Performance of Outlier Detection on Smartwatch Data in Single and Multiple Person Environments

An analysis of the performance of different outlier detection methods on consumer-grade wearable data in environments with single and multiple subjects

Luuk Wubben[1]

Supervisor(s): David Tax[1]**, Arman Naseri Jahfari**[1]**, Ramin Ghorbani**[1]

[1]EEMCS, Delft University of Technology, The Netherlands

A Thesis Submitted to EEMCS Faculty Delft University of Technology,
In Partial Fulfilment of the Requirements
For the Bachelor of Computer Science and Engineering
June 24, 2023

Name of the student: Luuk Wubben
Final project course: CSE3000 Research Project
Thesis committee: David Tax, Arman Naseri Jahfari, Ramin Ghorbani, Guohao Lan

An electronic version of this thesis is available at http://repository.tudelft.nl/.

Abstract

Outlier detection is an essential part of modern systems. It is used to detect anomalies in behaviour or performance of systems or subjects, such as fall detection in smartwatches or voltage irregularity detection in batteries. This provides early indications of something of potential problems.

A part of outlier detection that is not often analysed is the performance of algorithms in environments with data from only one subject, versus environments with data from multiple subjects. This paper aims to answer the questions regarding the performance of Gaussian Mixture Models (GMM) and DBSCAN in these different environments. This paper focuses on time series data collected from consumer-grade wearables like smartwatches. In this paper, the outliers are defined manually, as the used data set did not contain predefined outliers. This research considers both outliers defined within the subject data, and the use of other subjects as outliers.

Results from this paper indicate that the amount of subjects in the environment is not the sole factor in the performance of these algorithms. Rather, it is a combination of the amount of subjects in the environment and the type of outlier to be detected. Results show that a GMM has difficulty distinguishing subjects that are similar when using another subject as outlier data. On average, DBSCAN outperforms a GMM in almost all cases, and DBSCAN is a lot more consistent in its performance than a GMM.

1 Introduction

Outlier and anomaly detection is a topic within machine learning with many uses. Examples of these use cases include finding unusable data in preprocessing [1], or detecting outliers in the voltage of a battery as an early indicator of a defect [2]. In the scenario of consumer-grade wearable data, outlier detection can be used to indicate heart problems at early stages [3], or a noticeable change in a pilot's habits, which might hint at a lesser degree of preparedness to perform their job properly [4].

A question that is not often answered in the study of outlier detection is whether the models work better in an environment with data from a single subject, or an environment with data from multiple subjects. An answer to this question could prove useful for developers aiming to implement outlier detection for their smartwatch data. These environments will from hereon out be referred to as 'single person environment' and 'multiple person environment' respectively. The set of subjects in a multiple person environment will be referred to as a 'group'. This paper aims to answer the question *"Do outlier detection methods perform better in a single person environment, compared to in a multiple person environment"*.

The choice of a single or multiple person environment is an influential one, as the optimized parameters for one are unlikely to work optimally on the other. Furthermore, subject data might be difficult to separate or combine in certain data sets, for example in unlabelled data of multiple subjects or in separate data sets of very similar subjects. This paper aims to provide a clearer, academically backed, reason to choose one environment over the other, by providing evidence of the performance of different algorithms in these environments.

To analyse this question, two models have been chosen for comparison: Gaussian Mixture Models (GMM) [5] and Density Based Spatial Clustering of Applications with Noise (DBSCAN) [6]. These will be tested on time series data measured from consumer-grade wearables, and this data consists of heart rate and step count measurements. Based on the main research question, the data set, and these models, four valuable sub-questions to the main research question have been identified:

- *"When performing outlier detection in a single person environment, does a Gaussian Mixture Model perform better than DBSCAN?"*
- *"When performing outlier detection in a multiple person environment, does a Gaussian Mixture Model perform better than DBSCAN?"*
- *"Does exclusion of heart rate data affect performance of a Gaussian Mixture Model when compared to DBSCAN?"*
- *"Does exclusion of step count data affect performance of a Gaussian Mixture Model when compared to DBSCAN?"*

The chosen algorithms are popular in the industry for this task and are thus more easily comparable to current research and systems. This paper aims to answer the above questions for these algorithms. This will be done by testing the performance of these algorithms in the two environments and comparing based on factors such as accuracy and popular evaluation metrics like Area Under the Curve (AUC) and Silhouette Coefficient scores[7]. This will create an overview of which algorithm performs best in the different environments, when using a similar data set to the one used in this paper.

The outliers used for this research were manually defined based on the available data. This was done, as the data set used for this study did not include annotated outliers. Outliers were determined based on a measure defined for this study, which is described in section 3. Furthermore, outliers were determined from window data summarized with features. This research focuses on outliers on a bigger timescale, and no conclusions will be drawn on the efficacy of these algorithms in finding smaller timescale outliers.

2 Methodology

In this section, the following topics will be covered. The papers that were used to choose the algorithms will be discussed in subsection 2.1. The area of research will be discussed in 2.2. Finally, the analysis methods will be discussed in subsection 2.3.

