Department of Computer Science
Winter term 2021/2022

Automated classification of activity groups based on empirical motion data

Research Project
Data Science Master

Advisors:

Prof. Dr. Ingo J. Timm
Christian Lohr

submitted by:

Anastasia Firsova
Student number: 1548050

Jibin Sebastian
Student number: 1570020

March 20, 2022

Contents

List of Figures

1 Introduction

The given research paper is dedicated to the attempts of automated classification of activity groups based on a small set of empirical motion data collected in the "SiNuS-Pflege" project. The underlying data set complications include very specific nature of data such as a high number of attributes while keeping the number of observations relatively small as well as the necessity for indirect extraction of information from various sensor devices including smartphone, fitness bracelet (Fitbit), and accelerometer. While dealing with an unconventional data set, feature extraction from sensors records and overall transformation are executed in separate steps. In pursuit of application of different dimensionality reduction techniques, several clustering algorithms must be consequently applied and the results are to be tracked down to reveal the possibility of presence of sustainable clusters among participants of the experiment. Parallelly, a comparison between pure algorithmic approach and application of heuristics takes place in order to reveal effects of the usage of subsets obtained from two approaches. During the performance of our case studies, we will consider several critical issues of data quality and their impact on the final results.

2 Literature Review

The specialty of the given data set dictated the search of relevant literature and methods. The size of our dataset was defined by 42 objects with 149 features including those extracted from sensors records. In this case, the potential impact of outliers is significant, and overfitting is to be expected while performing clustering algorithms. The article by A. Bradford, T. Yellamraju, and M. Boutin from Purdue University was dedicated to a similar problem (Bradford A., Yellamraju T., Boutin M. (2020)). As it was pointed out by the authors, it is typically not the case that small data sets in high-dimension form clusters in their original space so that application of dimensionality reduction techniques is required. After application of n-TARP algorithm, they proposed the usage of k-Means, BIRCH and DBSCAN. However, in order to apply the proposed techniques, the authors recommend to have at least 100-200 points. An additional problem consists of impossibility of checking the clustering results by drawing new samples or performing cross-validation in our situation. The majority of authors see small data sets as ones with about 100-300 observations and at most half this number of features. So are the historical research cases described in the article dedicated to the application of clustering methods to Alzheimer‘s disease by H. Alashwal et al (Alashwal H., El Halaby M., Crouse J., Abdalla A., Moustafa A. (2019)). Among the methods methodologically best suitable for small data sets, they name hierarchical clustering and DBSCAN, pointing out that extreme dimensionality reduction is still a prerequisite for both better understandability of results and their visualisation. Another research by P. Alvarez Gonsalez and F. Forsberg supports the idea of harsh dimensionality reduction as a preparation step for a more powerful tool like k-Means algorithm (Forsberg F., Alvarez Gonzalez P. (2018)). While comparing the performance of DBSCAN and k-means algorithms on small data sets ("less than 500 observations"), they admit results to be mixed. According to their findings, k-Means is better in broader generalizations, while DBSCAN detects anomalies better and

succeeds in isolating them from other patterns discovered.

3 Methods and Data Preparation

The literature dedicated to clustering algorithms on small data usually does not cover issues of dimensionality reduction, so there is no information on compatibility of the tools. Keeping this in mind, rather classical or more general methods were chosen. Data preparation was conducted in several steps.
First of all, selection of additional information from the sensors was fulfilled. Correct time periods of six minutes and one minute had to be located inside the records from each sensor. Having taken advantage of all the data to be synchronized, the time intervals only had to be located once. Since the timestamps were only present in smartphone records, this procedure was firstly performed here approximating one row to one second, which allowed us to extrapolate time feature for other devices‘ records. Smartphone records allowed to define minimum, maximum as well as deviation of acceleration for both tasks; fitbit data contained tracks of heart frequency and were summarized in form of minimum, maximum, deviation and respective amounts of confidence (which was present as ordinal variable with range from 0 to 3) for both . Accelerometer measurements were represented as three-dimensional points where rate of velocity was recorded. The strong fluctuations of values during each task which can be explained by the feeling of weariness as well as extemporaneousness of participants‘ movements led to the solution of splitting time of performance into two segments for one minute task and three for six minutes task respectively. For each of these splits, minimum, maximum as well as deviation were determined. Since it is impossible to combine rates in velocity in different dimensions, they were designed as separate variables. More detailed description of feature extraction is documented in the R script ”full_automated_r_skript.R”.
The main file included basic demographic information in anonymised form (”ageGroup”, ”gender”, ”bmi”, ”smoker”, ”employed”, ”sports”), general information about tasks performance (”timeOfDay”, ”arrival”, ”testSuit”, ”firstTask”, ”dist_6mwt”, ”dist_sct”) and variables from the questionnaires: swe (self-efficacy expectation), bsa (movement and sports activity), abi_p (anxiety management inventory), wkv (perceived physical condition), mood (perceived psychological condition), borg (sense of exertion), panas (positive and negative affect schedule), hee and facts (unknown to us). The majority of variables was coded as ordinal variables.
After feature extraction was completed, the data was combined, treated for missing values and normalised using MinMaxScaler from Python‘s sklearn library. Correlation check allowed to reveal that there were groups of variables with high correlation inside: these variables stem from the same questionnaires like swe, wkv, panas. Features extracted from sensors also show high correlation since they encode information about the same events. More unexpected was the lack of high correlation between demographic and movement variables. We potentially explained this circumstance as the lack of heterogeneity among participants (all of them are young and healthy people with similar socio-economic background) as well as a consequence of excessive data anonymization (the differences are now mitigated, the nuances are lost). Some features of questionnaires are named similarly which may imply that some responses contain

