Journal of Biomedical Informatics 50 (2014) 32–45

Contents lists available at ScienceDirect

Journal of Biomedical Informatics

journal homepage: www.elsevier.com/locate/yjbin

# A data recipient centered de-identification method to retain statistical attributes

Tamas S. Gal [a,*], Thomas C. Tucker [a], Aryya Gangopadhyay [b], Zhiyuan Chen [b]

[a] University of Kentucky, 2365 Harrodsburg Rd., Suite A230, Lexington, KY 40504, USA
[b] University of Maryland at Baltimore County, 1000 Hilltop Circle, Baltimore, MD 21250, USA

ARTICLE INFO

*Article history:*
Received 28 August 2013
Accepted 3 January 2014
Available online 10 January 2014

*Keywords:*
Privacy
Utility based privacy preserving data mining
Statistical analysis

ABSTRACT

Privacy has always been a great concern of patients and medical service providers. As a result of the recent advances in information technology and the government's push for the use of Electronic Health Record (EHR) systems, a large amount of medical data is collected and stored electronically. This data needs to be made available for analysis but at the same time patient privacy has to be protected through de-identification. Although biomedical researchers often describe their research plans when they request anonymized data, most existing anonymization methods do not use this information when de-identifying the data. As a result, the anonymized data may not be useful for the planned research project. This paper proposes a data recipient centered approach to tailor the de-identification method based on input from the recipient of the data. We demonstrate our approach through an anonymization project for biomedical researchers with specific goals to improve the utility of the anonymized data for statistical models used for their research project. The selected algorithm improves a privacy protection method called *Condensation* by Aggarwal et al. Our methods were tested and validated on real cancer surveillance data provided by the Kentucky Cancer Registry.



## 1. Introduction

The advances in Information Technology and the recent push from the federal government [1] made Electronic Health Records (EHR) systems widespread in the United States. Based on a survey by the American Medical Association (AMA), 42% of physicians use some kind of EHR system, and it is estimated, that by 2015 the coverage will grow to over 80% [2]. Electronically collected biomedical data needs to be made available for research but at the same time patient privacy must be protected. This is a major challenge for the Healthcare Data and Knowledge Management field that has technical, management and policy implications.

Various approaches have been proposed to address privacy issues regarding publicly released data. A popular solution is to mask the original values of the attribute that could be used to identify individuals. Perturbation based masking methods add random noise to the original data values [3–8]. Data swapping techniques exchange attribute values between different records [9,10]. Generalization methods replace original values with more general ones [11–13]. Suppression is a special format of generalization when the value of an attribute is removed from the record. These masking methods can be used by themselves or as parts of more complex anonymization schemas.

Microaggregation and *k*-anonymity are two grouping based de-identification approaches that gained considerable popularity in recent years [14–17]. The main idea behind them is to partition the data into groups of similar records and then mask the quasi identifier attributes at group level so the records within a group become indistinguishable. Multiple solutions have been proposed to used as partitioning and masking methods to optimize these anonymization methods [18,12,19,20].

The process of privacy preservation causes information loss, which can be considered as loss of *utility*. To produce useful output the data publisher has to balance the competing requirements of sufficient privacy protection and maximum possible utility. Table 1 shows an example of utility loss in privacy preservation [21]. {*Age, Insurance, Zip*} can be used to identify individuals in the dataset (quasi identifiers). *Diagnosis* is a sensitive attribute. *Screening* shows whether the individual is targeted for colon cancer screening or not. Suppose that, in order to protect the sensitive attribute (*Diagnosis*), 2-diversity is required, so the quasi identifiers need to be modified in such a way that based on the quasi identifiers {*Age, Insurance, Zip*} each individual in the dataset would be indistinguishable from at least one other person. Tables 1(A) and (B) are both valid 2-anonymizations of the original data (records sharing the same quasi identifiers have the same Group IDs). However, Table 1(A) provides more accurate results than Table 1(B) when answering the following queries:

* Corresponding author.
*E-mail addresses:* tamas.gal@uky.edu (T.S. Gal), tct@kcr.uky.edu (T.C. Tucker), gangopad@umbc.edu (A. Gangopadhyay), zhchen@umbc.edu (Z. Chen).

1532-0464/$ - see front matter © 2014 Elsevier Inc. All rights reserved.
http://dx.doi.org/10.1016/j.jbi.2014.01.001

**Table 1**
Utility loss in privacy preservation.

| ID | Age | Insurance | Zip | Diagnosis | Screening |
|---|---|---|---|---|---|
| *Original data:* | | | | | |
| 1 | 54 | No | 40504 | HIV | Y |
| 2 | 55 | No | 40509 | HEP-B | Y |
| 3 | 60 | HMO | 40512 | SM | N |
| 4 | 60 | HMO | 40517 | HEP-B | N |
| 5 | 62 | HMO | 40524 | HEP-B | N |
| 6 | 62 | PPO | 40525 | Prostate cancer | N |

| Group ID | ID | Age | Insurance | Zip | Diagnosis | Screening |
|---|---|---|---|---|---|---|
| *De-identified data (A):* | | | | | | |
| 1 | 1 | [54–55] | No | 4050X | HIV | Y |
| 1 | 2 | [54–55] | No | 4050X | HEP-B | Y |
| 2 | 3 | 60 | HMO | 4051X | SM | N |
| 2 | 4 | 60 | HMO | 4051X | HEP-B | N |
| 3 | 5 | 62 | Private | 4052X | HEP-B | N |
| 3 | 6 | 62 | Private | 4052X | Prostate cancer | N |
| *De-identified data (B):* | | | | | | |
| 1 | 1 | [54–60] | Any | 405XX | HIV | Y |
| 2 | 2 | [55–62] | Any | 405XX | HEP-B | Y |
| 3 | 3 | [60–62] | HMO | 405XX | SM | N |
| 1 | 4 | [54–60] | Any | 405XX | HEP-B | N |
| 3 | 5 | [60–62] | HMO | 405XX | HEP-B | N |
| 2 | 6 | [55–62] | Any | 405XX | Prostate cancer | N |

Q1: *How many patients under age 59 are there in the data set?*
Q2: *Is an individual with Age = 55, Insurance = No, Zip = 40509 targeted for colon cancer screening?*

According to Table 1(A) the answer to Q1 is 2 and to Q2 is "Y". But according to Table 1(B), the answer to Q1 is an interval [0, 4], because 59 falls in the age range of record 1, 2, 4, and 6. The answer to Q2 is "Y" or "N" with 50% probability each.

Two conclusions can be drawn from this example:

- Different anonymization leads to different information loss. Tables 1(A) and (B) are on the same anonymization level but Table 1(A) provides better results. Therefore, utility loss should be minimized in privacy preserving.
- Data utility depends on the application. Q1 is an aggregate query, so the data is more useful if the values are more accurate. Q2 is a classification query so the utility of the data depends on how much the classification model is preserved in the de-identified data. Utility is the quality of the data for the intended use.

To decide whether one de-identification method preserves utility better than another, we need to measure utility of the de-identified data compared to the utility of the original data. In practical terms it means that we need to define a *distance measure* between the original data and the de-identified data based on utility. The content of this distance measure depends on the use of the data.

The followings are examples of utility measures used in the literature:

- *Query answering accuracy*: Answering queries such as *count, average* and *sum* is the most common use of published data. The quality of query answering depends on the distance of each original value from the corresponding value in the anonymized dataset. A quantitative measure was introduced by Xu et al., which uses the normalized interval size to measure the utility loss for numeric attributes and normalized number of descendants in the generalization hierarchy to measure the utility loss for categorical attributes [22,23].
- *Classification accuracy*: The published data is often used to train classifiers, therefore the data quality depends on how well the class structure is preserved in the anonymized data. Fung et al. propose a metric that measures entropy change during anonymization [24,25]. Ideally, the entropy of an equivalence class with respect to class label distribution should be minimized in the published data.
- *Distribution similarity*: Statistical distribution is an important characteristic of a dataset. A model which measures the difference between the distribution of the original and the anonymized data has been developed by Kifer et al. [26].
- *Discernibility measure:* Bayardo and Agrawal consider a discernibility measure as a utility measure as they try to minimize the equivalence class size while anonymizing the data [27]. The more records are in an equivalence class, the less specific information is preserved for those records.
- *Generalization measures* include *Generalization Height* [28], which measures the total number of generalization steps applied in the anonymization process. The idea behind this measure is that generalization causes information loss and the total number of generalization steps represents the total amount of loss. The *Loss Metric* penalizes the generalization made in that entry according to the size of the generalized subset [29,30]. *Ambiguity Metric* is the average size of the Cartesian products of all generalized entries in each record in the table [30].
- *Entropy based measures:* Gionis and Tassa introduced entropy as Mutual Information Utility Measure [31]. Private Mutual Information Utility Measure builds on the previously mentioned entropy measure and it quantifies the mutual information between the generalized public data and the private data [19].

The same de-identified dataset might be useful for one purpose but useless for another. When researchers request de-identified biomedical data, they already have a plan how they want to use it. Yet, these research plans are rarely utilized when choosing the de-identification method. We believe that de-identification methods should be tailored to the specific needs of the data recipient when possible and that this customization should reflect in utility measurements as well.

We present a de-identification framework to address the need for customized anonymization. Our approach investigates the requirements of the data recipient and selects a suitable de-identification method that is specific to the requirements. We evaluated our method by comparing it to three general purpose de-identification algorithms using utility measures that were specific to the data recipient's requirements.

Our experiments used real cancer surveillance data provided by the Kentucky Cancer Registry.

The rest of the paper is organized as follows: Section 2 gives a detailed review of related work. Section 3 explains the materials and methods used in our experiments. Section 4 describes our results. Section 5 discusses some of the issues that arose during our experiments and Section 6 concludes the paper and provides directions for future work.

