# An approach for anonymization of sensitive clinical and genetic data based on Data Cube Structures

Ntalaperas Dimitrios
Ubitech Ltd, Greece
e-mail: dntalaperas@ubitech.eu

## Abstract

This paper presents an approach for anonymizing data containing sensitive patient information. The approach is based on the usage of a Data Cube structure that stores only aggregate information of the various phenotypic and genetic values of variables of interest. The Data Cube is further perturbed in a manner that does not affect the statistical characteristics of the original dataset, thus further enhancing the fidelity of the anonymization procedure. The anonymized data can be semantically enhanced and transferred over the network in plain format without the risk of violating patients' personal rights.

## Keywords

Anonymization, Security, Algorithms

## 1. Introduction

As Electronic Health Records (EHR) are adopted by an ever growing number of clinical health care providers, institutes can easily interchange information electronically, which can then be statistically analyzed, thus increasing research output. However, interchange of data concerning clinical trials and genetic data are subject to protection laws, which inhibit the disclosure of sensitive patient information. To this extent, data should be anonymized, not only to ensure that sensitive data is not eavesdropped during transmission, but also to ensure that the party legitimately receiving the data cannot have access to sensitive personal data.

In this paper, we present a methodology for transforming data corresponding to patient clinical and genetic data to an equivalent data set that contains the informational content of the original data set regarding the medical data, but from which all information regarding personal data has been filtered out. The methodology builds upon the Data Cube approach as this has been adopted by the Linked2Safety[1]consortium (Perakis et al. 2013) combined with cell-suppression and perturbation techniques which ensure that the produced data cube cannot be reversed engineered (Forgó et al. 2012), while also retaining the same statistical characteristics as the original data set.

The resulting methodology is being used in the context of the SAGE-CARE[2] project, in order to be able to transmit data that combine phenotypical characteristics and counts of gene expressions of a patient in a safe manner, retaining patients' anonymity while at the same time allowing researchers to search for correlations between these data in the context of melanoma research.

## 2. Theory

A dataset containing medical data can have mixed information with sensitive data being present at various locations of the source files. Moreover, the informational content may be in semi-structured or unstructured format (e.g. free text). An anonymized dataset must fulfil the following criteria:

[1] http://www.linked2safety-project.eu/

[2] http://www.sage-care.eu/

1. Any information of personal data must be filtered out
2. Information that can be used to reverse engineer the anonymized data must be filtered out. If, for example, a combination of phenotypic characteristics is extremely rare and this data is present in the anonymized set, an eavesdropper can use this information to identify a patient.
3. Any statistical analysis performed on the anonymized data should yield the same results as for the original set.