redundant information because participants do not anticipate the differences. In order to mitigate correlation related problems, we dropped 26 features exceeding the threshold of 0.9 and proceeded to application of dimensionality reduction algorithms: principal component analysis (PCA), isometric feature mapping (Isomap), and uniform manifold approximation and projection for dimension reduction (UMAP) done in parallel.
PCA is one of the most frequently used methods for dimensionality reduction by projecting data onto its orthogonal feature subspace. As a prerequisite, the data features must be normalized, and, for the better performance, be linearly connected. The application of PCA on our data set allowed us to see that features are connected in a non-linear fashion, and the percentage of explained variance proved to be relatively small for the first three components (always below 50 %) and then evenly low spread among other components. However, PCA enabled us to analyse the impact of single features inside of components which is otherwise impossible when performing dimensionality reduction techniques.
Isomap based on geodesic distances works better at discovering the true low dimensional geometry of manifolds (Tenenbaum J.B.,de Silva V., Langford J.C. (2000)). It can be seen as an extension of MDS method. Although Isomap preserves geometry better, it is not robust to the noise because of the selection method of the neighbouring points and shows unsatisfactory results in case the given manifold was sampled poorly. It also does not allow to analyse the impact of features constituing the resulting components.
The final dimensionality reduction technique employed for the given research is UMAP. This method is founded on three assumptions about the underlying dataset: it must be uniformly distributed on Riemannian manifold, Riemannian metric should be locally constant or approximated as such, and the manifold must be locally connected (McInnes L, Healy J (2018)). For our dataset, these assumptions may not be fulfilled completely, but the advantages of UMAP are that, according to the authors of the methods, it provides better or comparable quality of embeddings when reducing to two or three dimensions among its class. However, as in the case of Isomap, UMAP lacks the strong interpretability of PCA, so that we are unable to track down the features‘ impact (McInnes L, Healy J (2018)).

4 Data Analysis

Our research is divided into two parts: in the first part, we attempted to find clusters inside of our data with pure instrumental approach whereas the second part introduces our tries to apply heuristics while constructing subsets and choosing the methods of data preprocessing (sticking to PCA only as the one enabling interpretability by humans). As clustering algorithms, k-Means, DBSCAN, Gaussian mixture, and hierarchical agglomerative clustering were chosen to represent different classes of clustering techniques. All algorithms are contained in Python‘s sklearn library. For the determination of optimal number of clusters several metrics were used: Elbow method, Calinski-Harabasz method, Silhouette score and gap statistics. Since there is no strict opinion on which metrics works better, the final decision on the number of clusters was constitued by simple majority or/and common sense. For DBSCAN, a build-in metrics for epsilon definition and some heuristics were used. Due to the

presence of a randomization component in the mentioned algorithms and the impossibility of fixing it completely in Jupiter notebook, the reproducibility of the obtained results may vary to an extent explicable by this technical circumstance.

4.1 Pure Instrumental Approach

4.1.1 Principal Component Analysis

The components obtained as a result of PCA attributed to a rather small share of explained variance. After the second component, it was below 8% and smeared over the rest. Due to convenience of representation, we picked up two first components which added up to 25.38% of explained variance (see Table 1). The variable "wkv_a_t4" had the strongest impact on the first principal component, whereas the variable "firstTask" influenced most the second principal component.

PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8	PC9	PC10
16.05	9.33	7.63	5.85	5.58	5.12	4.51	4.23	3.73	3.44

Table 1: Explained variance by first 10 principal components, in %

Since it is impossible to track down the impact of separate variables on cluster labels after the application of dimensionality reduction techniques, we proposed the usage of correlation analysis as an attempt to understand if there are any connections between labels and initial data sets. To distinguish between random correlations and potentially interesting correlations, a threshold of 0.5 was chosen. K-Means and Gaussian mixture method both discovered 2 clusters and assigned objects to them identically. Cluster labels resulted from k-Means/GMM algorithms application correlated with the following variables:

swe_b_t4	-0.513395	wkv_t_t2	-0.526923
hee_t1	-0.569155	wkv_t_t3	-0.548890
hee_t4	-0.505723	wkv_t_t4	-0.514137
wkv_a_t1	-0.647946	mood_v_t1	-0.540659
wkv_a_t2	-0.555872	mood_v_t3	-0.571360
wkv_a_t4	-0.549860	mood_v_t4	-0.580936
wkv_b_t1	-0.576640	mood_c_t1	-0.510247
wkv_g_t1	-0.660310	mood_ea_t1	-0.593075
wkv_g_t2	-0.531002	panas_pos_t1	-0.540183
wkv_g_t3	-0.608785	borg_6mwt_pre	0.509776
wkv_g_t4	-0.557346	swe_6mwtask_before	-0.580881
wkv_t_t1	-0.689614		

HAC discovered 3 clusters and the following variables have got over the threshold of 0.5 correlation level:

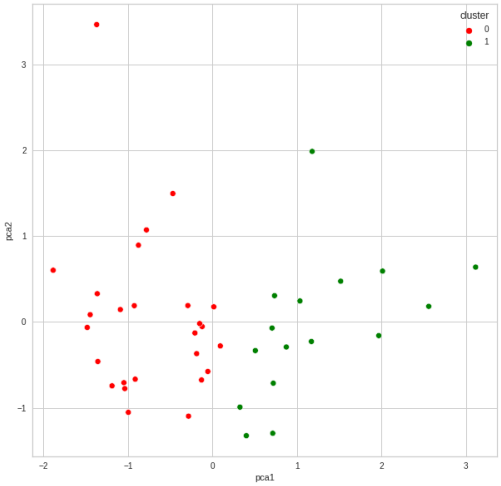

Figure 1: K-Means on PCA

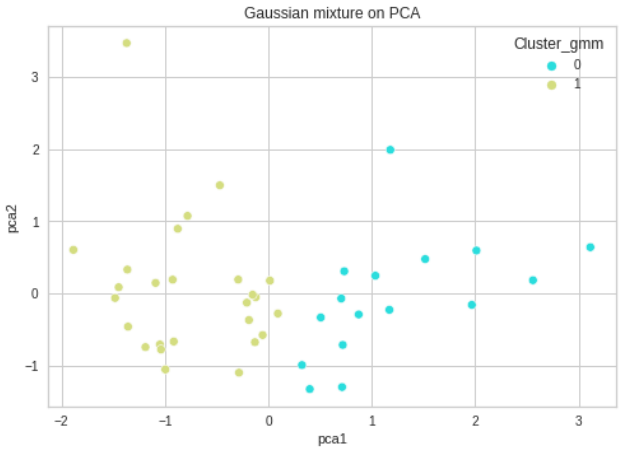

Figure 2: GMM on PCA

wkv_a_t1	0.547359	wkv_g_t4	0.505760
wkv_a_t2	0.503792	wkv_t_t1	0.620538
wkv_b_t1	0.605580	mood_ea_t1	0.500145
wkv_b_t2	0.532576	panas_pos_t1	0.501929
wkv_g_t1	0.621737	swe_6mwtask_before	0.532980
wkv_g_t3	0.548384		

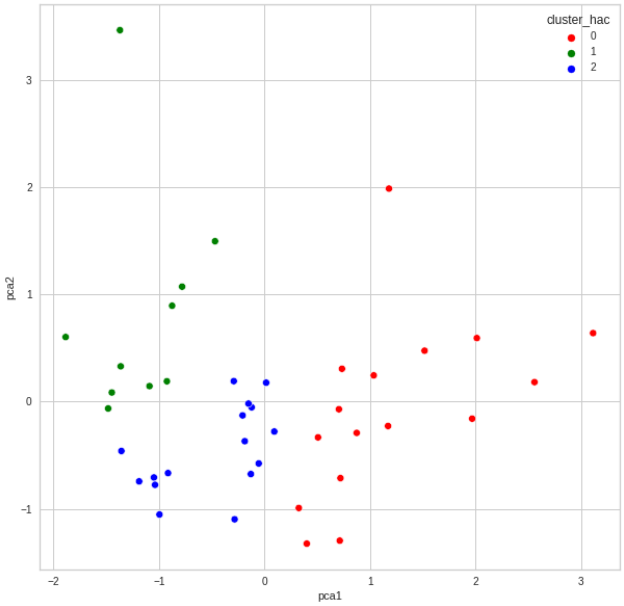

Figure 3: HAC on PCA

DBSCAN also discovered 3 clusters with the respective correlations between original variables and labels:

swe_b_t4	0.538575	wkv_g_t1	0.556086
facts_t4	-0.518128	wkv_g_t2	0.556950
wkv_a_t1	0.560037	wkv_g_t3	0.501974
wkv_a_t2	0.680793	wkv_t_t1	0.661285
wkv_a_t3	0.567684	wkv_t_t2	0.655188
wkv_a_t4	0.596235	wkv_t_t3	0.591373
wkv_b_t1	0.511212	wkv_t_t4	0.508879
wkv_b_t2	0.650794	mood_ea_t1	0.510674
wkv_b_t3	0.535382	mood_ea_t2	0.554564
wkv_b_t4	0.521542	swe_6mwtask_before	0.555814

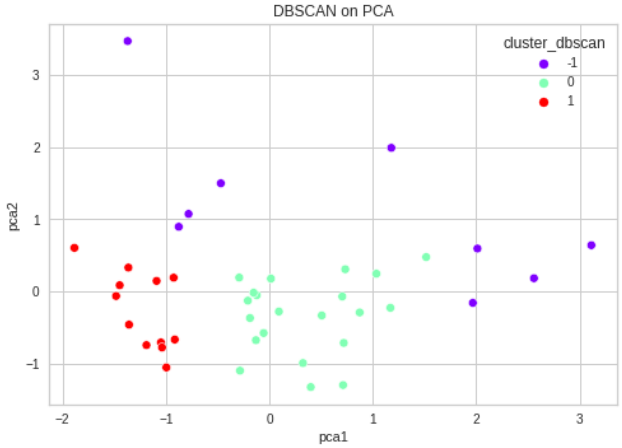

Figure 4: DBSCAN on PCA

Judging by the visualisation of found clusters, we can say that k-Means and GMM algorithms provide results where the division between two clusters is easily recognisable. Although one cannot use correlation as a tool for drawing conclusions about the causality, we still need to point out a connection between cluster labels and variables from wvk, mood and borg questionnaires which contain self-assessment of physical and psychological condition by experiment participants.

4.1.2 Isomap

Isomap does not allow us to assess the quality of performance on the small set of data (almost impossible to make cross-validation) and without true knowledge of manifold geometry our sample was drawn from. However, the results are of interest for comparison purposes.
K-Means algorithm detected 4 clusters, however, none of the original variables demonstrated a level of correlation with labels above 0.5. The division of points into clusters is not understandable at the first glance, and gap statistic providing all negative values supports the idea that no clusters are to be found.

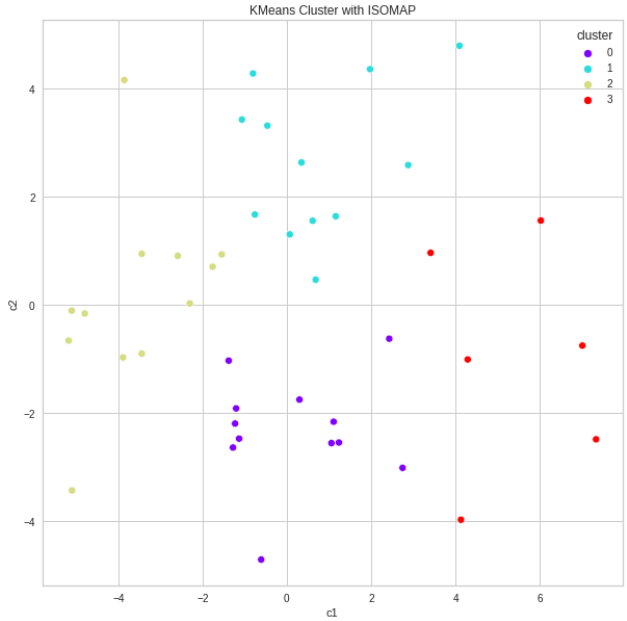

Figure 5: K-Means on Isomap

HAC method discovered 3 clusters, and the resulting labels correlate with the following variables:

wkv_a_t1	-0.610808	wkv_t_t2	-0.615801
wkv_a_t2	-0.618654	wkv_t_t3	-0.568927
wkv_a_t4	-0.524305	wkv_t_t4	-0.638848
wkv_g_t1	-0.649657	mood_v_t2	-0.537950
wkv_g_t2	-0.618399	mood_ea_t1	-0.652498
wkv_g_t3	-0.578661	mood_ea_t4	-0.531126
wkv_g_t4	-0.588213	panas_pos_t1	-0.586731
wkv_t_t1	-0.672398		

