer et al. assessed the impact of sampling rate on the accuracy of the HAR model and claimed that there was no noticeable enhancement when the sample rate increased from 20 Hz to 100 Hz [15]. With the sample rate 100 Hz and the 1200 min dataset, the complete picture of the dataset is that there are four Excel files ascending, cycling, elliptical, running) saving the vibrations in three directions. In each Excel file, it shapes as 300 min × 60 s × 100 Hz rows and four columns—timesteps and vibrations in x, y, and z directions defined by the phone coordinate system in Figure 2.

2.3. Handcrafted Features

To extract the 38 features in Table 2, Figure 3 takes the ascending as an example to illustrate the procedure generating the handcrafted feature dataset from the raw sensory dataset. The remaining three activities follow the same techniques. Eight window sizes are employed in our work, and the raw data are sliced to the corresponding windows and converted to the 38 features using MATLAB. The eight post-processed datasets are run by the four algorithms defined in Section 2.5. The Excel sheet of the raw dataset is now transformed to 38 columns representing 38 features and an additional column to show the type of activity. Regarding the row of the transformed dataset, the number of the rows can be calculated from the equation in Figure 3. Table 2 summarizes the features used in this study, and the detail of each attribute is reviewed. Extensive evaluation on a wide range of handcrafted feature types has been conducted in [16]. Most of the features from [16] are also available in our research. From Table 2, the features are computed in three directions except for the spectral energy, the standard deviation of signal magnitude vector (SMV), an average of SMV, a maximum of SMV, and a minimum of SMV. The correlation feature is calculated between two directions, which results in correlation (x, y), correlation (x, z), and correlation (y, z). Therefore, once every feature has been computed and stacked in a row, there are 38 features for each sample. Figure 4 presents a detailed view of a 2 s window. With the 100 Hz sample rate, each time step stands for 1 millisecond. The dataset's representation is now transformed based on the window sizes of 2 s and 40 s to the description shown in Table 3.

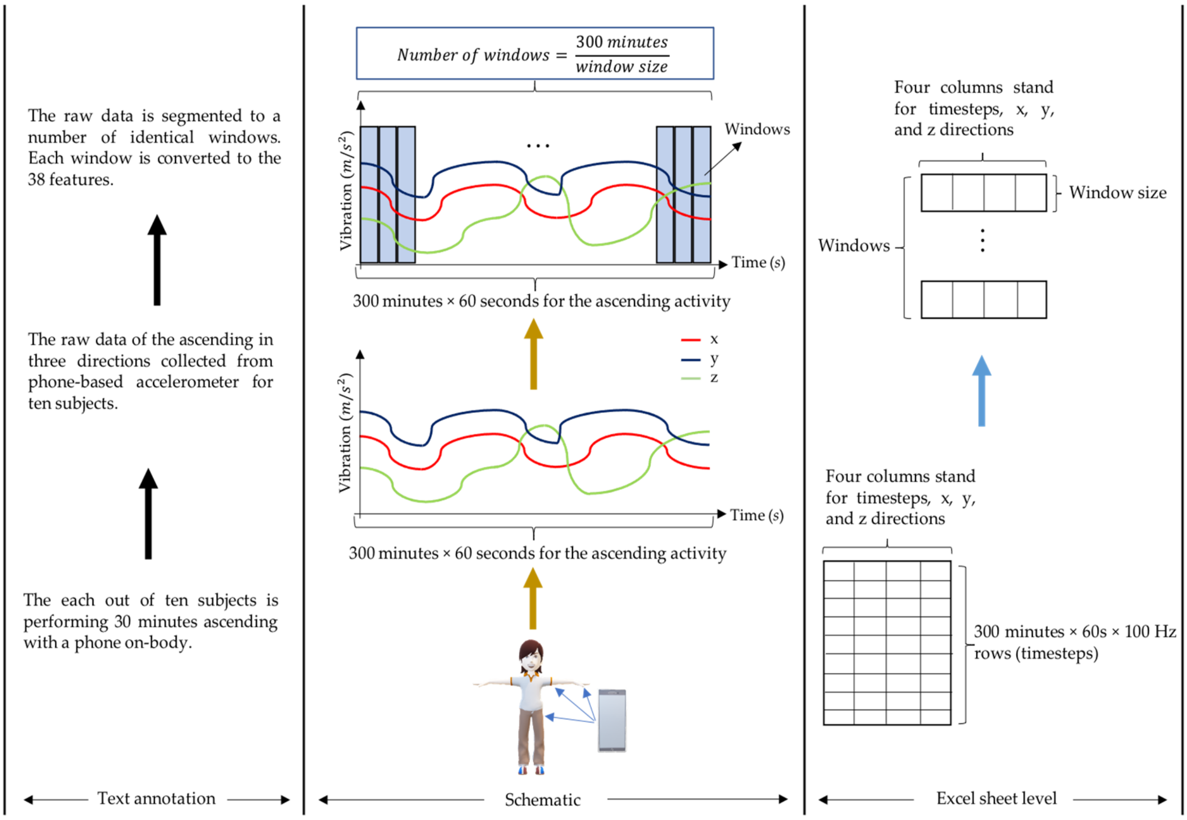
The raw data is segmented to a number of identical windows. Each window is converted to the 38 features.
The raw data of the ascending in three directions collected from phone-based accelerometer for ten subjects.
The each out of ten subjects is performing 30 minutes ascending with a phone on-body.
Text annotation
$Number\ of\ windows = \frac{300\ minutes}{window\ size}$
Windows
Vibration (m/s^2)
Time (s)
300 minutes × 60 seconds for the ascending activity
x
y
z
Vibration (m/s^2)
Time (s)
300 minutes × 60 seconds for the ascending activity
Schematic
Four columns stand for timesteps, x, y, and z directions
Window size
Windows
Four columns stand for timesteps, x, y, and z directions
300 minutes × 60s × 100 Hz rows (timesteps)
Excel sheet level

Figure 3. Handcrafted features computed from raw sensory data.

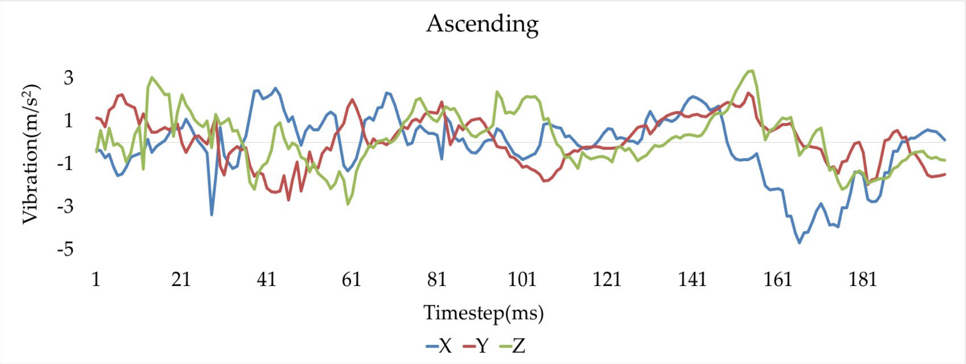

Figure 4. One window of ascending raw data.

Table 3. The overview of the dataset extracted from the raw data with window sizes of 2 s and 40 s.

Window Size	**2 s**			**40 s**		
Type of Activity	**No. Samples**	**No. Features**	**Total Subjects**	**No. Samples**	**No. Features**	**Total Subjects**
Ascending on a treadmill	9000	38	10	450	38	10
Cycling	9000	38	10	450	38	10
Elliptical	9000	38	10	450	38	10
Running on a treadmill	9000	38	10	450	38	10

After the handcrafted features are extracted, the dataset is split into 80% for the training set and 20% for the testing set. Min–Max normalization is deployed. In particular, the testing set is normalized using the maximum and minimum of the training set to avoid information leakage to the testing set.