## 2. Related work

Most medical providers follow the Safe Harbor standard [32] in the US when releasing data which removes 18 well defined identifiers from the dataset. Sweeney showed that removing obvious identifiers does not provide protection against privacy attacks [33]. As a solution, *k-anonymity* was proposed by Samarati and Sweeney [11]. *k*-anonymity divides the data attributes into *quasi identifiers*, *sensitive attributes* and *non-sensitive attributes* and creates equivalence classes by masking quasi identifier attributes in such a way that the quasi identifier attributes of any record would be identical to quasi identifier attributes of at least $k-1$ other records. Achieving optimal *k-anonymity* is NP-hard

[34,27,35] and even though it effectively prevents identity disclosure, it does not prevent attribute disclosure. To address this weakness Machanavajjhala et al. introduced *l-diversity* [36]. *l*-diversity, in addition to *k*-anonymity, requires each equivalence class to have at least *l* unique values for the sensitive attribute. *k*-anonymity and *l*-diversity protect privacy effectively enough when the sensitive attribute is a categorical variable. With a numerical sensitive attribute however, an adversary might infer a "close enough" value to jeopardize privacy. To address this issue Li et al. introduced *t-closeness* [37], which requires that, for each equivalence class, the distance between the distribution of the sensitive attribute in the class and the distribution of the sensitive attribute in the whole dataset must be smaller than a preset value *t*.

*k*-anonymity based techniques have multiple practical pitfalls. They focus on datasets with categorical quasi identifier attributes such as race or gender. When the dataset contains numerical quasi identifier attributes, they are converted to categorical variables (e.g. age can be converted to age intervals) [38]. Furthermore, the quasi identifier values are generalized or suppressed to achieve homogeneity of the equivalence classes [28,18,39]. These conversions lead to considerable loss of information in the de-identified dataset [40,26]. Implementation of *k*-anonymity, *l*-diversity and *t*-closeness together leads to competing requirements resulting in large class sizes with undesirable amount of information loss [36,37]. Furthermore, most implementations of *l*-diversity and *t*-closeness assume that there is only one sensitive attribute in the dataset. Extension of these principals to multiple sensitive attributes is not trivial and, again, leads to unreasonably large class sizes with significant information loss [41–43].

Microaggregation is another privacy preserving approach comparable to *k*-anonymity. Microaggregation focuses on numerical quasi identifier attributes, though categorical quasi identifier attributes can be converted to binary dummy variables and handled as numerical attributes [38]. Microaggregation clusters records in the dataset such that similarity among data points inside the clusters is minimized and similarity among data points in different clusters is maximized. Each cluster contains at least *k* records, just like in *k*-anonymity. The quasi identifier values are masked in a way that is relevant to the cluster, they can be replaced with the cluster averages for example [44–46]. This way the quasi identifier values become uniform, making individuals indistinguishable in a cluster. Achieving optimal multivariate microaggregation is NP-hard [47].

Microaggregation methods consist of two main steps, clustering and masking. These steps can be manipulated independently to minimize information loss. Clustering algorithms aim to achieve optimal microaggregation and the level of information loss is usually the result of a trade-off between performance and time complexity of the algorithm. Several heuristic clustering algorithms have been proposed for microaggregation. Laszlo and Mukherjee used a minimum spanning tree based method for clustering [44]. Chang et al. introduced a two-phase algorithm called Two Fixed Reference Points (TFRP) [48]. Panagiotakis et al. proposed a successive group selection method based on sequential minimization of SSE (sum of the within-partition squared error) [49]. The most common way of masking in microaggregation is to replace the quasi identifier attribute values with the cluster averages [44–46] which reduces variance and distorts covariance in the data. To address this issue Domingo-Ferrer et al. proposed *R-microhybrid*, a method that replaces the original data with synthetic data generated based on the mean and covariance of the original data in each group. Since the mean and covariance are preserved in each group, the mean and covariance of the entire data set are also preserved [50]. Li et al. offer a microperturbation based solution by replacing the data for each group using a statistical distribution with the mean equal to the group average and some random noise that represents the distortion in variance–covariance statistics caused by the group average substitution [20].

There has been some research on privacy protection techniques specialized for medical data and e-health. El Emam et al. published extensively about privacy protection in the medical field [51–53,15,54,16,55–59,17]. Benitez et al. described a lattice based automatic policy discovery algorithm which creates optimal de-identification policies to replace HIPAA's static Safe Harbor [60]. A specialized anonymization technique was proposed to prevent patient re-identification through linking standard diagnosis codes by Chen et al. [61]. Durham et al. published about their research on using cryptographic techniques to link data across different health care providers [62]. There is little privacy research in the medical field that shows interest in the utility of the de-identified data based on the needs of the data recipient.

## 3. Material and methods

To evaluate our data recipient centered framework we selected a test project where we worked together with biomedical researchers at the University of Kentucky. These researchers requested de-identified cancer surveillance data to evaluate risk factors in cancer. Together with the researchers we created a set of requirements to measure utility. These requirements were very specific to the statistical analyses outlined in their research plans:

- The de-identified data should be in the same data space and should use the same dimensions as the original dataset. Many anonymization techniques transform the data into new data space using PCA [63,64]. While the reduced dimensionality might make data mining easier for a data mining specialist, it adds an extra layer of complexity that makes the dataset useless for the medical researcher. They prefer to use their standard variables with their original permissible values.
- Basic statistics (average, sum, median) should not differ significantly after de-identification.
- Selected statistical analysis should give *similar* results when performed on the original data and on the anonymized data. The definition of *similar* in this case is the following:
  - Variables should not change significance in a statistical model. In other words, if a variable is significant when an analysis is performed on the original data, it should remain significant when the same analysis is performed on the anonymized data. The same way, if a variable was not significant for the original data, it should not be significant for the de-identified data either when the same analysis is performed.
  - Coefficients should not change direction for significant variables. This means that if a coefficient is positive for a significant variable when the targeted statistical analysis is performed on the original dataset, it should be positive when the same analysis is performed on the de-identified dataset. It should be true the same way for negative coefficients.
  - The values of the corresponding coefficients should be "close" for significant variables when the same analysis performed on the original and on the de-identified data. The definition of "close" is arbitrary in this case and depends on the actual value of the coefficient. In our experimental evaluation we checked whether the value of a coefficient from the statistical analysis performed on the de-identified data was in the 95% confidence interval of the corresponding coefficient from the same statistical analysis performed on the original data.

Following the research plans of the data recipients, three statistical methods were examined:

- Linear regression
- Logistic regression
- Cox's proportional hazards model

These models are frequently used in biomedical research.

We considered *k*-anonymity and microaggregation based methods when designing the customized de-identification approach for this project. *k*-anonymity based methods use generalization and suppression to mask data values, it was unacceptable based on our requirements. We decided to use a modified version of the *Condensation* method introduced by Aggarwal et al. [46]. Condensation clusters similar records into groups just like microaggregation techniques do. However, instead of masking only the values of quasi identifier attributes in the groups, condensation replaces the values of all attributes with synthetic data that was randomly generated based on the statistical attributes of the original data. This decision is justified by the observation that the traditional classification of attributes as Quasi Identifiers, Sensitive Attributes and Non-Sensitive Attributes is not always trivial. For example, let us imagine a scenario where a celebrity is admitted to a hospital. Let us also imagine that a pre-existing medical condition of this celebrity is public knowledge. If a dataset, that was de-identified using a *k*-anonymity or microaggregation based method, is released from the hospital for this time period, we would probably be able to find the group where the celebrity belongs based on the publicly known demographics of the celebrity. If both the known pre-existing condition and the new diagnosis are listed in the dataset without masking then we could infer the cause of the hospitalization with high probability.

Our new anonymization techniques were evaluated by creating de-identified datasets and performing the above listed statistical analyses both on the original and the de-identified datasets. The results of the analyses were compared.

### 3.1. The original condensation method

Aggarwal described the condensation method in [46]. The technique generates a synthetic dataset based on the distribution of the original dataset. Similarly to previously discussed models the condensation algorithm creates groups in which the records would be indistinguishable. Instead of masking the values inside the groups, the condensation model creates synthetic random data based on the statistical characteristics of the original data.

**Definition 1** (*Indistinguishability level* [65]). A pseudo-dataset $D$ generated from the original dataset $D$ is said to be *k-indistinguishable*, if every record $\overline{X}$ in $D$ can be mapped to at least $K$ records $M(\overline{X})$ in $D$. The record $\overline{X}$ is generated from $M(\overline{X})$ using a randomized algorithm which treats all records in $M(\overline{X})$ symmetrically. Therefore, $\overline{X}$ is equally related to all records in $M(\overline{X})$.

Next, we will describe the algorithm. It consists of two main steps.

- First, the data is condensed into multiple groups with size of at least $K$,which is referred to as the *indistinguishability level.* The greater the indistinguishability level, the greater the amount of privacy. At the same time, a greater amount of information is lost because of the condensation of a larger number of records into a single statistical group entity. The groups need to be created in such way that the data points in the same group are close to each other.
- Second, mean and covariance statistics are computed for each cluster (condensation unit). The statistics of the cluster are used to create pseudo data that preserves the mean and covariance statistics of the original data.

**Generalization of the group creation problem:** We would like to create groups of records in such way that each group represents a tight cluster of data points, with each cluster containing at least $K$ data points. We would like to minimize an objective function $W()$ which measures the average tightness within the clusters. An example of such function could be the average intra-cluster distance between the data points, or the average centroid radius of the clusters. For a given database $D$, partition it into $s = \lfloor N/K \rfloor$ groups $C_1 \dots C_s$ of at least $K$ data points each, so that the objective function $W(C_1 \dots C_s)$ is minimized.

**Theorem 1** (*Condensation Problem*). *The condensation problem is NP-hard.*

**Proof.** This is an instance of the *balanced clustering* problem, which is NP- hard [66,67], even for $K = 3$ for minimizing the intra-cluster distance.

A heuristic algorithm has been designed to overcome the hardness problem **[65]**. The method is shown in Algorithm 1. The input of the algorithm is the dataset $D$ and indistinguishability level $K$. In step 2a, a random seed is chosen from $D$ to start a group. In step 2b the $K-1$ closest data points are added to the group. In step 2c the group is removed from the dataset. This process is repeated $s = \lfloor N/K \rfloor$ times until the dataset is empty. Note, that a maximum of $K-1$ records can remain after the last iteration. In step 4 these records are added to their closest clusters. In step 5 the mean and covariance statistics are calculated for each cluster and in step 6 the pseudo data is generated based on those statistics. Note, that step *7* was added to transform the resulted pseudo data to the same format as the original data, which is a requirement in our experiments.