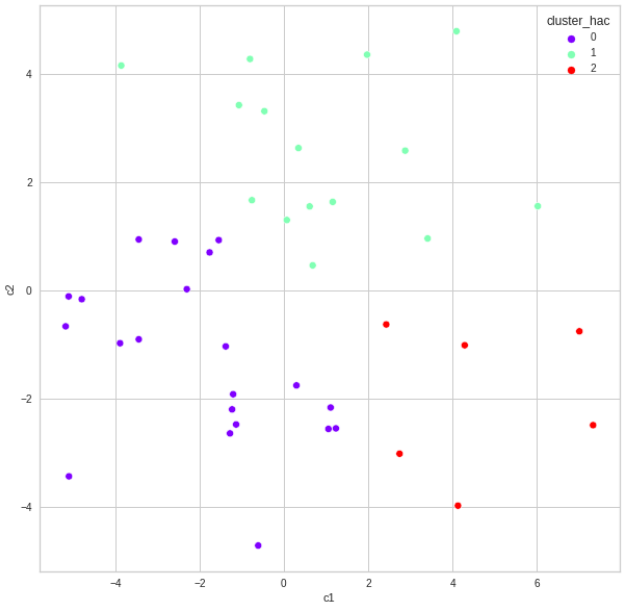

Figure 6: HAC on Isomap

DBSCAN provided visibly unsatisfactory results by finding 3 cluster, whose labels lack meaningful correlation with original.

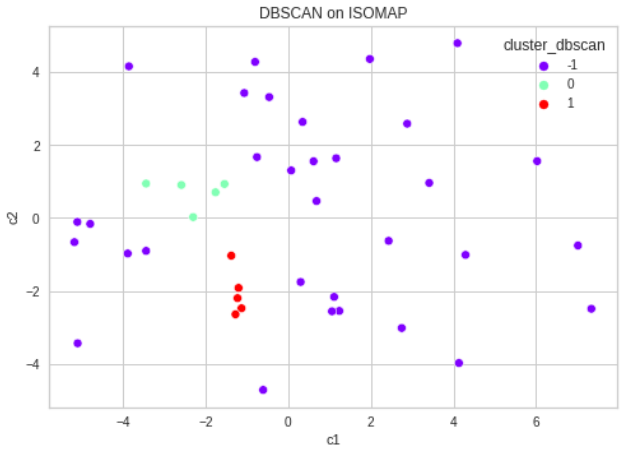

Figure 7: DBSCAN on Isomap

GMM found 2 visually recognisible clusters. Their labels correlated with the following variables:

swe_b_t1	-0.577906	wkv_t_t1	-0.569537
swe_b_t4	-0.671047	wkv_t_t3	-0.653223
swe_m_t1	-0.501209	mood_v_t1	-0.509513
swe_f_t1	-0.534557	mood_v_t3	-0.601686
swe_f_t4	-0.559682	mood_v_t4	-0.534309
wkv_a_t1	-0.581446	mood_c_t1	-0.511983
wkv_b_t1	-0.537528	mood_ea_t1	-0.676045
wkv_g_t1	-0.567792	panas_pos_t1	-0.569254
wkv_g_t3	-0.552188	borg_6mwt_pre	0.527567

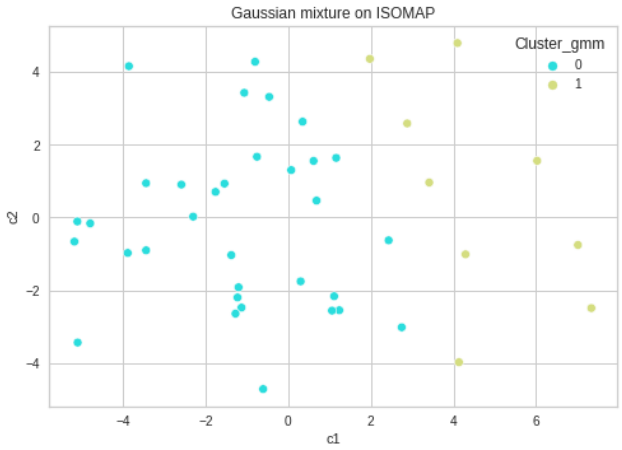

Figure 8: GMM on Isomap

The labels resulting from HAC algorithm have connection to the variables from wkv and mood questionnaires describing perceived physical and psychological well-being of participants. The labels from GMM algorithm additionally demonstrate the link to some variables from swe and borg questionnaires, aimed to assess self-efficacy expectations and sense of exertion respectively.
Intuition behind the clusters found on the basis of Isomap components is more unclear in comparison to those from PCA part. The reason may lie in a sub-optimal approximation of nonlinear connections in the data.