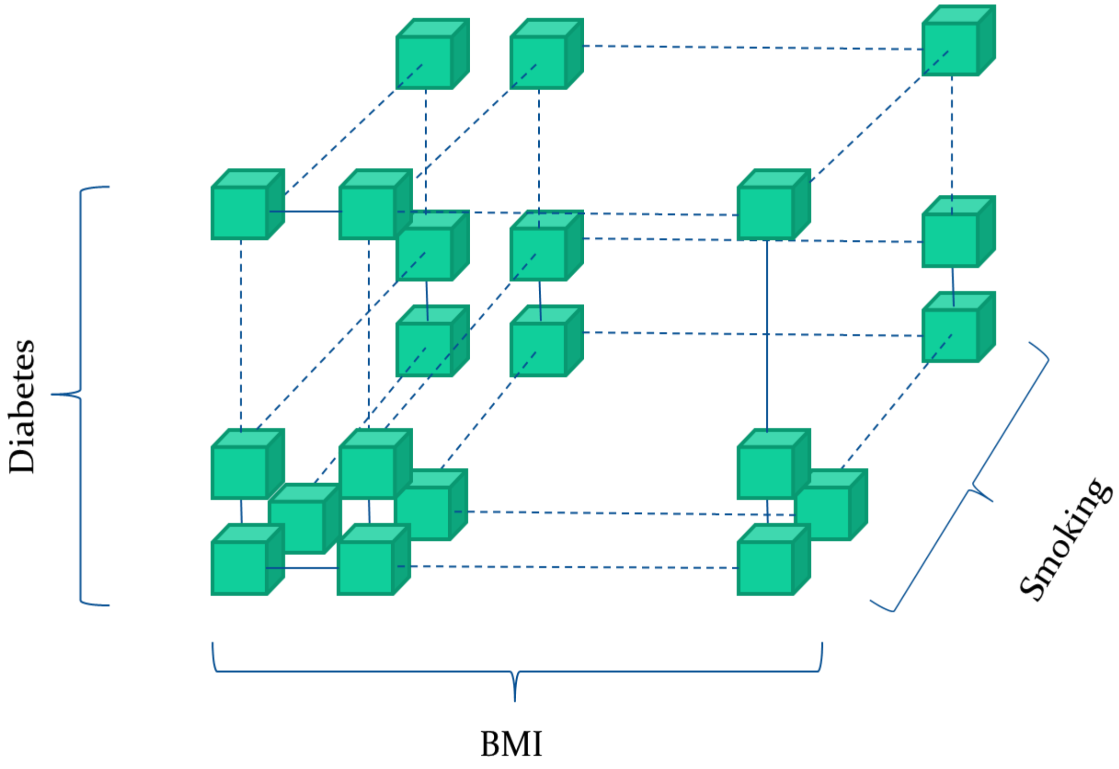


**Figure 1 Example of a Data Cube Structure.**

A data structure that can fulfil the above criteria is a Data Cube. A Data Cube can be defined as an $m_1$x$m_2$x...x$m_n$ array, with $n$ being the number of selected variables and $m_i$ being the number of distinct values the variable indexed by $i$ can take. Each cell contains the number of total counts with the combination of values defined by the index of the Data Cube. Figure 1 depicts an example of a three-dimensional Data Cube. In this example, each variable is supposed to be categorical (take values from a district set of integers). The value *0* for BMI for example can correspond to low BMI, value *0* for diabetes to low blood sugar levels and so on. If the Data Cube is named *DC*, then the cell *DC[i][j][k]*, will have the total count of patients having BMI equal to *i,* Smoking equal to *j* and

Diabetes equal to *k*. Since the cube contain only aggregate data, it fulfils criterion 1. Moreover, all data correlations of the original dataset concerning these variables are not altered, thus criterion 2 is satisfied.

A special case is that of *degenerate* data cubes, that can arise from non-categorical data. If a variable is not categorized (e.g. has continuous values) and a data cube is constructed naively, then it may be the case that each distinct value will correspond to a very small number of patients. In this case, the data cube will have a size of the order of the size of the original data set with each containing a small number. Criterion 2 in this case will be violated.

The case of degenerate cells is handled by adopting a combination of cell-suppression and perturbation techniques as well as imposing categorization on the data. These two techniques are described in the following sub-sections.

## 2.1. Cell-suppression

Cell-suppression is the procedure that forces cells with a low value to be discarded from the dataset. By discarding these cells, the process of distinguishing non-existent combinations from combinations with a low count becomes impossible. Perturbation imposes an extra layer of security by imposing a random, low amplitude noise to each cell so they are no longer guaranteed to contain the exact number of patients sharing the characteristics described by the cell. The distribution of the noise can always be chosen as to not alter the statistical characteristics of the original dataset. A typical configuration is to randomly select a value from the set *{-1,0,1}* to add to the final count with equal probability. This combination ensures in most cases that the statistical characteristics of the anonymized dataset will remain unaltered (Antoniades et al. 2012) thus ensuring that Criterion 3 is still being fulfilled.

Considering the specific distribution that occur in each data set, the user may opt to increase the perturbation noise to further obfuscate the data or to decrease it to keep the fidelity of the data as high as possible. In general, as the ratio of distinct combinations to total patient grows, so does the range of perturbation noise.

