FOUNDATIONS OF COMPUTING AND DECISION SCIENCES
Vol. 39 (2014) No. 4

DE GRUYTER OPEN

DOI: 10.2478/fcds-2014-0015 ISSN 0867-6356
e-ISSN 2300-3405

# Subpopulation Discovery in Epidemiological Data with Subspace Clustering

Uli Niemann[a], Myra Spiliopoulou[a], Henry Völzke[b], and Jens-Peter Kühn[b]

[a]Otto-von-Guericke University Magdeburg, Germany
[b]University Medicine Greifswald, Germany

**Abstract.** [1] [2] A prerequisite of personalized medicine is the identification of groups of people who share specific risk factors towards an outcome. We investigate the potential of subspace clustering for finding such groups in epidemiological data. We propose a workflow that encompasses clusterability assessment *before* cluster discovery and quality assessment *after* learning the clusters. Epidemiological usually do not have a ground truth for the verification of clusters found in subspaces. Hence, we introduce quality assessment through juxtaposition of the learned models to "models-of-randomness", i.e. models that do not reflect a true cluster structure. On the basis of this workflow, we select subspace clustering methods, compare and discuss their performance. We use a dataset with hepatic steatosis as outcome, but our findings apply on arbitrary epidemiological cohort data that have tenths of variables and exhibit class skew.

## 1 Introduction

A major objective of epidemiology is to identify risk factors for diseases [1], thus contributing to the advancement of prevention strategies, of diagnostics and therapies. For the purposes of personalized medicine [2, 3], it is further necessary to identify subpopulations that are characterized by the same risk or protective factors with respect to an outcome. Subspace clustering methods lend themselves naturally to this task: they scan the high-dimensional feature space and find clusters of individuals who are similar to each other for some variables (e.g. medical test results) but not for others (e.g. age and gender). In this work, we investigate the potential of subspace clustering on data from the epidemiological study SHIP [4] for the disorder "hepatic steatosis".

[1] Part of this work was supported by the German Research Foundation project SP 572/11-1 "IMPRINT: Incremental Mining for Perennial Objects".

[2] The data used in this work were made available through the cooperation SHIP/2012/06/D "Predictors of Steatosis Hepatis".

Subspace clustering algorithms build subsets of the original set of dimensions (i.e. build "subspaces") and construct clusters in these subspaces. This means that when the similarity of two objects is computed in a subspace, all dimensions except those in the subspace are ignored (i.e. "projecting away", in the sense of the relational algebra operation "projection"). But why not use traditional clustering for the discovery of small subpopulations? In [5, 6, 7] we find three limitations of traditional clustering: (i) curse of dimensionality, (ii) correlations among features, and (iii) sensitivity to irrelevant or noise-prone features. Epidemiological data are high-dimensional, since they contain recordings on sociodemographics, assessments and laboratory tests. Some of these variables are naturally correlated (e.g. menopause with age) [3]. The third limitation of traditional clustering also applies on epidemiological data: many of the variables in the high-dimensional space are only filled for some of the study participants (e.g. menopause and pregnancy only applies for women), thus making participants with NULL values artificially different from the other participants. This might suppress similarities with respect to e.g. laboratory test results. Such caveats are elegantly overcome by subspace clustering.

Our contribution is twofold. First, we propose a workflow for studying the potential of subspace clustering in the very important area of personalized medicine. We do so for an example disorder, but our findings are relevant for high-dimensional epidemiological data on disorders that exhibit a skewed distribution, where different subpopulations are characterized by different risk factors. Second, we propose a new approach for the evaluation of subspace clustering algorithms *without* using a ground truth.

The paper is organized as follows. We next present related work on finding subpopulations in epidemiological data with supervised methods, and we give an overview of unsupervised algorithms that identify subspaces and build clusters on them. In section 3, we describe the epidemiological study SHIP and the sample SHIP2·578.In section 4, we present our workflow for assessing the potential of subspace clustering on epidemiological data. In Section 5, we apply this workflow on SHIP2·578, assess the model quality of different subspace clustering algorithms and report on our findings. We then summarize the insights we won and the lessons-learned, and we present a list of tasks for future work.

## 2 Related Work

In epidemiological mining, data analysis is performed with respect to a target outcome – an impairment or an intervention. Therefore, data analysis mostly concerns supervised learning. However, it has been recognized [8] that the complete population used for learning can be very heterogeneous, affecting classifier performance negatively. In [8], Zhang and Kodell first train an ensemble of classifiers, then associate with each training instance the predictions made on it by each ensemble member, thus creating

[3] In the following, we use the term "feature" to describe a variable associated with a value range, e.g. body mass index larger than 28. However, we acknowledge that the term "feature space" is often used to denote the set of variables; we avoid this term whenever possible.

a new set of dimensions/variables, where the variables are the predictions. They then perform hierarchical clustering on the instances, thus building three subpopulations: one where the prediction accuracy is high, one where it is intermediate and one where it is low [8].

Zhang and Kodell perform clustering *after* classification [8]. In [9], clustering is done *before* classification, to partition tumors into regions before classifying them on malignancy; the motivation is that a tumor may be very heterogeneous. After clustering, variables are derived for each region, and are then used for classification [9].

In [10, 11], we improve model quality by identifying subpopulations in a supervised way. In [10], we report that for the population under study, the two genders show different class distributions and that for one of the genders (female), age is also modulating the outcome. In [11], we use information gain to partition the population on gender and then on a specific age value. While it is possible to select features on the basis of their "merit" [12], as we did in [11], it is more reasonable to consider clustering instead of a brute-force approach.

Klemm et al. [13] analyse epidemiological data on pains of the spine, applying conventional clustering. They report that all tested algorithms are very sensitive to parameter settings and to the choice of the distance measure [13].

In his survey [7], Arthur Zimek elaborates on the advantages of subspace clustering over traditional clustering for high-dimensional data. He uses following terminology: "subspace clustering" algorithms find potentially overlapping groups of objects whereas "projected clustering" algorithms partition the objects into non-overlapping groups.This distinction is not always retained in the literature, but in [7] it is properly justified by pointing to the pioneering algorithms *CLIQUE* [14] for subspace clustering and *PROCLUS* [15] for projected clustering. In our work, we study subspace clustering and projected clustering algorithms, using these terms as defined in [7].

Studies of clustering in subspaces of medical data are rare Damian et. al apply COSA (Clustering Objects on Subsets of Attributes) on different datasets, and group objects on metabolic and/or physiological similarities [16]. They show that better clusters can be found in a subset of the attributes than in the original high-dimensional space [16].

Similarly to [5, 6, 7], we compare subspace clustering algorithms. However, our goal is not to highlight their general advantages and disadvantages, but to investigate their potential for high-dimensional epidemiological data.

## 3 Materials

Hepatic steatosis (informally fatty liver) is the condition of raised fat concentration inside the liver cells. According to [17], around 20-30% of adults have developed hepatic steatosis. Hence, it is considered as the most common liver disorder [17], with increasing prevalence tendency for western countries [18]. Hepatic steatosis itself is a reversible disorder but can evolve into a more severe disease, for instance steatohepatitis, fibrosis, cirrhosis and liver cell cancer [19]. A timely diagnosis is difficult since hepatic steatosis commonly has no symptoms. Epidemiological studies on this disorder focus on discovering risk factors that hold for the whole population and for specific subpopulations, and on specifying reliable indices for diagnosis. For example, indices of fat storage in the body (BMI, waist circumference) and the liver enzyme GGT have been proposed in [20] as part of a "Fatty Liver Index". The role of the SNPs rs11597390, rs2143571 and rs11597086 is highlighted in [21] (cf. Table 3 of [21]). In [22], it has been shown that menopausal status is associated with hepatic steatosis, thus referring to a variable that concerns only one subpopulation (female persons).

To study the potential of subspace clustering for the discovery of subpopulations sharing similar characteristics, we use hepatic steatosis as outcome on a data sample from the population-based Study of Health in Pomerania (SHIP) [4], as described hereafter.

## 3.1 On the Examination Programme of SHIP

SHIP encompasses two independent cohorts of residents in the region Pomerania, in Northeast Germany. Inclusion criteria are residency and age (between 20 and 79 years). The set of dimensions is very large, because the examination programme contains interviews, exercise tests, laboratory analyses, somatometric and blood pressure measurements, dental, dermatological, cardio-metabolic and various ultrasound examinations, sleep monitoring and whole-body magnetic resonance tomography (MRT). SHIP is a longitudinal study: baseline examinations for the first SHIP cohort were performed between 1997 and 2001 (SHIP-0, n= 4308), follow-up examinations were done in 2002-2006 (SHIP-1, n= 3300) and 2008-2012 (SHIP-2, n= 2333) [4]. SHIP data are being used in numerous independent epidemiological studies, including studies on hepatic steatosis (e.g. [22], [23]).

We analyze on a random sample of 578 SHIP-2 participants for which we have received the recordings of the target variable, and we denote as SHIP2·578 and describe hereafter.

## 3.2 Class Distribution in the SHIP2·578 Dataset

SHIP2·578 have 56 variables, of which we use the 26 numerical ones; they are depicted in Table 1. We omit categorical variables, because in our preliminary experiments we found that subspace clustering algorithms favored categorical variables with very skewed value distribution and thus produced subspaces of very low quality. We use the fat concentration identified in the MR images of the liver to derive the target variable: participants with a fat concentration of less than 10% are assigned to class A (negative); participants with fat concentration values between 10 and 25% are assigned to class B, and those with values higher than 25% are assigned to class C. The classes B and C are both positive. In our experiments, we consider the classes

[4] The second SHIP cohort, SHIP-TREND, is not relevant for this work, so we skip its description.

