Published in final edited form as:

Priv Stat Databases. 2018 January ; ?: . doi:10.1007/978-3-319-99771-1_13.

Grouping of variables to facilitate statistical disclosure limitation methods in multivariate data sets

Anna Oganian[1], **Ionut Iacob**[2], **Goran Lesaja**[2]

Anna Oganian: aoganyan@cdc.gov; Ionut Iacob: ieiacob@georgiasouthern.edu; Goran Lesaja: goran@georgiasouthern.edu

[1]National Center for Health Statistics 3311 Toledo Rd Hyattsville, MD, 20782, U.S.A.

[2]Georgia Southern University Department of Mathematical Sciences P.O. Box 8093, Statesboro, GA 30460-8093, U.S.A.

Abstract

Data sets that are subject to Statistical Disclosure Limitation (SDL) often have many variables of different types that need to be altered for disclosure limitation. To produce a good quality public data set, the data protector needs to account for the relationships between the variables. Hence, ideally SDL methods should not be univariate, that is, treating each variable independently of others, but multivariate, handling many variables at the same time. However, if a data set has many variables, as most government survey data do, the task of developing and implementing a multivariate approach for SDL becomes difficult. In this paper we propose a pre-masking data processing procedure which consists of clustering the variables of high dimensional data sets, so that different groups of variables can be masked independently, thus reducing the complexity of SDL. We consider different hierarchical clustering methods, including our version of hierarchical clustering algorithm, that we call *K-Link*, and outline how the data protector can define an appropriate number of clusters for these methods. We implemented and applied these methods to two genuine multivariate data sets. The results of the experiments show that *K-Link* has a potential to solve this problem efficiently. The success of the method, however, depends on the correlation structure of the data. For the data sets where most of the variables are correlated, clustering of variables and subsequent independent application of SDL methods to different clusters may lead to attenuated correlation in the masked data, even for efficient clustering methods. Thereby, the proposed approach is a trade-off between the computational complexity of multivariate SDL methods and data utility loss due to independent treatment of different clusters by SDL methods.

Keywords and phrases: Statistical disclosure limitation (SDL), hierarchical clustering, dimensionality reduction.

1 Introduction

Data sets that are released to the public by the data collecting organizations often contain many variables of different types. For example, U.S. government surveys such as the National Health Interview Survey, the Behavioral Risk Factor Surveillance System, the Current Population Survey and American Community Survey are high dimensional. Data collecting organizations have an obligation by law to protect the privacy and confidentiality of responses provided by individuals or enterprises. This is usually accomplished by altering —we use the term *masking*—the original data before release, for example, by aggregating categorical values, swapping data values for selected records, or adding noise to numerical values. See [10, 11] for more details. These methods limit disclosure risk by reducing the information available to intruders attempting to identify individuals in the released data. Data can also be synthesized, however, to do so one needs to come up with a good data generation model which is a complex task. As the dimensionality of the data increases, model estimation becomes more and more difficult. In case of the big governmental surveys mentioned above, model estimation can become extremely difficult and time consuming. Finding the best strategy for joint masking of many variables at a time is not a straightforward task either. Whatever approach for SDL is chosen, the organizations that disseminate the data strive to release data products with high utility - a goal competing with confidentiality protection, because any data alteration done to thwart identification will negatively impact at least some statistical properties of the data.

On the other hand, independent masking of different variables, or univariate masking in other words, may lead to attenuation of correlation structure of the data, particularly for those variables which receive SDL with higher intensity. At the same time for independent variables in the original data, univariate and joint masking are essentially the same from the utility perspective.

In this paper we propose a pre-masking procedure of clustering the variables into groups with the objective of increasing the separation between the groups as much as possible. Separation is viewed in terms of how related the variables in different groups are and we want to make the variables in different groups as unrelated as possible, so that SDL can be applied independently to different clusters with minimal loss of data utility.

1.1 Contribution and plan of the paper

The main contribution of the paper is a pre-masking procedure of clustering variables in the data set that can help government agencies reduce the complexity of SDL methods. In Section 2 we describe our clustering approach. We propose a variant of hierarchical clustering method, that we call *K-Link*, which can serve this purpose. In Section 3 we present numerical experiments with genuine data sets. Our results show that *K-Link* compares favorably to other hierarchical clustering methods. Concluding remarks are given in Section 4.