2.1 Related Works

Current literature on the subject of outlier detection on consumer-grade wearable data was used to select suitable

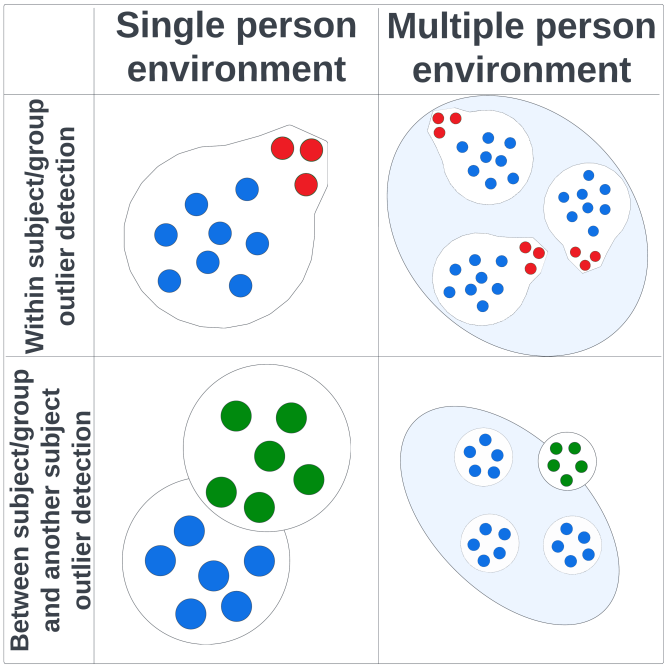

Figure 1: The different outlier definitions for the two environments under test to be considered in this paper. Blue data points are normal data, red data points are outlier points within a subject, while green data points are of a subject treated as an outlier

models for this research. To assure recency of the literature used, only papers published in the last three years were considered. Additionally, search terms relating to outlier detection, heart rate or step count data, and consumer-grade wearables were utilized to quickly narrow down the search. This yielded six key papers.

Sunny et al. provide an overview of the status quo of outlier detection methods [8]. This paper, along with Huang et al. [9] indicate the effectiveness of clustering algorithms in detecting outliers in consumer-grade wearable data. Nanehkaran et al. and Fitriyani et al. confirmed the efficacy of DBSCAN as a choice for outlier detection on such data [10; 11]. DBSCAN is a clustering algorithm that works on the assumption that clusters are regions of dense data points, separated by lower density space. Data points in these lower density spaces are automatically classified as outliers, which makes DBSCAN very suited for outlier detection.

Dwivedi et al. directly applied a Gaussian-based outlier detection method and achieved an improvement in precision over other models [12]. Yang et al. applied a Gaussian-based algorithm as part of a greater outlier detection system and achieved good results in finding and removing outliers [13]. These papers indicate that a Gaussian Mixture Model is suitable for outlier detection on biometric data. A Gaussian Mixture Model is a probabilistic model that assumes the data corresponds to a combination of a finite amount of Gaussian distributions, of which the parameters aren't known. By computing the log-likelihood score of test samples on a fitted model and assigning a threshold value, data points can be marked as normal or as outliers.

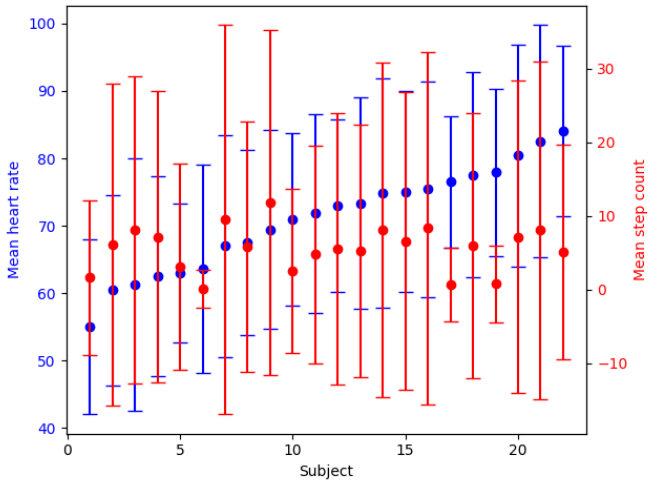

Figure 2: Error plot of the mean heart rate and steps for the 22 chosen subjects under test, with the error being the standard deviation. A lot of variation in their values can be seen

2.2 Area of Research

The goal of this research is to analyse whether different outlier detection methods perform better in a single person environment, compared to a multiple person environment. Figure 3 visualizes the focus area of this research in the process of training and implementing an outlier detection algorithm.

Definition of Outliers

The data set used for this study is from the ME-TIME study, registered at ClinicalTrials.gov with ID NCT05802563 [14]. In this data set, no outliers were annotated in the data. Thus, the choice was made to define outliers based on the available data. The outliers were defined on windowed data, summarized by features. This is further discussed in detail in section 3.

For this study, two definitions of outliers were considered. The first is outlier detection within the subject's data, where outliers are defined in the subject data. In a multiple person environment, the outliers are defined separately per subject, instead of from the group as a whole. The second scenario is outlier detection between subjects. In this scenario, the outlier data is collected from another subject and added to the data set of the subject under test as outlier data. In case of an environment with multiple person environment, another subject not part of the group is selected to collect outlier data from. These two outlier definitions will from hereon out be referred to as 'within subject' outlier detection and 'between subjects' outlier detection, respectively. Variations might be formulated based on the relevant environment, such as 'between group and another subject' outlier detection referring to the 'between' case in a multiple person environment as described in section 1.

Figure 1 provides a visual representation of these definitions in the two environments this paper will consider. The semicircular boundaries show the limits of a subject's data, with the inner coloured dots being data points of that subject. For the multiple person environment, an underlying blue