2.4. PCA

PCA is probably the most popular multivariate technique that aims to reduce the dimensions of the data using some new orthogonal variables (principal components) to represent the original data [17]. It uses an orthogonal transformation to reconstruct a series of linearly related variables into a set of linearly unrelated new variables which are the principal components. The small structure of the data resulting from PCA provides the benefit of accurate and straightforward analysis [18]. To deliver the new representation of the dataset, PCA is used to reduce the dimension of the 38-feature dataset with the following goals [17]:

1. Obtain the most crucial information from the 38-feature dataset;
2. Exclude the less critical information to achieve a smaller dataset;
3. Simpify the description of the dataset;
4. Assess the attributes of the new representation.

To deliver these four goals, PCA maps n-dimensional features to k-dimensions. This k-dimension is a new orthogonal feature called principal components. PCA sequentially finds a set of mutually orthogonal coordinate axes from the original space. The first new coordinate axis shows the most considerable variance in the original data, and the second new axis is the plane orthogonal to the first new axis that maximizes the variance, while the third new axis is the enormous variance in the plane orthogonal to the first and second axes, and so on, until n such coordinate axes can be obtained. K coordinates with non-zero variances can be obtained, and the latter coordinate axes are almost zero variances, which can be discarded to achieve the goal of dimensional reduction. An explained variance ratio is provided in Appendix B to analyze the variance of components.

2.5. Machine Learning Algorithms

KNN, in pattern recognition, classifies samples based on the distance to the closest training samples in the feature map exemplified in Figure 5. It is noticeable that two axes are used to reflect two attributes of the sample in Figure 5. The meaning of the axis is also applicable in Figures 6 and 7. An unknown sample is assigned to one of the groups according to a majority vote of its k nearest neighbors. As can be seen in Figure 5, the k nearest neighbors is set to 5 (the k must be a positive integer and can be tuned to obtain the optimal performance), which implies that the five nearest points are identified by measuring the Euclidean distance between two points, and the unknow point is compared to the 5 points to determine which group the unknown point is allocated to. During the five closest samples, the majority are yellow samples, so that the unknown sample is classified to the yellow group.

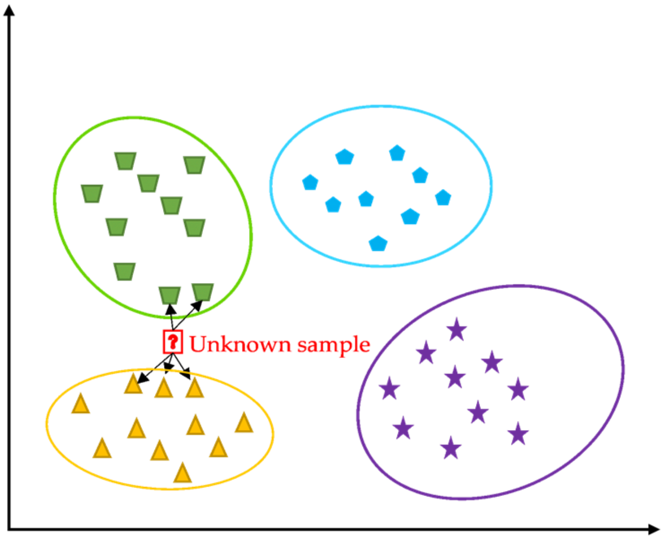

Figure 5. KNN classification example for k = 5. Four different shapes point to four different groups.

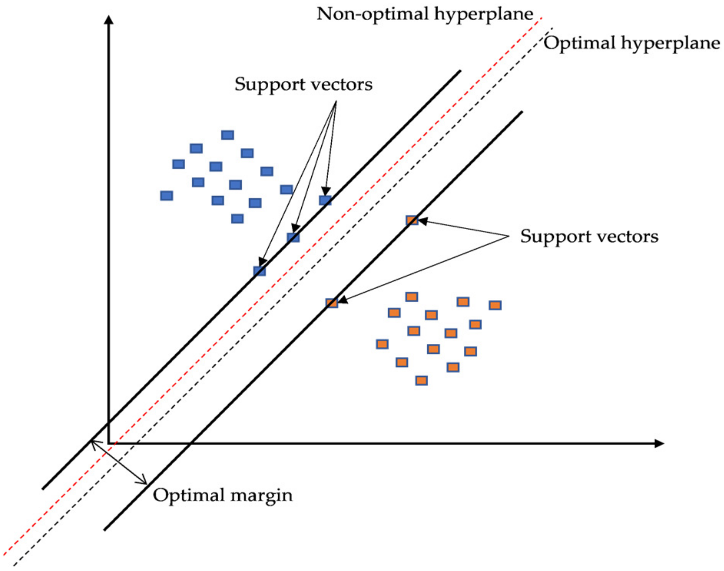

Figure 6. SVM classification example for two-group problems.

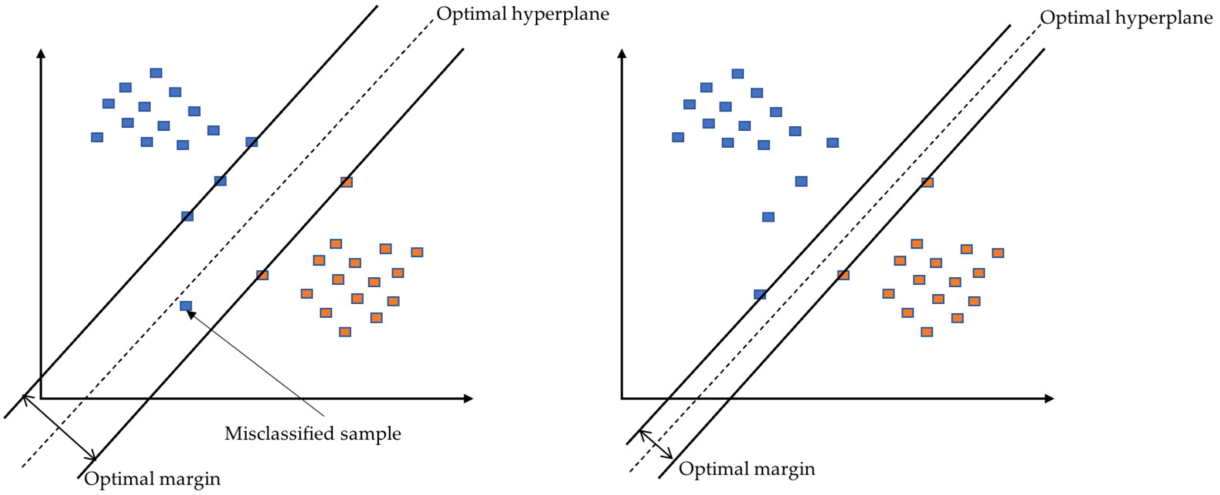

Figure 7. An example of the impact of low C (**left**) and high C (**right**).

SVM was first introduced for 2-group classification problems [19]. The binary input vectors are mapped to a feature map, and then the SVM is used to determine the hyperplane that can separate the binary input vectors seen from Figure 6. The support vectors characterize the most significant margin between the two groups. Many hyperplanes can separate the two classes, such as the two hyperplanes shown in Figure 6. SVM strives to find an optimal hyperplane that is the most far away (optimal margin) from any samples in the feature space. If the misclassified samples cannot be avoided, a constant C is introduced to define the tolerance of the misclassified samples. From Figure 7 (left), a low value of C ignores the misclassified sample to achieve a large optimal margin. In contrast, the right of Figure 7 provides a small optimal margin due to the immense value of C not allowing an error. The extension of SVM from binary classification to multiclass classification has been well developed and assessed in [20]. An artificial neural network (ANN) is inspired by the human brain structure [21], which is the fundamental reason it has intelligence. Figure 8 provides a typical example of SNN with only one hidden layer between the input and output layers. W, b, and z are the weights, biases, and the output of the hidden layer; the activation function denotes tanh, relu, or identity; and X and y are the input and output vectors, respectively. DNN shares the characteristics with SNN but more than one hidden layer.