2 Clustering of variables for disclosure limitation

2.1 Measures of proximity

In order to design any clustering procedure, first it is necessary to define measures of proximity of the objects being clustered. In case of clustering variables these are the measures of similarity/dissimilarity between the variables which are often based on some form of correlation ([18],[5],[2]). different types of variables require different metrics. For quantitative variables some function of the correlation coefficients is used while for categorical variables many association measures exists, such as χ^2, Jaccard, Rand and others. When there are both types of variables in the data, a metric that can be computed for both types of variables is necessary. Similar to [2] we will use squared canonical correlation as such a metric. It can be computed as a first eigenvalue of the product $X'YY'X$ for two data matrices $X_{n \times d1}$ and $Y_{n \times d2}$ for which $min(n, d_1, d_2) = d_1$. As shown in [2] this metric is equivalent to a squared Pearson correlation for two quantitative variables. In the case of one quantitative X and one categorical variable Y, it is a correlation ratio which takes values in the interval from 0 to 1 and is defined as follows:

$$\eta^2 = \frac{\sum_{c \in Cat_y} (\bar{x}_c - \bar{x})^2}{\sum_{i=1}^{n} (x_i - \bar{x})^2} \qquad (2.1)$$

where Cat_y is the set of categories of Y, n_c is the number of observations in category c, $\bar{x}_c$ is the mean value of X computed on the observations belonging to category c, $\bar{x}$ is the mean of X, n - total number of observations.

For the case of two categorical variables squared canonical correlation does not correspond to any well known association measure, but nevertheless, it can be interpreted geometrically according to [2]: the closer to one it is, the closer are the two linear subspaces spanned by the matrices representing these categorical variables, which means that the two qualitative variables bring similar information.

Hence, the dissimilarity matrix of variables is created as a lower triangular matrix DM with elements $DM[i, j] = 1 - r[i, j]$ where $r[i, j]$ is a squared canonical correlation between variables x_i and x_j.

2.2 Clustering methods

Once the dissimilarity matrix is established a clustering method that fits our goals can be chosen. In [2] a hierarchical clustering method suited for clustering variables based on homogeneity criteria is proposed. In the sequel we will refer to this method as *Homclust*. Homogeneity of a cluster is calculated as the sum of squared canonical correlations between each variable of the cluster and the central synthetic variable of that cluster. Such a synthetic variable of the cluster is defined as a quantitative variable “most linked” to other variables in the cluster and computed as the first principal component of the variables in the cluster. The goal of this method is to produce the most homogeneous clusters, so that the variables within the cluster are strongly related to each other. However, in our case when the purpose of clustering the variables is to find groups of variables to which SDL can be applied independently with minimal loss of correlation in the masked data, the objective will be different: the variables in different groups should be as uncorrelated as possible, so that independent masking of different groups of variables would not lead to significant correlation loss comparative to joint masking of all the variables at the same time. On the other hand, from a utility prospective it is not problematic if some, but not all, variables in the same group have little association. Indeed, if the multivariate SDL method preserves correlation structure, application of such method to the cluster of variables in the original data will produce masked cluster with similar associations, strong or week. Thereby, our goal is not necessarily to produce homogenous clusters, that is, clusters with highly correlated variables, but to maximize the separation between the clusters. Because the method described in [2] is focused on the homogeneity and not the separation, the clusters in the resulting partition may not be very far apart, so the variables assigned to different clusters may still be highly correlated. This will be demonstrated in Section 3. Hence, the methods based on the dissimilarities between the variables in different clusters may better suite our goals then the methods based on the proximities within the clusters such as *Homclust* or k-means. We will call such methods tentatively "separation clustering methods".

One type of such methods is divisive or "top down" hierarchical clustering. This approach starts with all the objects in one cluster and at each subsequent step, the largest available cluster is split into two clusters until finally all clusters comprise of single objects. One of the well known divisive methods is *Diana*(DIvisive ANAlysis) [12] implemented in R [3] as well as in other packages. *Diana* starts from finding a data point that has the highest average dissimilarity to all other objects. This object initiates a new cluster, that is called a splinter group. Then remaining objects are assigned either to the splinter group or to the complementary group based on average distance to the objects in these groups. Splitting clusters continues until all the objects end up in different clusters.

Contrary to divisive clustering, the agglomerative hierarchical approach starts with every object being a separate cluster and at each iteration the closest clusters are merged together building clustering hierarchy until all the objects end up in the same cluster. Some simple examples of such methods are *Single-Link*, *Complete-Link*, *Average*(see [6] and references therein). These algorithms differ in the way how they define distance between clusters. For example, for *Single-Link* it's a distance between the two closest objects in the respective clusters. Distances between the clusters increase from iteration to iteration, so for *Single-Link* the separation is guaranteed to increase as one goes up the dendrogram which is a tree diagram illustrating the arrangement of the clusters produced by hierarchical clustering. For *Complete-Link* method the distance between clusters is defined as the distance between the two farthest points in the respective clusters. In some sense *Complete-Link* method is complementary to the *Single-Link* method. For *Average* method it is measured as an average of pairwise distances of all points between two clusters.