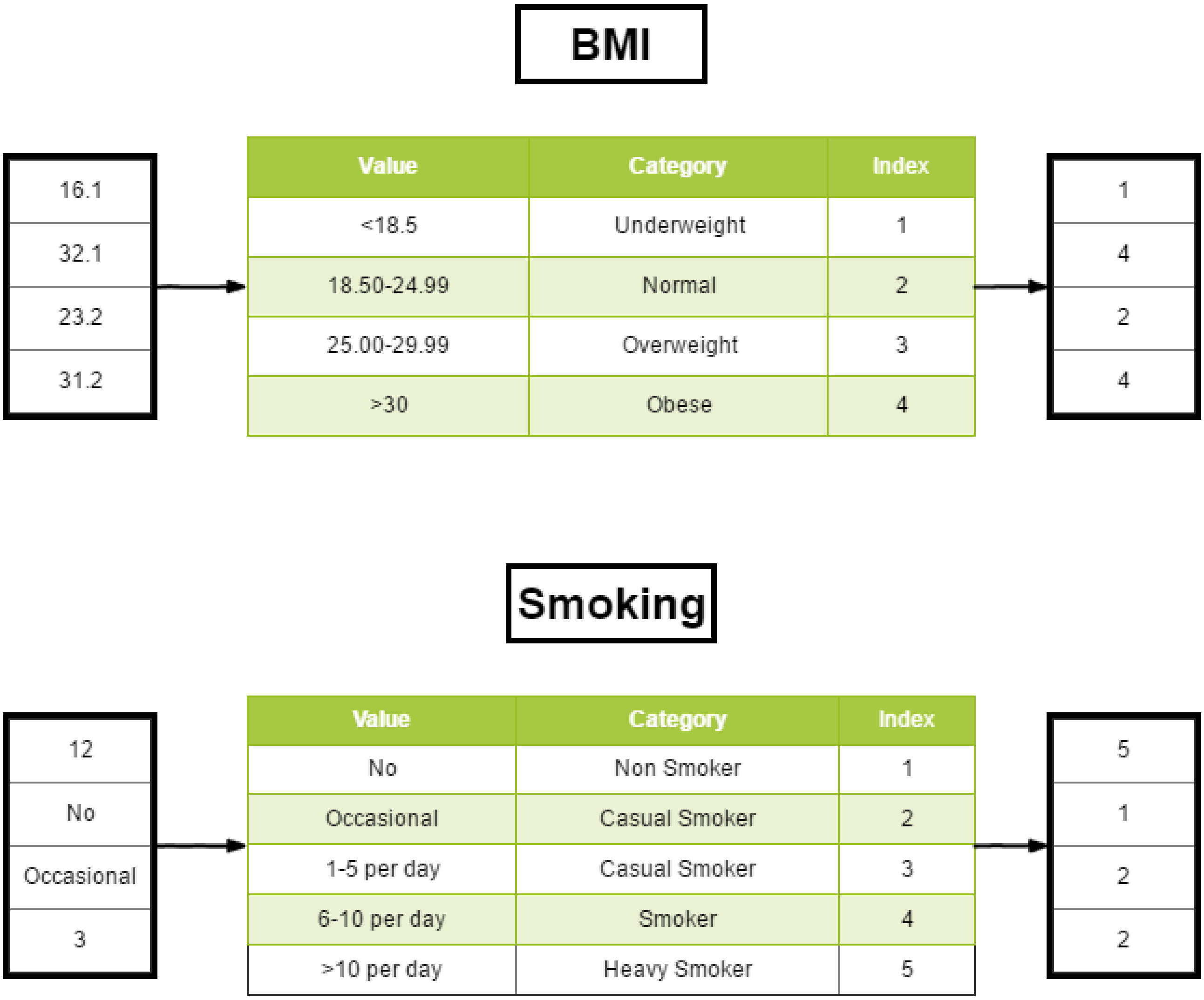
## 2.2. Data Categorization

Data Categorization refers to the process of mapping values of source data, that take continuous or arbitrary values, to values of a finite set. The mapping does not need to be isomorphic, in fact, in the case of continuous mapping, isomorphic mapping is impossible. Figure 2 depicts an example for each of the cases of Data Categorization.

In the case of BMI the source data contain entries that describe the BMI of each patient and the categorization is performed according to the international classification (WHO 1995). This is often the case of numerical data, since these usually correspond to phenotypic variables that can be categorized according to medical standards or according to various taxonomies and ontologies like MedRA[3] or SEHR (Sahay et al. 2013). In the case of Smoking on the other hand, the values are arbitrary and can be numerical or text. The mapping in this case must be custom and be provided explicitly via a mapping file. A mapping file is also needed in the case of numerical data, if the user does not wish to conform to a standard ontology.