**Algorithm 1.** The original Condensation Algorithm [46,65]

Condensation (dataset $D$,$K$)
1. Let $n$ be the size of $D$. Let $s = \lfloor n/K \rfloor$ be the number of clusters
2. For $i = 0$ to $s$
   (a) Randomly select $\overline{X_i}$ seed from the $D$. Move $\overline{X_i}$ to $C_i$ cluster.
   (b) Select $K-1$ data points closest to $\overline{X_i}$. Move them to $C_i$ cluster.
   (c) $D = D - C_i$
3. End For
4. Add the remaining (max $K-1$) data points to their closest cluster.
5. Compute mean and covariance statistics for each cluster.
6. Generate synthetic data for all attributes in each cluster using the computed statistics.
7. (Addition by our requirements): Transform the data values in the synthetic data to their original permissible values.

**Categorical variables:** Aggarwal et al. extend the model to datasets with categorical attributes by using histograms instead of means and frequencies of co-occurrences instead of covariance [65]. However, this method assumes, that all attributes are categorical in the dataset because it is not possible to mix scalar covariances with categorical frequencies to calculate dataset wide statistics. Therefore, even though our data included categorical attributes, we did not use this method. Instead, we converted the categorical attributes to binary dummy variables which were afterwards used as numerical attributes both at synthetic data generation and in our empirical evaluation with the regression models.

### 3.2. The improved method

To be able to design a de-identification algorithm that preserves data utility specific to the statistical models listed previously, we need to examine those models closely. We would like to find the statistical attributes which need to be preserved so these models would give similar results when run on both the original data and on the synthetic data.

- *Linear regression* is an approach to model the relationship between a scalar variable $y$ and one or more variables denoted as $X$. Let us suppose that we have a patient dataset which contains a number of attributes ($m$) about patients, such as age, gender, race, geographic location, cancer stage, etc. (independent variables). Let us also suppose, that we have a scalar variable $y$, such as one that measures a diagnostic bio-marker in the blood. We are interested in the relationship between the independent variables and this diagnostic variable. In linear regression, the model is:

$$y = X\beta + \varepsilon \tag{1}$$

where:
  - $y$ is the observations of the dependent response variable,
  - $X$ is a $N \times m$ data matrix as observations of independent variables,
  - $\beta$ is the coefficient of the model that we are trying to find,
  - $\varepsilon$ is the error.

  If ordinary least square method is used, the estimated $\hat{\beta} = (X^T X)^{-1} X^T y$. Here $X^T X$ is the covariance matrix of the data if the mean of each variable is set to zero. $X^T y$ is the covariance between every column vector of $X$ and the response variable column $y$. So if we can preserve the covariance of the whole data set (which includes both independent and response variables), we can preserve the linear regression model.
- *Logistic regression model* can be used in similar situations as the linear regression model, the difference being that the the response variable is binary (with value 0 or 1). Let $p_i$ be the probability of response 1 for patient $i$ with independent variable vector $X_i$. The logistic model is:

$$logit(p_i) = ln\left(\frac{p_i}{1-p_i}\right) = X_i\beta + \varepsilon \tag{2}$$

This can be seen as a modified linear regression model so the requirement is the same as for the linear regression model.
- *Cox's proportional hazards model* is widely used in biostatistics. It allows analysis of the effect of several risk factors on survival. Let us suppose, that we have a patient dataset which contains a number of attributes ($m$) about patients, such as age, gender, race, geographic location, cancer stage, etc. (independent variables). Let us also suppose, that the dataset also contains survival status and survival time. The attribute values, excluding survival status and survival time, for a patient $i$ are represented by an $m \times n$ matrix $X$. The probability of the endpoint (death, or any other event of interest, e.g. recurrence of disease) is called the hazard. The hazard is modeled as:

$$h(t|X) = h_0(t)e^{\beta^T X} \tag{3}$$

where:
  - $t$ is the time of the endpoint (survival time),
  - $h(t)$ is the hazard of dying at time $t$,
  - $X$ is the matrix of the independent variables (covariates) that affect the hazard,
  - $\beta$ is the coefficient of the model that we are trying to find,
  - $h_0(t)$ is the baseline hazard. It indicates the instantaneous risk for the respective individual when all independent variable values ($X_i$) are equal to zero.

  We can easily linearize this model:

$$ln\frac{h(t)}{h_0(t)} = \beta^T X \tag{4}$$

We now have a fairly "simple" linear model that can be readily estimated.

**Assumptions.** While no assumptions are made about the shape of the underlying hazard function, the model equations shown above do imply two assumptions.

  - They specify a multiplicative relationship between the underlying hazard function and the log-linear function of the covariates. This assumption is also called the *proportionality assumption*. In practical terms, it is assumed that, given two observations with different values for the independent variables, the ratio of the hazard functions for those two observations does not depend on time.
  Consider, two observations $i$ and $i'$ that differ in their $X$ values, with the corresponding linear predictors:

$$\eta_i = \sum_{i \in D} \beta X_i \tag{5}$$

and

$$\eta_{i'} = \sum_{i \in D} \beta X_{i'} \tag{6}$$

The hazard ratio for these two observations is:

$$\frac{h_i(t)}{h_{i'}(t)} = \frac{h_0(t)e^{\eta_i}}{h_0(t)e^{\eta_{i'}}} = \frac{e^{\eta_i}}{e^{\eta_{i'}}} \tag{7}$$

As we can see, the ratio of the hazard functions for those two observations does not depend on time.
  - The second assumption of course, is that there is a log-linear relationship between the independent variables and the underlying hazard function.
  To build the model we need to compute $\beta$. Suppose $R_j$ is the set of patients who are at risk at time $j$ (i.e., they died at some time $t_k > t_j$) and patient $i$ died at time $t_i$. The conditional probability of patient $i$ died at time $t_i$ given one of the patients at risk at $t_i$ died equals

$$P(i \text{ died at } t_i | \text{someone died at } t_i) = \frac{h_i(t)}{\sum_{l \in R_j} h_l(t)} = \frac{e^{\beta^T X_i}}{\sum_{l \in R_j} e^{\beta^T X_l}} \quad (8)$$

It is important to note that the baseline hazard function is canceled out. We can find the coefficient vector $\beta$ that maximizes the product of these conditional probabilities (i.e., maximum likelihood). To approximately preserve the model ($\beta$), we need to ensure that:

1. The risk set $R_j$ is preserved.
2. The relative distance (similarity) between at risk patients are approximately preserved.

The first condition is straightforward because $R_j$ is used in computing the conditional probability. Preserving the risk set $R_j$ means that the survival status and survival time attributes need to remain unchanged.

We use an example to illustrate the second condition. Consider three patients $P_1$, $P_2$, and $P_3$. Suppose their survival times are 12 months, 13 months, and 30 months, respectively. Patient $P_1$ and $P_2$ also have more similar attributes compared to $P_3$. Clearly, when we anonymize the data, we can blur the difference between $P_1$ and $P_2$ which will not introduce much distortion, but we should not blur the difference between $P_1$and $P_3$.

### 3.3. Implementation of our model

We developed the following privacy protection method (Algorithm 2) that achieves the above conditions. This method first divides the data set into two subsets $D_0$ and $D_1$, where $D_0$ contains the patients who died and $D_1$ contains those who are still alive. Only patient records in $D_0$ will have impact on the proportional hazard model because we do not know the outcome for those who are still alive. Thus, it does not make sense to mix patients in $D_0$ with $D_1$ and the method will anonymize $D_0$ and $D_1$ separately.

The method then runs the k-means clustering algorithm [68] to generate $\lfloor n_i/k \rfloor$ clusters on each subset $D_i$ $(i = 0, 1)$. The objective of k-means clustering is to minimize the average squared Euclidean distance of data points from their cluster centers, where a cluster center is defined as the mean or centroid of the data points in a cluster.

Survival time is the response variable in survival analysis with proportional hazards model and it is important that patients with similar survival time be clustered together. For that reason, we used weighted Euclidean distance function to calculate the distances among the data points, with weight $w$ assigned to survival time and the rest of the weight evenly distributed across other attributes. Here, $w$ is a parameter that can be fine tuned to perfect the clustering process. In our experiments, $w = 0.5$ gave good results. Different models with different response variables might need to set the weights accordingly.

Since some clusters may have fewer than $k$ points and some may have more, in step 2c the method moves points from larger clusters to smaller ones to make sure each cluster has at least $k$ points. Step 2d ensures privacy protection by generating synthetic data preserving the mean and covariance statistics of each cluster. Thus, the difference between patients in the same cluster is blurred. Since there are at least $k$ patients in each cluster $k$-anonymity (or $k$ level indistinguishability by Definition 1) is satisfied. Next, we show that this method satisfies the two conditions necessary to approximately preserve the proportional hazard model.