A and notA to make the unsupervised learning task easier. Findings on supervised learning for the three classes can be found in [10].

| Name | Description |
|---|---|
| age_ship_s2 | Age at examination [$years$] |
| menopaus_s2 | Age at menopause [$years$] |
| alkligt_s2 | Daily Alcohol intake [$g/day$] |
| sleeph_s2 | # of sleep hours per day |
| som_bmi_s2 | Body Mass Index [$kg/m^2$] |
| som_tail_s2 | Waist circumference [$cm$] |
| som_huef_s2 | Hip size [$cm$] |
| hgb_s2 | Haemoglobin [$mmol/l$] |
| hba1c_s2 | Glycated haemoglobin [%] |
| quick_s2 | Thromboplastin time Quick[%] |
| fib_cl_s2 | Fibrinogen (Clauss) [$g/l$] |
| crea_s_s2 | Serum creatinine [$\mu mol/l$] |
| hrs_s_s2 | Serum uric acid [$\mu mol/l$] |
| gluc_s_s2 | Serum glucose [$mmol/l$] |
| asat_s_s2 | Serum ASAT [$\mu mol/sl$] |
| ggt_s_s2 | Serum GGT [$\mu mol/sl$] |
| lip_s_s2 | Serum lipase [$\mu mol/sl$] |
| chol_s_s2 | Serum cholesterol [$mmol/l$] |
| tg_s_s2 | Serum triglycerides [$mmol/l$] |
| hdl_s_s2 | Serum HDL [$mmol/l$] |
| ldl_s_s2 | Serum LDL [$mmol/l$] |
| tsh_s2 | TSH [$mU/l$] |
| jodid_u_s2 | Lodide (urine) [$\mu g/dl$] |
| crea_u_s2 | Creatinine (urine) [$mmol/l$] |
| sd_volg_s2 | Total thyroid volume [$ml$] |
| mrt_liverfat_s2 | Liver fat concentration [%] |

**Table 1**: Overview of the numerical variables in SHIP2·578

The fat concentration values acquired from the MR images are preliminary: quoting [10], "the MR technique used to compute the values of the original target variable mrt_liverfat_s2 included a correction of $T2\star$ effects, but other confounders for chemical shift MR fat quantification, such as multi-spectral complexity of fat and T1 effects were ignored. However, as shown in [24], these latter confounders behave linearly with respect to the target. Through conservative choice of the cut-off value (...) and discretization, this problem was partially amended, so that the mining methods still behave reliably." The cut-off values we use are 10% and 25%, as mentioned earlier.

SHIP2·578 contains 314 female and 264 male participants. These two subsets have different class distributions, as shown in Table 2 (repeating part of a table from [10]). Distribution differences do not necessarily mean that the outcome is predicted by different variables in each subset, though. We introduce a *clusterability assessment* step in our workflow (cf. 4.2), in which we seek indicators of clusters over different subsets of variables.

| | | absolute | | | relative | | |
|---|---|---|---|---|---|---|---|
| SHIP2·578 data | total | A | B | C | A | B | C |
| All | 578 | 438 | 108 | 32 | 76 % | 19 % | 6 % |
| "M" (gender=male) | 264 | 183 | 66 | 15 | 69 % | 25 % | 6 % |
| "F" (gender=female) | 314 | 255 | 42 | 17 | 81 % | 13 % | 5 % |

**Table 2**: Class Distribution on Gender in SHIP2·578

## 4 Subspace Clustering Workflow on Epidemiological Data

Our workflow is depicted in Figure 1. It encompasses components for following major tasks: dataset preparation, clusterability assessment, model learning with subspace clustering and projected clustering algorithms, and model quality assessment.

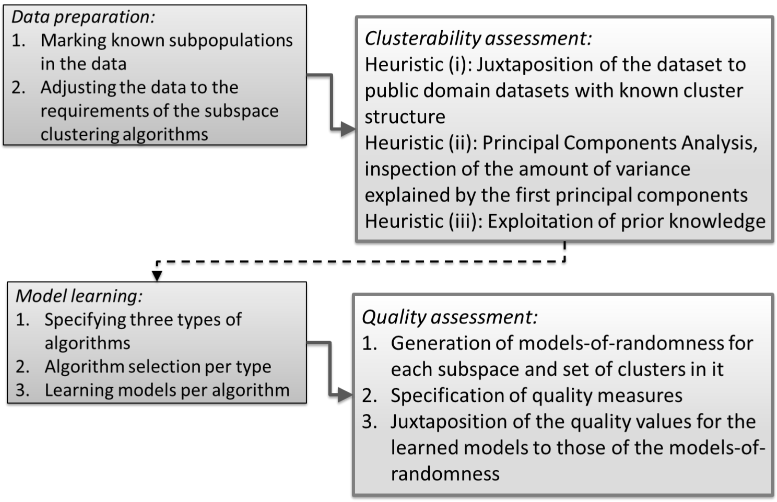
Data preparation:
1. Marking known subpopulations in the data
2. Adjusting the data to the requirements of the subspace clustering algorithms
Clusterability assessment:
Heuristic (i): Juxtaposition of the dataset to public domain datasets with known cluster structure
Heuristic (ii): Principal Components Analysis, inspection of the amount of variance explained by the first principal components
Heuristic (iii): Exploitation of prior knowledge
Model learning:
1. Specifying three types of algorithms
2. Algorithm selection per type
3. Learning models per algorithm
Quality assessment:
1. Generation of models-of-randomness for each subspace and set of clusters in it
2. Specification of quality measures
3. Juxtaposition of the quality values for the learned models to those of the models-of-randomness

**Figure 1**: Workflow of our approach on assessing the merit of subspace clustering and projected clustering on epidemiological data

Two of these tasks (on the right of Figure 1) build the core of our approach: under *clusterability assessment* we define indicators on the existence of clusters in subspaces; under *quality assessment* we generate models governed by randomness ("models-of-randomness", cf. 4.4) and check whether the quality of the "true models" significantly exceeds the quality achieved by noise. This latter task is neccessary, because our goal is to identify subpopulations that have not been already discovered. We describe all tasks hereafter.

## 4.1 Data Preparation

As shown in Figure 1, data preparation in our workflow covers two aspects. First, we check for subpopulations that are characterized by different sets of variables. The identification of such subpopulations delivers a first indication on the existence of potentially interesting subspaces. Moreover, subpopulations can be used for the evaluation of the clusters.

As can be seen in Table 2, SHIP2·578 has two subpopulations that differ in the class distribution. Moreover, several variables are only filled for the female participants. So, subspace clustering algorithms may identify subspaces containing some variables filled solely for female participants.

Data preparation further encompasses the alignment of the epidemiological data to the requirements of the algorithms – normalization and treatment of missing values. Normalization is necessary for all algorithms that use a distance that is sensitive to differences among value ranges (e.g. Euclidean distance). Epidemiological data have variables with very different value ranges (e.g. age in years, sleep hours per day). Hence, for each variable $V$ with original range $R_V = [\min_V, \max_V]$, we apply min-max normalization [25], mapping each value $w \in R_V$ into $w' = \frac{w - \min_V}{\max_V - \min_V}$.

Treatment of missing values is necessary for algorithms that cannot compute distances between objects that have missing values. We consider two options, *REPLACEMENT* and *MAX DISTANCE*. For *REPLACEMENT*, we replace each missing value of variable $V$ with the average of the observed values if $V$ is numerical, and with the mode of $V$, if $V$ is categorical. For *MAX DISTANCE*, we specify that the distance between NULL and any value is 1.0.

In our experiments, we opted against *MAX DISTANCE*, because it tends to artificially bring objects with few missing values closer to each other and it turns objects with many missing values into outliers. However, it is possible that *REPLACEMENT* hinders the identification of subspaces containing only male/only female participants.

## 4.2 Clusterability Assessment

To acquire an indication that there are clusters in the data, we use three heuristic indicators: (i) similarity of the dataset under study to datasets known to contain clusters, (ii) amount of variance captured by combinations of variables under Principal Component Analysis and (iii) prior knowledge on the existence of clusters in subspaces.

### 4.2.1 Heuristic (i): Clusterability Assessment through Dataset Comparison

As our first heuristic, we propose to compare the dataset under study to datasets known to have a cluster structure. For the comparison, we define *similarity between datasets* on the basis of (a) cardinality, (b) number of dimensions and (c) number of classes: we select (i.1) datasets that have comparable cardinality, number of dimensions and number of classes as our dataset, as well as (i.2) datasets with different values for some of the three properties.

As clusterability indicator for heuristic (i), we propose to use the *$K$-dist curve* of [26]: the "$K$-dist" of an object is its (Euclidean) distance to its k$^{th}$ nearest neighbour. For our heuristic, we compute the "$K$-dist curve" or "$K$-dist graph" of a dataset by sorting the objects on increasing $K$-dist, and then we juxtapose the curves of different datasets.

The motivation of using the $K$-dist curves is that if a cluster structure exists in the full-dimensional dataset, then it will be reflected as an early steep increase in the $K$-dist curve. In contrast, a slow increase means that most objects are at the same distance to each other; then, we infer that clustering struggles to find representative clusters in full space. However, we do not know in advance when an increase is "early steep" vs "slow". Therefore, we compare the curve of the dataset under study to datasets, where clusters are easy-to-find (according to literature). If the curve of our dataset is similar to curves of datasets with easy-to-find clusters, this is an indicator of clusterability in the complete set of dimensions. The choice of $k$ does not critically affect the trend of the curves, therefore we use a low value to reduce complexity in calculating the distance values.

In Figure 2, we see the $K$-dist curves for SHIP2·578 (578 objects, 26 dimensions, 3 classes) and for the UCI datasets *Iris* (150 objects, 4 dimensions, 3 classes) [27], *Ionosphere* (351 objects, 34 dimensions, 2 classes) [28] and *Libras Movement* (360 objects, 90 dimensions, 15 classes) [29], denoted as "Libra" in the figure. With respect to cardinality and number of dimensions, the SHIP2·578 dataset under study is most similar to the *Ionosphere* dataset (selection i.1), and is dissimilar to *Iris*, which contains much less elements and has only 4 dimensions, and to *Libras Movement* which has far more dimensions (selection i.2). The *Iris* clusters are easy to find (e.g. by K-Means or DBSCAN) on the complete set of dimensions.

Before computing Euclidean distances, we normalize all variable ranges. Since each dataset has a different number of dimensions $d$, we compute the $K$-dist as the sum of distances over all dimensions and then divide by $d$. We set $k = 4$, as in [26].

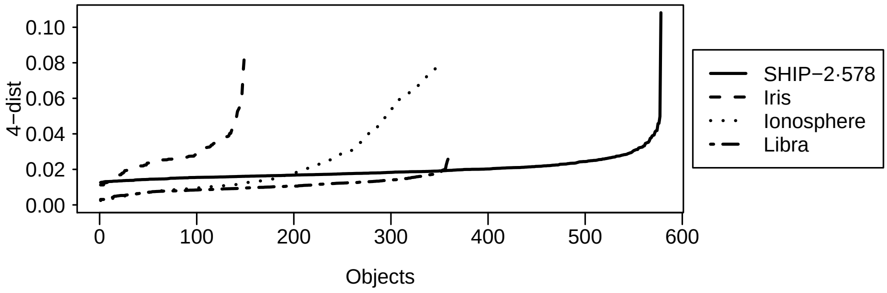

**Figure 2**: The 4-dist curve of the dataset SHIP2·578 under study, juxtaposed to the curves of datasets, which have a known cluster structure

As we see on Figure 2, the 4-dist curves of *Iris* and *Ionosphere* are the most similar to each other, although they are dissimilar in number of objects and dimensions. Both datasets exhibit an early upward trend (after reading the first few elements). The curve of *Libras Movement* is very different from these two curves, and is the one most similar to the curve of SHIP2·578: both of them exhibit a slow upward trend, indicating that most objects are equi-distant from each other, except for a few outliers that are very far from all other elements; these outliers are reflected in a steep raise for the last dataset elements.