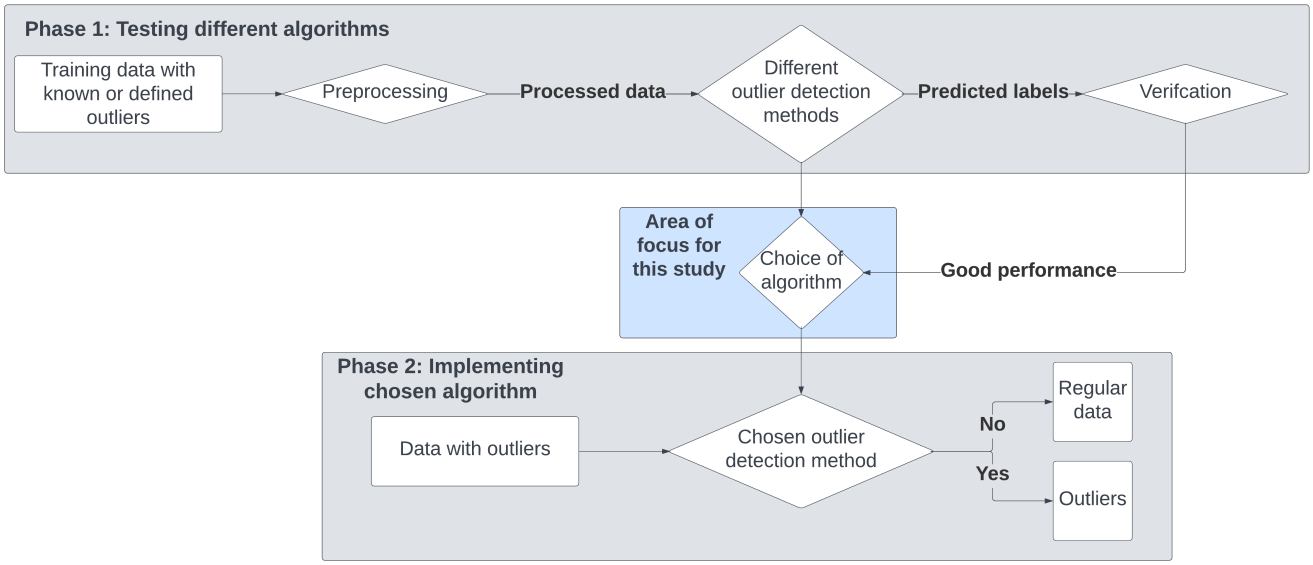

Figure 3: An overview of the process of training and implementing an outlier detection method, with the area of focus for this study highlighted in blue

oval indicates the boundary of the group under test. Red data points indicate data points belonging to the subject that were marked as outliers. Green data points are data points of another subject that are treated as an outlier. The overlap in the subject or group under test and the outlier subject is to indicate a scenario where the outlier detection problem can prove to be more difficult, but they are not required to overlap. This representation is in two dimensions, but subject data can have any number of dimensions.

2.3 Analysis Methods

To answer the research question and sub-questions of this paper, data was gathered from test runs where a few key factors were varied between. All combinations of these factors were used and data was collected from a set of test runs with these different factors. The factors are:

- Single person environment or multiple person environment
- Outlier detection within subject, or between subjects
- Exclusion of heart rate or step count data

The data from these tests will include accuracy data for both GMM and DBSCAN. For GMM, the AUC score will be included, while for DBSCAN the Silhouette Coefficient score will be included. Additionally, in the case of between subjects outlier detection, the distance to the outlier subject will be recorded by means of taking the Euclidean distance between the 4-dimensional coordinates consisting of the subjects' mean heart rate, standard deviation of their heart rate, mean step count, and standard deviation of their step count. This was chosen, as analysis of these values on the selected subjects showed clear differences between subjects, as can be seen in Figure 2.

3 Experimental Setup

This section covers the experimental setup of the preprocessing of the data, the models, and the data collection. The data preprocessing, outlier definitions and result collection will be discussed in subsection 3.1. The models will be covered in subsection 3.2.

3.1 Data Processing

The data set used in this study [14] consists of data on 54 subjects. These subjects used a smartwatch during their participation in the study, which recording a time series of their heart rate and step count. Heart rate data was recorded every 5 seconds, while step count was recorded every minute. The amount of data and the amount of gaps in the data differed per subject.

It was decided to take the 22 subjects which had at least 100 windows of six hours with no stride. This would limit the subjects under test to subjects with enough data to work with, and thus provide more consistent results. The data of these subjects was first normalized[1] in its entirety for both heart rate and steps. After this, windows were created of six hours, and the normalized heart rate and step count data within these windows were separately summarized using a set of features. The features used, along with their parameters, can be found in Table 1.

After all windows were processed, if the within subject outlier definition was used, the outliers were defined using the values from the following formula:

$$outlier\_indicator = \sqrt{\sum_{f \in F} f^2}$$

where:

[1] Standard score normalization was used

Table 1: Features and outliers setup & parameters

Window size	6 hours
Window stride	None
Minimum amount of windows	100
Features	mean, standard deviation, minimum, 25th quantile, 50th quantile, 75th quantile, maximum, nth step, MFCC
nth step parameters	$moment = 6$, $nan\_policy = omit$
MFCC parameters	$sr = \|window\|/window\_time$, $n\_mfcc = 1$, $n\_mels = 1$
Amount of subjects	*single person environment:* 1 & *multiple person environment:* $[2, 6]$
Amount of outlier windows (between subjects outliers)	$0.1 \cdot \|subject\_windows\|$ per subject in environment
Percentage of windows marked as outliers (within subject outliers)	10% per subject in environment

f = a feature's value in this window's feature space
F = this window's feature space

The windows with the 10% highest values for $outlier\_indicator$ were marked as outliers.

If the between subjects outlier definition was used, $0.1 \cdot |subject\_windows|$ random windows from one other subject were randomly selected and added as outliers, where $subject\_windows$ are the windows of the subject under test. In case of a multiple person environment, this was done separately for every subject in the environment using the same outlier subject, after which all regular data and outliers were combined.

The $0.1 \cdot |subject\_windows|$ additional windows from other subjects used in between subjects outlier detection was used to prevent overfitting in the within subject outlier detection scenarios and thus marked as normal data.

3.2 Outlier Detection Models

The Gaussian Mixture Model was implemented using `scikit-learn`'s `GaussianMixture` class and its related methods (for a full overview of the manually installed external libraries, see Appendix A). Additionally, a few functions were implemented to facilitate the creation of training sets without outliers, finding the optimal threshold for the log-likelihood scores, and testing the accuracy of the model.