[3] http://www.meddra.org/

In the context of SAGE-CARE, both methods have been implemented. Phenotypic variables are typically mapped by using standard taxonomies, whereas genetic data (typically counts of gene expressions) are mapped in a custom way, since the needs of categorization depend both on the specific gene, as well as the context of the specific correlations the user wishes to derive.



BMI

16.1<br>32.1<br>23.2<br>31.2

| Value | Category | Index |
|---|---|---|
| <18.5 | Underweight | 1 |
| 18.50-24.99 | Normal | 2 |
| 25.00-29.99 | Overweight | 3 |
| >30 | Obese | 4 |

1<br>4<br>2<br>4

Smoking

12<br>No<br>Occasional<br>3

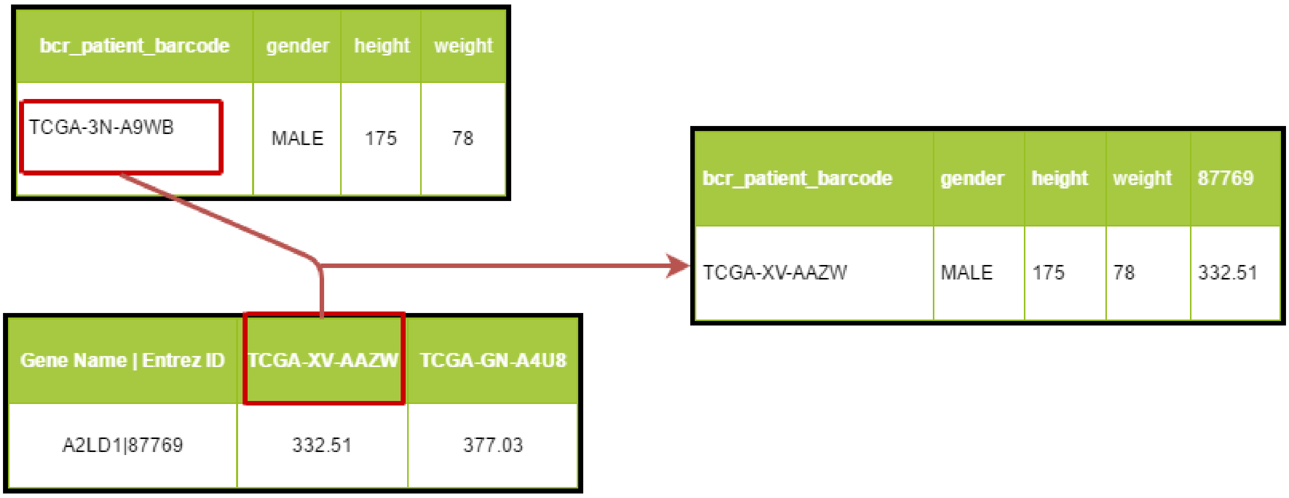
| Value | Category | Index |
|---|---|---|
| No | Non Smoker | 1 |
| Occasional | Casual Smoker | 2 |
| 1-5 per day | Casual Smoker | 3 |
| 6-10 per day | Smoker | 4 |
| >10 per day | Heavy Smoker | 5 |

5<br>1<br>2<br>2

**Figure 2 Categorization Examples**

## 3. Implementation

The above general techniques presented above were implemented in the context of the SAGE-CARE project to create anonymized clinical (phenotypical) data and genetic data recording the expression of genes of each patient. The methodology pipeline is depicted in Figure 4. The component at first accepts the two files that contain the source data; a .csv file that contains phenotypical data with each row corresponding to each patient and a .tsv file that contains genetic data with each row corresponding to a gene and each column to a patient. The .tsv file contains the patient code at the top cell of each column thus facilitating the linking of patients between the two datasets. The component constructs an intermediate structure which aligns the two data set so that operations, such as aggregation which is required for data cubes, can be easily performed. An example depicting how data alignment is performed can be seen in Figure 3, where the phenotypic variables of a fake patient is matched against the file containing genetic information. As can be seen the barcode of the patient is used to obtain the column, indexed *j*, of the genetic data file. Supposing that we need to construct a cube containing expression counts of the gene with Entrez id equal to *87769*, the corresponding row, indexed *i* of the *.tsv* file needs to be found by matching the id against the values of the cells of the first column. The cell with the coordinates *(i,j)* of the *.tsv* file contains the value for the expression count that is to be stored to the data aligned data structure. The header of the column, which is to be used to identify the genetic variable, will be the Entrez id of the gene.