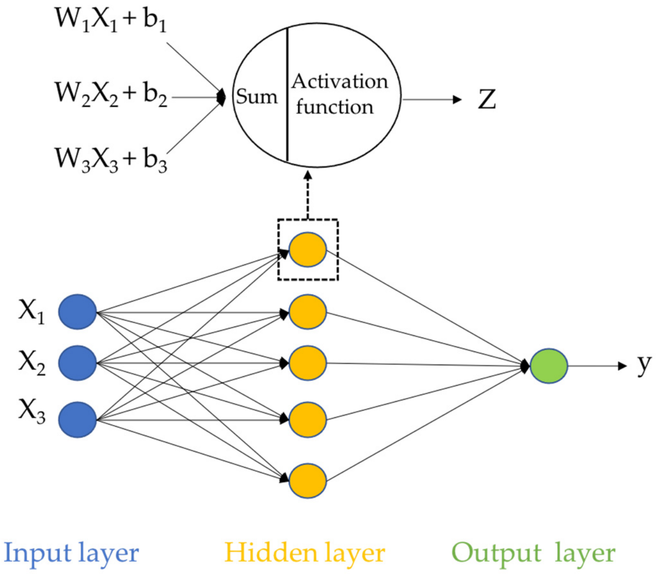

Figure 8. An example of an SNN network.

2.6. Evaluations Measures of Machine Learning Algorithms

To compare the performance of different machine learning classifiers, the most common method accuracy lacks focusing on a class. The introduction of a new learning problem inevitably focuses on its field but ignores the detailed analysis. Therefore, the main disadvantage of the most used empirical measure, accuracy, is no discrimination between the sum of appropriate labels of diverse classes. In [21], apart from accuracy, three additional measures were introduced—precision, recall, and F-measure, which can be calculated from Equations (1)–(4). In the present study, we also exploit these four measures since the accuracy is used to assess the model's overall performance, and the other three evaluate the model in each activity. Figure 9 unveils terms used in Equations (1)–(4), where the column exposes the actual labels, and the row represents the values classified. Accuracy is the number of samples correctly predicted divided by the number of all samples. Precision describes the relationship between correctly classified samples (true positives) and the ones inappropriately identified as positives (false positives). At the same time, the recall highlights the function of true positives and the falsely classified samples (false negatives). Sometimes, the precision and recall indicators are contradictory, so we need to consider them comprehensively using the F-measure, which balances the precision and recall. The worst to perfect cases of F-measures can be shown by the value from zero to one of the F-measure.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

$$recall = \frac{TP}{TP + FN} \tag{3}$$

$$F - measure = \frac{2 \times precision \times recall}{precision + recall} \tag{4}$$

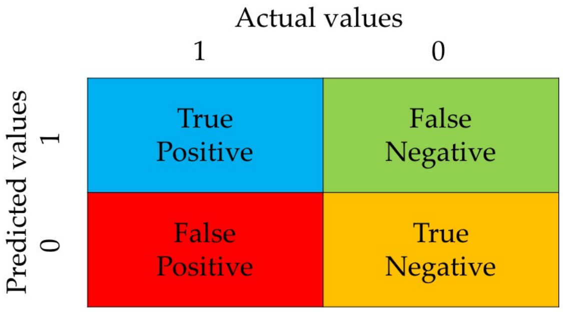

Figure 9. Confusion matrix.

3. Results

This chapter aims to find the optimal window size to compute handcrafted features, the optimal components to represent the original dataset, and the fine-tuned models producing optimal performance. The overall performance is presented in this section. More details are available in the Appendix A section.

3.1. Window Sizes

The measures mentioned in Section 2.5 generated from KNN, SVM, SNN, and DNN employing eight window sizes are presented in Table A1. Appendix A provides details of the evaluation of the window sizes providing the optimal window size for KNN and SVM is 2 s that offers the most accurate scenario claimed in [12]. However, the optimal window size for SNN and DNN is 40 s. Therefore, the following experiments are conducted under the window sizes of 2 s and 40 s depending on the algorithms. The hyper-parameter of each model is not tuned, and no PCA occurs at this stage. The hyper-parameters corresponding to each ML model are tabulated in Table 4.

Table 4. Hyper-parameter of the Four models.

Type of Algorithms	Parameter
KNN	1. Number of neighbors = 3
SVM	1. C = 10
SNN	1. Number of hidden layers = 1 2. Number of Neurons = 40 3. Activation function = tanh
DNN	1. Number of layers = 4 2. Number of neurons for each layer: layer 1 = 20, layer 2 = 20, layer 3 = 15, layer 4 = 15 3. Activation function = tanh

3.2. Explained Variance Ratio

In this section, PCA is implemented to extract the most informative features. Figure A1 shows the explained variance ratio that presents the percentage of the original information that can be described by the extracted features using PCA. There is evidence that the 18 features and 15 features configurations offer more than 99% information of the original datasets with a 2 s window size and a 40 s window size.

3.3. Hyper-Parameter Tunning

In this section, we present the process of hyper-parameter tuning to all four algorithms. The number of neighbors is tuned for KNN; the value of C is tuned for SVM; the number of neurons and the activation function are tuned for SNN and DNN; an additional hyper-parameter, the number of hidden layers, is also adjusted for DNN.

Figure 10 offers how the testing accuracy of KNN responses to the change of the number of neighbors. The testing accuracy starts at 0.961 and suffers an overall drop to 0.957 at the number of neighbors of 10. The trend of the graph shows the optimal number of neighbors is one.

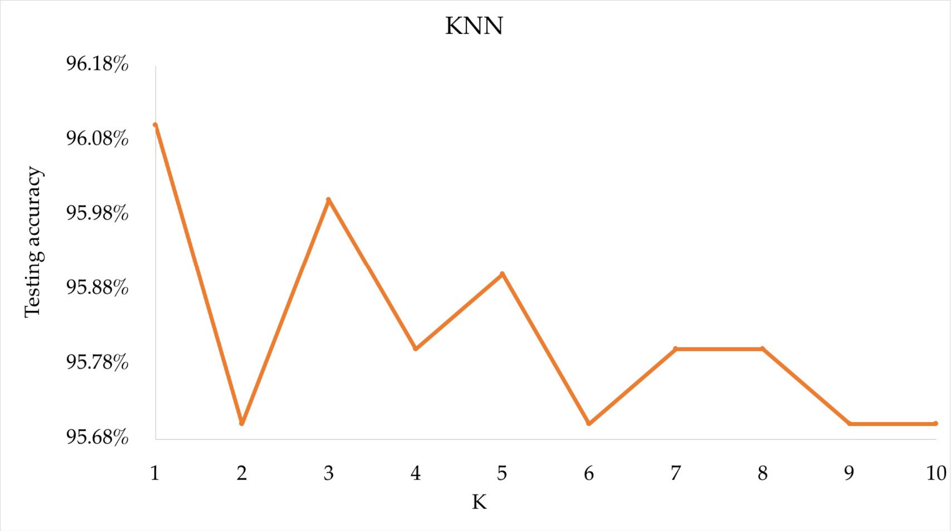

Figure 10. The testing accuracy of KNN for k = 1 to k = 10.

The result obtained from Figure 11 is that when C is equal to 180, the testing accuracy reaches its peak of testing accuracy 0.970 and remains at that level, although C is turned to 200.