**Algorithm 2.** Improved Anonymization Algorithm for Proportional Hazards Analysis

Improved-Condensation (dataset $D, k$)

1. Divide $D$ into two data sets $D_0$ and $D_1$ such that $D_0$ contains patient records with survival status equals 0 and $D_1$ contains patient records with survival status equals 1. Let $n_0$ and $n_1$ be the size of $D_0$ and $D_1$.
2. For $i = 0$ to *1*
   (a) Run k-means clustering on $D_i$ to generate $\lfloor n_i/k \rfloor$ clusters using weighted Euclidean distance, where weight $w$ is assigned to the response variable in the statistical model targeted, and the rest of the attributes receive equal weight.
   (b) Sort the clusters in ascending order of cluster size. Let them be $C_1, C_2, \ldots, C_s$.
   (c) For each cluster $C_j$ that contains less than $k$ patients, find $k - |C_j|$ patients closest to the center of $C_j$ that lie in clusters that contain more than $k$ patients. Move these patients to cluster $C_j$.
   (d) For each cluster $C_j$
      i. Synthetic-Data-Generation: Compute mean and covariance statistics. Generate synthetic data for all attributes in each cluster using the computed statistics except that the survival status attribute is not changed. (See Algorithm 3)
      ii. Sort the newly created synthetic data clusters by Survival Time. Also sort the original Survival Time values the same way. Assign back the original Survival Time values to the synthetic dataset, replacing the smallest synthetic Survival Time value with the smallest original Survival Time, so on. This way we ensure that the synthetic dataset preserves the original Survival Time values and also the correlations are preserved in the synthetic dataset.
      iii. Transform the data values in the synthetic data to their original permissible values.
   (e) End For
3. End For

Note, that for proportional hazards analysis, the risk set $R_j$ depends on both survival status (only those patients who already deceased are in the risk set) and survival time (any patient in the risk set who survives longer than time $t_j$ belongs to $R_j$). The above algorithm does not modify survival status and survival time, thus the risk set $R_j$ is preserved. For condition 2, the distance function used in the clustering step considers both the similarity based on survival time and the similarity based on other attributes, so patients with similar survival time and other similar attributes are more likely to be assigned to the same cluster. Thus this condition is satisfied as well. For example, using the example from the previous section, patients $P_1$ and $P_2$ are likely to be assigned to the same cluster while $P_3$ is likely to be assigned to another.

Our method bears some similarity with the condensation approach [46,65]. However, the condensation approach has three drawbacks that needed to be improved:

- The condensation algorithm picks cluster centers randomly, so it may generate inferior clusters. We improved the algorithm by using k-means clustering, which is reasonably efficient in the sense of within-class variance [68]
- The condensation algorithm assigns equal weight to all attributes, including survival time, so patients with similar survival time may not be assigned to the same cluster. Our algorithm

uses weighted Euclidean distance at clustering, with significantly larger weight assigned to survival time than other attributes to ensure that records with similar survival time are clustered together.
- The condensation algorithm does not consider the difference between patients who are in the risk set (deceased) and patients who are not (alive). Our algorithm separates these two classes and de-identifies them independently.

The synthetic data generation method (point 2di in Algorithm 2) is described in Algorithm 3. The input is a cluster that contains similar data points. In step 1 the data in the cluster is shifted to a new data space using Principal Component Analysis (PCA). In this space, the data components $Z_1, Z_2, \ldots Z_p$ are independent. In step 2 random data $Z'_i$ is generated with the statistical characteristics of $Z_i$. In step 3 the random independent $Z'_1, Z'_2, \ldots Z'_p$ components are combined into one dataset $Z'$. Finally in step 4 $Z'$ is shifted back to the original data space using reverse PCA and $C'_j$ is created. $C'_j$ has the same attributes as $C_j$ had. The means and the covariances among the attributes are preserved.

**Algorithm 3.** Synthetic Data Generation

Synthetic-Data-Generation ($C_j$)
1. Using Principal Component Analysis (PCA) shift the data in the cluster into a new space ($C_j \rightarrow Z$), creating independent components $Z_1, Z_2, \ldots Z_p$.
2. For each independent component $Z_i$
   (a) Generate random data $Z'_i$ with normal distribution such way that
   $|Z'_i| = |Z_i|$,
   $\mu_{Z'_i} = \mu_{Z_i}$ and
   $\sigma_{Z'_i} = \sigma_{Z_i}$
3. Combine $Z'_1, Z'_2, \ldots Z'_p$ into one dataset $Z'$ in an orderly manner
4. Using reverse PCA shift $Z'$ back to the original data space ($Z' \rightarrow C'_j$)
5. Return $C'_j$

The algorithm was implemented using the R statistical software. The data was stored in a MySQL database.

### 3.4. Complexity of our model

Let $n$ be the number of records, $m$ be the number of attributes and $s = \lfloor n/K \rfloor$ the number of clusters. The cost of k-means clustering is $O(Isnm)$ [69], where $I$ is the number of iterations run to refine clustering. Even though the complexity is linear to the number if iterations, the number of clusters, the number of records and the number of attributes, it can be costly if the number of iterations gets out of control, so statistical software packages usually offer a parameter to set the maximum number of iterations to a reasonable limit. In the R statistical software it is set to 10 by default. The cost of PCA for one cluster is $O(Km^2)$, as $K$ is the approximate number of records in the cluster. This needs to be done $s$ times, so the cost of computing PCA for the whole dataset is $O(sKm^2) = O(nm^2)$. So the cost of the whole algorithm is $O(Isnm + nm^2) = O((m + Is)nm)$.

PCA is causing the algorithm to be square in terms of the number of attributes. This is usually not a problem in patient datasets where $n \gg m$. For high dimensional datasets where $n \ll m$, such as genetic or imaging data, we can use a method that is referred to as the *PCA transpose trick* and has been used when generating eigenfaces [70–72]:

**Proposition 1.** *Let us suppose that we have an $m \times n$ observation matrix A, where $n \ll m$. To find the PCA of A, we need to compute the eigenvectors of the large $m \times m$ covariance matrix $A^TA$, which is computationally difficult. Instead, we can compute the eigenvectors of the $n \times n$ matrix $AA^T$, because if $v$ is an eigenvector of $AA^T$, then $A^Tv$ is an eigenvector of $A^TA$.*

**Proof.** Let $v$ be an eigenvector of $AA^T$ with eigenvalue $\lambda$. Then

$$(AA^T)v = \lambda v$$
$$A^T(AA^Tv) = A^T(\lambda v)$$
$$(A^TA)(A^Tv) = \lambda(A^Tv)$$

so $A^Tv$ is an eigenvector of $A^TA$, with eigenvalue $\lambda$. Therefore, instead of computing the eigenvectors of $A^TA$ directly, we can compute the eigenvectors of $AA^T$ and multiply those from the left by $A^T$.

This way the complexity of PCA can either be $O(nm^2)$ or $O(n^2m)$, depending on the dimensionality of the data. So the cost of the whole algorithm is either $O((m + Is)nm)$ or $O((n + Is)nm)$.

### 3.5. Experimental evaluation

We developed a utility based de-identification method based on the preservation of specific statistical qualities of the data. The promise of doing this is that statistical models based on these preserved statistics would return comparable results when run on the anonymized dataset, as if they were run on the original data. In the experimental evaluation we prove that our model:

- *Works reliably and is able to produce statistically consistent output.*
- *Is scalable in terms of cluster size (K).*
- *Is scalable in terms of* the number of attributes.
- *Is scalable in terms of the size of the dataset.*

The experimental evaluation was conducted with the following setup:

- **The computational environment:** The tests were run on a virtual machine in a Dell PowerEdge R610 virtual environment with 4 CPU cores and 8 GB RAM assigned to the virtual machine. The operating system was Ubuntu Linux 10.04 Server. The algorithms were implemented in the R Statistical Package. The data was stored in a MySQL database.
- **The data:** Two cancer surveillance datasets were used for testing. The first contained data of colon cancer ($N = 9,552$), and the second of lung cancer ($N = 17,421$) patients who were diagnosed between 2004 and 2009. The variables in the datasets were:
  - VitalStat (vital status): Whether the patient was alive at the time of last contact (categorical)
  - SurvInterval (survival time): The time elapsed between the time of the diagnosis and either the time of death or the time of last contact (scalar – integer)
  - DiagAge (diagnosis age): Age at diagnosis (scalar – integer)
  - Gender: The gender of the patient (categorical)
  - Race: The race of the patient (categorical)
  - Stage: Cancer stage at the time of the diagnosis (categorical)
  - Appalachia: Whether the patient lives in Kentucky's Appalachian region. The Appalachian region is a predominantly rural area with high poverty rate, lower education levels and less access to health care. Cancer rates are the highest in the Appalachian region in the whole United States (categorical)
  - TobaccoUse (tobacco usage): Whether the patient used tobacco products or not (categorical)

- PrimaryPayor: Who pays for the patient's medical care. Permissible values include {private insurance, federal program, uninsured}.

- **Statistical models:** The following statistical models were used in the R statistical software:
  - Linear regression:
    Stage, Gender, Race, Appalachia, TobaccoUse and PrimaryPayor were used as predicting variables with SurvInterval (survival time) as the response variable. The research question was: *How do cancer stage, gender, race, geographic location, tobacco usage and health insurance status affect the survival time of the patients?* Categorical attributes (Stage, Race and PrimaryPayor) were changed to binary dummy variables. Only the records where *VitalStat = 0* (deceased patients) were used for this analysis.
  - Logistic regression:
    The same variables were used as for linear regression with the difference of variable SurvInterval, which was converted to binary variable here (low or high survival time). The rational here is to build a model that predicts whether a patient would have low or high survival time. Only the records of deceased patients were used for this analysis as well.
  - Cox's Proportional Hazards Model:
    DiagAge, Gender, Race, Appalachia, Stage, TobaccoUse and PrimaryPayor were used to build a model to assess the risk factors contributing to the death of cancer patients. SurvInterval (survival time) and VitalStat (vital status) form a survival object in this survival analysis.

- **Metrics**: Our requirement was to preserve information in the data, such that specific statistical analyses would yield similar results when run on the original and the de-identified data. We evaluated our method by measuring the change in the parameters of the statistical models (coefficients) before and after anonymization. The followings metrics were reported after the statistical models were built based on both the original and the synthetic datasets:
  - Percentage of coefficients changed significance.
  - Percentage of significant coefficients changed direction.
  - Percentage of the new coefficients were out of the 95% confidence interval of the original coefficients.

We used *conditional privacy* to measure the privacy of the de-identified data [73]. Conditional privacy is an average measure of privacy that was originally proposed in context of distribution reconstruction after additive perturbation. The measure is based on the differential entropy of a random variable. The differential entropy of $A$, given $B = b$ is:

$$h(A|B) = -\int_{\Omega_{A,B}} f_{A,B}(a,b)\log_2 f_{A|B=b}(a)\,da\,db \tag{9}$$

where $A$ is a random variable describing the data, and $B$ is another random variable giving information on $A$. $\Omega_{A,B}$ defines the domain of $A$ and $B$.
The average conditional privacy of a random variable $A$, given $B$, is:

$$\Pi(A|B) = 2^{h(A|B)} \tag{10}$$

This measure will be used in the context of $A$ to be a random variable in the original data and $B$ the corresponding random variable in the de-identified data. If conditional entropy between A (original data) and B (synthetic data) is zero then A and B are identical so there is no privacy preserved. The greater the conditional privacy, the greater the privacy protection is.