The curve for *Libras Movement* might be due to the impact of its 90 dimensions on the Euclidean distance values. However, the curse of dimensionality cannot be the sole cause for the curve of SHIP2·578, which has more elements and less dimensions, even less than *Ionosphere*. Hence, we expect that SHIP2·578 does not contain easy-to-find clusters, either due to the curse of dimensionality or because there are no clusters in the complete space.

To test this further, we run $k$-means and DBSCAN with different parameter settings on SHIP2·578. We acquired very poor results both for internal indices and with respect to class separation (using the classes as clusters).

### 4.2.2 Heuristic (ii) on the Existence of Interesting Subspaces

In this heuristic, we perform Principal Component Analysis and inspect the amount of variance explained by the first few principal components (PC). The motivation is as follows. Each principal component reflects correlations between some variables. The variables involved in any two principal components may well be the same. However, if several principal components are needed to explain most of the variance, then there may be some subsets of correlated variables that are only partially overlapping.

As we see in Figure 3, the first three principal components explain ca. 35% of the variance in SHIP2·578. Since SHIP2·578 has more than few dimensions (we consider 26), we expect that some of the subsets of correlated variables are not identical and may be found by a subspace clustering method.

### 4.2.3 Heuristic (iii): Exploitation of Prior Knowledge

This heuristic exploits independently acquired insights on variables that are associated with the outcome.

In [10, 11] we name variables that are known to be associated to hepatic steatosis (from epidemiology literature), and for which we found (on SHIP2·578) that they contribute to class separation for only part of the sample (e.g. specific age intervals, only one of the genders). In [10], we found the most important variable/valuerange-pairs for the subset of the female and the male participants and found them to overlap only partially. In [11], we ranked the variables on information gain and came to the same observation: the top-10 positions in the two subsets overlap only partially.

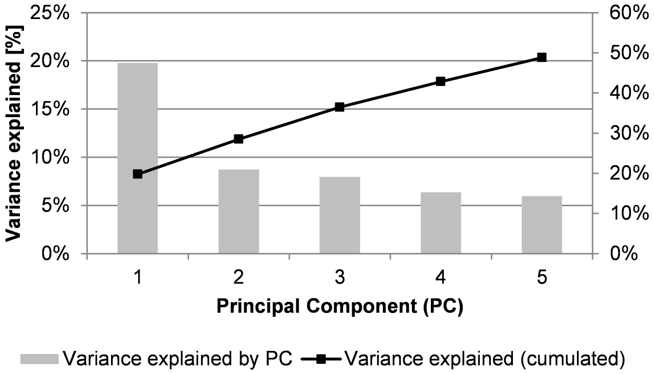

**Figure 3**: Pareto chart of the variance explained by the first 5 principal components on SHIP2·578

### 4.2.4 Summary of Clusterability Assessment Heuristics

We propose heuristics to a priori assess the appropriateness of subspace clustering for the dataset under study,. Heuristic (i) juxtaposes the $K$-dist curve of the dataset with the curves of a selection of public domain datasets with known cluster structure. This heuristic delivers a positive indication if the dataset under study has a $K$-dist curve that is similar to a dataset, whose clusters are known to be easy to find. This indication concerns the complete space, since we consider all dimensions for the computation of the $K$-dist values. Heuristic (ii) concerns the existence of interesting subspaces; itt delivers a positive indication if the amount of variance covered by the first principal componentsis comparatively low.

For SHIP2·578, heuristic (i) delivers a negative result: the dataset is not similar to datasets with easy-to-find clusters in the complete space. Heuristic (iii) delivers a positive result: the dataset may contain interesting subspaces. Heuristic (iii) on the exploitation of prior knowledge delivers a weakly positive result: the dataset contains subpopulations that overlap only partially in the sets of variables describing them.

## 4.3 Subspace & Projected Clustering

Subspace clustering methods find (possibly overlapping) groups of objects which are homogeneous in a subset of features, or in a combination of feature subsets [7]. More formally, let $X = \{x_1, \cdots, x_n\}$ be a set of objects and $F = \{f_1, \cdots, f_m\}$ the set of dimensions. Then, a subspace cluster $C = (O, S)$ encompasses a subset $O \subseteq X$ of objects that are similar to each other in the subspace $S \subseteq F$. Subspace clustering algorithms allow an object to belong to two clusters. In contrast, project clustering methods assign an object to exactly one cluster, thus avoiding redundancies.

As a (oversimplified) example, assume a set of 9 objects $X = \{x_1, \ldots, x_9\}$, where

$F = \{f_1, \ldots, f_4\}$. Further, assume that $x_2, \ldots, x_6$ are similar to each other across dimensions $f_1, f_2 \in F$, $x_4, \ldots, x_7$ are similar to each other across dimensions $f_3, f_4 \in F$; $x_1, x_9$ are similar to each other over dimensions $f_4, f_5$. Then, a subspace clustering algorithm would return three clusters, $C_1 = (\{x_2, \ldots, x_6\}, \{f_1, f_2\})$ and $C_2 = (\{x_4, \ldots, x_7\}, \{f_3, f_4\})$ and $C_3 = (\{x_1, x_9\}, \{f_4, f_5\})$. The first two clusters differ by only one object, $x_7$. A projected clustering algorithm would either create only two of these three clusters, or split the objects $x_2, \ldots, x_7$ into two distinct clusters, suppressing the fact that some of the objects would belong to both clusters. Which approach is best depends ultimatively on the semantics of the subspaces, the objects and the application itself. [5]

Therefore, we propose to consider both subspace clustering algorithms (type I) and projected clustering algorithms (type II). However, we do not focus solely on finding clusters in subspaces. Rather, we recognize that it is no less appealing to identify *interesting* subspaces over the epidemiological data, i.e. subspaces that are expected to contain clusters, as the two subspaces $\{f_1, f_2\}$ and $\{f_3, f_4\}$ in the example above. Once the subspaces are found, a conventional clustering algorithm can be used to find clusters in them. Hence, we also consider algorithms that find all interesting subspaces (type III).

For our workflow, we consider the type I algorithm RIS [30], the type II algorithm DUSC [31] and the type III algorithm PROCLUS [15]. We outline these algorithms below. A more detailed discussion of all algorithms we investigated is in [32].

**Type I - Subspace Clustering:** We choose the "Dimensionality Unbiased Subspace Clustering" algorithm (DUSC) [31]. Earlier subspace clustering algorithms use a single density threshold for all subspaces; this is problematic for large subspaces (curse of dimensionality). DUSC alleviates this problem by using kernel estimators and computing density thresholds that depend on the cardinality of each subspace [31].

**Type II - Projected Clustering:** We choose the algorithm PROCLUS [15] as typical member of the "projected clustering" family. PROCLUS has two major parameters, the number of clusters and the *average* number of cluster dimensions. In our example above, the first parameter would determine whether PROCLUS would build three or two clusters (assuming that the average number of dimensions were set to 2). If the second parameter were set to 3, PROCLUS may have attempted to build a 4-dimensional cluster after building one (or two) 2-dimensional cluster(s).

**Type III - An algorithm that ranks subspaces on interestingness:** We choose the subspace ranking algorithm RIS [30]. RIS processes each object $x$ in turn, and works bottom-up to find the set of subspaces where $x$ is a "core point", according to [26]; these are the subspaces "relevant for" $x$. To rank all subspaces, RIS uses a quality function that considers the number of objects for which a subspace is relevant and the number of dimensions of the subspace (higher numbers preferred), as well as the volume covered by the hypersphere spanning all objects in this subspace (lower volumes preferred) [30]. RIS prunes away a subspace $S$, either if there is a subspace

[5] This is a very simplistic example, intended solely to highlight the differences between subspace clustering and projected clustering. Some subspace clustering algorithms may disallow overlaps among the subspaces. Other algorithms may reject some of these clusters as two small.

containing it and having higher quality (thus promoting larger subspaces), or if all subsets of $S$ with size $|S| - 1$ have higher quality than $S$ (thus promoting smaller subspaces) [30].

For our workflow, we speed up RIS as follows. Instead of iterating over each object, we invoke RIS to find each subspace $S$ that has at least one core point. We store $S$ and the number of core points in it, $count(S)$, and compute the RIS quality function [30] for it. We then rank the subspaces on their RIS quality score and choose the top-ranked ones.

### 4.3.1 Distance Measure

All three algorithms we consider use Euclidean distance [31, 15, 30], and this is the distance we use hereafter, since we consider only the numerical variables of SHIP2·578 (cf. 3.2). If categorical variables are also used, measures like Heterogeneous Euclidean-Overlap Metric (HEOM) or Heterogeneous Value Difference Metric (HVDM) [33] should be considered. HVDM is supervised, though: it takes the class distribution into account; in our workflow, we do not disclose the class labels. For HEOM, we note that if the epidemiological dataset has many categorical variables with several distinct values, HEOM may "push apart" objects that are similar with respect to the numerical variables.

## 4.4 Quality Assessment

The last step of our workflow is a new approach for the evaluation of subspace clustering algorithms in the absence of a ground truth. This approach is not peculiar to epidemiological mining; it holds for any case of cluster evaluation without ground truth.

Our approach is inspired by [34], Ch. 8, section 8.5.8 on "Assessing the Significance of Cluster Validity Measures", where the quality of a set of clusters is compared to the quality of clusters built over a dataset of random data points in the same set of dimensions. However, we do not generate random data. Rather, we use the original dataset but organize its members randomly into groups, simulating the behaviour of the learning algorithm to a limited extend, as we explain hereafter.

### 4.4.1 Core Idea for Models-of-Randomness

Our core idea is to compare the models learned by the algorithms to "models-of-randomness". We define as *model-of-randomness* a set of $k$ clusters $\zeta = \{Z_i, i = 1 \dots k\}$ over over the dataset $X$ and feature space $F$, built in such a way that the assignment of objects to clusters is random. For each true model $\xi$ generated by a clustering algorithm, we generate $N$ models-of-randomness (in our experiments, $N = 500$), compute the quality of all models-of-randomness and build a histogram of the observed quality values. These are the quality values likely to be observed on these data through random assignment of objects to clusters. If the quality achieved by $\xi$ is better than the values in the histogram, then we have an indicator that $\xi$ has found structure in the data. If the quality of $\xi$ is within the histogram, then $\xi$ is no better than a random structure.

The algorithms DUSC, PROCLUS and RIS are very dissimilar: DUSC builds overlapping clusters, PROCLUS does not. Hence, a model-of-randomness that partitions the data is fair for PROCLUS but not for DUSC. Thus, we build for each of DUSC, PROCLUS and RIS (a set of $N$) dedicated models-of-randomness with comparable behaviour.