The difference in definition of distances between the clusters may have significant effect on the form, size and, especially on the separation between the clusters, as it can be seen in Figure 1 of the Appendix. In particular, *Single-Link* may be the best choice if the goal is to achieve good separation between the clusters. Indeed, in the case of *Single-Link* two clusters S and L for which the gap between the closest points $i \in L$ and $j \in S$ is the smallest are joined at each iteration. Thus, unlike *Complete-Link* or *Average*, *Single-Link* will not create a partition where the gap between the borders of the clusters is smaller than the gap between the points of the same cluster (see Figure 1 in Appendix).

However, the shortest distance between points $i \in L$ and $j \in S$ may not always be a good measure of a gap between two clusters. Points i and j can be relatively close to each other but far away from the rest of the points in their respective clusters, so with the exception of these two points the gap between L and S may be larger than it is assessed by *Single-Link*. To mitigate this issue we propose a simple modification of *Single-Link*: we measure the distance between clusters L and S as an average of k shortest distances between points in L

and *S*. We call such a distance *k*-distance. This approach is in some way 'midway' between *Single-Link* and *Average* clustering method. The *Average* takes the average of pairwise distances of all the points between two clusters which may be big simply because two clusters are spread out, while the actual gap between the clusters may be small. Therefore, *Single-Link* may be 'too little' while the *Average* may be 'too much' since we want to focus on the gap between two clusters, so it makes sense to concentrate on the few points that are close to the boundary. We call this intermediate approach *K-Link*.

2.3 Number of clusters in a partition

Once the hierarchy of clusters is built we need to cut the dendrogram at some height to obtain an actual partition. In our clustering application, cutting height and the number of clusters may be determined based on the data protector preferences for the maximal utility loss due to independent masking of clusters. Indeed, the vertical axis of the dendrogram is a measure of closeness between the clusters. In other words, cutting the dendrogram at a particular height *h* sets up a lower bound on the distance between pairs of clusters in the partition which, in turn, corresponds to the upper bound on the allowed correlation between different clusters. Correlation between the variables in different clusters may be attenuated or lost after the SDL method is applied independently to different clusters. Thus, the data protector can set up an upper bound on maximal loss of correlation by choosing the acceptable value of *h*. The exact interpretation of *h* may differ for different clustering methods as it is based on the definition of a distance between the clusters. For example, for the *Single-Link* method this is a maximal correlation between two variables in different clusters. For *K-Link* method, *h* is an average of a few largest correlations that can be observed between the variables in different clusters, and so on. However, for all of the methods it is essentially a summary of the observed correlation between the variables in different clusters.

It should be noted that for *K-Link*, cutting the dendrogram at a particular height may in some rare cases lead to several solutions, that is, several clustering partitions with different number of clusters. This might happen because the sequence of *k*-distances is not strictly monotone as in *Single-Link*, although, there is a clear overall increasing trend. In particular, the *k*-distance may slightly decrease from one iteration to another which results in merging of next closest clusters slightly lower in the dendrogram tree.

We would also like to note, that one of the reasons why in this paper we haven't considered such algorithms as k-means, k-medoids or some model-based clustering algorithms, such as [7], is due to the fact that all these methods require the number of clusters as an input parameter. To successfully apply these algorithms, one often needs to compare many clustering partitions corresponding to different numbers of clusters. Another reason, is that many of these algorithms, by design specifically target the homogeneity of clusters. For example, k-means minimizes the sum of squares within the cluster on each iteration, and, thus, may create a partition with poor separation between the clusters. We believe, that the approach outlined above for the hierarchical agglomerative clustering methods allows for a more straightforward way of determining the number of clusters. It is important to mention that this approach produces clusters of variables that are a suitable input for the subsequent use of SDL methods.

3 Numerical experiments

3.1 Data sets

We applied our approach for clustering the variables to two real multivariate data sets. One of them is the National Health Interview Survey 2015 fourth quarter sample adult component public file [16]. In the sequel we will refer to it as NHIS. This is a public use file that has already undergone disclosure limitation. It has 6213 records. For our experiments we selected 86 variables. Their summary description is given in the Appendix. When the correlations were computed for NHIS data, sampling weights and design structure were taken into account. Package R Survey was used for that.