- **Algorithms:** We compare our algorithm to a commercially available anonymization system, the original condensation method and the TFRP algorithm. The following naming conventions were used:
  - *Commercial*: We used a commercially available de-identification software which achieves $k$-anonymity through a heuristic algorithm using generalization and suppression.
  - *Condensation*: The original condensation algorithm [46,65].
  - *TFRP*: Two Fixed Reference Points (TFRP) method [48]. We chose TFRP in our comparison as it is one of the fastest microaggregation algorithms that achieved similar utility as other slower algorithms in the field. TFRP is a two-phase method for microaggregation. In the first phase, TFRP uses the pre-computing and median-of-medians techniques to shorten its running time. In the second phase, TFRP generates variable-size groups by removing the lower homogeneous groups to reduce the number of groups and to improve the data quality. The time complexity of this algorithm is $O(n^2/k)$.
  - *Improved*: This is our algorithm, which is an improved version of the condensation method.

To make sure that our algorithm performs consistently and that the resulting datasets are similar, we repeated each test cycle one hundred times including the synthetic data generation. Each cycle was evaluated independently and the averages are reported here.

## 4. Results

### 4.1. Utility preservation

For this first test we used $K = 100$. We ran the three statistical analyses on all original and de-identified datasets and compared the coefficients from the models resulted by analyses run on the original datasets to the coefficients from the models resulted by analyses run on the anonymized datasets. For the proportional hazards model we compared the $exp(coef)$ values (the exponential values of the coefficients) as they give the hazard ratios and those are the parameters from the model that are used in practice.

- **The percentages of the coefficients that changed significance:** Fig. 1 shows the percentages of coefficients that changed significance when anonymized. For this measure, the smaller percentage means better utility preservation. Preserving significance is important for researchers to decide which variables have real effect on the outcome.
  As Fig. 1 shows, our improved algorithm performed the best, with performance between 1.85% and 26%.

- **The percentage of the significant coefficients that changed direction:** Fig. 2 shows the percentages of significant coefficients that changed direction after de-identification. For this measure again, the smaller percentage means better utility preservation. The direction of a coefficient gives information about the direction of the change in the outcome given the corresponding attribute changes and as such it is important to preserve after de-identification. It would tell us for example, whether smoking has positive or negative effect on survival time. For the proportional hazards model the exp (coef) values are used in practice. The exp (coef) values show how the change of an attribute affects the overall risk. When the coefficients change direction in this case, the exp (coef) values change whether they are smaller or larger than one. Since the hazards are proportional to the exp (coef) values, they change the risk

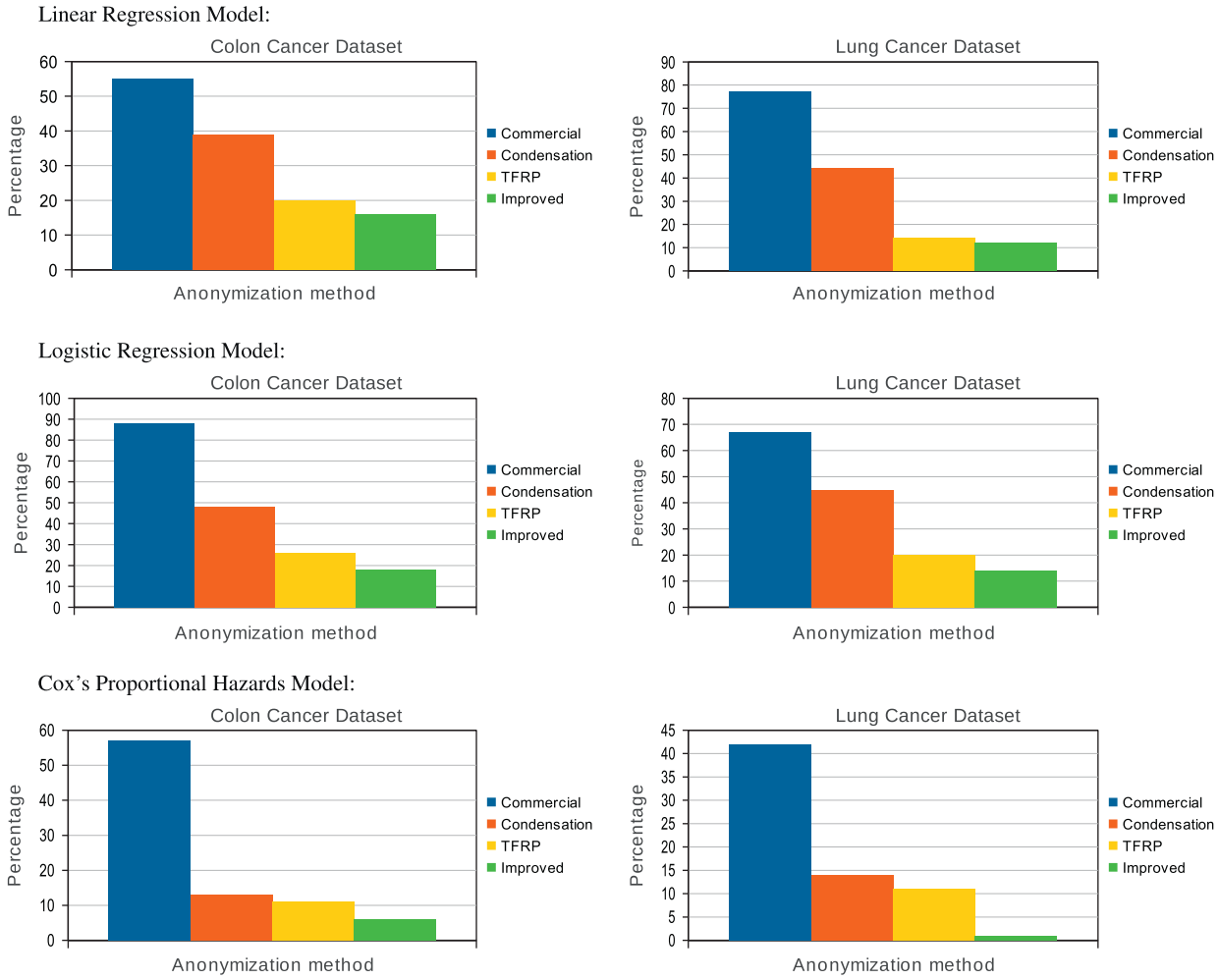


**Fig. 1.** Coefficients changed significance.

in a positive way if they are larger than one and negative way if they are smaller.

As Fig. 2 shows, our improved method performs best among the de-identification methods, and it works significantly better for the lung cancer datasets. A possible explanation is that since lung cancer is much more aggressive and shorter in duration, these effects might be more exaggerated with less co-morbidity. Therefore the relations are clearer and they are easier to pick up for analysis.

- **The percentage of the coefficients that were outside of the 95% confidence intervals of the original coefficients:** The actual value of the coefficients tell the multiplicative effect of the variable on the outcome (Again, the smaller the percentage, the better the utility preservation). We consider it significant distortion in the statistical model if a coefficient in the de-identified model deviates from its twin coefficient in the original model by more than the 95% confidence interval of the original coefficient. Fig. 3 shows, that our improved method performs best among the algorithms.

Although TFRP and our improved method have similar utility for linear regression and logistic regression, our method significantly outperforms TFRP for the Cox regression model, probably because our clustering algorithm is specifically tuned to preserve Cox regression model (e.g., our method puts more weight on survival status and survival interval and never puts patients with different survival status into the same cluster).

- **Privacy:** Fig. 4 shows the conditional privacy measures for the de-identified datasets. Based on this measure the Condensation, TFRP and Improved algorithms provide similar privacy protection.

### 4.2. Scalability

We compared the scalability of the tested de-identification methods in three areas:

- *Cluster size* (varying $K$)
- *Number of variables* (varying $m$)
- *Number of records* in the dataset (varying $n$)

- **Scalability in terms of cluster size**
  In this test, we varied the cluster size ($K$ = {10, 20, 50, 100, 200, 500}), while keeping the number of the variables and the dataset size unchanged. The upper limit for the commercial de-identification tool was $K$ = 100, that is why the blue line ends before the others. We report the execution time as a measure in Fig. 5. Cluster size ($K$) should not affect the execution time of our model significantly as it was canceled out in the complexity calculation. However, as Fig. 5 shows, the execution time