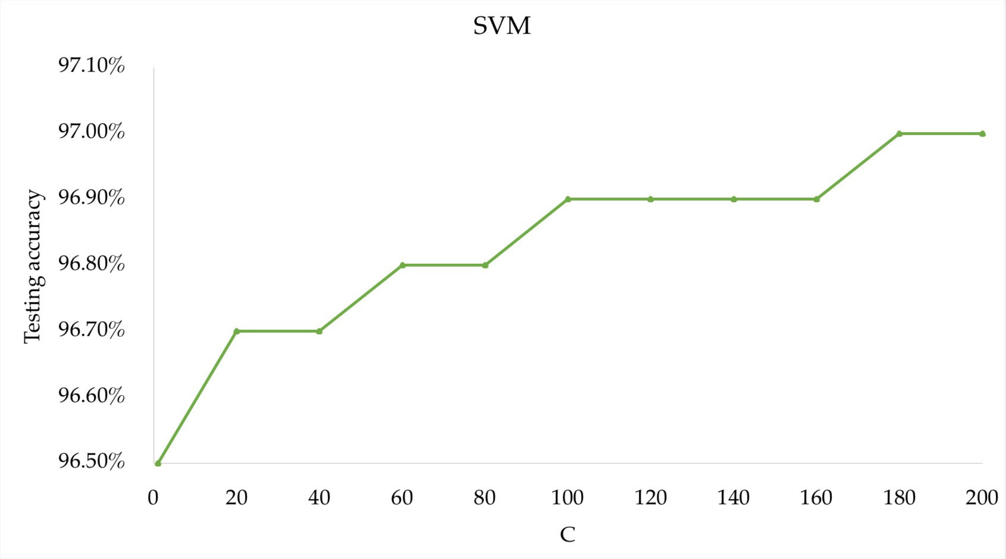

Figure 11. The testing accuracy of SVM associated with C.

What stands out from Figure 12 is that the optimal result of 0.972 is provided by the relu and 80 neurons configuration. We can see a higher performance reported by the tanh and relu compared to identity.

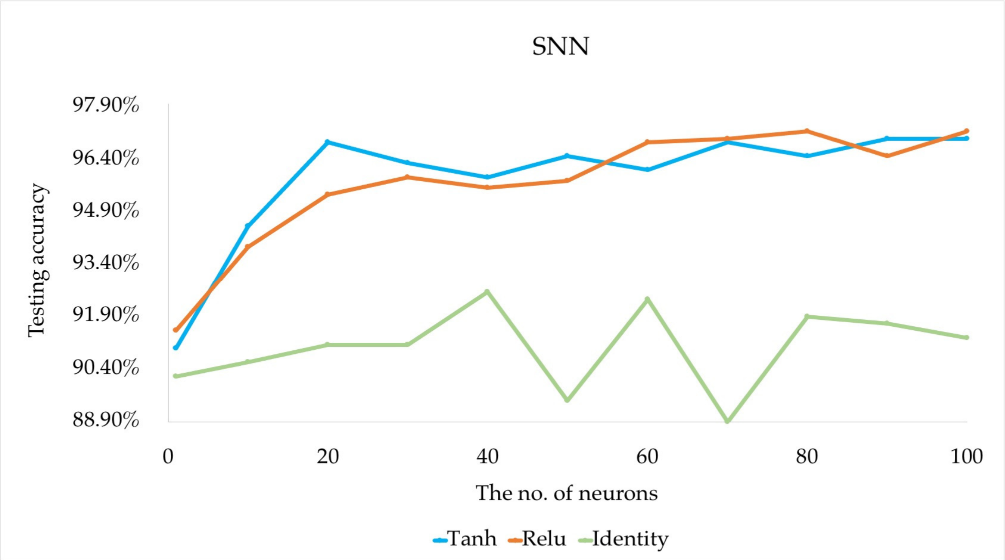

Figure 12. The testing accuracy of SNN responses to the three types of activation function and the number of neurons.

From Tables A4–A6, we already know that the 80 neurons provide the optimal performance for SNN with the relu activation function. As aforementioned, the DNN features two or more hidden layers compared to the SNN. Therefore, the DNN is tuned by adding more hidden layers to the optimal SNN to see any improvements. Figure 13 presents the optimal capability is 0.974 procured by 3 hidden layers with 80, 10, and 10 neurons in each hidden layer. As can be seen from Tables A5 and A7, there is a slight rise in the testing accuracy of 0.002 from SNN to DNN. Overfitting refers to the trained model having poor performance on the testing dataset but good performance on the training dataset [22]. There is no significant gap between them the training accuracy and testing accuracy in each model. The model does not encounter an overfitting problem.

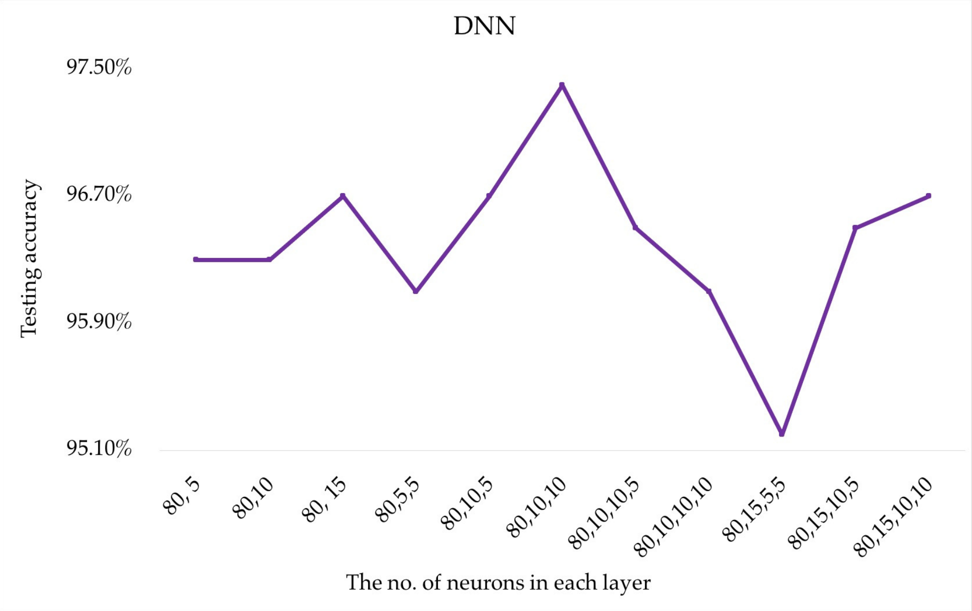

Figure 13. The testing accuracy of DNN responses to the No. layers, the No. neurons in each layer, and the Relu function.

4. Discussion

This study set out to develop a high-performance gym activity classifier for smartphones. The primary constraint is the storage and the computational capability of the phone. There is a requirement for phones that need to sense the subject's movement continuously. In addition, the model implemented on the phone needs to update constantly. These two actions require immense computing resources. Therefore, the proposed model is designed to use one sensor and a dataset with a small size. It is not practical for a phone to use raw sensory data because of the large requirement of computation energy. We strive to transform the raw data into a dataset that is small and keeps as much helpful information as possible. First of all, the variable—window size—is analyzed, and it is inferred that 40 s is favorable. Every 40 s, raw data are converted to a wide range of handcrafted features (38 features) followed by a PCA technique to further reduce 38 elements to 15 elements. Table A1 tells that the 38-feature dataset is sensible due to the satisfaction performance offered by the four models. However, the size of the dataset can be reduced more by PCA. DNN can produce the same result with a 15-element dataset resulting from PCA.

The DNN model procures an error of 0.026 using a 360-sample testing set. We can calculate the 95% confidence interval with a z-value of 1.96 producing the confidence interval [0.0096, 0.0426]. It implies that there is a 95% possibility that the interval contains the true classification error of the DNN model on unseen data.

Qi et al. [3] proposed a state-of-the-art technique to classify gym aerobic and weightlifting activities with wearable sensors. Without a throughout window size analysis, they also derived data from 10 subjects but with multiple sensors. However, only an accelerometer was used to detect the aerobic exercises of running, cycling, and ascending [3,23–25]. In addition, PCA was also exploited to reduce the 88 handcrafted features to 36 features. The optimal results from us and [3,26] are listed together in Table 5.