### 4.4.2 Algorithm-Specific Models-of-Randomness

Let $\xi = \{Y_i | i = 1, \ldots, k\}$ be the set of $k$ subspace clusters over the dataset $X$ and feature space $F$, as returned by one of the algorithms invoked in the previous workflow step. We build a set $G(\xi)$ of $N$ algorithm-specific "models-of-randomness" for $\xi$ as follows.

**Models-of-Randomness for DUSC.** DUSC builds a set of clusters that may overlap partially; each cluster is in a different subspace. Let $S_i$ be the subspace for cluster $Y_i$ of the model $\xi$ built by DUSC. We create a set $Z_i$ that has the same cardinality as $Y_i$ but contains randomly selected objects from $X$. Then, we project away all but the $S_i$ dimensions of $Y_i$, so that $Z_i$ in the same subspace as $Y_i$. Note that we randomly select the objects *with replacement*, since the clusters in DUSC may overlap. We repeat this process for all clusters in $\xi$, building a model-of-randomness $\zeta = \{Z_i | i = 1, \ldots, |\xi|\}$ and denote it as $MRND_{DUSC}(\xi)$. We generate $N$ such models.

**Models-of-Randomness for PROCLUS.** PROCLUS takes as input two parameters, the number of clusters $k$ and the average number of dimensions per cluster $l$, cf. [15]. To build a model-of-randomness *zeta* for a model $\xi$ of PROCLUS, we first randomly select $k$ subsets of variables $R_i \subset F, i = 1 \ldots k$, such that $\sum_{i=1}^{k} |R_i| = k * l$; these subsets may overlap. Then, we partition $X$ into $k$ sets of objects $Z_i, i = 1, \ldots, k$ by random selection *without replacement*, since the clusters built by PROCLUS do not overlap. For $Z_i$ we retain only the dimensions in the subspace $R_i$ (of the same index $i$). We denote model $\zeta = \{Z_i | i = 1 \ldots k\}$ as $MRND_{PROCLUS}(k, l)$. We generate $N$ such models.

It is noted that the function $MRND_{PROCLUS}(\xi, k, l)$ takes as input the two input parameters of PROCLUS, but not the model $\xi$, since all information needed to build $MRND_{PROCLUS}()$ is encapsulated in these two parameters.

**Models-of-Randomness for RIS.** RIS returns a set of ranked subspaces only. To build a model $\xi$ for RIS, we invoke DBSCAN in each of the top-M subspaces (for an input value M) returned by RIS. Let $S$ be such a subspace and let $k_S$ be the number of clusters returned by DBSCAN in it. We partition $X$ into $k$ sets of objects $Z_i, i = 1, \ldots, k_S$ by random selection *without replacement* (since DBSCAN does not return overlapping clusters). Thus, for RIS+DBSCAN we acquire a model-of-randomness *per subspace.* We denote it as $MRND_{RIS}(S, k_S)$ per subspace. We generate $N$ such models per subspace.

**Complete Generation Process for Models-of-Randomness.** The complete generation process goes as follows:

1. For a model $\xi$ of the algorithm under study, we generate a set of $N$ models-of-randomness $G = \{\zeta_1, \ldots, \zeta_N\}$ as described above.
2. If the clusters in $\xi$ do not cover all objects in $X$, i.e. if the clustering algorithm has identified noise points and skipped them (as in [26]), then we randomly choose and remove $n$ elements from the clusters in $\zeta_i$ (for $i = 1, \ldots, N$), where $n$ is the cardinality of the set difference $\cup_{Y \in \xi} Y \setminus \cup_{C \in \zeta_i} C$.
3. We compute the histogram $hist(G, q)$ of quality values of the models in $G$ using the quality function $q()$.

### 4.4.3 Quality Functions

For quality evaluation, we consider both internal and external indices, across the guidelines of [34]. For evaluation against a ground truth (external index) we use the F-measure $F_1 = 2 \cdot \frac{precision \cdot recall}{precision + recall}$, with $precision = \frac{TP}{TP+FP}$ and $recall = \frac{TP}{TP+FN}$. For SHIP2·578, we consider the class A as negative (N) and the classes B and C taken together as positive (P), deriving the numbers of true positives (TP), false positives (FP) etc accordingly. To associate each cluster with a class we use majority voting, i.e. we assign each cluster to the class of the majority of its elements.

For internal quality evaluation, we adjust the silhouette coefficient measure, as described in [34] for clusters over the whole set of dimensions. This measure captures compactness within each cluster and separability among clusters. Given a subspace $S$, let $a(o)$ be the average distance between object $o$ and all other objects in its cluster and $b(o)$ be the minimum of the average distances from $o$ to all clusters not containing $o$. The smaller the value of $a(o)$ is, the more close $o$ is to all other objects of its cluster. The higher the value of $b(o)$ is, the more well-separated $o$ is from other clusters. Then, the silhouette coefficient of $o$ with respect to $S$ is defined as $Silh_S(o) = \frac{b(o) - a(o)}{\max(a(o), b(o))}$. The value of the silhouette coefficient is between -1 and 1, higher values are better. The silhouette coefficient of a clustering is the average of the silhouette coefficient of all data objects involved in the clustering, i.e. excluding noise points if any.

## 5 Experiments and Findings on SHIP2·578

We apply our workflow on SHIP2·578, using our own MATLAB implementations of DUSC, PROCLUS and RIS. We initially experimented also with the subsets of female and of male participants (cf. Table 2). However, in most of the experiments, the quality of the models did not vary much. Hence, we only report on the findings for the complete dataset.

Unless otherwise specified, we disclose to the algorithm the variable mrt_liverfat_s2, i.e. the fat concentration in the liver (cf. subsection 3.2).

## 5.1 Preparatory Steps and Parameter Tuning

According to the first step of our workflow, we normalize all variables to the interval $[0, 1]$. For the treatment of missing values, we opt for the *REPLACEMENT* strategy (cf. subsection 4.1), i.e. we replace missing values of numerical variables with the mean, and missing values of nominal variables with the mode of the distribution. In the following, we denote the cardinality of the dataset as $\mathcal{N}$ and the number of dimensions as $d$.

**Parameter settings for DUSC:** We focus on small subspaces to alleviate the side-effects of the curse of dimensionality on the similarity between the objects in the clusters of each subspace. Hence we experiment with low values for the global density threshold $\mathcal{F}$. We conducted preliminary runs for values between 0.1 and 2 and set $\mathcal{F}$ to 0.7 for the complete dataset, and to 1.5 for the subsets of female and of male participants. Similarly to RIS, we set the minimum size of a cluster *minSize* to 6, which is equal to $round(ln(\mathcal{F}))$ and approx. 1 % of the total dataset size. For the parameter $\epsilon$, which controls the kernel density [31], we empirically set it to 0.02. Parameter $\eta$ guarantees that very sparse regions are not considered as dense; we set it to 2, as suggested in [31]. For both $\mathcal{F}$ and $\epsilon$, we tune the parameter values such that for the cardinality *dim* of a created subspace $S$ it holds that it lies between 1 and half of the feature space $\mathcal{F}$, i.e. $1 < dim(S) < 0.5 \cdot d$.

Finally, parameter $r$ influences the pruning of redundant subspaces: given a subspace cluster $C' = (O', S')$ if there exists another subspace cluster $C = (O, S)$ with $O' \subseteq O \wedge S \subset S', |O'| > r \cdot |O|$, then we prune $C$. We set $r$ to 0.2; this is admittedly very strict, but considerably reduces complexity in our experiments.

**Parameter settings for PROCLUS:** The main parameters input to PROCLUS are the number of projected clusters $k$ and the average dimensionality $l$. For $k$, we consider $\{2, 3, \ldots, 15\}$; for $l$, we employ $\{3, 5, \ldots, 15\}$. For each run, PROCLUS stops after 10 iterations. We set the parameters $A$ and $B$, which control the complexity of the initialization phase, to 10 and 5, respectively. Parameter *minDeviation*, needed to identify too small clusters, is set to 0.1 which means that a cluster medoid is declared as *bad*, if the cluster contains less than $\frac{\mathcal{N}}{k} \cdot minDeviation$ objects.

**Parameter settings for RIS:** RIS uses the concept of density introduced in DBSCAN [26]. To set the density threshold *minPts* and the neighborhood radius $\epsilon$ for DBSCAN, we adapt the heuristic of [30], and set $minPts = \text{round}(\ln(\mathcal{N}))$. Also, we utilize the upper bound heuristic proposed in [30] to set $\epsilon$ for a $\mathcal{N} \times d$-dimensional dataset as $\epsilon^{\gamma}_{minPts} = \frac{1}{2} \cdot \gamma \cdot \sqrt[d]{\frac{minPts}{\mathcal{N}}}$ with $\gamma \in \{0.05, 0.06, \ldots, 0.1\}$.

We prune subspaces with fewer core objects than 90% of the dataset's size. From the ranked list of subspaces returned by RIS, we select the top-$M$, for $M = 6$. Then, we apply DBSCAN with the same $minPts$ and $\epsilon$ to obtain a clustering.

### 5.2 Setting Up the Models-of-Randomness

We create models-of-randomness for each of DUSC, PROCLUS and RIS, as described in subsubsection 4.4.2. We set the number of models $N = 500$, i.e. we generate 500 models-of-randomness for each model of DUSC and of PROCLUS.

For RIS, we set $M = 6$, i.e. we consider the top-6 subspaces, as already mentioned at the end of Section 5.1. We invoke DBSCAN for clustering in each subspace, as explained in the corresponding part of subsubsection 4.4.2.

When we invoked PROCLUS in our preliminary experiments, we found that it returns non-unique results for a given setting of the parameters $k, l$. A possible explanation is that the average number of dimensions per subspace, i.e. the parameter $l$, can be realized in different ways. For example, if the total number of dimensions is 6 and if the parameter settings are $l = 3$ and $k = 2$, then RIS has several ways of building two clusters with an average of 3 dimensions. Therefore, we first select the best runs, according to our quality measures (cf. 4.4.3) and identify the best values for the parameters $k, l$. We fix $k, l$ to these values and invoke *both* PROCLUS and $MRND_{PROCLUS}(k, l)$ $N = 500$ times. We thus generate the histogram of quality values for both PROCLUS and for its models-of-randomness, and we juxtapose the average quality value of the PROCLUS histogram with the values in the histogram of $MRND_{PROCLUS}(k, l)$.