4.1.3 UMAP

UMAP is a qualitatively different method of approximation of nonlinear connections inside data sets. As in case of Isomap, we are unable to look inside of the components produced by the algorithm or to assess its efficiency beyond computational time, so that only visual comparison of the differences is possible.

K-Means algorithms found out 2 clusters, and the only variable which exceeds the threshold of 0.5 correlation with the labels is bsa_B (-0.60543) which describes movement activity at work.

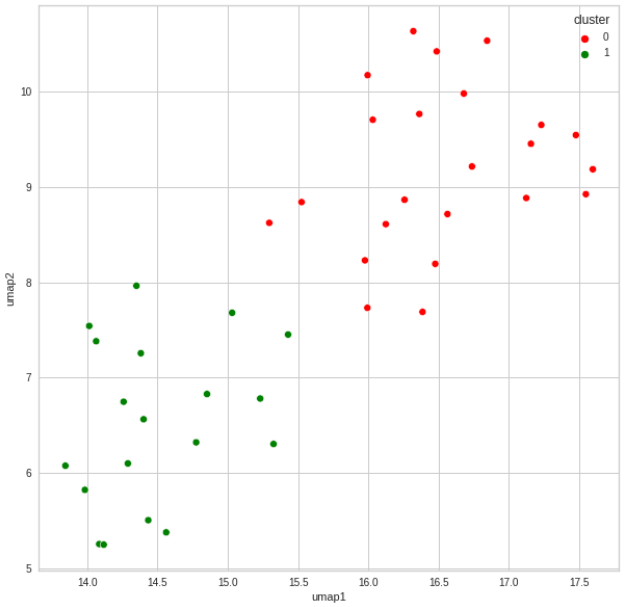

Figure 9: K-Means on UMAP

HAC detected 3 clusters, but the correlation between labels and original variables was overall quite low.

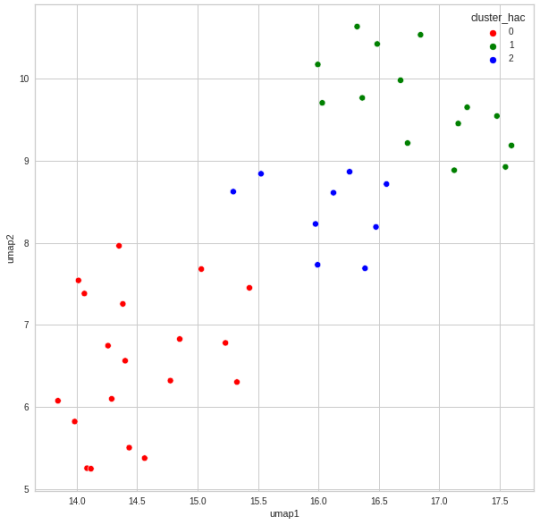

Figure 10: HAC on UMAP

No clusters were found by DBSCAN, which can be partially explained by the incompatibility problems of UMAP with density-based algorithms claimed by the authors (McInnes L, Healy J (2018)).
GMM discovered 2 clusters, however, with no strong or moderate correlations between labels and variables.

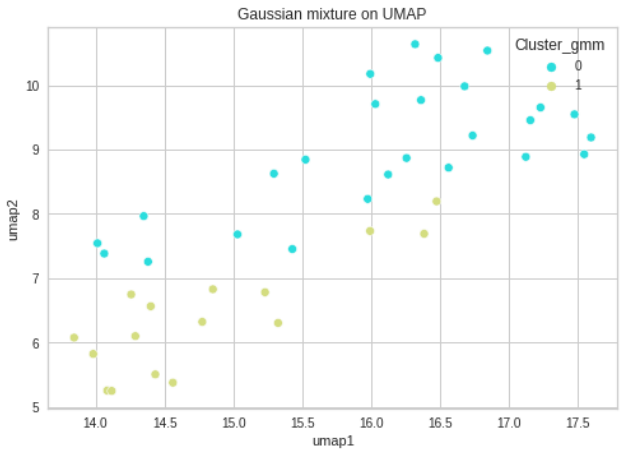

Figure 11: GMM on UMAP

This, again, may point out the problems with the data sample available or the insufficient quality of approximation of nonlinear relationships between its variables.