Our second data set was downloaded from the UCI Machine Learning Repository [4]. This is a sample drawn from the Public Use Microdata Samples (PUMS) person 1990 US Census file. We will refer to this file as Census in the paper. It has 68 categorical variables and about 2.5 million records. Full description of the variables can be found in [1].

We applied *Diana*, *Single-Link*, *Average*, *Complete-Link*, *Homclust* as well as our *K-Link* method to these data sets.

3.2 Clustering criterion

In order to assess and compare the quality of partitions obtained by different methods we need to choose appropriate clustering criterion. Because clustering of variables is the first step and application of SDL to clustered data is the second step, ideally the clustering criterion should be in concordance with the SDL procedure which has to produce masked data with good utility. In this regard, we want to note that the clustering procedure and subsequent independent application of SDL methods to different clusters will not have any effect on univariate statistics of the masked data. These statistics will depend only on the properties of SDL method applied to the variables. Furthermore, clustering does not affect any relationships between the variables that belong to the same cluster. The only influence clustering may have is on the relationships between the variables that belong to different clusters. For example, clustering may have an effect on correlations between the variables that belong to different clusters. The worst case scenario or the worst output corresponds to the case when all the correlations between the variables that belong to different clusters are lost in the masked data because of the independent application of SDL methods to these clusters. That is why we base the assessment criterion on the separation between the clusters, which measures correlation between the variables in different clusters -the correlation which can be lost in the worst case scenario. The smaller the correlation between the variables in different clusters - the better the output from the utility prospective.

Many clustering criteria were proposed in the literature, some examples are [15, 8, 14, 13]. Many of them, however, are focused on the compactness of the clusters. However, as we mentioned above, compactness of the clusters is not an important quality for masking or synthesis of the variables in the clusters, however, separation between the clusters is. Several separation indexes were proposed in the literature. For example, [9] mentions an index that is computed as the ratio of the shortest distance between two clusters S and L (computed as the shortest distance between two points $i \in S$ and $j \in L$) and the maximal cluster diameter in the partition. A similar metric was proposed in [17], in particular, the gap between two clusters is divided by the total spread of both clusters. However, the spread of a cluster or it's diameter is a measure of cluster compactness. It was incorporated in the aforementioned separation indexes in order to give preference to the partitions with compact and well separated clusters, which makes sense if the ultimate goal of clustering is to detect the "true" cluster structure of the data. However, when diameters of clusters increase, such separation indexes become smaller indicating that the quality of the partition becomes worse. But for the purpose of masking clustered data such a partition is not worse than a partition with compact clusters if the gap between the clusters is the same. Therefore, we will include only the separation component into our clustering criterion, but not the spread. Thus, the separation criterion that we will use to compare different methods is as follows: first, separation between any two clusters S, L is computed as an average of the smallest s distances/dissimilarities $d(i, j)$ between the elements $i \in L$ and $j \in S$. Next, minimum of separations between all pairs of clusters in a partition is found.

The elements, located in different clusters at shortest distances from each other essentially represent the borders of these two clusters and s can be thought as a parameter of "thickness" of the border. In general, s is data dependent and can be set to different values, for example, s can be equal approximately to the 5^{th} percentile of the number of distances between the elements in different clusters $i \in S$ and $j \in L$. One of the topics of our future research is to investigate further the best ways of defining s. For simplicity, in our experiments we set s to be equal to 5 for all the pairs of clusters with 5 or more pairwise distances between the elements in different clusters. For the most populated pairs of clusters, 5 is approximately a 5^{th} percentile of distances for our data sets. For the pairs of small clusters, for which there are less than 5 distances between the variables in different clusters, we consider all the distances. Further in the text when we refer to the distance between two clusters, we mean average of the s shortest distances between the variables in different clusters.

3.3 Results

Table 1 shows the results of the comparison of *Diana*, *Single-Link*, *Average*, *Complete-Link*, *Homclust* and *K-Link* methods which were applied to NHIS data. For *K-Link* method we experimented with different values of the parameter k. The best results in terms of separation were obtained for $k = 3$ and we present these results in Table 1.

The minimal distance between any two clusters in the partition for each of the methods does not give a full picture of the partition. We get a more realistic impression of the composition of the partition by considering several closest distances between pairs of clusters, not just the shortest one. In Table 1 we listed distances between 10 closest pairs of clusters for *3-Link*, *Single-Link*, *Complete-Link*, *Average*, *Homclust* and *K-Link*. Column *3-Link-10* denotes *3-Link* method where the size of the cluster was enforced not to exceed a predefined limit, in this case, 10 variables per cluster. This method will be discussed later in the paper.