The windowed data was split into a train, validation, and test set consisting of 70%, 15%, and 15% of the total windows respectively. The train set was normalized[2] and its mean and standard deviation were used to normalize the validation and test set. The model was trained with no outliers to provide a better fit for the normal data. The validation set was used to find the optimal parameters for the model and the threshold for the log-likelihood scores as calculated by the `score_samples` method. Using a grid search to find the maximum of the negated Akaike Information Criterion (AIC) [15], the best covariance type and number of components were selected. To do this, `scikit-learn`'s `GridSearchCV` function was used, which finds the maximum score for a given model, set of possible parameters, and scoring function. AIC scores are negative and lower scores are better, therefore the scores were negated to provide the function with positive scores.

The optimal threshold for the log-likelihood scores was found using an ROC curve, where the threshold with the largest absolute difference between the false positive rate and true positive rate was selected as the best candidate. AUC scores were also computed from the ROC curve. Finally, the test set, along with the best found threshold, would be used to test the model's accuracy by scoring the test set samples. Using the found threshold to mark windows as outliers or normal windows and calculating the accuracy by comparing it to the actual labels that were determined during preprocessing.

The DBSCAN algorithm was implemented using `scikit-learn`'s `DBSCAN` class. Since DBSCAN is unsupervised, no train test split was made in the windowed data. Two important parameters, epsilon and minPts, were determined based on the conclusions from two papers. Sander et al. stated the best minPts value is $2 * |dimensions|$ [16]. Thus, the value for minPts was set to twice the amount of features used per window. Rahmah et al. concluded that the best value for epsilon could be found by finding the y-coordinate of the point of maximum curvature in the K-distance graph from applying K-means to the data set with $K = minPts$ [17]. The `kneebow` library was used to automatically find this point and retrieve its y-coordinate. These parameters were then used to fit and predict the cluster labels of the windowed data.

DBSCAN assigns the label -1 to outliers. The accuracy was calculated by comparing the actual labels to the DBSCAN predicted labels. Finally, the Silhouette Coefficient score was calculated using `scikit-learn`'s `silhouette_score` function. The Silhouette score shows whether clusters are dense and well-defined, or sparse and overlapping. Scores range from -1 to 1, where positive scores indicate good clusters and negative scores indicate bad clusters.

4 Results

This section covers the results of running a series of tests on the implemented models. The test results are showcased and analysed in subsection 4.1. These tests consisted of running a set of test runs, as described below in Table 2. The table shows the total runs per algorithm for each outlier definition, as well as which factors were varied in testing. For single person environments, all possible combinations of subjects under test and outlier subjects were tested. For multiple person environments, the choice was made to limit the group sizes and membership combinations, as running all membership

[2] Standard score normalization was used

combinations and possible sizes against all their possible outliers would have taken too long in the time available for this research. All tested groups were tested against all possible outlier subjects. For every test iteration, the best parameters were recalculated for each model to get the best possible performance. This was done to provide a fair comparison of test iterations.

Table 2: Test runs specifications

	Single Person Environment	**Multiple Person Environment**
Total runs per algorithm, within subject outliers	462	270
Total runs per algorithm, between subjects outliers	462	270
Subjects/Groups tested	All 22 subjects	3 random groups for every group size
Outlier subjects tested against	All 21 other subjects	All $22 - \|group\|$ other subjects

4.1 Analysis of Results

The means and standard deviations of the chosen performance scores, per the four scenarios as showcased in Figure 1, are shown in Table 3 and Table 4. Table 3 shows the accuracy and AUC score for GMM. Table 4 shows the accuracy and Silhouette score for DBSCAN. All values have been rounded to two decimals.

Table 3: Mean and standard deviation (std) of accuracy and AUC score for GMM per scenario

	Single Person Environment	**Multiple Person Environment**
Within Subject	Accuracy mean: 0.84 Accuracy std: 0.08 AUC mean: 0.92 AUC std: 0.07	Accuracy mean: 0.87 Accuracy std: 0.04 AUC mean: 0.95 AUC std: 0.03
Between Subjects	Accuracy mean: 0.94 Accuracy std: 0.13 AUC mean: 0.96 AUC std: 0.11	Accuracy mean: 0.77 Accuracy std: 0.20 AUC mean: 0.80 AUC std: 0.19

This performance indicates that, while a Gaussian Mixture Model can outperform DBSCAN in outlier detection in some cases, the performance is not consistent across the test runs. In terms of accuracy, DBSCAN has lower standard deviations in every scenario, and is thus more consistent. The Silhouette scores are also consistently higher than 0.5, indicating that the clusters are dense and well-defined. The high AUC score for GMM gives credibility to its accuracy. The low standard deviation for both accuracy and AUC scores in the within subject scenarios, shows it performs well in those consistently. However, for between subjects, it appears to have difficulty, as seen by the standard deviation in its accuracy and AUC scores in those test cases.

Table 4: Mean and standard deviation (std) of accuracy and Silhouette Coefficient (SC) score for DBSCAN per scenario

	Single Person Environment	**Multiple Person Environment**
Within Subject	Accuracy mean: 0.91 Accuracy std: 0.05 SC mean: 0.77 SC std: 0.13	Accuracy mean: 0.92 Accuracy std: 0.01 SC mean: 0.89 SC std: 0.11
Between Subjects	Accuracy mean: 0.82 Accuracy std: 0.08 SC mean: 0.77 SC std: 0.13	Accuracy mean: 0.88 Accuracy std: 0.04 SC mean: 0.91 SC std: 0.09

When analysing the performance of DBSCAN and GMM in both environments on between subjects outliers, it becomes clear what causes the deviation in accuracy for GMM. In Figures 4 and 5, the accuracy and AUC score of GMM are graphed against the accuracy and Silhouette score of DBSCAN. The x-axis is the distance ranking, which is a ranking of the test measurements based on the distance between subjects. This distance rank was determined for every subject under test and then aggregated per rank. DBSCAN's trend is mostly constant, while GMM's trend is increasing with distance. This indicates that distance impacts a GMM's performance, where closer subjects are harder for it to distinguish than further subjects. DBSCAN's consistency indicates it has no difficulty making these distinctions.