Table 5. Comparison analysis between [3] (green) and the proposed model (red).

	Ascending	Ascending	Cycling	Cycling	Running	Running
Precision	0.824	0.935	0.826	0.993	0.904	0.993
Recall	0.788	0.963	0.812	0.978	0.882	0.983
F-measure	0.805	0.949	0.898	0.985	0.912	0.988

Regarding the three measures in Table 5, our model outperforms the baseline model classifying ascending, cycling, running in the gym. However, the potential performance of the proposed model classifying other gym exercises remains unknown. Future investigations might be possible to quantify how much computational cost can be saved by reducing the dataset. Further studies will need to be undertaken which take the computational time and the battery used if the model is implemented to a smartphone into account.

5. Conclusions

Ensuring healthy styles and promoting wellbeing at all ages is critical to sustainable development. This study addresses one of the United Nation's sustainable development goals, with respect to Goal #3 'Good health and wellbeing'. Active lifestyles and physical activities (or exercises) can help in increasing life expectancy and reducing some of the common risk factors associated with obesity and heart diseases. However, more efforts and incentives are required to fully eradicate a wide range of diseases and address many different persistent and emerging health issues. By fostering on providing more efficient recognition, classification, and quantification of physical activities, improved understanding, and increased access to fitness activities, significant progress can be made in helping to improve the quality of lives of millions. In this study, novel deep neural networks have been established to predict the most common fitness activities, including four gym aerobic activities ascending on a treadmill, cycling, elliptical, and running on a treadmill to prompt more physical activities in gyms.

In order to aid the data collection and quantification of fitness activities, an AI-based mobile app integrated with smartphone sensors has been developed. An accelerometer embedded in a phone has been calibrated with a Brüel & Kjær industrial transducer and then used to collect the time series data of gym activities. The smartphone has been adopted since mobile phones have been used in the daily lives of potential end-users nowadays, and the novel AI-based apps developed can be implemented within smartphones for daily usage.

In addition, this research examines various ML models classifying four gym cardio activities. The experiments on data fusion and assimilation techniques have contributed to a new understanding of the impact of the window size during feature extraction from raw data to handcrafted features, the effectiveness of the feature extraction to further reduce the data dimensions, and how we tune the hyper-parameter to obtain optimal ML models. The proposed method is compared with an advanced solution to gym activity recognition to show outstanding performance. This development can be further implemented in the App Store in the future to enhance the public usage of the apps so that active physical human activities can be promoted to enhance 'Good health and Well-being' in accordance with United Nation's sustainable development goals (Goal #3).

Author Contributions: Conceptualization, S.K.; methodology, S.K.; software, J.H. and J.N.; validation, J.H. and J.N.; formal analysis, J.N.; investigation, S.K., J.H. and J.N.; resources, S.K.; writing—original draft preparation, J.H.; writing—review and editing, S.K. and J.H.; visualization, J.N.; supervision, S.K.; project administration, S.K.; funding acquisition, S.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by European Commission, grant numbers: H2020-MSCA-RISE No. 691135 and H2020-S2R No. 730849. The APC was funded by MDPI's Invited Paper Program.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data can be made available upon reasonable request.

Acknowledgments: Special thanks to European Commission for H2020-MSCA-RISE Project No. 691135 "RISEN: Rail Infrastructure Systems Engineering Network" (www.risen2rail.eu (accessed on 29 December 2021)). Partial support from H2020 Shift2RailProject No 730849 (S-Code) is acknowledged. In addition, the sponsorships and assistance from Ministry of National Education (Turkey), Network Rail, RSSB (Rail Safety and Standard Board, UK) are highly appreciated.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A. Window Sizes

Table A1 gauges the performance of four models in terms of accuracy, recall, precision, and F-measure. The precision, recall, and F-measure of the four activities are ordered as ascending, cycling, elliptical, and running activities. The highest testing accuracy is marked in red to assist reading, which is also applicable in Tables A1–A7. To access the optimal window size, the first indicator for the overall performance is the testing accuracy. The remaining three measures, precision, recall, and F-measure are used to confirm that the values in red are the best without any potential biases.

Table A1. Detail performances of four models by using different window sizes.

Window Sizes Measures	1 s	2 s	5 s	10 s	20 s	40 s	60 s	80 s
KNN training accuracy	0.979	0.981	0.977	0.972	0.966	0.961	0.939	0.930
KNN testing accuracy	0.959	0.965	0.959	0.952	0.937	0.904	0.848	0.856
KNN precision	0.931 0.981 0.939 0.987	0.939 0.984 0.957 0.980	0.945 0.976 0.940 0.976	0.923 0.971 0.936 0.976	0.875 0.992 0.891 0.989	0.811 0.935 0.919 0.967	0.778 0.897 0.783 0.931	0.747 0.891 0.845 0.984
KNN recall	0.943 0.957 0.964 0.971	0.952 0.967 0.963 0.976	0.928 0.964 0.971 0.974	0.911 0.960 0.945 0.990	0.941 0.889 0.933 0.989	0.896 0.841 0.913 0.975	0.811 0.805 0.783 0.979	0.903 0.814 0.790 0.909
KNN F-measure	0.937 0.969 0.952 0.979	0.946 0.976 0.960 0.978	0.937 0.970 0.956 0.975	0.917 0.966 0.940 0.983	0.907 0.938 0.912 0.989	0.851 0.885 0.916 0.971	0.794 0.848 0.783 0.954	0.818 0.851 0.817 0.945
SVM training accuracy	0.953	0.961	0.958	0.942	0.926	0.904	0.835	0.770
SVM testing accuracy	0.950	0.961	0.952	0.945	0.917	0.869	0.814	0.793
SVM precision	0.905 0.979 0.935 0.986	0.922 0.983 0.960 0.982	0.907 0.985 0.949 0.972	0.906 0.975 0.915 0.982	0.824 0.992 0.869 0.985	0.782 0.940 0.803 0.992	0.885 0.784 0.685 0.947	0.884 0.762 0.642 0.968
SVM recall	0.930 0.957 0.945 0.969	0.952 0.960 0.960 0.973	0.935 0.952 0.958 0.965	0.900 0.953 0.948 0.976	0.882 0.885 0.933 0.968	0.776 0.797 0.927 0.983	0.568 0.874 0.892 0.938	0.528 0.914 0.839 0.909
SVM F-measure	0.917 0.968 0.940 0.978	0.937 0.971 0.960 0.977	0.921 0.968 0.954 0.968	0.903 0.964 0.931 0.979	0.852 0.936 0.900 0.977	0.779 0.863 0.861 0.987	0.692 0.826 0.775 0.942	0.661 0.831 0.727 0.937
SNN training accuracy	0.978	0.991	0.995	0.991	0.992	0.994	0.995	0.968