For all the methods we partitioned the data into 25 clusters which corresponds to cutting the dendrograms approximately at height 0.8, so that the maximum correlation loss between two clusters does not exceed 0.2.

Results for the Census data are shown in Table 2. Parameters were set the same as for the NHIS data. Both tables show similar patterns in terms of relative closeness between the clusters for different methods.

As it can be seen from Tables 1 and 2, *3-Link* has the largest separation for all ten closest pairs of clusters. It is followed by *Single-Link* and then by *Average*. Consistently worse are *Diana* and *Complete-Link*. The worst separation among the methods that have no limits on the cluster size is observed for *Homclust* which agrees with our assumption that the algorithms based on clusters' homogeneity, which groups the most correlated variables in the same clusters, may not be appropriate for those cases when the objective is to create maximal separation, or minimal correlation, between the variables in different clusters.

Moreover, it is worth noting that we were not able to apply *Homclust* to the Census data set, which has 2.5 million records. The implementation of *Homclust* provided in package "ClustOfVar" [2] by its authors was not able to handle data set of this size. Thus, poor scalability is an additional issue of the *Homclust* method.

The entries in Tables 1 and 2 are the averages over the shortest s distances between two clusters. We also compared the actual correlations between the variables in different clusters. We observed that some of the variables which were placed in different clusters by *Complete-Link*, *Homclust* and *Diana* were very close in terms of correlation. For example, for the NHIS data set, the shortest distance between the variables in the two closest clusters is 0.81 for *K-Link*, 0.78 for *Single-Link*, 0.70 for *Average*, however, they are about 0.1 for *Complete-Link* and *Homclust* and $2.33E-16$ for *Diana*. The same was true for the next closest pair of clusters as well, that is, the the actual distances between the variables were considerably smaller for *Diana*, *Complete-Link* and *Homclust* comparative to *K-Link* and *Single-Link*. A similar pattern was observed for the Census data.

Regarding partitions obtained by the applications of the clustering methods mentioned above, we observed that *Single-Link* and *K-Link* may lead to a partition where one of the clusters contains many (or a majority) of the variables and a number of small clusters with one or two variables, while for methods that lead to more compact and homogenous clusters such as *Complete-Link*, *Average*, *Diana* and *Homclust*, the largest cluster has less variables and overall partition is slightly more balanced. For example, when we partitioned the NHIS data into 25 clusters, the largest cluster contains 9 variables for *Diana*, 11 for *Average* and 11 variables for *Complete-Link*. However, for *Single-Link* and *3-Link* the largest clusters contain 50 and 58 variables respectively. In the case of Census data the largest cluster contains 14 variables for *Diana*, 28 variables for *Complete-Link*, *Average*, but 38 variables for *K-Link* and 35 for *Single-Link*.

Since the main reason of clustering the variables here is to reduce the complexity of joint masking or joint synthesis, clustering partitions with one or few very big clusters may not serve the main purpose very well. That is why we implemented a modification of *K-Link* method that incorporates an upper bound on the cluster size: as soon as the cluster size reaches *n* variables, the cluster cannot "accept" any new members. We will refer to this modification as *k-Link-n* further in the text. Another possibility to solve a "big cluster" problem is to split the biggest cluster in two or three smaller ones. There is, however, no guarantee that the obtained partition would not result in one big and another small cluster again. Moreover, incorporation of the restriction *n* during the merging process, as opposed to cutting the biggest cluster after clustering hierarchy is complete, may lead to better results because the variables that cannot join the cluster any longer that reached the maximum number of variables, can still join any other cluster which is closest to it. This will occur in the earlier stages of clusters formation, that is, as soon as the limit *n* was reached for the big cluster. Partitioning the biggest cluster in two or three after the dendrogram was finished, would limit the possibilities of grouping the variables only with those that are part of this cluster while all other clusters remain unchanged, which may not be the best solution.

For NHIS data *K-Link* with the limit of *n* = 10 variables per cluster still compares reasonably well with other methods that do not have restrictions. It is the second method after *Single-Link*. Recall, however, that *Single-Link* creates a cluster of 50 variables, while *3-Link-10* has only 10. Performance of *3-Link-10* is very similar now to *Average*, for which the largest cluster in the partition has 10 variables as well.