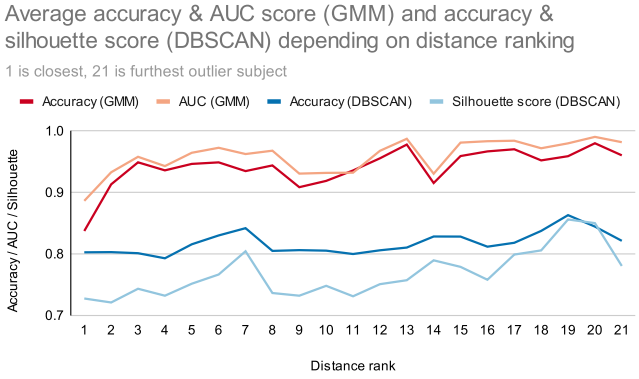

Figure 4: Performance scores of GMM (red) and DBSCAN (blue) for the single person environment with the between subjects outlier definition

The within subject outlier definition has been analysed in Figures 6 and 7, with subject number or group size instead of distance ranking on the x-axis. This analysis indicates that there were subjects that were more difficult for one or both of the models to properly identify outliers for. This could be caused by a lack of data, or inconsistency in the data. DBSCAN had less of these difficult subjects than the GMM, only

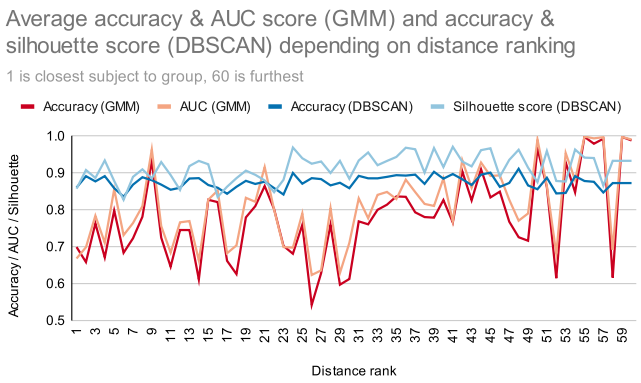

Figure 5: Performance scores of GMM (red) and DBSCAN (blue) for the multiple person environment with the between subjects outlier definition

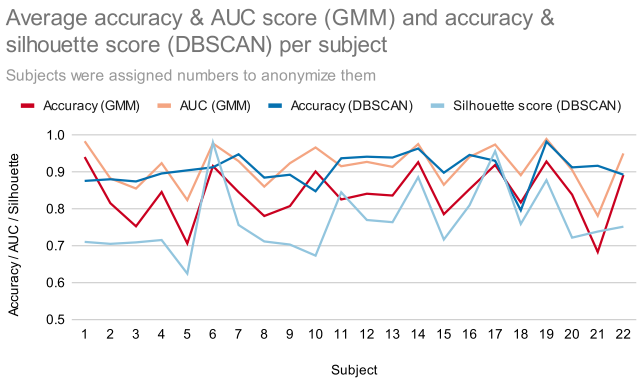

Figure 6: Performance scores of GMM (red) and DBSCAN (blue) for the single person environment with the within subject outlier definition

When analysing the effect of excluding heart rate or step count from the data, two key things can be noted. Performance in terms of accuracy and AUC or Silhouette score is generally affected little. The only exception to this is the GMM in between subjects outlier detection for both environments. In these scenarios, the mean accuracy and AUC score drop to around 0.60.

The standard deviations of accuracy and AUC/Silhouette score are considerably lower in almost every case when excluding step count, often being about half of the original. The standard deviations when using both heart rate and step count

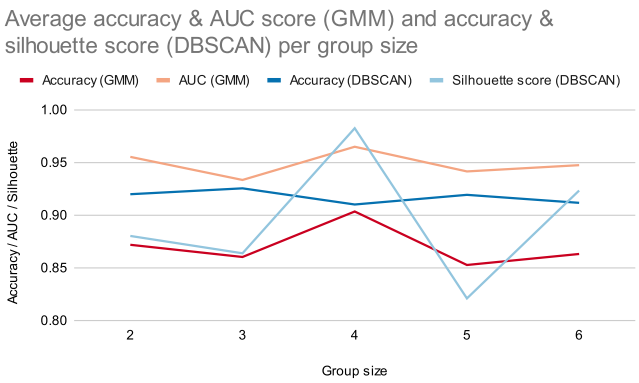

Figure 7: Performance scores of GMM (red) and DBSCAN (blue) for the multiple person environment with the within subject outlier definition

are similar to the standard deviations when only using step count. This indicates that step count is most likely responsible for most of the standard deviation seen in the results discussed earlier. Standard deviations are higher for the GMM in between subjects outliers when only using heart rate, which is likely caused by the inaccuracy of the GMM on similar subjects. For readability purposes, the performance scores have not been listed here, but have been added to Appendix B.

5 Responsible Research

This section is intended to cover the ethical aspects and implications of this research, as well as the reproducibility of the methods used. The ethical implications will be covered in subsection 5.1. The reproducibility of this research will be covered in subsection 5.2.

5.1 Ethical Implications

Any research involving human data has various ethical implications that should be addressed. These include, the privacy of subjects, the possibility of direct or indirect (re)identification, the potential of using the data or its results against individuals or groups, for example through discrimination. This research is no different in that regard and thus, this section aims to address these concerns by detailing what has been done to mitigate risks.