| bcr_patient_barcode | gender | height | weight |
|---|---|---|---|
| TCGA-3N-A9WB | MALE | 175 | 78 |

| bcr_patient_barcode | gender | height | weight | 87769 |
|---|---|---|---|---|
| TCGA-XV-AAZW | MALE | 175 | 78 | 332.51 |

| Gene Name \| Entrez ID | TCGA-XV-AAZW | TCGA-GN-A4U8 |
|---|---|---|
| A2LD1\|87769 | 332.51 | 377.03 |

**Figure 3 Data Alignment example for one patient.**

At the second state, the component receives a mapping file in .xml format which contains the variables and genes for which a data cube is to be created. The mapping file also contains information for grouping values to categories; the categorization can be performed either by using value ranges or by using enumerations (see Listing 1 for a sample mapping file). Alternatively, as discussed in Section 2, a taxonomy can be used.

The data are transformed according to the categorization schema and are then fed as input to the Data Cube Creation component. The Data Cube Creation component performs aggregation on the data (see Listing 2 for pseudocode describing the algorithm).

Having the Categorization being performed after the Data Alignment can have a detrimental effect on performance. This can easily be seen from the fact that the tables containing the aligned data need to be parsed again in order to convert original values to categorical ones, something that could have been done during Data Alignment process. Keeping however the original (uncategorized) data separate, has the advantage that these can be reused with different categorization schemas, something that reduces execution times of subsequent calls to the component.

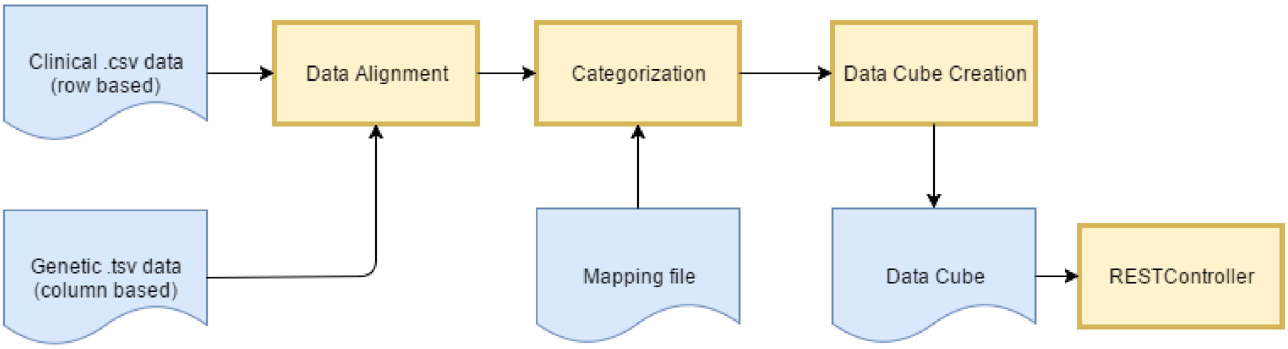


**Figure 4 Methodology Pipeline**

```
<?xml version="1.0" encoding="UTF-8"?>
<root>
    <phenos>
         <pheno name="gender" />
         <pheno name="height" round="1d" />
    </phenos>
     <genes>
          <gene name="155060" round="3d" />
     </genes>
</root>
```