For the Census data, enforcing the limit on cluster size had a larger effect on the separation between the clusters than for NHIS data. In fact, in the Census data there is a big group of correlated variables. Thus, *Single-Link*, *Average*, *Complete-Link* form a big cluster in their corresponding partitions. The size of the largest cluster varies from 28 to 38 variables among these methods. Thus, by enforcing the limits on the cluster size for *K-Link* we inevitably reduced the separation between the clusters in the obtained partition. Columns *3-Link-15*, *3-Link-25* and *3-Link-28* of Table 2 show the separation for *3-Link* with limits *n* = 15, 25 and 28. It can be seen that separation is not very good especially for the lower values of *n*.

It is important to observe that, it is not possible to enforce *n* till the top of the dendrogram. Our implementation of *k-Link-n* reports the minimal number of clusters when *n* can still be enforced. After that, to complete the hierarchy, clustering process continues as in the original version without restriction until the dendrogram is complete.

We conclude that clustering of variables can help reduce the complexity of SDL methods. However, there is a trade-off between complexity reduction and data utility, which depends on the correlation structure of the original data.

Finally, we want to note that the grouping of variables produced by clustering makes scientific sense. For example, for NHIS data, a group of food availability questions were placed in the same cluster, a group of questions about health care availability were together, exercising questions about exercising patterns were in the same cluster, and so on. Similar pattern was observed for Census data.

4 Concluding remarks and future work

In this paper we propose a pre-masking clustering procedure that can be used by data publishing organizations that release data sets with many attributes of different types, such as big government surveys. Joint masking of data sets with many variables may be complicated and computationally involved. To reduce the complexity of the problem we outline a procedure of grouping variables into clusters in such a way that data utility loss due to independent application of SDL methods to these groups is limited. An upper bound on utility loss can be set up by the data protector. The value of this bound determines the parameters of the clustering procedure. Furthermore, we present a hierarchical clustering method, that we call *K-Link*, that can be suitable for the purpose of subsequent independent application of SDL to these clusters of variables. In our experiments *K-Link* compares favorably with a number of existing hierarchical agglomerative and divisive clustering methods. In our future research we plan to consider a wider range of clustering methods that may be used for this purpose.

It is worth mentioning that we focus on the correlation-based utility loss due to clustering. In the future research we plan to expand the study of utility loss by considering other types of associations between the variables in the masked data.

In this paper we do not specify, neither do we focus on any particular SDL method as we believe that in general our clustering approach for variables should help to reduce complexity of any multivariate SDL method which preserves correlation structure of the data. As we mentioned earlier in the paper, it may be particularly beneficial for synthetic methods. Clustering of variables can also be helpful for developing multivariate analogs of some commonly used univariate SDL procedures, for example top-coding. Extreme values of some continuous variables are often top coded. For example, weight, height or income can be top-coded. However, if the upper bound of top-coded variable is determined independently from other variables, protection may be inadequate for different groups of individuals. For example, assume that the data protector sets the upper bound for weight to be equal to 300 pounds for all the respondents. However, a female respondent with such a top-coded weight whose race/ethnicity is Asian is more extreme as opposed to respondent with the same weight who is male Caucasian. A multivariate approach to top-coding could be considered. While grouping race/ethnicity, gender, weight, height together may seem intuitive, there may be other, much less obvious combinations of variables especially in big survey data sets with hundreds of variables. Clustering of variables may be helpful for finding such groups.

Acknowledgments

The authors would like to thank Van Parsons from the National Center for Health Statistics (US) for providing cleansed version of NHIS public use sample file for our experiments and for his valuable suggestions. Also the authors would like to express their appreciation to Donald Malec also from the National Center for Health Statistics for his careful reading of the paper and many useful suggestions. The findings and conclusions in this paper are those of the authors and do not necessarily represent the official position of the Centers for Disease Control and Prevention.

Appendix

Part A: different partitions obtained by *Single-Link*, *Average*, *Complete-Link* and *Diana*

Figure 1 illustrates how differences in definition of distance between the clusters for *Single-Link*, *Average*, *Complete-Link* and *Diana* may influence the form and separation between the clusters. For this data set *Single-Link* was able to capture the structure of the data and created the most separated clusters. Separation between the clusters for partitions obtained by *Complete-Link*, *Average* and *Diana* is poor. These methods cut the vertical cluster in two or three parts very close to each other. On the other hand, a distant group of four point to the right of vertical cluster is merged with it.