(Re)identification of subjects

With the use of data from subjects, the possibility of (re)identification exists. Heart rate and step count data has been proven to be very effective for the task of reidentifying individuals, with some papers claiming up to 99.7% accuracy on the reidentification of subjects [18].

To mitigate this possibility, data should be anonymized. This helps to prevent reidentification of individuals, as no direct identification remains. Subject data might still be separated, but without an identifying label, linking these unnamed subjects to individuals will prove difficult.

The data used in this research was anonymized before being provided to the researchers of this paper. Furthermore,

instead of using the study IDs of the subjects, the 22 subjects used in the experiments of this paper were instead assigned a number between 1 and 22 inclusive at random. Therefore, reidentifying the individuals without prior knowledge of their daily activity, fitness, or potential heart conditions is made more difficult. The (processed) data used in this paper will also not be published with the paper. This further limits the possibility of (re)identification. However, if the original study the data originates from releases the data, (re)identification might become possible, since the code used for this paper will be published.

Privacy

Privacy is a large issue when using data from individuals. Their explicit consent should be received before their data is used in any (machine learning) algorithm or data analysis. This can be accomplished through a privacy policy, Terms of Service (TOS), or a consent form. To mitigate the possibility of using data from individuals who declined the use of their data, or were not informed at all, only data from reputable sources should be used. These sources should be able to prove the individuals their data originates from have consented to its use, including the way the receiving party intends to use or manipulate the data.

The data used for this research has been collected during a clinical trial [14]. The individuals the data has been collected from have given their explicit consent for their data to be used for research purposes, including extended research like this paper. All researchers who work with this data must first sign a Non-Disclosure Agreement (NDA), which specifies the legal limitations of using, sharing and storing data.

Discriminatory Use of Data or Results

The potential exists for the data or its results to be used in discriminatory ways, if some aspects of the individual are known. For example, discriminatory wrong conclusions could be drawn from patterns in the data or results if certain information about the subject is known, like race, age, gender, occupation, etc.

To mitigate such a risk, the data should be anonymized as much as possible, to prevent false conclusions from being drawn. Any metadata on the subject that is not relevant to the results of the research or its official conclusions should be left out of the data set. Metadata that does not exist in the data set cannot be used for nefarious purposes, and only direct knowledge of which subjects participated could circumvent that.

For this research, the data set will not be published with this paper. The data set did include some metadata about the subjects. However, this metadata was heavily scrubbed of anything not relevant to any potential research and was not used in this research at any point. Therefore, discrimination against persons or social groups of any kind based on the results from this research isn't possible. If the data is published at any point, there is the potential of someone using the published code with this paper to link results to subjects. In this scenario, the scrubbing and limited collection of the metadata will mitigate discriminatory use.

5.2 Reproducibility of Research

This research has the aim to be reproducible. The implementations used for this research will be publicly available[3] and are explained in a concise and clear manner in this paper. Any terms that might not be familiar to a reader have been clarified and explained and any parameters, settings, assumptions, or specific values have been covered in concise detail.

A limitation of the reproducibility of this research is the fact that the data set, used and provided by the ME-TIME study [14], is not publicly available. To work with this data set, the researchers involved with this paper had to sign an NDA. Therefore, the data set cannot be published with the paper. This might make the results of this paper not reproducible, if the methods used are very specific to this data set.

6 Discussion

In this section, the research methods and results will be discussed. This will be done in two parts. The first part will cover the limitations of the research in subsection 6.1. The second part will discuss the results from this research in subsection 6.2.

6.1 Limitations of the Research

One big limitation of this research is the choice to work with features from windows. This approach makes data more robust and potentially easier to analyse for larger anomalies that manifest over longer periods of time. However, anomalies that happen in a fraction of the window's timespan go undetected.

Only two algorithms were compared in this study, which limits the general applicability of the results. Patterns and performances for these algorithms do not necessarily reflect onto the bigger picture. Therefore, conclusions on a larger scale cannot be made with confidence. However, this paper can provide motivation to perform further research on a larger scale into the performance of outlier detection methods in different environments.

Furthermore, the automated parameter optimization might not always find the best parameters when compared to other methods, such as handcrafted parameters. A middle ground had to be found between speed, performance, and quality of the parameters. This can create a false picture of the performance of an algorithm, as it might perform differently with other, more common or robust parameter optimization methods.

6.2 Discussion of Results

The conclusions drawn from the results in section 4 paint a clear picture of consistently good results for DBSCAN, regardless of the environment or outlier definition. The GMM implemented for this research shows good performance for within subject outlier detection, but has trouble with closer

[3]The repository can be found here (hyphens are all part of URL): https://github.com/ATicklishTomato/Performance-of-Outlier-Detection-on-Smartwatch-Data-in-Single-and-Multiple-Person-Environments

outlier subjects in the between subjects outlier definition scenarios, showing considerable standard deviations on both accuracy and AUC scores. Finally, exclusion of step count data appears to negatively impact the GMM's performance in between subjects outliers and be responsible for most of the deviations in performance for all scenarios. These differences in performance indicate that further research could be warranted, to better map out where the strengths and weaknesses lie of different outlier detection algorithms on consumer-grade wearable data.

While these results are promising, the limitations of the results are important to address. Foremost, the limitation of subjects, especially in the multiple person environment, plays an important factor in the limitations of the results. Only some possible groups were tested, as there wasn't enough time during the research period of this paper to try them all for both algorithms. The limiting group size also limits results to only small group sizes, meaning no conclusions can be drawn for larger groups. The amount of subjects could be increased in further research, as well as the amount of group sizes and membership combinations tried. This would provide a clearer picture of the general performance and also provides the possibility to focus on certain subjects, for example those who are similar or dissimilar, or those with certain medical conditions.

7 Conclusions and Future Work

This section will cover the conclusions of this work, as well as suggest future work that should be performed to improve understanding and results. The conclusions will be covered in subsection 7.1. The future work will be covered in subsection 7.2.