### 5.3 Results for DUSC

In Table 3, we show the 13 subspace clusters produced by DUSC on SHIP2·578. The largest subspace has three dimensions, 3 clusters are in one-dimensional spaces. All subspaces come from a set of four variables: alkligt_s2, crea_s_s2, ggt_s_s2 and sleeph_s2. The cluster sizes vary from 6 ($\approx$ 1 %) to 533 members ($\approx$ 92 %). The minimum silhouette score is 0.40 (Cluster #9), the maximum is 0.97 (Cluster #1). These scores are much higher than those of the models-of-randomness, where mean and median are between -0.08 and -0.01. Hence, DUSC discovers cluster structure in the data. However, no subspace cluster achieves an F1 score above 0, i.e. the clusters do not separate between classes.

In Figure 4, we depict clusters #11, #12, #13 in their joint 4-dimensional space, as well as 12 noise points: we draw three-dimensional scatterplots to highlight the relationship among each three dimensions.We see that no cluster is very compact and that each cluster contains objects that are close to objects of the other clusters. Noise points are not very far from the cluster members either. Hence, albeit the silhouette scores of all clusters are higher than in the model-of-randomness, the clusters do not yield much insight on the similarities and differences among their members.

| Cluster No. | #1 | #2 | #3 | #4 | ... | #11 | #12 | #13 |
|---|---|---|---|---|---|---|---|---|
| Subspace | alkligt_s2 | crea_s_s2 | ggt_s_s2 | sleeph_s2 | ... | alkligt_s2 | alkligt_s2 | alkligt_s2 |
| | - | - | - | crea_s_s2 | ... | sleeph_s2 | sleeph_s2 | sleeph_s2 |
| | - | - | - | ggt_s_s2 | ... | crea_s_s2 | crea_s_s2 | crea_s_s2 |
| | - | - | - | - | ... | ggt_s_s2 | ggt_s_s2 | ggt_s_s2 |
| Cardinality of $C$ | 84 | 403 | 317 | 197 | ... | 533 | 27 | 6 |
| Silhouette | 0.97 | 0.70 | 0.86 | 0.58 | ... | 0.60 | 0.42 | 0.50 |
| F1 | 0 | 0 | 0 | 0 | ... | 0 | 0 | 0 |
| $MRND_{DUSC}$: | | | | | | | | |
| Mean of Silhouette | -0.04 | -0.05 | -0.05 | -0.02 | ... | -0.02 | -0.02 | -0.01 |
| Median of Silhouette | -0.02 | -0.08 | -0.08 | -0.04 | ... | -0.02 | -0.01 | -0.01 |
| (Mean, Median) of F1 | (0,0) | (0,0) | (0,0) | | ... | (0,0) | (0,0) | (0,0) |

**Table 3**: Excerpt of the clustering obtained by DUSC on SHIP2·578 – subspace variables, cluster sizes, silhouette & F1 scores, and (in the last four rows) performance of $MRND_{DUSC}$

On the SHIP2·578 subset of female participants, DUSC returned 16 subspace clusters (results without figures), among them 7 four-dimensional ones. As for the complete dataset, the variables sleeph_s2 and ggt_s_s2 were also part of these subspaces. However, DUSC also included the variable menopaus_s2 (age at onset of menopause), as well as the fat liver concentration variable mrt_liverfat_s2 in the subspace. Hence, DUSC did manage to exploit a gender-associated variable. The liver fat concentration variable did not contribute to class separation though: the F1 score is as low as for the models-of-randomness. The silhouette scores are also low, though still higher than in the models-of-randomness.

On the SHIP2·578 of male participants (results without figures), DUSC returned 8 clusters with at most three dimensions (in 6 of the 8 clusters). As for the complete dataset, the variables crea_s_s2, ggt_s_s2 and sleeph_s2 constitute a subspace, but not the variable ggt_s_s2 alone. The silhouette scores are higher than for the subset of female participants, while the F1 score is again as low as for the models-of-randomness.

## 5.4 Results for PROCLUS

As pointed out in subsection 5.2, PROCLUS runs in a non-deterministic way, so we performed several experiments to identify the best parameter configuration: the best silhouette score value (0.56) is achieved for $k^{opt} = 2$ and $l^{opt} = 3$, i.e. two clusters in a subspace with three dimensions *on average*. The histograms of the silhouette scores for these parameter settings is depicted on the upper part of Figure 5: PROCLUS at the left-hand side, $MRND_{PROCLUS}(k^{opt}, l^{opt})$ at the right-hand side.

In the PROCLUS histogram on the left upper part of Figure 5, the median is between 0.25 and 0.5, i.e. lower than the best observed silhouette score of 0.56. Still, the histogram is shifted to the right in comparison to the histogram of the models-of-randomness on the right upper part of Figure 5, where all values are between -0.25 and 0.25 (arithmetic mean: 0.003, median: 0.004). Hence, PROCLUS does find a clustering structure in the subspace. The F1-score for $k^{opt}$ and $l^{opt}$ is 0 though (not shown), indicating that the two clusters do not reflect the classes in the data. It is noted that PROCLUS exploited the fat liver concentration variable mrt_liverfat_s2

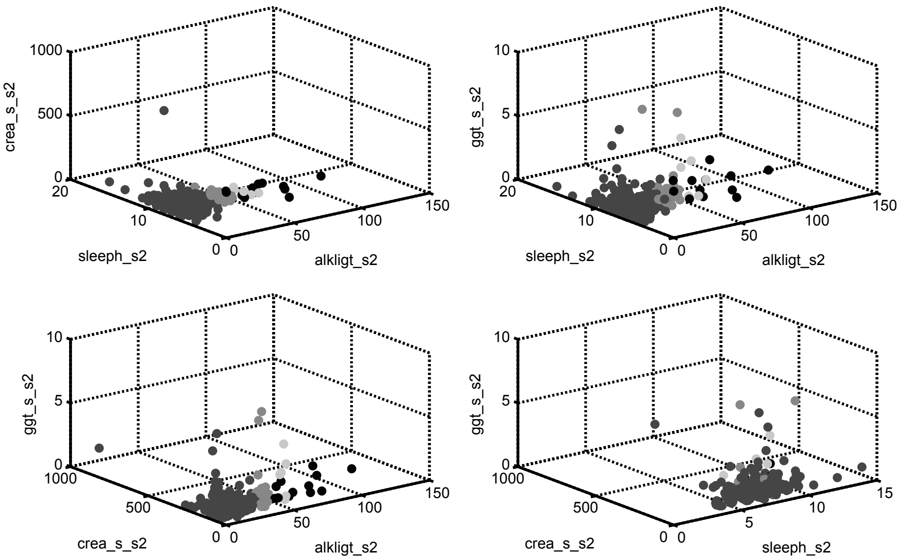

**Figure 4**: Scatterplots for the 4-dimensional clusters #11-13 (cf. Table 3) returned by DUSC on SHIP2·578: the largest cluster (dark grey, low values for alkligt_s2) is rather compact, but cluster members and noise points are close to each other, making interpretation difficult.

when building those clusters, although this variable is in neither of the subspaces. In the lower left part of the figure we show the best silhouette score achieved when hiding this variable; this experiment is described later.

In Table 4, we show a description of the two clusters that lead to the best observed silhouette score on the upper left part of Figure 5. Cluster $C_1$ contains 523 of 578 objects in a two-dimensional projection, $C_2$ the remaining 55 objects in a 4-dimensional projection. $C_1$ is in a two-dimensional subspace, $C_2$ in a four-dimensional one, whereby the average subspace size is indeed 3, as dictated by $l^{opt}$. The mean and standard deviation values of each variable inside and outside a cluster are normalized; we skip the term "normalized" hereafter when referring to the values of the variables in these two clusters.

The variables of the subspace of cluster $C_1$ have a lower standard deviation inside $C_1$ than outside it, and their mean values are shifted to the left, i.e. $C_1$ contains participants with lower crea_s_s2 and ggt_s_s2 values on average. However, the standard deviation outside $C_1$ is larger than the mean for both variables, implying that lower values can also be seen in $C_2$. The scatterplot in Figure 6 verifies this observation, by showing the values of crea_s_s2 and ggt_s_s2 in $C_1$ (in black) and in $C_2$ (in grey); despite the shift of the crea_s_s2 values in $C_1$, the overlap is substantial. Essentially, $C_2$ is a subcluster of $C_1$.

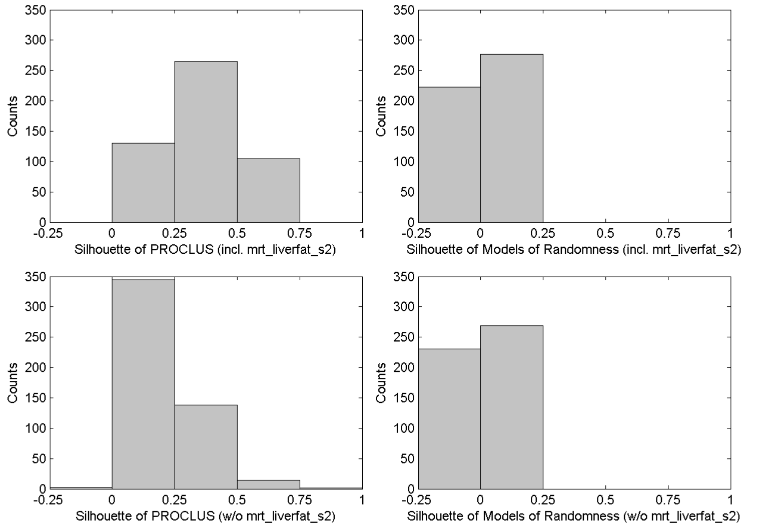

**Figure 5**: Histogram of silhouette scores (over 500 runs) for PROCLUS (left-hand side) and corresponding models-of-randomness (right-hand side). The models depicted in the upper part of the figure exploited the predictive variable mrt_liverfat_s2. When mrt_liverfat_s2 is disclosed, PROCLUS achieves a silhouette score of 0.56 for $k = 2$ clusters and $l = 3$ average number of dimensions (upper part). When this variable is hidden, PROCLUS achieves an even higher silhouette score of 0.65 – this time for $k = 2, l = 15$ (lower part) – but this value is an outlier: as can be seen in the lower part of the figure, the average cluster quality of PROCLUS is the same as for the models-of-randomness.

Indeed, cluster $C_2$ is less compact than $C_1$, as we see in Table 4: for two of the four variables constituting the subspace of $C_2$, namely for tsh_s2 and hba1c_s2, the mean values are very close and the standard deviation values are so large that an overlap of the clusters is certain. The mean of asat_s_s2 is larger inside $C_2$ than outside it, but the two mean values are less than one third of the standard deviation (almost the same inside and outside $C_2$) away from each other. The overlap of values for crea_s_s2 is evident in Figure 6. Since the values of all $C_2$ dimensions overlap inside and outside $C_2$, we conclude that $C_2$ is essential an area inside $C_1$ that is less dense than the rest of $C_1$.