Table A1. *Cont.*

Window Sizes **Measures**	**1 s**	**2 s**	**5 s**	**10 s**	**20 s**	**40 s**	**60 s**	**80 s**
SNN testing accuracy	0.967	0.973	0.974	0.974	0.973	0.982	0.967	0.933
SNN precision	0.953 0.975 0.957 0.983	0.963 0.980 0.978 0.972	0.959 0.983 0.974 0.982	0.967 0.964 0.987 0.976	0.943 0.997 0.969 0.982	0.943 0.993 0.993 0.999	0.948 0.999 0.962 0.960	0.930 0.999 0.900 0.903
SNN recall	0.943 0.974 0.972 0.976	0.957 0.979 0.972 0.986	0.966 0.984 0.969 0.978	0.956 0.977 0.970 0.990	0.976 0.970 0.965 0.982	0.985 0.978 0.973 0.992	0.958 0.989 0.916 0.999	0.917 0.957 0.871 0.985
SNN F-measure	0.948 0.975 0.964 0.980	0.960 0.979 0.975 0.979	0.962 0.984 0.972 0.98	0.962 0.971 0.979 0.983	0.959 0.983 0.967 0.982	0.964 0.985 0.983 0.996	0.953 0.994 0.938 0.980	0.923 0.978 0.885 0.942
DNN training accuracy	0.979	0.984	0.980	0.986	0.990	0.995	0.992	0.944
DNN testing accuracy	0.967	0.970	0.963	0.961	0.957	0.974	0.942	0.896
DNN precision	0.949 0.981 0.953 0.985	0.941 0.982 0.985 0.975	0.925 0.989 0.975 0.968	0.943 0.971 0.963 0.967	0.937 0.983 0.925 0.978	0.949 0.999 0.973 0.975	0.937 0.977 0.924 0.929	0.882 0.957 0.864 0.877
DNN recall	0.945 0.972 0.975 0.975	0.966 0.973 0.956 0.987	0.959 0.967 0.949 0.978	0.939 0.966 0.959 0.979	0.929 0.956 0.965 0.975	0.970 0.971 0.967 0.992	0.937 0.989 0.880 0.958	0.833 0.957 0.823 0.970
DNN F-measure	0.947 0.977 0.964 0.980	0.953 0.978 0.970 0.980	0.942 0.978 0.962 0.973	0.941 0.969 0.961 0.973	0.933 0.969 0.944 0.977	0.959 0.985 0.970 0.983	0.937 0.983 0.901 0.944	0.857 0.957 0.843 0.921

Appendix B. PCA

An index named explained variance ratio in PCA represents the percentage of the variance of each retrained component. As can be seen from Figure A1, it shows a relationship between the number of principal components and the explained variance ratio. The larger the percentage, the more information is retained in the original data.

The line marked in blue implies that 15 components can represent almost 100% of the original dataset for the 40 s window size, while 18 components are employed when the window size is 2 s. Therefore, the original 38-feature dataset is reduced to a 15-feature dataset for SNN and DNN and an 18-feature dataset for KNN and SVM.

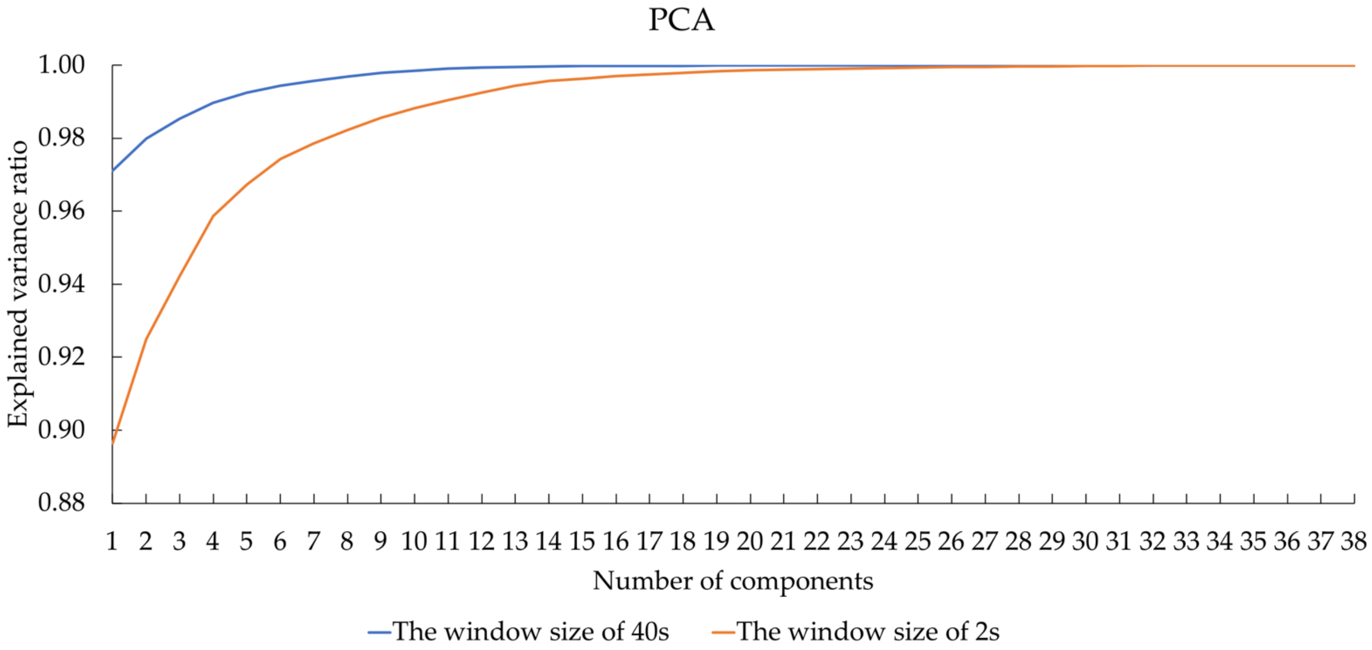

Figure A1. The dimensionality reduction driven by PCA for the two datasets with different window sizes.

Appendix C. Hyper-Parameter Tuning

For KNN, the number of neighbors is tuned to deliver optimal performance. The result is summarized in Table A2. KNN is good at classifying running during the four activities since the highest precision, recall, and F-measure are yielded for running. On the contrary, KNN is not sensitive to ascending.

Table A2. KNN outcomes for the number of neighbors tuned.

Number of Neighbors	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Training accuracy	0.990	0.980	0.978	0.972	0.971	0.967	0.966	0.964	0.963	0.962
Testing accuracy	0.961	0.957	0.960	0.958	0.959	0.957	0.958	0.958	0.957	0.957
Precision	0.939 0.974 0.952 0.979	0.908 0.975 0.959 0.989	0.935 0.979 0.949 0.980	0.924 0.981 0.948 0.982	0.936 0.980 0.943 0.979	0.926 0.981 0.943 0.981	0.934 0.981 0.940 0.980	0.928 0.981 0.942 0.982	0.931 0.982 0.937 0.981	0.928 0.981 0.938 0.983
Recall	0.943 0.968 0.957 0.975	0.969 0.964 0.935 0.960	0.948 0.962 0.959 0.974	0.954 0.960 0.949 0.969	0.944 0.963 0.957 0.972	0.948 0.961 0.952 0.969	0.942 0.961 0.959 0.970	0.948 0.961 0.953 0.968	0.942 0.959 0.957 0.970	0.946 0.959 0.954 0.968
F-measure	0.941 0.971 0.955 0.977	0.938 0.970 0.947 0.974	0.941 0.970 0.954 0.977	0.939 0.970 0.948 0.975	0.940 0.971 0.950 0.975	0.937 0.971 0.948 0.975	0.938 0.971 0.949 0.975	0.938 0.971 0.947 0.975	0.937 0.970 0.947 0.975	0.937 0.970 0.946 0.975

Table A3 tunes a wide range of values of C to get the optimal SVM model. When the C is set to 180, SVM outperforms the model with other C settings. Like KNN, SVM is also sensitive to running. It is understandable since the running spurs the most dramatic movement compared to ascending, cycling, and elliptical.

Table A3. SVM outcomes for the number of C.