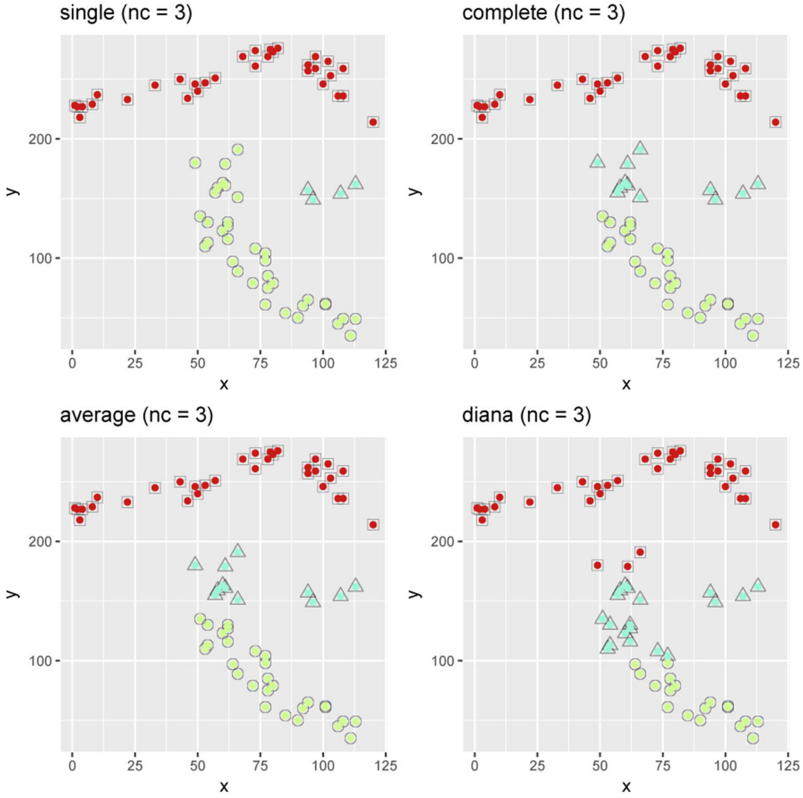

Fig. 1.

Partitions for *Single-Link*, *Complete-Link*, *Average* and *Diana* for an artificial data set of points with coordinates (*x, y*). Red, blue and green colors indicate cluster memberships for the points.

Part B: Summary description of NHIS variables

The NHIS data set contains 86 variables. The variables are the respondents' answers to the questions in the following categories: health conditions, mental and emotional health, health behavior, affordability and accessibility of health care services, health insurance coverage, food availability and accessibility, employment status, income and education. The group of health related variables include presence or absence of asthma, diabetes, bronchitis, other pulmonary diseases and high blood pressure. Health behavior group of variables are the answers to the questions about cigarette smoking, alcohol use, leisure-time physical activity and exercising. Mental and emotional health variables are the answers about feeling hopeless, nervous, restless and fidgety, and also feeling worthless and sad. NHIS file also includes height, weight, body mass index of the respondents as well as demographic variables, such as race, age, marital status and region.

Most of the categorical variables are binary (Yes or No answers).There are also few continuous variables in the file, for example, age, weight, height and BMI. For details see [16].

References

1. Census: US census (1990) data set. UCI Machine Learning Repository (2017), https://archive.ics.uci.edu/ml/datasets/US+Census+Data+%281990%29
2. Chavent M, Kuentz-Simonet V, Liquet B, Saracco J: ClustOfVar: An R Package for the Clustering of Variables. Journal of Statistical Software 50(i13), 1–16 (2012) [PubMed: 25317082]
3. Cluster R package, https://cran.r-project.org/web/packages/cluster/cluster.pdf
4. Dheeru D, Karra Taniskidou E: UCI machine learning repository. University of California, Irvine, School of Information and Computer Sciences (2017), http://archive.ics.uci.edu/ml
5. Dhillon I, Marcotte E, Roshan U: Diametrical clustering for identifying anti-correlated gene clusters. Bioinformatics 19(13), 1612–1619 (2003) [PubMed: 12967956]
6. Everitt B, Landau S, Leese M, Stahl D: Cluster Analysis Series in Probability and Statistics, Wiley, fifth edn (2011)
7. Fraley C, Raftery A: MCLUST version 3 for R: Normal mixture modeling and model-based clustering. Tech. rep., Department of Statistics, University of Washington (2006), http://cran.r-project.org/web/packages/mclust/index.html
8. Halkidi M, Batistakis Y, Vazirgiannis M: On clustering validation techniques. Journal of Intelligent Information Systems 17, 107–145 (2001)
9. Höppner K, Klawonn F, Runkler T: Fuzzy Cluster Analysis: Methods for Classification, Data Analysis and Image Recognition. Wiley, New York (1999)
10. Hundepool A, DomingoFerrer J, Franconi L, Giessing S, Nordholt ES, Spicer K, de Wolf P: Handbook on Statistical Disclosure Control (version 1.2). ESSNET, SDC project (2010), http://neon.vb.cbs.nl/casc
11. Hundepool A, DomingoFerrer J, Franconi L, Giessing S, Nordholt ES, Spicer K, de Wolf P: Statistical Disclosure Control. Wiley (7 2012)
12. Kaufman L, Roussew P: Finding Groups in Data - An Introduction to Cluster Analysis. A Wiley-Science Publication John Wiley & Sons (1990)
13. Kim JJ: A method for limiting disclosure in microdata based on random noise and transformation. In: Proceedings of the ASA Section on Survey Research Methodology pp. 303–308 (2002)
14. Lin C, Chen M: A robust and efficient clustering algorithm based on cohesion selfmerging. In: Proceedings of Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining pp. 582–587. Edmonton, Alberta, Canada (2002)
15. Milligan G: A monte carlo study of thirty internal criterion measures for cluster analysis. Psychometrika 46(2), 187–199 (1981)
16. NHIS: National Health Interview Survey. National Center for Health Statistics (2015), https://www.cdc.gov/nchs/nhis/nhis_2015_data_release.htm
17. Qiu W: Separation Index, Variables Selection and Sequential Algorithm for Cluster Analysis. Ph.D. thesis, The University of British Columbia (2004)
18. Vigneau E, Qannari E: Clustering of variables around latent components. Communications in Statistics Simulation and Computation 32(4), 1131–1150 (2003)