In the next experiment with PROCLUS, we hide mrt_liverfat_s2. The best silhouette score for this set of variables is 0.65, achieved for $k = 2, l = 15$. The histograms for PROCLUS and for the models-of-randomness with the same $k, l$ settings are in the left lower part, respectively right lower part of the figure. Differently from the histograms in the upper part of the figure (where mrt_liverfat_s2 is disclosed), the mean and median silhouette scores of PROCLUS are as for the models-of-randomness; the best score (0.65) is an outlier. Hence, the liver fat concentration variable is exploited by PROCLUS.

| Cluster $C_i$ | $\lvert C_i \rvert$ | Variable $V$ | $\mu(V_{C_i})$ | $\sigma(V_{C_i})$ | $\mu(V_{C \setminus C_i})$ | $\sigma(V_{C \setminus C_i})$ |
|---|---|---|---|---|---|---|
| 1 | 523 | crea_s_s2 | 0.0617 | 0.0239 | 0.1103 | 0.1295 |
| | | ggt_s_s2 | 0.0580 | 0.0788 | 0.1512 | 0.1658 |
| 2 | 55 | tsh_s2 | 0.1048 | 0.0513 | 0.1103 | 0.0860 |
| | | hba1c_s2 | 0.3348 | 0.0528 | 0.3344 | 0.0839 |
| | | asat_s_s2 | 0.1620 | 0.0745 | 0.1329 | 0.0767 |
| | | crea_s_s2 | 0.1103 | 0.1295 | 0.0617 | 0.0239 |

**Table 4**: Clustering result for PROCLUS with $k$=2 and $l$=3 for on SHIP2·578, depicting each variable $V$ that contributes to each cluster, the variable's normalized mean $\mu(\cdot)$ and standard deviation $\sigma(\cdot)$ inside and outside each cluster

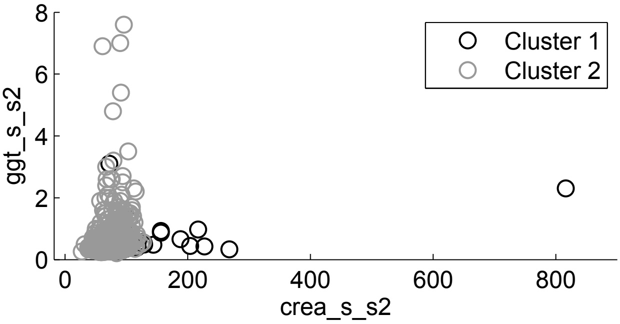

Figure 6: Scatterplot of PROCLUS for $k^{opt} = 2$, $l^{opt} = 3$, across the variables crea_s_s2 and ggt_s_s2, i.e. the subspace of $C_1$; crea_s_s2 is also in the subspace of $C_2$: the crea_s_s2 values in $C_2$ are more shifted to the right than in $C_1$, but the overlap between the two clusters is substantial.

The mean and median F1-scores for PROCLUS have been consistently equal to zero (and thus equal to the values for the models-of-randomness), independently of the $k, l$ values. We observed an F1-score of 0.45 for $k = 3$, $l = 15$, but this value appeared only once; the corresponding PROCLUS histogram over 500 runs had again a mean and a median equal to zero. Thus, we conclude that no class separation is achieved.

The silhouette scores of the clusters are influenced by the values of $k$ and $l$, as we see in Figure 7: we vary the number of clusters $k$, and, for each value we increase the average number of dimensions $l$ and depict the line of the silhouette values. We observe that the parameter $l$ has a large impact: for small $l$ values, the silhouette values are widely scattered; as $l$ increases, the silhouette lines coerce.

### 5.5 Results for RIS

In Table 5 we show the six best subspaces returned by RIS, ordered by position in the ranked list (i.e. subspace #1 is the best one). The first rows of the table describe these subspaces, while the last four rows describe the quality of the models-of-randomness.

Each subspace returned by RIS consists of two variables only, which we depict in the second row of Table 5. We observe that one of the variables, crea_s_s2 (serum

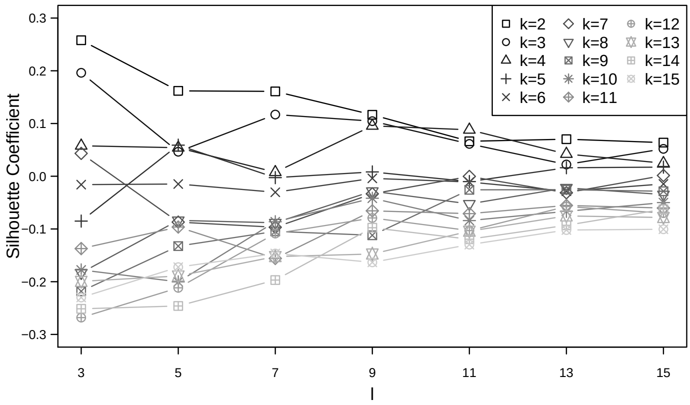

Figure 7: PROCLUS' silhouette coefficient values for different parameter settings over 5 runs; values close to 1 are best. All line graphs converge towards 0 as the average number of dimensions increases, indicating that the curse of dimensionality affects the quality of PROCLUS, even for as few as 15 dimensions.

| Subspace No. | #1 | #2 | #3 | #4 | #5 | #6 |
|---|---|---|---|---|---|---|
| Subspace Variables | sleeph_s2<br>crea_s_s2 | crea_s_s2<br>tsh_s2 | crea_s_s2<br>ggt_s_s2 | crea_s_s2<br>asat_s_s2 | crea_s_s2<br>lip_s_s2 | quick_s_s2<br>crea_s_s2 |
| Number of Clusters | 7 | 1 | 1 | 2 | 1 | 1 |
| Number of Outliers | 13 | 21 | 26 | 23 | 25 | 28 |
| RIS Quality Score | 703.5 | 688.2 | 678.1 | 676.8 | 671.7 | 669.2 |
| Silhouette | 0.74 | 0.78 | 0.85 | 0.61 | 0.77 | 0.61 |
| F1 Score | 0 | 0 | 0 | 0 | 0 | 0 |
| Mean(Silhouette$_{MRND}$) | -0.02 | -0.06 | -0.06 | -0.04 | -0.05 | -0.04 |
| Median(Silhouette$_{MRND}$) | -0.06 | -0.09 | -0.10 | -0.08 | -0.09 | -0.09 |
| Mean(F1$_{MRND}$) | 0 | 0 | 0 | 0 | 0 | 0 |
| Median(F1$_{MRND}$) | 0 | 0 | 0 | 0 | 0 | 0 |

**Table 5**: The six highest-rated subspaces of RIS for SHIP2·578: for each subspace we show its RIS quality score, contributing variables and basic information on clustering, including the number of clusters, the number of outliers, silhouette and F1 score. The last four rows depict the performance of the models-of-randomness.

creatinine concentration, see Table 1 for all variable names) is contained in each of them; given that the subspaces consists only of two variables, there is not much variety among the subspaces.

The best subspace, #1, consisting of the variables sleeph_s2 and crea_s_s2, has a RIS quality score of 703.5. On this subspace, DBSCAN returns 7 clusters and 13 outliers. The silhouette score of this clustering is 0.74, clearly higher than the models-of-randomness, where the silhouette mean is -0.02 and the median is -0.06. The clusterings in the other subspaces have even higher silhouette scores (cf. Table 5, 5th row).

The clusterings in the subspaces found by RIS have an F1 score of zero, similarly to the models-of-randomness. This is expected for the subspaces #2, #3, #5 and #6, because DBSCAN builds only one cluster in each of them; this cluster is naturally assigned to the majority class. The 7 clusters in subspace #1 and the two clusters in subspace #4 are also characterized by the majority class, though, hence DBSCAN cannot reconstruct the classes in the subspaces found by RIS.

In Figure 8 we use scatterplots to give more insights of the clusters found in each of the 6 subspaces. The "Quality score" mentioned above each subspace is the RIS quality score; we repeat the value, together with the id of the subspace. In each of the figures (marked in different shades of gray, and the outliers).

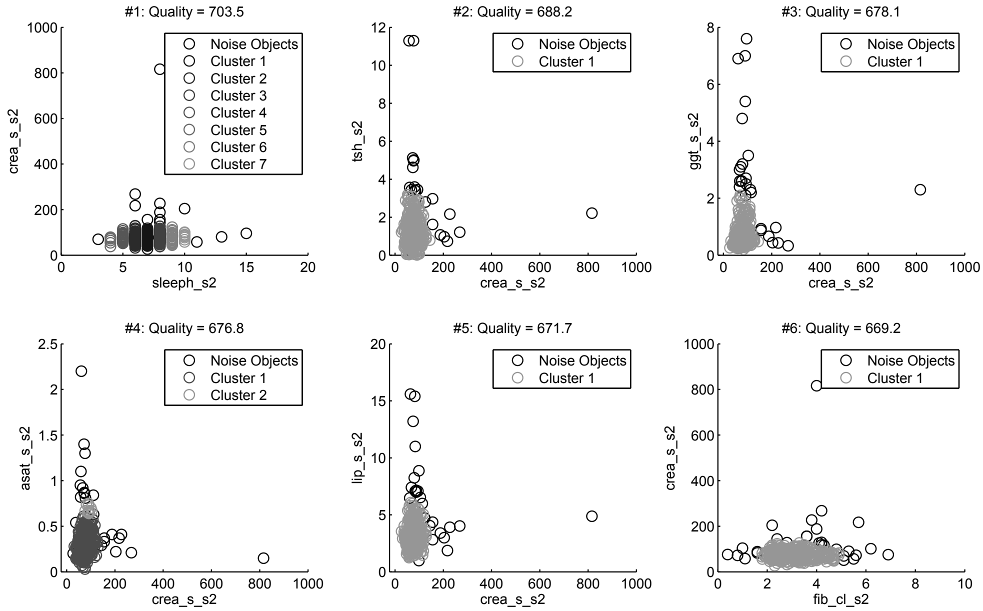

**Figure 8**: Scatterplots for the clusterings in the top-6 subspaces found by RIS in SHIP2·578

## 5.6 Findings, Insights and Restrictions

The models learned by all algorithms are superior in internal quality to the models-of-randomness, implying that all algorithms identified some cluster structure in subspaces of the data. On the negative side, the quality with respect to the ground truth is unsatisfactory for all algorithms: no learned model has achieved a separation between the classes. However, our clusterability assessment step already gave an indication that SHIP2·578 is not easy-to-cluster (cf. 4.2.1, especially Figure 2), and we already know from prior research that class separation in SHIP2·578 is not easy either [10, 11].