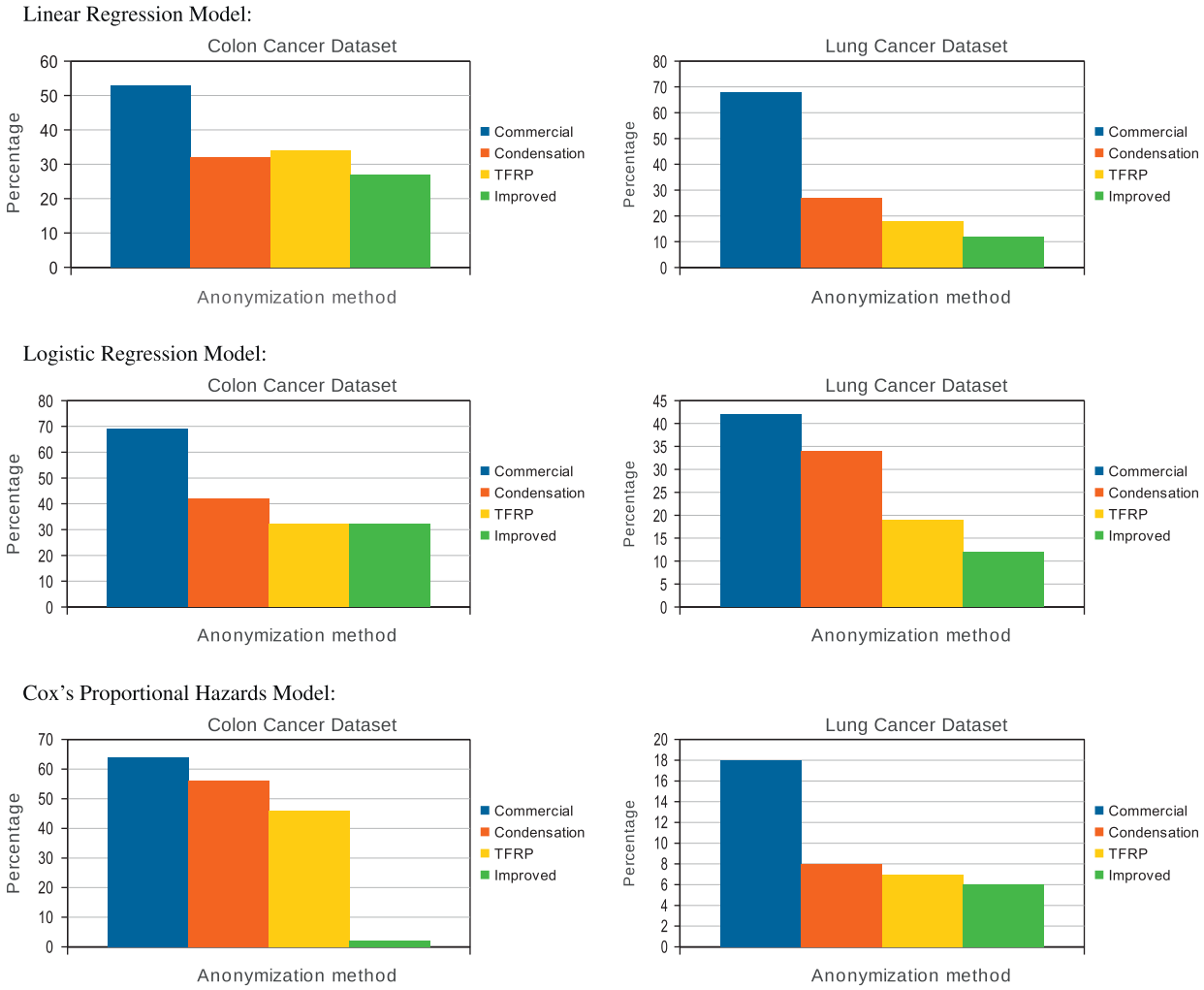


**Fig. 2.** Significant coefficients changed directions.

decreased at first as $K$ increased until about $K = 50$, then it increased again. The reason for this can probably be found in the implementation of the algorithm. Execution time consists of CPU (central processor unit) time and disk read/write time. In the complexity calculation we only examined CPU time. Our algorithm was implemented utilizing a database system. Database systems utilize many techniques that affect execution time, such as caching or indexing. Database systems usually use hard disk drives to store data, which makes reading and writing operations time consuming. To overcome this handicap, memory tables were used in the database. Writing to and reading from the memory is considerably faster than disk operations, yet it still requires time. This could be the reason behind the shape of the curve for the improved algorithm in Fig. 5. The majority of the read and write operations occur at two phases in the improved algorithm (Algorithm 2):

- Cluster creation (2a in Algorithm 2): Even though overall the same amount of data is read and written when we process the same dataset using different $K$ values, it is done in different packaging. When $K$ is small, the size of the data is smaller, but the number of clusters is larger; so smaller amount of data need to be read and written many more times. When $K$ is large on the other hand, there are fewer clusters, so larger amounts of data are read and written less times. This can conflict with the caching policy of the database system making it more time consuming to write small amounts of data repeated many times.
- Moving data points from larger clusters to smaller ones to ensure *K-indistinguishability*. In this case disk reading and writing time depends on the number of data points that need to be moved among clusters to make the clusters $n_c \geqslant qK$ (2b and 2c in Algorithm 2).

To asses how read/write times affected the execution time, we measured the time needed for cluster creation and moving records from larger clusters to smaller ones separately (Fig. 6). Fig. 6 shows the averages of ten measurements for various $K$ values ($K = \{10, 20, 50, 100, 200, 500\}$). The time used for read and write operations was almost constant as K varied, with almost zero standard deviation. On the other hand, the time needed for moving records from larger clusters to smaller ones closely imitated the shape of the overall chart in Fig. 5. It is not a requirement for K-means clustering to create clusters with equal number of records and it does not seem to do well when the cluster size is too small or too large. To compensate this weakness, the improved algorithm needs to move more records from larger clusters to smaller ones when $K$ is too small or too large. Another interesting fact to notice is the large standard deviation in this second line. This can be explained by the random nature of K-means clustering.