7.1 Conclusions

To conclude, this paper shows that DBSCAN shows consistent good performance in both environments and with both outlier definitions. Its accuracy and Silhouette score are high and both have low standard deviations. The GMM outperforms DBSCAN in a single person environment with the between subjects outlier definition, but performs worse in all other scenarios. The GMM only beats DBSCAN a few times at high distance in the between subjects and multiple person environment scenario. A GMM is consistently accurate and gets high AUC scores in both environments for within subject outlier detection. In the between subjects case, the distance between the subject under test and the outlier subject appears to be of great importance to the overall performance of the GMM, with a clear upwards trend being seen when comparing distance to accuracy and AUC scores. DBSCAN does not show this difficulty and performs well consistently regardless of environment or outlier definition. Furthermore, step count data appears to cause most of the deviation in performance for both models. Exclusion of either heart rate or step count data has little effect, except in between subjects outliers for GMM, where accuracy drops drastically.

To answer the main research question of this paper, some outlier detection methods do appear to perform better in a single person environment than in a multiple person environment. This is not purely due to environment, however, as the outlier definition also plays a role. DBSCAN has similar accuracies in both environments for both of the outlier definitions. The GMM has similar accuracies for both environments when using within subject outliers, but average accuracy drops quite significantly when using between subjects outliers, from 94% in a single person environment (standard deviation of 13%) to 77% in a multiple person environment (standard deviation of 20%).

7.2 Future Work

A major limitation of this work is the small amount of algorithms that were compared. For future research, more algorithms should be considered in the comparison. This can improve provide answers whether results from this paper are specific to the models tested, or if the trend continues on a larger scale. For this paper, Bayesian Outlier Detection [19] was considered, but was not included due to time constraints. However, it showed promise in the single person environment during small scale testing and would be worth a consideration for future research on this topic. When its prior mean and standard deviation can be estimated or are known, this algorithm could perform well in a more detailed study.

Future work should also analyse the performance of algorithms when trying to find small scale anomalies that windowed data will not reflect. For example, sudden peaks in someone's heart rate might indicate heart problems, but this would not be reflected well if the frequency of these peaks within a window is low. Thus, it is important to analyse, as windowed data is not applicable to every scenario where outlier detection is needed.

Additionally, future work might want to use well understood data, for which parameters of the algorithms under test can be more efficiently tweaked. If the data is understood well, parameters can more easily be verified manually using knowledge of the data set. Thus, the final results might be more robust and more accurate conclusions may be drawn. Future work should also consider using a public data set to improve reproducibility of the research and its results.

References

[1] Jie Sun, Bo Shen, and Yufei Liu. A resilient outlier-resistant recursive filtering approach to time-delayed spatial-temporal systems with energy harvesting sensors. *ISA TRANSACTIONS*, 127:41–49, AUG 2022.

[2] Zonghai Chen, Ke Xu, Jingwen Wei, and Guangzhong Dong. Voltage fault detection for lithium-ion battery pack using local outlier factor. *MEASUREMENT*, 146:544–556, NOV 2019.

[3] Nan Yue and Stephan Claes. Wearable sensors for smart abnormal heart rate detection during skiing. *Internet Technology Letters*, 4(3):e230, 2021.

[4] Weiwei Yuan, Li Zhou, Donghai Guan, Guangjie Han, and Lei Shu. Anomaly detection for civil aviation pilots using step-sensors. *IEEE ACCESS*, 5:11236–11243, 2017.

[5] Douglas Reynolds. *Gaussian Mixture Models*, pages 659–663. Springer US, Boston, MA, 2009.

[6] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. A density-based algorithm for discovering clusters. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96)*, pages 226–231. Institute for Computer Science, University of Munich, AAAI Press, 1996.

[7] Peter J. Rousseeuw. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53–65, 1987.

[8] Jithin S. Sunny, C. Pawan K. Patro, Khushi Karnani, Sandeep C. Pingle, Feng Lin, Misa Anekoji, Lawrence D. Jones, Santosh Kesari, and Shashaanka Ashili. Anomaly detection framework for wearables data: A perspective review on data concepts, data analysis algorithms and prospects. *SENSORS*, 22(3), FEB 2022.

[9] Geyu Huang, Zhiming Zhang, and Wenxin Yang. Outlier detection method based on improved two-step clustering algorithm and synthetic hypothesis testing. In B Xu, editor, *PROCEEDINGS OF 2019 IEEE 8TH JOINT INTERNATIONAL INFORMATION TECHNOLOGY AND ARTIFICIAL INTELLIGENCE CONFERENCE (ITAIC 2019)*, pages 915–919. IEEE; IEEE Beijing Sect; Chongqing Global Union Acad Sci & Technol; Chongqing Univ Technol; Chengdu Global Union Acad Sci & Technol; Chongqing Geeks Educ Technol Co Ltd, 2019. IEEE 8th Joint International Information Technology and Artificial Intelligence Conference (ITAIC), Chongqing, PEOPLES R CHINA, MAY 24-26, 2019.

[10] Y. A. Nanehkaran, Zhu Licai, Junde Chen, Ahmed A. M. Jamel, Zhao Shengnan, Yahya Dorostkar Navaei, and Mohsen Abdollahzadeh Aghbolagh. Anomaly detection in heart disease using a density-based unsupervised approach. *WIRELESS COMMUNICATIONS & MOBILE COMPUTING*, 2022, MAR 26 2022.

[11] Norma Latif Fitriyani, Muhammad Syafrudin, Ganjar Alfian, and Jongtae Rhee. Hdpm: An effective heart disease prediction model for a clinical decision support system. *IEEE ACCESS*, 8:133034–133050, 2020.