It is remarkable that some variables are present in many subspaces produced by different algorithms. Most striking is the variable `crea_s_s2` (serum creatinine, cf. Table 1), which appears in subspaces of all algorithms. High values of this variable indicate kidney dysfunction. Such a dysfunction is a common complication of diabetes mellitus. It is also reported [35] that there is an association between sleep disorders and chronic kidney disease. The variable `sleeph_s2` (sleep hours per night, cf. Table 1) appears together with `crea_s_s2` in some of the subspaces found by DUSC (cf. Table 3), and also constitutes the best subspace found by RIS (cf. Table 5), i.e. the one with the highest RIS quality score. The variable `hba1c_s2` (cf. Table 1, high amounts are associated with diabetes mellitus) appears together with `crea_s_s2` in the PROCLUS cluster $C_2$, Table 4).

These subspaces are not associated with the outcome "hepatic steatosis" which we study. Do they reflect associations between other outcomes and risk factors, though? Next to scrutinizing cluster content (PROCLUS cluster $C_2$, the 7 clusters of DBSCAN in subpace #1 of RIS, the clusters in the DUSC subspaces), it must be investigated whether these subspaces and clusters persist for different parameter settings of the algorithms. This is necessary before stating that such subspaces *might* reflect known variable associations. Ultimately, such associations must be validated in independent studies, of course.

Concerning the potential of subspace clustering algorithms for epidemiological mining in general, our experiments delivered insights on several restrictions that need to be taken into account when using these algorithms or preparing epidemiological data for them.

**Setting the parameters** has been a challenge. For some parameters, it is even unclear what would be a proper range of values: this holds for the global density threshold $F$ required by DUSC, which is less intuitive than e.g. neighbourhood cardinality. Setting the average dimensionality $l$, as required by PROCLUS, is also less intuitive than specifying a constant number of dimensions. Assigning values to non-intuitive parameters requires several experimental runs. Moreover, as stated in [6], such parameters are often set to minimize run time or result size; this does not imply that the cluster structure is good.

Even for conceptually intuitive parameters like neighbourhood cardinality and radius, specifying values has been far from easy. For RIS, we adapted the heuristics proposed in [30], but recognize that this heuristic is brute force and does not necessarily reflect the peculiarities of a specific dataset. For PROCLUS, we faced the additional problem of non-deterministic output, a fact that further impeded the tuning of the input parameters.

The algorithms require more parameters than we mentioned in our experiments. For example, PROCLUS requires values for the cut-off percentage to identify "bad" medoids (PROCLUS uses k-medoids for clustering [15]), as well as two integers $A$ and $B$ that tune the complexity of the random initialization phase. We have used the parameter settings suggested in the literature, but we recognize that different values may have a substantial impact on the results. Ultimately, this problem can only be solved by methods that reduce the number of parameters, as e.g. in [36, 37].

**Exploitation of domain knowledge.** It is often pointed out that parameter tuning requires some knowledge on the application. However, we found it difficult to guide the algorithms with use of our prior knowledge, because there was no match between prior knowledge and algorithmic parameters.

For example, we disclosed to the algorithms the variable mrt_liverfat_s2 that determines the class, yet no algorithm exploited this variable during subspace construction and learning. How to inform the algorithms that this variable is more important than others? The supervised distance measure HVDM [33] would not help in that case, because we did not disclose the discrete target variable, only the mrt_liverfat_s2 from which we derived the class labels. Re-weighting this or other variables in the distance function might have been effective. However, it is not certain that assigning a high weight to some variables would make the algorithm prefer these variables when building subspaces; insider knowledge of the algorithm is necessary to decide that. Algorithms should preferably allow for direct guidance on the importance of some variables.

As another example, the experimentation with PROCLUS revealed that the algorithm can be trapped to favor variables for which most objects have the same value, although those variables do not contribute to well-separated clusters. An explanation is that PROCLUS attempts to build a subspace of variables for which the standard deviation is minimal. A variable that gets the same value for most of the objects does satisfy this objective, although it is not informative with respect to clustering. We know that epidemiological data contain many such variables, e.g. biomarkers, categorical variables on medication intake, variables associated to the presence of a rather rare impairment. How to instruct the algorithm to consider them but not concentrate *exclusively* on them?

**Dimensionality bias.** Algorithms like CLIQUE [14], SUBCLU [30] and RIS use a static density threshold. This is problematic for subspaces of different cardinalities (curse-of-dimensionality). This affects the performance of DUSC, too: DUSC uses a dimensionality-unbiased density threshold to find clusters in subspaces, but it uses a global density threshold to prune irrelevant subspaces during subspace generation [7]. Distance functions that are not sensitive to the number of dimensions seem to be the sole remedy to this problem.

**Cluster redundancy.** Algorithms like DUSC build many overlapping clusters. So, the application expert must decide which clusters are interesting and which object-cluster memberships are important. Algorithms like RESCU [38], OSCLU [39] and STATPC [40] compute not only the quality of a subspace cluster with respect to a certain score but also scores for the complete clustering, e.g. “relevance” [38] or “orthogonality” [39]. Such concepts seem promising for ranking clusters. However, these algorithms seek the best candidate subspaces and the best candidate clusters in them; although they employ heuristics to avoid generating and scanning all subspaces, their execution time is very high. In our preliminary experiments, RESCU failed to produce results for SHIP2·578 within a reasonable time. The advantage of cluster ranking must hence be considered in the light of execution time overhead.

**Mix of numerical and categorical variables.** Subspace clustering algorithms exhibit an almost exclusive preference for the Euclidean distance and often assume that all variables are numeric. Epidemiological data contain many categorical variables, though. Our dataset contains more categorical than numerical variables. At the beginning of our study, we attempted to alleviate the problem by discretizing the numerical variables with a technique proposed in [41], and then run the algorithms with a feature space consisting solely of categorical variables. This was motivated by the fact that there are algorithms like CLICKS [42] and SUBCAD [43], designed for feature spaces with categorical variables only. However, PROCLUS, which we wanted to include in our evaluation, has concentrated on categorical variables with very skewed distribution of values, and produced very unsatisfactory results. Thus, we discarded this approach. Nonetheless, it may be worth pursuing for other algorithms.

A candidate for subspace clustering on a mix of numerical and categorical variables is HSM [44], which extracts subspace clusters by using the density-based mechanism of DUSC for numerical variables and frequent pattern mining for categorical variables. The exploitation of categorical variables could compensate for the inconclusive performance of DUSC on the numerical variables, as we observed it for SHIP2·578. However, our study in [10] indicates that frequent pattern mining is most beneficial when exploiting both numerical and categorical data, hence treating the two data types separately might not bring great advantage. Yet, it might be worth investigating.

The more recent subspace clustering algorithm proposed in [45] uses k-modes clustering and a feature weighting distance measure to cover so-called "heterogeneous datasets", which contain both numerical and categorical variables. However, this algorithm builds "soft" subspaces [45], and might thus exacerbate the problem of too many candidates to be inspected by the human expert.

**Impact of design decisions on algorithm performance.** Some of our design decisions may have influenced the performance of the subspace clustering algorithms. First, the treatment of missing values with the *REPLACEMENT* strategy has possibly prevented the recognition of male and female subpopulations. However, the alternative strategy *MAX DISTANCE* (cf. subsection 4.1) may exacerbate object dissimilarity and lead to more noise points, and it would be incompatible with the constraint that at least 90% of the objects be present in a subspace. Hence, a future step would be to allow for subspaces with less than 50% of the objects and then use the *MAX DISTANCE* strategy. Approaches that generate too many alternatives are impractical for PROCLUS (because of its non-deterministic nature), hence we would consider this option only for RIS and DUSC.

## 6 Conclusion

We presented a workflow that investigates the potential of subspace clustering on epidemiological data, using the example of hepatic steatosis as ground truth outcome for the evaluation and the SHIP2·578 population-based dataset for the experiments.

Our workflow encompasses steps for preparing the data according to the demands of the algorithms and for selecting methods that find interesting subspaces, clusters within a subspace and clusters across subspaces. The core steps of our workflow are *clusterability assessment* and *quality assessment*. In the first core step, we seek for indicators on whether the dataset contains clusters or interesting subspaces, by performing statistics on it and comparing it to public domain datasets. In the second core step, we evaluate the quality of the clusters learnt by the subspace clustering algorithms against the quality of models-of-randomness. We introduce the concept of *model-of-randomness* as a model describing data that do not have a clustering structure at all. This concept is needed to evaluate in the absence of ground truth: albeit epidemiological data have a ground truth with respect to a target variable, they have no ground truth with respect to subspace clusters.

In accordance to our prior knowledge, our workflow has verified that the example epidemiological dataset might contain subspaces but finding cluster structure in it is not easy. This was reflected in the experiments with all algorithms: the internal quality of the clusters found was higher than the quality of models-of-randomness, but the subspaces were very small, some models consisted of only one cluster and there was no class separation.

Our experiments revealed several insights on the potential of subspace clustering for epidemiological datasets. Parameter setting is a major issue, because many parameters are not intuitive, so the expert does not even know what would be a proper *range* for them. The exploitation of available domain knowledge is difficult, because prior knowledge does not translate into the available set of parameters. Some properties of the epidemiological data, namely a mix of categorical and numerical variables, a large set of dimensions and a skewed distribution of many categorical variables (some of them have the same value for most of the cohort participants), seem to disagree with the design of the algorithms. A distance function that is insensitive to the number of dimensions, covers both categorical and numerical variables, and allows for missing values, seems to be key for the success of subspace clustering on epidemiological data. So, we will work next towards extending our distance measures [11] in this direction.

The interplay of internal quality and class separation deserves some discussion in the context of epidemiological mining. In our experiments, none of the subspace clustering algorithms managed to separate the classes, but all of them produced models which, despite the shortcomings identified above, were of higher internal quality than randomness. Färber et al. point out that the identified clusters may mirror meaningful, previously undiscovered subpopulations of the dataset, yet common external evaluation measures discriminate these findings when the algorithms either split classes into multiple clusters or combine two or more classes into a single cluster [46]. Therefore, Färber et al. speak against juxtaposition of classes to clusters, stating that "the already annotated classes are not even interesting in terms of finding new, previously unknown knowledge. And this is, after all, the whole point in performing unsupervised methods in data mining."

In the light of the demands of personalized medicine, it seems reasonable to keep track of groups of variables that appear together in subspaces, and to consider independent validation for them. A further, perhaps even more promising next step is the design of semi-supervised variants of the subspace clustering algorithms for better and transparent exploitation of already secured prior knowledge.

## References

[1] B. Preim, P. Klemm, H. Hauser, K. Hegenscheid, S. Oeltze, K. Toennies, and H. Völzke, *Visualization in Medicine and Life Sciences III*, ch. Visual Analytics of Image-Centric Cohort Studies in Epidemiology. Springer, 2014.