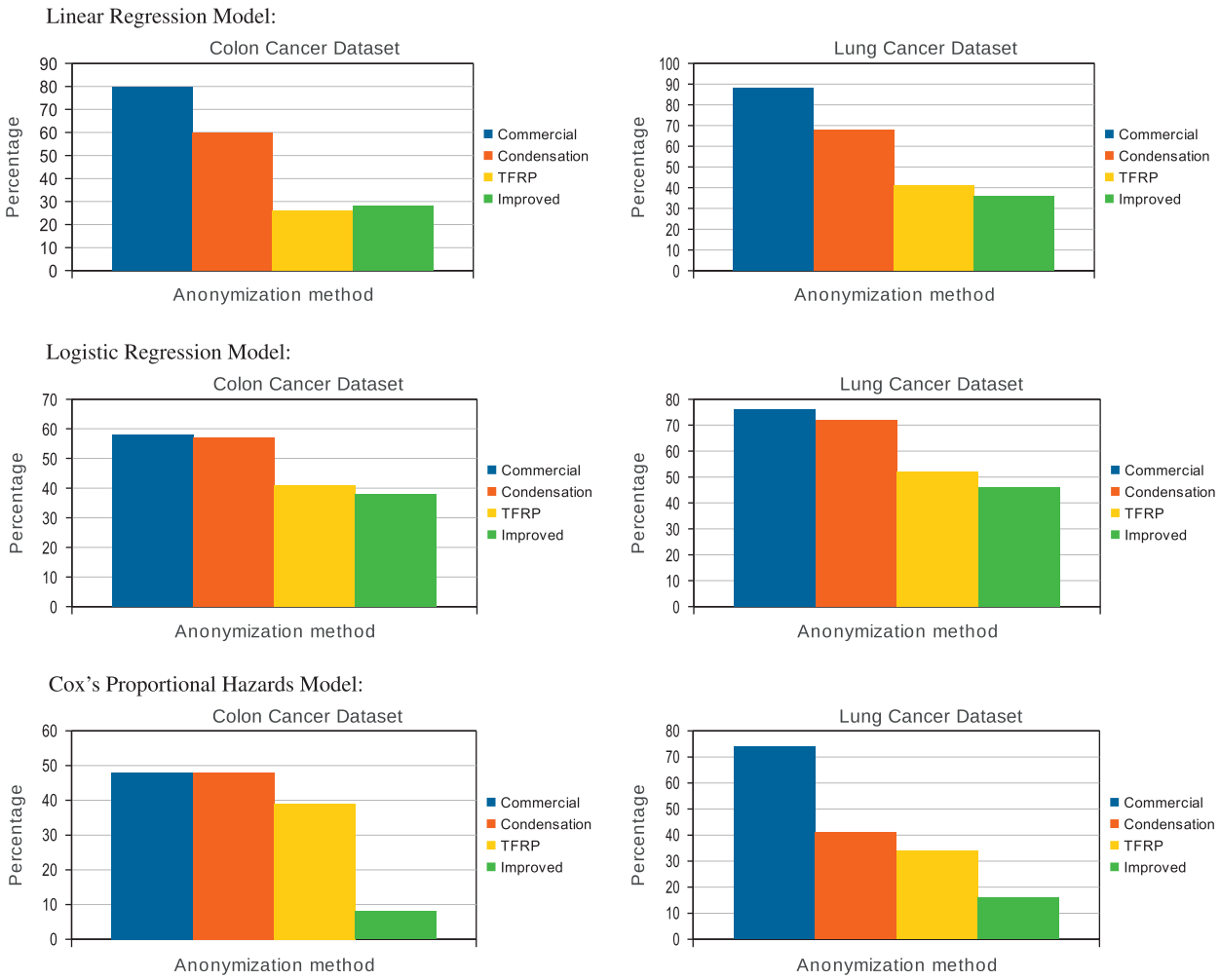


**Fig. 3.** Coefficients outside of the 95% confidence interval of the originals.

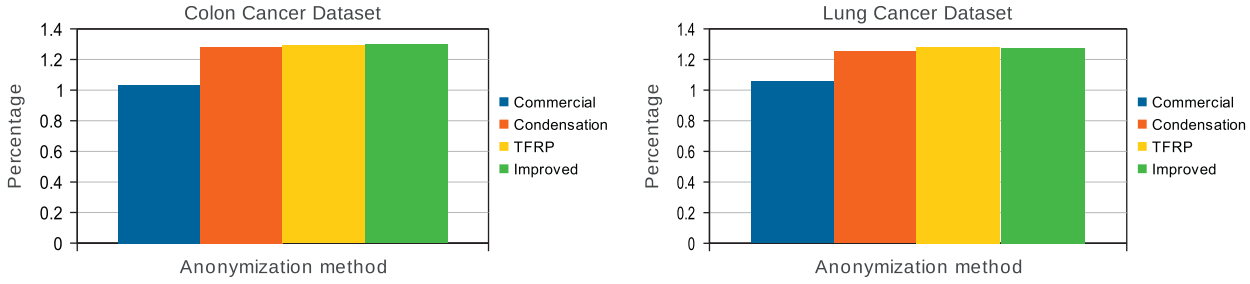


**Fig. 4.** Average conditional entropy.

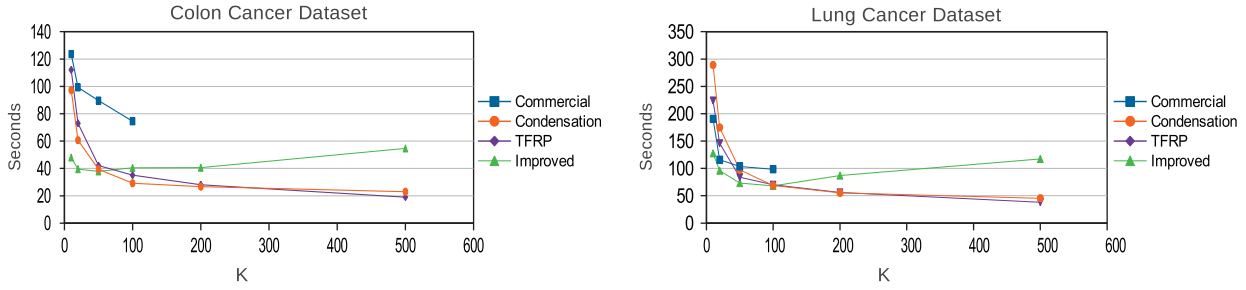


**Fig. 5.** Execution time when varying *K*.

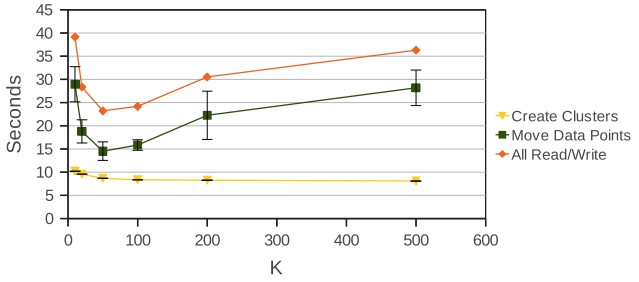


**Fig. 6.** The effect of read and write operations when varying *K*.

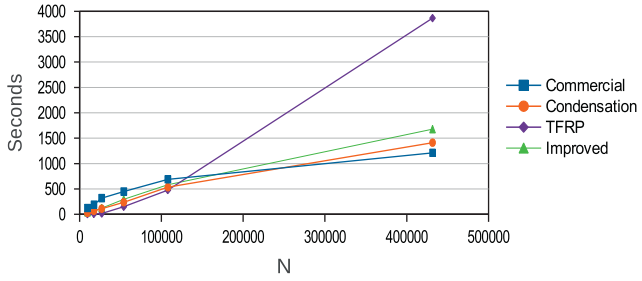


**Fig. 8.** Execution time when varying the size of the dataset.

In real life, *K* is usually set to a value between 20 and 100, where execution time does not change significantly for the improved method. It also performs well compared to the two other algorithms in this interval.

- **Scalability in terms of the number of variables**
  For this test we varied the number of variables between 2 and 13 ($m = \{2, 4, 8, 13\}$) while keeping $K = 100$ and the size of the dataset unchanged. As we showed earlier, our improved method has square relationship in time complexity in terms of the number of dimensions because of the PCA calculation. We also showed the PCA transpose trick, which can be used for datasets with high dimensionality. Our empirical results did not expose the square relationship. A possible reason for this is that the overhead coming from disk input–output operations probably dwarfs the effects of the PCA calculation. Our improved algorithm performed well, especially when compared to the commercial de-identification tool in higher dimensions. (Fig. 7). The commercial tool is a proprietary software. It implements *k*-anonymity through a heuristic lattice search which seems to become complex at higher dimensions. It also uses a database system to store the data during de-identification which means disk operations and other overhead. On the other hand, *k*-anonymity only masks quasi identifiers. The number of quasi identifiers is usually no more than five or six, in which territory the commercial tool performs well.
- **Scalability in terms of dataset size**
  For this test we varied the size of the dataset between about 10,000 and 430,000, while keeping *K* and the number of variables unchanged. We created larger datasets by combining the original datasets with synthetic data. According to our complexity calculation, dataset size is in linear relationship with time complexity, unless we use the PCA transpose trick for high dimensional data. Our empirical results show an almost linear relationship between the size of the dataset and execution time except for the TFRP algorithm, which has a square relationship between *N* and time complexity. The improved method showed considerable variation in terms of execution time when experiments were repeated with the same settings. Just as previously in the cluster size experiments, this variation can be explained by the random nature of K-means clustering.
  Another interesting observation is the slightly sub-linear nature of the curves in Fig. 8. Since all these models were implemented utilizing database systems, the explanation for this anomaly can probably be found in the decreasing overhead in database operations at larger datasets. In other words, database optimization seems to work better when larger chunks of data are moved.

## 5. Discussion

De-identification is a balancing act between privacy and utility preservation. The most common approaches to de-identification are generalization and suppression, which, by definition, operate on the basis of information loss. As we showed in this project, an alternative solution is synthetic data generation. The promise of this approach is to generate synthetic records that might have come from the same population as the original records. Yet, usage of synthetic data is not widespread, possibly because of the resistance of the biomedical research community. It needs to be studied whether researchers are willing to work with synthetic data and whether the biomedical research community is ready to accept the results produced using synthetic data as equivalent to results coming from original data. This is an area where research can be extended.

Although our anonymization approach was purposefully customized to the needs of the data recipient, we can still investigate whether it is generalizable or not. Generalization of our methods can be addressed on two levels:

- Our customization approach can be generalized and used for other projects, namely:
  - Ask for input from the data recipients about their data usage plans (mining methods, statistical analyses, etc.).
  - Analyze the proposed data mining and statistical models.
  - Design a de-identification method that will minimally obscure the data while ensuring privacy.
- Our actual anonymization method designed for this particular project can be generalized to the use of any covariance based statistical models as we showed that our method preserves covariance in the data.

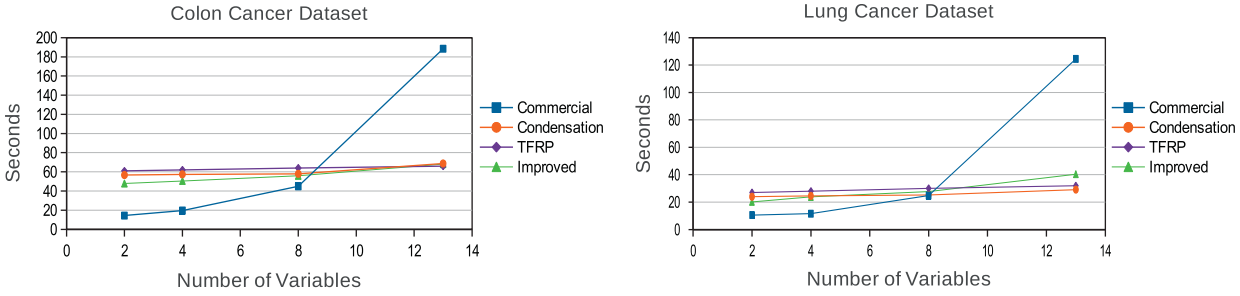


**Fig. 7.** Execution time when varying the number of variables.

Although our de-identification method was presented and examined in the biomedical domain it can be readily applied in other areas where privacy is a concern.

## 6. Conclusions

This paper proposed a data recipient centered utility based de-identification framework. In this framework we ask the data recipient about their plans regarding the data, carefully analyze the proposed data mining and statistical models and design a customized de-identification approach that is specific to the needs of the data recipient. In our test project the requirements were to preserve statistical attributes specific to three statistical models. After analyzing these models, we designed a microaggregation method where both the clustering and the algorithms were specific to the project requirements. We measured the performance of our method using utility metrics that were specific to the data recipient's requirements as well and showed that our customized method performed better than other general de-identification algorithms.

We will continue working with biomedical researchers to further explore the benefits of providing customized de-identification solutions. We are planning to extend the scope of this research to other statistical models and classification algorithms.