4.2 Heuristics Approach

The reliance on pure algorithmic approaches may be impractical for complex cases. In this part, we tried to apply heuristics and common sense to obtain better results. Because of time and paper limitations, we only worked with one subset. Our choice of relevant variables was dictated by subjective assessment of their relevance to movement description itself. We decided to reduce our dataset by including some basic demographic variables, variables stemming from swe, bsa and wkv questionnaires and simplified movement data. For movement data, we included information about global minima and maxima from smartphone, fitbit and accelerometer in order to significantly reduce dimensionality. However, we still had 86 variables, so that the resulting subset was undergone PCA. We decided to concentrate on PCA only due to its explicability potential. The resulting principal components demonstrated slightly improved values for explained variance in comparison to inclusion of all variables but still did not manage to reach at least 50% as cumulative value, so we worked only with two first components (see Table 2). The variable ”wkv_a_t4” had again the strongest impact on the first principal component, whereas the variable ”arrival_car/motorcycle” influenced most the second principal component.

PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8	PC9	PC10
17.04	11.46	8.27	7.53	6.81	5.70	5.54	4.59	4.49	3.89

Table 2: Explained variance by first 10 principal components for the subset, in %

K-Means algorithm found 4 clusters in the subset. Their labels correlated mostly with the following variables:

wkv_a_t1	0.511166
wkv_a_t2	0.544428
arrival_on foot	-0.722351

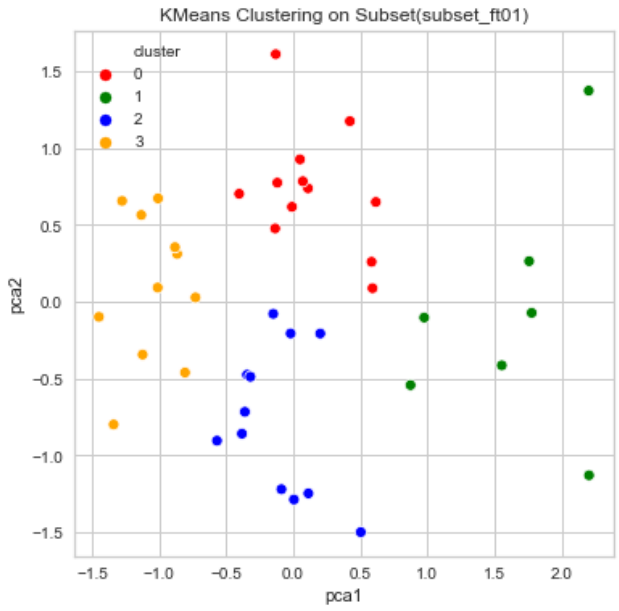

Figure 12: K-Means on PCA, subset

HAC algorithm discovered two clusters, however, almost all points were signed to one cluster. The highest correlation of the labels was with bsa_F variable describing movement activity during free time as number of minutes per week: 0.860901.

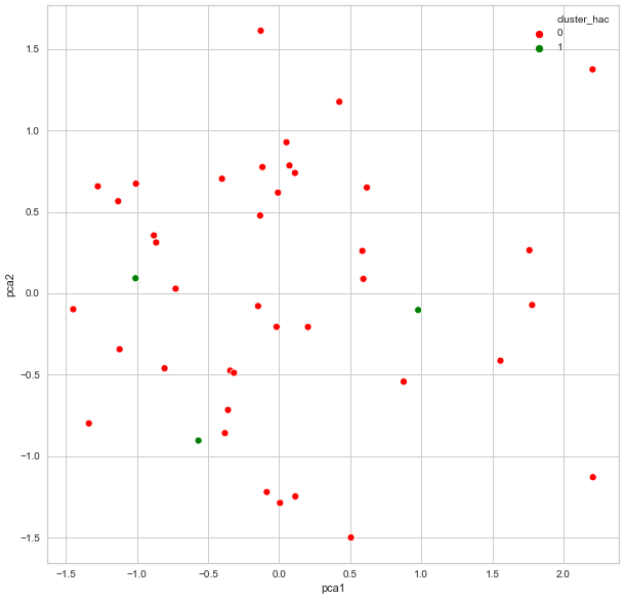

Figure 13: HAC on PCA, subset

Three clusters were established by DBSCAN, where the only variable which exceeded the threshold of 0.5 correlation with the labels was wkv_g_t1 (0.528682).