[2] A. D. Hingorani, D. A. van der Windt, R. D. Riley, (...), W. Sauerbrei, D. G. Altman, and H. Hemingway, "Prognosis research strategy (PROGRESS) 4: Stratified medicine research," *BMJ: British Medical Journal*, vol. 346, no. e5793, 2013.

[3] H. Völzke, C. Schmidt, K. Hegenscheid, J. Kühn, F. Bamberg, W. Lieb, H. Kroemer, N. Hosten, and R. Puls, "Population imaging as valuable tool for personalized medicine," *Clin Pharmacol Ther*, vol. 92, no. 4, pp. 422–424, 2012.

[4] H. Völzke, D. Alte, . . . , R. Biffar, U. John, and W. Hoffmann, "Cohort profile: the Study of Health In Pomerania," *International Journal of Epidemiology*, vol. 40, no. 2, pp. 294–307, 2011.

[5] L. Parsons, E. Haque, and H. Liu, "Subspace Clustering for High Dimensional Data: A Review," *ACM SIGKDD Explorations Newsletter*, vol. 6, pp. 90–105, 2004.

[6] K. Sim, V. Gopalkrishnan, A. Zimek, and G. Cong, "A survey on enhanced subspace clustering," *Data mining and knowledge discovery*, vol. 26, pp. 332–397, 2013.

[7] A. Zimek, *Data Clustering: Algorithms and Applications*, ch. Clustering High-Dimensional Data, pp. 201–230. CRC Press, 2013.

[8] C. Zhang and R. L. Kodell, "Subpopulation-specific confidence designation for more informative biomedical classification," *Artificial Intelligence in Medicine*, vol. 58, no. 3, pp. 155–163, 2013.

[9] S. Glaßer, U. Niemann, B. Preim, and M. Spiliopoulou, "Can we Distinguish Between Benign and Malignant Breast Tumors in DCE-MRI by Studying a Tumor's Most Suspect Region Only?," in *26th International Symposium on Computer-Based Medical Systems (CBMS)*, pp. 77–82, 2013.

[10] U. Niemann, H. Völzke, J.-P. Kühn, and M. Spiliopoulou, "Learning and inspecting classification rules from longitudinal epidemiological data to identify predictive features on hepatic steatosis," *Expert Systems with Applications*, vol. 41, pp. 5405–5415, September 2014.

[11] T. Hielscher, M. Spiliopoulou, H. Völzke, and J.-P. Kühn, "Using participant similarity for the classification of epidemiological data on hepatic steatosis," in *Proc. of the 27th IEEE Int. Symposium on Computer-Based Medical Systems (CBMS'14)*, pp. 1–7, IEEE, 2014.

[12] M. A. Hall, "Correlation-based feature selection for discrete and numeric class machine learning," in *Proc. of 17th Int. Conf. on Machine Learning*, pp. 359–366, Morgan Kaufmann, 2000.

[13] P. Klemm, L. Frauenstein, D. Perlich, K. Hegenscheid, H. Völzke, and B. Preim, "Clustering Socio-demographic and Medical Attribute Data in Cohort Studies," in *Bildverarbeitung für die Medizin (BVM)*, pp. 180–185, Springer Berlin Heidelberg, 2014.

[14] R. Agrawal, J. Gehrke, D. Gunopulos, and P. Raghavan, "Automatic subspace clustering of high dimensional data for data mining applications," in *Proceedings of the ACM International Conference on Management of Data (SIGMOD)*, pp. 61–72, 1998.

[15] C. C. Aggarwal, C. Procopiuc, J. L. Wolf, P. S. Yu, and J. S. Park, "Fast Algorithms for Projected Clustering," in *Proceedings of the ACM International Conference on Management of Data (SIGMOD)*, pp. 61–72, 1999.

[16] D. Damian, M. Orešič, E. Verheij, J. Meulman, J. Friedman, A. Adourian, N. Morel, A. Smilde, and J. van der Greef, "Applications of a new subspace clustering algorithm (COSA) in medical systems biology," *Metabolomics*, vol. 3, no. 1, pp. 69–77, 2007.

[17] L. S. Friedman and E. B. Keeffe, *Handbook of Liver Disease.* Library of Congress Cataloging-in-Publication Data, 2011.

[18] A. P. Levene and R. D. Goldin, "The epidemiology, pathogenesis and histopathology of fatty liver disease," *Histopathology*, vol. 61, pp. 141–152, 2012.

[19] S. Bellentani, G. Bedogni, L. Miglioli, and C. Tiribelli, "The epidemiology of fatty liver," *European Journal of Gastroenterology & Hepatology*, vol. 16, pp. 1087–1093, 2004.

[20] G. Bedogni, S. Bellentani, L. Miglioli, F. Masutti, M. Passalacqua, A. Castiglione, and C. Tiribelli, "The Fatty Liver Index: a simple and accurate predictor of hepatic steatosis in the general population," *BMC Gastroenterology*, vol. 6, no. 33, 2006.

[21] X. Yuan, D. Waterworth, J. R. Perry, (...), T. M. Frayling, J. S. Kooner, and V. Mooser, "Impact of fatty liver disease on health care utilization and costs in a general population: A 5-year observation," *Gastroenterology*, vol. 134, no. 1, pp. 85–94, 2008.

[22] H. Völzke, S. Schwarz, S. E. Baumeister, H. Wallaschofski, C. Schwahn, H. J. Grabe, T. Kohlmann, U. John, and M. Dören, "Menopausal status and hepatic steatosis in a general female population," *Gut*, vol. 56, pp. 594–595, 2007.

[23] S. Baumeister, H. Völzke, P. Marschall, U. John, C. Schmidt, and D. Alte, "Impact of fatty liver disease on health care utilization and costs in the general population: a 5-year observation," *Gastroenterology*, vol. 134, pp. 85–94, 2008.

[24] J.-P. Kühn, D. Hernando, B. Mensel, (...), J. Mayerle, N. Hosten, and S. B. Reeder, "Quantitative chemical shift-encoded MRI is an accurate method to quantify hepatic steatosis," *Journal of Magnetic Resonance Imaging*, vol. 39, no. 6, pp. 1494–1501, 2014.

[25] J. Han, M. Kamber, and J. Pei, *Data Mining: Concepts and Techniques Third Edition.* Morgan Kaufmann Publishers, 2012.

[26] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96)*, pp. 226–231, 1996.

[27] R. A. Fisher, "The use of multiple measurements in taxonomic problems," *Annals of eugenics*, vol. 7, no. 2, pp. 179–188, 1936.

[28] V. G. Sigillito, S. P. Wing, L. V. Hutton, and K. B. Baker, "Classification of radar returns from the ionosphere using neural networks," *Johns Hopkins APL Tech. Dig*, vol. 10, pp. 262–266, 1989.

[29] D. Dias, R. Madeo, T. Rocha, H. Biscaro, and S. Peres, "Hand movement recognition for brazilian sign language: A study using distance-based neural networks," in *International Joint Conference on Neural Networks (IJCNN 2009)*, pp. 697–704, 2009.

[30] K. Kailing, H.-P. Kriegel, and P. Kröger, "Density-Connected Subspace Clustering for High-Dimensional Data," in *Proc. SIAM Int. Conf. on Data Mining (SDM'04)*, pp. 246–257, 2004.

[31] I. Assent, R. Krieger, E. Müller, and T. Seidl, "DUSC: Dimensionality Unbiased Subspace Clustering," in *ICDM*, pp. 409–414, 2007.

[32] U. Niemann, "The potential of high-dimensional clustering for subpopulation discovery in epidemiological datasets." Otto-von-Guericke University Magdeburg, Faculty of Computer Science, 2014. Master Thesis.

[33] D. R. Wilson and T. R. Martinez, "Improved heterogeneous distance functions," *J. Artif. Int. Res.*, vol. 6, pp. 1–34, Jan. 1997.

[34] P.-N. Tan, M. Steinbach, and V. Kumar, *Introduction to Data Mining.* Pearson/Addison-Wesley, 2006.

[35] P. J. Hanly and S. B. Ahmed, “Sleep Apnea and the Kidney: is sleep apnea a risk factor for chronic kidney disease?,” *CHEST Journal*, vol. 146, no. 4, pp. 1114–1122, 2014.

[36] J. Zhao, “Subspace clustering with gravitation.,” in *Grundlagen von Datenbanken*, 2010.

[37] J. Zhao and S. Conrad, “Automatic subspace clustering with density function.,” in *DATA*, pp. 63–69, 2012.

[38] E. Müller, I. Assent, S. Günnemann, R. Krieger, and T. Seidl, “Relevant subspace clustering: Mining the most interesting non-redundant concepts in high dimensional data,” in *Ninth IEEE International Conference on Data Mining (ICDM’09)*, pp. 377–386, IEEE, 2009.

[39] S. Günnemann, E. Müller, I. Färber, and T. Seidl, “Detection of orthogonal concepts in subspaces of high dimensional data,” in *Proceedings of the 18th ACM conference on Information and knowledge management*, pp. 1317–1326, ACM, 2009.

[40] G. Moise and J. Sander, “Finding non-redundant, statistically significant regions in high dimensional data: a novel approach to projected and subspace clustering,” in *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining: ACM*, pp. 533–541, 2008.

[41] U. Fayyad and K. Irani, “Multi-interval discretization of continuous-valued attributes for classification learning,” in *Proc. of 17th Int. Conf. on Machine Learning*, pp. 1022–1029, Morgan Kaufmann, 1993.

[42] M. J. Zaki, M. Peters, I. Assent, and T. Seidl, “Clicks: An effective algorithm for mining subspace clusters in categorical datasets,” *Data & Knowledge Engineering*, vol. 60, no. 1, pp. 51–70, 2007.

[43] G. Gan and J. Wu, “Subspace clustering for high dimensional categorical data,” *ACM SIGKDD Explorations Newsletter*, vol. 6, no. 2, pp. 87–94, 2004.

[44] E. Müller, I. Assent, and T. Seidl, “HSM: Heterogeneous subspace mining in high dimensional data,” in *Scientific and Statistical Database Management*, pp. 497–516, Springer, 2009.

[45] F. Cao, J. Liang, D. Li, and X. Zhao, “A weighting k-modes algorithm for subspace clustering of categorical data,” *Neurocomputing*, vol. 108, pp. 23–30, 2013.

[46] I. Färber, S. Günnemann, H.-P. Kriegel, P. Kröger, E. Müller, E. Schubert, T. Seidl, and A. Zimek, “On using class-labels in evaluation of clusterings,” in *MultiClust: 1st International Workshop on Discovering, Summarizing and Using Multiple Clusterings Held in Conjunction with KDD*, 2010.

*Received August, 2014*