Number of C	1	20	40	60	80	100	120	140	160	180	200
Training accuracy	0.967	0.972	0.974	0.975	0.977	0.978	0.978	0.979	0.979	0.980	0.980
Testing accuracy	0.965	0.967	0.967	0.968	0.968	0.969	0.969	0.969	0.969	0.970	0.970
Precision	0.935 0.978 0.961 0.986	0.938 0.978 0.963 0.988	0.940 0.979 0.963 0.987	0.941 0.980 0.964 0.987	0.943 0.980 0.964 0.987	0.944 0.981 0.964 0.987	0.944 0.981 0.965 0.987	0.943 0.982 0.966 0.987	0.944 0.980 0.965 0.987	0.945 0.981 0.966 0.989	0.945 0.981 0.966 0.989
Recall	0.955 0.966 0.962 0.976	0.957 0.968 0.962 0.979	0.957 0.971 0.962 0.980	0.957 0.971 0.962 0.981	0.958 0.971 0.963 0.982	0.959 0.971 0.962 0.982	0.959 0.971 0.964 0.982	0.960 0.972 0.963 0.982	0.959 0.972 0.963 0.982	0.961 0.974 0.963 0.983	0.961 0.974 0.963 0.983
F-measure	0.945 0.972 0.962 0.981	0.948 0.973 0.963 0.984	0.948 0.975 0.963 0.984	0.949 0.975 0.963 0.984	0.950 0.976 0.963 0.984	0.951 0.976 0.963 0.985	0.951 0.976 0.964 0.984	0.951 0.977 0.964 0.985	0.951 0.976 0.964 0.985	0.953 0.977 0.965 0.986	0.953 0.977 0.965 0.986

Tables A4–A6 present the tunning process for the number of neurons and the type of activation functions of SNN. Going deep to the activity level, SNN performs better when classifying cycling than KNN and SVM, leading to a slightly better overall performance.

Table A4. SNN outcomes for the No. neurons and the activation function tanh.

Number of Neurons	1	10	20	30	40	50	60	70	80	90	100
Training accuracy	0.944	0.985	0.990	0.990	0.994	0.995	0.998	0.999	0.999	0.999	0.999
Testing accuracy	0.910	0.945	0.969	0.963	0.959	0.965	0.961	0.969	0.965	0.970	0.970
Precision	0.857 0.985 0.858 0.944	0.904 0.971 0.938 0.967	0.934 0.993 0.966 0.983	0.915 0.985 0.966 0.992	0.906 0.985 0.966 0.983	0.914 0.985 0.979 0.983	0.913 0.985 0.966 0.983	0.927 0.993 0.973 0.983	0.908 0.999 0.959 0.999	0.921 0.993 0.979 0.992	0.934 0.978 0.980 0.992
Recall	0.806 0.964 0.887 0.992	0.910 0.978 0.913 0.983	0.955 0.971 0.960 0.992	0.963 0.964 0.940 0.992	0.940 0.971 0.940 0.992	0.55 0.971 0.947 0.992	0.940 0.971 0.947 0.992	0.948 0.978 0.960 0.992	0.963 0.971 0.940 0.992	0.963 0.978 0.953 0.990	0.955 0.978 0.960 0.992
F-measure	0.831 0.974 0.872 0.967	0.907 0.975 0.926 0.975	0.945 0.982 0.963 0.988	0.938 0.974 0.953 0.992	0.923 0.978 0.953 0.988	0.934 0.978 0.963 0.988	0.926 0.978 0.956 0.988	0.937 0.985 0.966 0.988	0.935 0.985 0.949 0.996	0.942 0.985 0.966 0.992	0.945 0.978 0.970 0.992

Table A5. SNN outcomes for the No. neurons and the activation function relu.

Number of Neurons	1	10	20	30	40	50	60	70	80	90	100
Training accuracy	0.927	0.979	0.973	0.986	0.983	0.988	0.990	0.993	0.992	0.995	0.992
Testing accuracy	0.915	0.939	0.954	0.959	0.956	0.958	0.969	0.970	0.972	0.965	0.972
Precision	0.833 0.954 0.925 0.959	0.882 0.956 0.940 0.983	0.924 0.978 0.924 0.999	0.914 0.985 0.959 0.983	0.899 0.978 0.952 0.999	0.913 0.978 0.966 0.975	0.947 0.985 0.954 0.992	0.941 0.985 0.966 0.992	0.955 0.993 0.961 0.983	0.914 0.993 0.966 0.992	0.955 0.978 0.961 0.999
Recall	0.896 0.906 0.900 0.967	0.896 0.942 0.940 0.983	0.903 0.964 0.967 0.983	0.948 0.971 0.947 0.975	0.933 0.971 0.933 0.992	0.940 0.964 0.947 0.983	0.940 0.978 0.967 0.992	0.955 0.978 0.960 0.992	0.940 0.978 0.980 0.992	0.955 0.971 0.947 0.992	0.948 0.978 0.973 0.992

Table A5. *Cont.*

Number of Neurons	1	10	20	30	40	50	60	70	80	90	100
F-measure	0.863	0.889	0.913	0.930	0.916	0.926	0.944	0.948	0.947	0.934	0.951
	0.929	0.949	0.971	0.978	0.975	0.971	0.982	0.982	0.985	0.982	0.978
	0.912	0.940	0.945	0.953	0.943	0.956	0.960	0.963	0.970	0.956	0.967
	0.963	0.983	0.992	0.979	0.996	0.979	0.992	0.992	0.988	0.992	0.996

Table A6. SNN outcomes for the No. neurons and the activation function identity.

Number of Neurons	1	10	20	30	40	50	60	70	80	90	100
Training accuracy	0.907	0.918	0.917	0.920	0.93	0.917	0.931	0.903	0.930	0.923	0.921
Testing accuracy	0.902	0.906	0.911	0.911	0.926	0.895	0.924	0.889	0.919	0.917	0.913
Precision	0.807	0.807	0.831	0.821	0.883	0.901	0.882	0.748	0.846	0.835	0.895
	0.963	0.992	0.985	0.992	0.999	0.978	0.992	0.992	0.993	0.993	0.978
	0.879	0.866	0.868	0.872	0.862	0.781	0.863	0.874	0.875	0.879	0.830
	0.967	0.975	0.975	0.975	0.975	0.967	0.975	0.975	0.975	0.975	0.975
Recall	0.813	0.843	0.843	0.858	0.843	0.679	0.836	0.866	0.858	0.866	0.761
	0.949	0.957	0.964	0.957	0.964	0.971	0.957	0.949	0.964	0.964	0.964
	0.873	0.860	0.873	0.867	0.920	0.953	0.927	0.787	0.887	0.873	0.947
	0.983	0.975	0.975	0.975	0.983	0.975	0.983	0.975	0.975	0.975	0.983
F-measure	0.810	0.825	0.837	0.839	0.863	0.774	0.858	0.803	0.852	0.850	0.823
	0.956	0.974	0.974	0.974	0.982	0.975	0.974	0.970	0.978	0.978	0.971
	0.876	0.863	0.870	0.870	0.890	0.859	0.894	0.828	0.881	0.876	0.885
	0.975	0.975	0.975	0.975	0.979	0.971	0.979	0.975	0.975	0.975	0.979

In Table A7, DNN, with additional layers, shows more precise performance to classify elliptical and running compared to SNN. DNN performs better at classifying elliptical and running but worse at ascending compared to SNN.

Table A7. DNN outcomes response to the No. layer, the No. neurons in each layer, and the relu function.