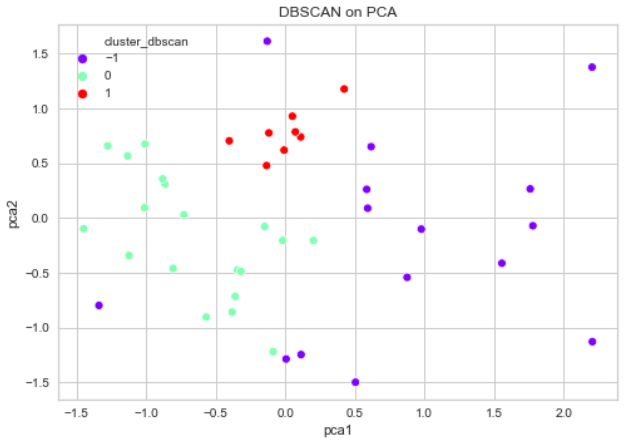

Figure 14: DBSCAN on PCA, subset

The final clustering algorithm, GMM, discovered 5 clusters, whose labels mostly correlated with the variables from wkv questionnaire:

swe_b_t4	0.581910	wkv_g_t2	0.593928
wkv_a_t1	0.625502	wkv_g_t3	0.546051
wkv_a_t2	0.638494	wkv_g_t4	0.561535
wkv_a_t4	0.558810	wkv_t_t1	0.650722
wkv_b_t1	0.547004	wkv_t_t2	0.668215
wkv_g_t1	0.676843	wkv_t_t4	0.706175

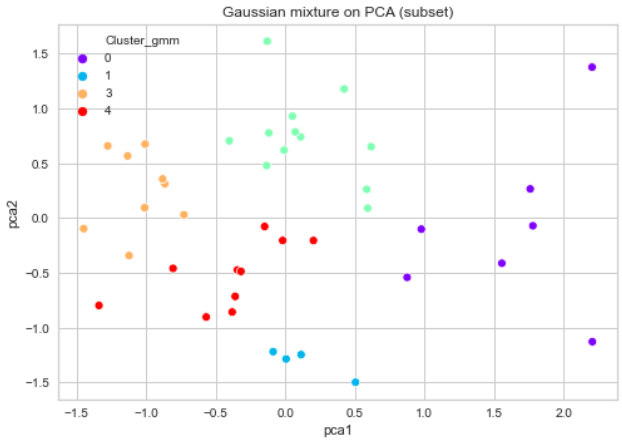

Figure 15: GMM on PCA, subset

The results obtained from the subset treatment do not resonate with the findings from the whole data set. There is no consent between different algorithms on how the clusters should look like or, what are the possible relations between labels and original variables.

5 Conclusion

The final conclusion on the conducted research is that we were unable to discover any persistent clusters. The only persistent result is that variables from wkv, mood and swe questionnaires may be involved into constituting of clusters although our methodology is unable to track it down precisely. There is a lot of fluctuation in the number and shape of the clusters depending on the dimensionality reduction technique applied, particular clustering algorithm and even the size of input sample. There are several possible reasons explaining this circumstance:

- Unorthodox shape of the data set (42 objects with 149 features) requiring a lot of domain knowledge and skills to work with;
- Complex geometry of a the data set and nonlinear relationships between the variables which are uneasy to discover and treat with the existing algorithms;
- A large number of ordinal variables encoding responses from questionnaires makes it difficult for clustering algorithms to detect clusters;

- Possible confusion of participants about the variety of similarly sounding questions from questionnaires;
- Possible imperfections of methods we used to encode movement data.

There are the following suggestions we would make to improve the results:

- Increase the number of participants to compensate for the number of features. The rule of thumb requires the number of objects to be at least double of that of features;
- Reconsider the experiment design and drop the parts of questionnaires including repetitive or similar information;
- Change the methodology of encoding the sensors information.

6 Technical information

The code is organized in five parts. The feature extraction from movement data was performed in R (version 4.1.2) and is contained in the file "full_automated_r_skript.R", while the rest of analysis was done in Python in Jupiter notebook (version 3.7.12) and is contained in the following files:

- "PCA_clustering.ipynb"
- "Isomap_clustering.ipynb"
- "UMAP_clustering.ipynb"
- "Heuristic_Approach.ipynb"

References

Alashwal H., El Halaby M., Crouse J., Abdalla A., Moustafa A. (2019). The application of unsupervised clustering methods to alzheimer's disease. *Front. Comput. Neurosci.* DOI: 10.3389/fncom.2019.00031.

Bradford A., Yellamraju T., Boutin M. (2020). Clustering small datasets in high-dimension by random projection. *arXiv:2008.09579 [stat.ML].* DOI: 10.48550/arXiv.2008.09579.

Forsberg F., Alvarez Gonzalez P. (2018). Unsupervised machine learning: An investigation of clustering algorithms on a small dataset. *Thesis no: URI: urn:nbn:se:bth-16300.*

McInnes L, Healy J (2018). Approximation and projection for dimension reduction. *ArXiv e-prints 1802.03426.* DOI: 10.48550/arXiv.1802.03426.

Tenenbaum J.B.,de Silva V., Langford J.C. (2000). Global geometric framework for nonlinear dimensionality reduction. *Science vol 290.* DOI: 10.1126/science.290.5500.2319.