[12] Rajendra Kumar Dwivedi, Rakesh Kumar, and Rajkumar Buyya. Gaussian distribution-based machine learning scheme for anomaly detection in healthcare sensor cloud. *International Journal of Cloud Applications and Computing*, 11(1):52–72, January 2021.

[13] Yuanjing Yang, Lianying Ji, and Jiankang Wu. Outlier detection in heart rate signal using activity information. In D Cheng, editor, *PROCEEDINGS OF THE 10TH WORLD CONGRESS ON INTELLIGENT CONTROL AND AUTOMATION (WCICA 2012)*, pages 4511–4516. Chinese Acad Sci, Acad Math & Syst Sci; IEEE Robot & Automat Soc; IEEE Control Syst Soc; Natl Nat Sci Fdn China; Chinese Assoc Automat; Chinese Assoc Artificial Intelligence; IEEE RACS Hong Kong Chapter; IEEE Control Syst Soc Beijing Chapter; IEEE Control Syst Soc Singapore Chapter, 2012. 10th World Congress on Intelligent Control and Automation (WCICA), Beijing, PEOPLES R CHINA, JUL 06-08, 2012.

[14] ClinicalTrials.gov. Machine learning enabled time series analysis in medicine (me-time). https://clinicaltrials.gov/ct2/show/NCT05802563, 2023.

[15] P. Stoica and Y. Selen. Model-order selection: a review of information criterion rules. *IEEE Signal Processing Magazine*, 21(4):36–47, 2004.

[16] Jörg Sander, Martin Ester, Hans-Peter Kriegel, and Xiaowei Xu. Density-based clustering in spatial databases: The algorithm gdbscan and its applications. *Data Mining and Knowledge Discovery*, 2(2):169–194, 06 1998.

[17] Nadia Rahmah and Imas Sukaesih Sitanggang. Determination of optimal epsilon (eps) value on DBSCAN algorithm to clustering data on peatland hotspots in sumatra. *IOP Conference Series: Earth and Environmental Science*, 31:012012, 01 2016.

[18] Arin Ghazarian, Jianwei Zheng, Daniele Struppa, and Cyril Rakovski. Assessing the reidentification risks posed by deep learning algorithms applied to ecg data. *IEEE ACCESS*, 10:68711–68723, 2022.

[19] Kathryn Chaloner and Rollin Brant. A bayesian approach to outlier detection and residual analysis. *Biometrika*, 75(4):651–659, DEC 1988.

A Python Libraries

Table 5: Libraries used, their official websites, and their purpose during development and testing

Name	**Website**	**Use**
NumPy	https://numpy.org/	Mathematical and scientific calculations, matrixes
pandas	https://pandas.pydata.org/	Dataset manipulation and visualisation
scikit-learn	https://scikit-learn.org/stable/	GMM and DBSCAN implementations, other data science tools (e.g. ROC-curves, AUC scores)
matplotlib	https://matplotlib.org/	Visualization of graphs
scipy	https://scipy.org/	Calculation of nth order moment feature
librosa	https://librosa.org/	Calculation of MFCC feature
tqdm	https://tqdm.github.io/	Easy tooling for progress bars
kneebow	https://pypi.org/project/kneebow/	Used to find elbow points in graphs

B Performance Scores when Excluding Data

Table 6: Mean and standard deviation (std) of accuracy and AUC score for GMM per scenario, when excluding heart rate or step count data

	Single Person Environment	**Multiple Person Environment**
Within Subject	*Heart rate excluded:* Accuracy mean: 0.87 Accuracy std: 0.06 AUC mean: 0.96 AUC std: 0.04 *Step count excluded:* Accuracy mean: 0.87 Accuracy std: 0.07 AUC mean: 0.92 AUC std: 0.07	*Heart Rate excluded:* Accuracy mean: 0.90 Accuracy std: 0.04 AUC mean: 0.96 AUC std: 0.03 *Step Count excluded:* Accuracy mean: 0.95 Accuracy std: 0.02 AUC mean: 0.99 AUC std: 0.01
Between Subjects	*Heart rate excluded:* Accuracy mean: 0.94 Accuracy std: 0.13 AUC mean: 0.96 AUC std: 0.11 *Step count excluded:* Accuracy mean: 0.62 Accuracy std: 0.11 AUC mean: 0.67 AUC std: 0.11	*Heart Rate excluded:* Accuracy mean: 0.81 Accuracy std: 0.23 AUC mean: 0.93 AUC std: 0.21 *Step Count excluded:* Accuracy mean: 0.55 Accuracy std: 0.10 AUC mean: 0.57 AUC std: 0.09

Table 7: Mean and standard deviation (std) of accuracy and Silhouette Coefficient (SC) score for DBSCAN per scenario, when excluding heart rate or step count data

	Single Person Environment	**Multiple Person Environment**
Within Subject	*Heart rate excluded:* Accuracy mean: 0.92 Accuracy std: 0.06 SC mean: 0.80 SC std: 0.14 *Step count excluded:* Accuracy mean: 0.96 Accuracy std: 0.02 SC mean: 0.88 SC std: 0.04	*Heart Rate excluded:* Accuracy mean: 0.92 Accuracy std: 0.01 SC mean: 0.89 SC std: 0.09 *Step Count excluded:* Accuracy mean: 0.94 Accuracy std: 0.02 SC mean: 0.92 SC std: 0.03
Between Subjects	*Heart rate excluded:* Accuracy mean: 0.83 Accuracy std: 0.08 SC mean: 0.80 SC std: 0.14 *Step count excluded:* Accuracy mean: 0.85 Accuracy std: 0.03 SC mean: 0.88 SC std: 0.04	*Heart Rate excluded:* Accuracy mean: 0.87 Accuracy std: 0.04 SC mean: 0.91 SC std: 0.08 *Step Count excluded:* Accuracy mean: 0.88 Accuracy std: 0.02 SC mean: 0.93 SC std: 0.03