Number of Neurons in Each Hidden Layer	80, 5	80, 10	80, 15	80, 5, 5	80, 10, 5	80, 10, 10	80, 10, 10, 5	80, 10, 10, 10	80, 15, 5, 5	80, 15, 10, 5	80, 15, 10, 10
Training accuracy	0.994	0.995	0.998	0.995	0.994	0.999	0.996	0.999	0.993	0.999	0.999
Testing accuracy	0.963	0.963	0.967	0.961	0.967	0.974	0.965	0.961	0.952	0.965	0.967
Precision	0.914	0.908	0.927	0.920	0.933	0.935	0.915	0.914	0.923	0.934	0.915
	0.985	0.978	0.978	0.999	0.985	0.993	0.978	0.993	0.964	0.964	0.985
	0.979	0.972	0.973	0.940	0.966	0.973	0.979	0.966	0.935	0.966	0.973
	0.975	0.999	0.992	0.992	0.983	0.993	0.992	0.975	0.992	0.999	0.999
Recall	0.955	0.955	0.948	0.940	0.940	0.963	0.963	0.955	0.896	0.948	0.963
	0.957	0.971	0.971	0.978	0.978	0.978	0.971	0.964	0.971	0.964	0.971
	0.953	0.940	0.960	0.933	0.960	0.967	0.940	0.940	0.953	0.953	0.947
	0.992	0.992	0.992	0.999	0.992	0.983	0.992	0.992	0.992	0.999	0.992
F-measure	0.934	0.931	0.937	0.930	0.937	0.949	0.938	0.934	0.909	0.941	0.938
	0.971	0.975	0.975	0.989	0.982	0.985	0.975	0.978	0.968	0.964	0.978
	0.966	0.956	0.966	0.936	0.963	0.970	0.959	0.953	0.944	0.960	0.959
	0.983	0.996	0.992	0.996	0.988	0.988	0.992	0.983	0.992	0.999	0.996

References

1. Blair, S.N. Physical inactivity: The biggest public health problem of the 21st century. *Br. J. Sports Med.* **2009**, *43*, 294–305.
2. Kohl, H.W., 3rd; Craig, C.L.; Lambert, E.V.; Inoue, S.; Alkandari, J.R.; Leetongin, G.; Kahlmeier, S.; Lancet Physical Activity Series Working Group. The pandemic of physical inactivity: Global action for public health. *Lancet* **2012**, *380*, 294–305. [CrossRef]
3. Qi, J.; Yang, P.; Hanneghan, M.; Tang, S.; Zhou, B. A Hybrid Hierarchical Framework for Gym Physical Activity Recognition and Measurement Using Wearable Sensors. *IEEE Internet Things J.* **2019**, *6*, 1384–1393. [CrossRef]

4. Koskimäki, H.; Siirtola, P.; Röning, J. Myogym: Introducing an open gym data set for activity recognition collected using myo armband. In Proceedings of the UbiComp ′17: 2017 ACM International Joint Conference on Pervasive and Ubiquitous Computing and 2017 ACM International Symposium on Wearable Computers, Maui, HI, USA, 11–15 September 2017.
5. Koskimäki, H.; Siirtola, P. Recognizing gym exercises using acceleration data from wearable sensors. In Proceedings of the 2014 IEEE Symposium on Computational Intelligence and Data Mining (CIDM), Orlando, FL, USA, 9–12 December 2014; pp. 321–328.
6. Khan, U.A.; Khan, I.A.; Din, A.; Jadoon, W.; Jadoon, R.N.; Khan, M.A.; Khan, F.G.; Khan, A.N. Towards a Complete Set of Gym Exercises Detection Using Smartphone Sensors. *Sci. Program.* **2020**, *2020*, 1–12. [CrossRef]
7. Li, K.; Habre, R.; Deng, H.; Urman, R.; Morrison, J.; Gilliland, F.D.; Ambite, J.L.; Stripelis, D.; Chiang, Y.-Y.; Lin, Y.; et al. Applying Multivariate Segmentation Methods to Human Activity Recognition from Wearable Sensors' Data. *JMIR Mhealth Uhealth* **2019**, *7*, e11201. [CrossRef] [PubMed]
8. Khalid, S.; Khalil, T.; Nasreen, S. A survey of feature selection and feature extraction techniques in machine learning. In Proceedings of the 2014 Science and Information Conference (SAI), London, UK, 27–29 August 2014; pp. 372–378.
9. Yang, J.; Zhang, D.; Frangi, A.F.; Yang, J.Y. Two-dimensional PCA: A new approach to appearance-based face representation and recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2004**, *26*, 131–137. [CrossRef] [PubMed]
10. Jokanovic, B.; Amin, M.; Ahmad, F.; Boashash, B. Radar fall detection using principal component analysis. In *Radar Sensor Technology*; International Society for Optics and Photonics (SPIE): Bellingham, WA, USA, 2016; Volume 20, p. 982919.
11. Song, F.; Guo, Z.; Mei, D. Feature selection using principal component analysis. In Proceedings of the 2010 International Conference on System Science, Engineering Design and Manufacturing Informatization, Yichang, China, 12–14 November 2010.
12. Banos, O.; Galvez, J.-M.; Damas, M.; Pomares, H.; Rojas, I. Window Size Impact in Human Activity Recognition. *Sensors* **2014**, *14*, 6474–6499. [CrossRef] [PubMed]
13. Weiss, G.M.; Timko, J.L.; Gallagher, C.M.; Yoneda, K.; Schreiber, A.J. Smartwatch-based activity recognition: A machine learning approach. In Proceedings of the 2016 IEEE-EMBS International Conference on Biomedical and Health Informatics (BHI), Las Vegas, NV, USA, 24–27 February 2016.
14. Ahmadi, M.; O'Neil, M.; Fragala-Pinkham, M.; Lennon, N.; Trost, S. Machine learning algorithms for activity recognition in ambulant children and adolescents with cerebral palsy. *J. Neuroeng. Rehabil.* **2018**, *15*, 105. [CrossRef] [PubMed]
15. Maurer, U.; Smailagic, A.; Siewiorek, D.P.; Deisher, M. Activity recognition and monitoring using multiple sensors on different body positions. In Proceedings of the International Workshop on Wearable and Implantable Body Sensor Networks (BSN'06), Cambridge, MA, USA, 3–5 April 2006.
16. Amjad, F.; Khan, M.; Nisar, M.; Farid, M.; Grzegorzek, M. A Comparative Study of Feature Selection Approaches for Human Activity Recognition Using Multimodal Sensory Data. *Sensors* **2021**, *21*, 2368. [CrossRef] [PubMed]
17. Abdi, H.; Williams, L.J. Principal component analysis. *Wiley Interdiscip. Rev. Comput. Stat.* **2010**, *2*, 433–459. [CrossRef]
18. Zhai, H.; Zhang, H.; Xu, X.; Zhang, L.; Li, P. Kernel sparse subspace clustering with a spatial max pooling operation for hyperspectral remote sensing data interpretation. *Remote Sens.* **2017**, *9*, 335. [CrossRef]
19. Cortes, C.; Vapnik, V. Support-vector networks. *Mach. Learn.* **1995**, *20*, 273–297. [CrossRef]
20. Weston, J.; Watkins, C. *Multiclass Support Vector Machines*; Technical Report CSD-TR-98-04; Royal Holloway University of London, Department of Computer Science: Egham, UK, 1998.
21. Sokolova, M.; Japkowicz, N.; Szpakowicz, S. Beyond accuracy, F-score and ROC: A family of discriminant measures for performance evaluation. In Proceedings of the AI 2006: Australasian Joint Conference on Artificial Intelligence, Canberra, Australia, 4–8 December 2006.
22. Hawkins, D.M. The problem of overfitting. *J. Chem. Inf. Comput. Sci.* **2004**, *44*, 1–12. [CrossRef] [PubMed]
23. Kaewunruen, S.; Lei, C. Smartphone Sensing and Identification of Shock Noise and Vibration Induced by Gym Activities. *Acoust. Aust.* **2020**, *48*, 349–361. [CrossRef]
24. Kaewunruen, S.; Huang, J.; Haslam, J. Insights into noise and vibration stemming from the gym's heavy lifting. *Sport Sci. Health* **2021**, 1–10. [CrossRef]
25. Kaewunruen, S.; Shi, Y. Impact Noise and Vibration Sources Induced by Heavy Gym Activities: Do They in Turn Unnecessarily, Indirectly Affect Our Health? *Appl. Sci.* **2021**, *11*, 11812. [CrossRef]
26. Kaewunruen, S.; Sresakoolchai, J.; Huang, J.; Harada, S.; Wisetjindawat, W. Human Activity Vibrations. *Data* **2021**, *6*, 104. [CrossRef]