Table 1.

NHIS data set: Minimal separation between the clusters for *3-Link*, *3-Link-10*, *Single-Link*, *Complete-Link*, *Average*, *Homclust* and *Diana*. Min_1, $Min_2 \cdots Min_{10}$ are the distances between the ten closest pairs of clusters.

	3-Link	3-link-10	Single-Link	Complete-Link	Average	Homclust	Diana
Min_1	0.8190	0.7501	0.7901	0.5734	0.7501	0.5717	0.6923
Min_2	0.8245	0.7583	0.7955	0.6321	0.7583	0.6128	0.7411
Min_3	0.8282	0.7701	0.8196	0.7501	0.7701	0.6764	0.7600
Min_4	0.8302	0.7802	0.8243	0.7599	0.7792	0.7432	0.7681
Min_5	0.8351	0.7900	0.8261	0.7701	0.7891	0.7523	0.7809
Min_6	0.8431	0.7925	0.8267	0.7735	0.7925	0.7606	0.7834
Min_7	0.8525	0.8001	0.8299	0.7810	0.8001	0.7783	0.7845
Min_8	0.8590	0.8019	0.8307	0.7880	0.8019	0.7801	0.7881
Min_9	0.8635	0.8038	0.8341	0.7903	0.8025	0.7834	0.7916
Min_{10}	0.8691	0.8200	0.8408	0.7999	0.8154	0.7999	0.8032

Table 2.

Census data set: Minimal separation between the clusters for *3-Link*, *3-Link-15*, *3-Link-25*, *3-Link-28*, *Single-Link*, *Complete-Link*, *Average*, *Homclust* and *Diana*. Min_1, $Min_2 \cdots Min_{10}$ are the distances between the ten closest pairs of clusters.

	3-Link	**3-Link15**	**3-Link-25**	**3-Link-28**	**Single-Link**	**Complete-Link**	**Average**	**Diana**
m1	0.5587	0.0228	0.1463	0.1883	0.2443	0.1967	0.1883	0.0000
m2	0.5914	0.0716	0.1883	0.2503	0.2503	0.2503	0.2503	0.1883
m3	0.5982	0.1883	0.4110	0.2790	0.2790	0.2790	0.2790	0.2790
m4	0.6250	0.3892	0.5914	0.4110	0.3314	0.4110	0.4110	0.3825
m5	0.6331	0.4110	0.6250	0.5210	0.4400	0.5148	0.5210	0.4731
m6	0.6351	0.5210	0.6392	0.5971	0.5587	0.5725	0.5971	0.5085
m7	0.6443	0.6405	0.6443	0.6392	0.5657	0.5971	0.6392	0.5631
m8	0.6627	0.7034	0.7034	0.6405	0.5971	0.6104	0.6405	0.5710
m9	0.6853	0.7050	0.7067	0.6740	0.6019	0.6378	0.6740	0.5971
m10	0.7034	0.7067	0.7080	0.7067	0.6351	0.6578	0.7067	0.6579