**Listing 1 Sample mapping file. Variable gender is defined without a mapping, which implies that the data are already categorized regarding gender values. Variable height is to be rounded to first digit which means that values within 10 cms will be grouped together. Similar for the gene with entrez code 155060 the expression counts within 1000 will be grouped together**

```
procedure dataCube
    foreach line
        computeunique_key
        foreach cell in line_cells
            cell_field++
            value[cell_field] = valueOf(cell)
            value[cell_field] = convertInRange(value, ranges[cell_field])
        Cube[unique_key][indexOf(value[cell_field])]++
    Foreach unique_key
        for i=1 to num_of_fields
            perturbe(Cube[unique_key][i])
            if Cube[unique_key][i] <= threshold
```

**Listing 2 Data Cube Creation Algorithm. Unique key is obtained by computing a unique hash from the combination of variables values corresponding to each cell.**

### 3.1. Special Case: Genetic Data

It is often the case that sets of phenotypic variables need to be checked for correlations against a large dataset of genetic data containing thousands of genes or DNA base pairs. Constructing a Data Cube for all these genes may cause degeneracy, long execution times and will require a vast amounts of storage space, considering that the Data Cubes will have a dimensionality on the order of thousands.

However, obtaining correlations between thousands of genes is seldom needed in practice; often, what the user needs is deriving correlations between a set of phenotypical variables and a gene, or between a small number of genes. The latter case can be handled by the techniques already discussed, for the former case the Data Cube component can be executed in a special mode. The algorithm of this mode is depicted in Listing 3. The procedure is simple: for each gene, a data cube is created using the set of phenotypic variables plus the gene. If the number of phenotypic variables is *n* and the number of genes is *m*, this procedure will create *m* data cubes of dimensionality *n+1* (*n* being the phenotypic variables plus one for the gene variable). Using the naïve approach a single data cube would be created with a dimensionality equal to *m+n*.

```
procedure geneticDataMode
        foreach gene
                call dataCubeGene
```

**Listing 3 Data Cube Creation Algorithm for Genetic data**

## 4. Conclusions

In this paper a methodology for ensuring the anonymization of data present in the context of the SAGE-CARE project was presented. The methodology is based on peer reviewed structures and techniques that are validated both technically and legally thus providing a framework for the required level of data security.

Future work consist of exploring ways to align data between heterogeneous data sources that are not covered by the mapping scheme presented here. Under consideration is the adoption of OpenRefine[4], which allows the filtering, faceting and noise extraction on data, as well as the definition of a common mapping ontology in a similar fashion that was followed by the Linked2Safety consortium.

Furthermore, there is ongoing research, with the collaboration of University of Naples[5], regarding ways to increase the performance of the algorithm by using High Performance Computing (HPC) techniques as to be able in the future to compute multidimensional data cubes and generate responses to queries in real time. The two points of the algorithms that are considered is the data alignment and the aggregation process; the tabular data that these two procedures are manipulating

[4] http://openrefine.org/

[5] http://www.unina.it/

do not, in general, have interdependence and thus parallelization of these two routines can lead to significant gains in performance.

## 5. Acknowledgements

We would like to thank Dr. Paul Walsh for providing guidance about what the user expectations of the component could be and for providing information about the representation of original source data. We would also like to thank Professor Bernhard Humm for providing insight on how to best use ontology for data mapping and categorization. Last, but not least, we would like to thank Professor Davide Marocco for his help regarding the parallelization of the algorithm.

## References

K.Perakis et. al. (2013). "Advancing Patient Record Safety and EHR Semantic Interoperability", in: IEEE International Conference on Systems, Man, and Cybernetics, IEE, 2013.

N. Forgó, M. Góralczyk, and C. Graf von Rex. (2012) "Security issues in research projects with patient's medical data", in: 12th Int. IEEE Conf. on Bioinformatics & Bioengineering (BIBE), IEEE, 2012

A. Antoniades et. al. (2012). "The effects of applying cell-suppression and perturbation to aggregated genetic data", in: 12th Int. IEEE Conf. on Bioinformatics & Bioengineering (BIBE), IEEE, 2012.

WHO (1995). Physical status: the use and interpretation of anthropometry. Report of a WHO Expert Committee. WHO Technical Report Series 854. Geneva: World Health Organization, 1995.

R. Sahay et. Al. (2013). "An Ontology for Clinical Trial Data Integration", in IEEE SMC 2013 - IEEE International Conference on Systems, Man, and Cybernetics, IEE, 2013