## References


[1] Centers for Medicare and Medicaid Services. Meaningful Use. Available from: http://www.cms.gov/Regulations-and-Guidance/Legislation/EHRIncentivePrograms/Meaningful_Use.html.
[2] American Medical Association. EHR survey 2011; 2011.
[3] Muralidhar Krishnamurty, Sarathy Rathindra. Security of random data perturbation methods. ACM Trans Database Syst 1999;24:487–93.
[4] Kargupta Hillol, Datta Souptik, Wang Qi, Sivakumar Krishnamoorthy. On the privacy preserving properties of random data perturbation techniques. In: ICDM; 2003. p. 99–106.
[5] Liu Kun, Kargupta H, Ryan J. Random projection-based multiplicative data perturbation for privacy preserving distributed data mining. IEEE Trans Knowl Data Eng 2006;18(1):92–106.
[6] Kargupta Hillol, Datta Souptik, Wang Qi, Sivakumar Krishnamoorthy. Random-data perturbation techniques and privacy-preserving data mining. Knowl Inf Syst 2005;7:387–414.
[7] Chen Kek, Liu Ling. A random rotation perturbation approach to privacy-preserving data classification. In: ICDM 2005, Houston, TX; November 2005.
[8] Li Xiao-Bai, Sarkar Sumit. A tree-based data perturbation approach for privacy-preserving data mining. IEEE Trans Knowl Data Eng 2006;18(9):1278–83.
[9] Dalenius T, Reiss SP. Data-swapping: a technique for disclosure control. J Stat Plan Inf 1982;6:73–85.
[10] Gomatam Shanti, Karr Alan F, Sanil Ashish P. Data swapping as a decision problem. J Official Statist 2003;21(4):635–55.
[11] Samarati Pierangela, Sweeney Latanya. Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression; 1998.
[12] Sweeney L. Achieving k-anonymity privacy protection using generalization and suppression. Int J Uncert Fuzziness Knowl-based Syst 2002;10(5):571–88.
[13] Wang Ke. Bottom-up generalization: a data mining solution to privacy protection. In: ICDM; 2004. p. 249–56.
[14] Defays D. Protecting micro-data by micro-aggregation: The experience in Eurostat. Questiio 1997;21:221–31.
[15] El Emam K, Dankar FK, Issa R, Jonker E, Amyot D, Cogo E, et al. A globally optimal k-anonymity method for the de-identification of health data. J Am Med Inform Assoc 2009;16:670–82.
[16] El Emam K, Brown A, AbdelMalik P, Neisa A, Walker M, Bottomley J, et al. A method for managing re-identification risk from small geographic areas in Canada. BMC Med Inform Decis Mak 2010;10:18.
[17] El Emam Khaled, Paton David, Dankar Fida, Koru Gunes. De-identifying a public use microdata file from the Canadian National Discharge Abstract Database. BMC Med Info Dec Making 2011;11(1):53.
[18] LeFevre Kristen, DeWitt David J., Ramakrishnan Raghu. Incognito: efficient full-domain k-anonymity. In: SIGMOD; 2005.
[19] Goldberger Jacob, Tassa Tamir. Efficient anonymizations with enhanced utility. Trans Data Privacy 2010;3(2):149–75.
[20] Li Xiao-Bai, Sarkar Sumit. Class-restricted clustering and microperturbation for data privacy. Manage Sci 2013;59(4):96–812.
[21] Aggarwal Charu C, Yu Philip S. Privacy-preserving data mining: models and algorithms. 1 edition. Springer Publishing Company, Incorporated; 2008.
[22] Xu Jian, Wang Wei, Pei Jian, Wang Xiaoyuan, Shi Baile, Fu Ada Waichee. Utility-based anonymization for privacy preservation with less information loss. ACM SIGKDD Explor 2006;8:2006.
[23] Jian Xu, Wei Wang, Jian Pei, Xiaoyuan Wang, Baile Shi, Ada Waichee Fu. Utility-based anonymization using local recoding. In: SIGKDD; 2006. p. 785–90.
[24] Fung Benjamin CM, Wang Ke, Yu Philip S. Top-down specialization for information and privacy preservation. In: Proc. of the 21st IEEE ICDE; 2005.p. 205–16.
[25] Fung Benjamin CM, Wang Ke, Yu Philip S. Anonymizing classification data for privacy preservation. IEEE Trans Knowl Data Eng 2007;19:711–25.
[26] Kifer Daniel, Gehrke Johannes. Injecting utility into anonymized datasets. In: Proceedings of the 2006 ACM SIGMOD international conference on management of data, SIGMOD 06. New York, NY, USA: ACM; 2006. p. 217–28.
[27] Bayardo Roberto J, Agrawal Rakesh. Data privacy through optimal k-anonymization. In: Proceedings of the 21st international conference on data engineering, ICDE 05. Washington, DC, USA: IEEE Computer Society; 2005. p. 217–28.
[28] Samarati Pierangela. Protecting respondents identities in microdata release. TKDE 2001;13(6):1010–27.
[29] Iyengar Vijay S. Transforming data to satisfy privacy constraints. In: Proceedings of the eighth ACM SIGKDD international conference on knowledge discovery and data mining, KDD 02. New York, NY, USA: ACM; 2002. p. 279–88.
[30] Ercan Nergiz M, Clifton Chris. Thoughts on k-anonymization. Data Knowl Eng 2007;63(3):622–45.
[31] Gionis A, Tassa T. k-anonymization with minimal loss of information. IEEE Trans Knowl Data Eng 2009;21(2):206–19.
[32] Office for Civil Rights. Guidance regarding methods for de-identification of protected health information in accordance with the health insurance portability and accountability act (HIPAA) privacy rule. Available from: http//www.hhs.gov/ocr/privacy/hipaa/understanding/coveredentities/De-identification/guidance.html.
[33] Sweeney L. K-anonymity: a model for protecting privacy. Int J Uncert Fuzziness Knowl-based Syst 2002;10(5):557–70.
[34] Meyerson Adam, Williams Ryan. On the complexity of optimal k-anonymity. In: Proceedings of the twenty-third ACM SIGMOD-SIGACT-SIGART symposium on principles of database systems, PODS 04. New York, NY, USA: ACM; 2004. p. 223–8.
[35] Aggarwal Charu C. On k-anonymity and the curse of dimensionality. In: Proceedings of the 31st international conference on very large data bases, VLDB 05, VLDB Endowment; 2005. p. 901–9.
[36] Machanavajjhala Ashwin, Gehrke Johannes, Kifer Daniel, Venkitasubramaniam Muthuramakrishnan. L-diversity: privacy beyond k-anonymity. In: 22nd IEEE international conference on data engineering (ICDE 2006), Atlanta, Georgia; April 2006.
[37] Li Ninghui, Li Tiancheng, Venkatasubramanian Suresh. t-Closeness: privacy beyond k-anonymity and l-diversity. In: Proceedings of the 23rd international conference on data engineering, ICDE 07. IEEE; 2007. p. 106–15.
[38] Domingo-Ferrer Josep, Torra Vicenc. Ordinal, continuous and heterogeneous k-anonymity through microaggregation. Data Min Knowl Discov 2005;11(2):195–212.
[39] LeFevre Kristen, DeWitt David J, Ramakrishnan Raghu. Mondrian multidimensional k-anonymity. In: ICDE; 2006.
[40] Brickell Justin, Shmatikov Vitaly. The cost of privacy: destruction of data-mining utility in anonymized data publishing. In: KDD 08: proceeding of the 14th ACM SIGKDD international conference on knowledge discovery and data mining. New York, NY, USA: ACM; 2008. p. 70–8.
[41] Gal Tamas, Chen Zhiyuan, Gangopadhyay Aryya. A privacy protection model for patient data with multiple sensitive attributes. Int J Info Secur Privacy 2008;2(3):28–44.
[42] Ye Yang, Liu Yu, Wang Chi, Lv Dapeng, Feng Jianhua. Decomposition: privacy preservation for multiple sensitive attributes. In: Proceedings of the 14th international conference on database systems for advanced applications, DASFAA 09. Berlin, Heidelberg: Springer-Verlag; 2009. p. 486–90.
[43] Li Zhen, Ye Xiaojun. Privacy protection on multiple sensitive attributes. In: ICICS; 2007. p. 141–52.
[44] Laszlo Michael, Mukherjee Sumitra. Minimum spanning tree partitioning algorithm for microaggregation. IEEE Trans Knowl Data Eng 2005;17:2005.
[45] Domingo-Ferrer J, Mateo-Sanz JM. Practical data-oriented microaggregation for statistical disclosure control. IEEE Trans Knowl Data Eng 2002;14(1):189–201.
[46] Aggarwal Charu C, Aggarwal Charu C, Yu Philip S, Yu Philip S. A condensation approach to privacy preserving data mining. In: EDBT; 2004. p. 183–99.
[47] Domingo-Ferrer Josep, Sebe Francesc, Solanas Agusti. A polynomial-time approximation to optimal multivariate microaggregation. Comput Math Appl 2008;55(4):714–32.
[48] Chang Chin-Chen, Li Yu-Chiang, Huang Wen-Hung. Tfrp: an efficient microaggregation algorithm for statistical disclosure control. J Syst Softw 2007;80(11):1866–78.
[49] Panagiotakis Costas, Tziritas Georgios. Successive group selection for microaggregation. IEEE Trans Knowl Data Eng 2013;25(5):1191–5.
[50] Domingo-Ferrer Josep, Gonzalez-Nicolas Ursula. Hybrid microdata using microaggregation. Inform Sci 2010;180(15):2834–44.
[51] El Emam K, Jabbouri S, Sams S, Drouet Y, Power M. Evaluating common de-identification heuristics for personal health information. J Med Internet Res 2006;8:e28.

[52] El Emam K, Dankar FK. Protecting privacy using k-anonymity. J Am Med Inform Assoc 2008;15:627–37.
[53] El Emam K, Brown A, AbdelMalik P. Evaluating predictors of geographic area population size cut-offs to manage re-identification risk. J Am Med Inform Assoc 2009;16:256–66.
[54] El Emam K, Neri E, Jonker E, Sokolova M, Peyton L, Neisa A, et al. The inadvertent disclosure of personal health information through peer-to-peer file sharing programs. J Am Med Inform Assoc 2010;17:148–58.
[55] El Emam K, Moreau K, Jonker E. How strong are passwords used to protect personal health information in clinical trials? J Med Internet Res 2011;13:e18.
[56] El Emam K, Hu J, Mercer J, Peyton L, Kantarcioglu M, Malin B, et al. A secure protocol for protecting the identity of providers when disclosing data for disease surveillance. J Am Med Inform Assoc 2011;18:212–7.
[57] El Emam K. Methods for the de-identification of electronic health records for genomic research. Genome Med 2011;3:25.
[58] El Emam K, Mercer J, Moreau K, Grava-Gubins I, Buckeridge D, Jonker E. Physician privacy concerns when disclosing patient data for public health purposes during a pandemic influenza outbreak. BMC Public Health 2011;11:454.
[59] El Emam K, Buckeridge D, Tamblyn R, Neisa A, Jonker E, Verma A. The re-identification risk of Canadians from longitudinal demographics. BMC Med Inform Decis Mak 2011;11:46.
[60] Benitez Kathleen, Loukides Grigorios, Malin Bradley. Beyond safe harbor: automatic discovery of health information de-identification policy alternatives. In: Proceedings of the 1st ACM international health informatics symposium, IHI 10. New York, NY, USA: ACM; 2010. p. 163–72.
[61] Chen T, Zhong S. An efficient privacy preserving method for matching patient data across different providers. In: Proceedings of the 34th annual symposium of American medical informatics association (AMIA); 2010. p. 1325.
[62] Durham E, Xue Y, Kantarcioglu M, Malin B. Private medical record linkage with approximate matching. In: 34th Annual symposium of American medical informatics association (AMIA); 2010. p. 182–6.
[63] Vidya Banu R, Nagaveni N. Preservation of data privacy using PCA based transformation. In: Proceedings of the 2009 international conference on advances in recent technologies in communication and computing, ARTCOM 09. Washington, DC, USA: IEEE Computer Society; 2009. p. 439–43.
[64] Vidyabanu R, Nagaveni N. A model based framework for privacy preserving clustering using SOM. Int J Comput Appl 2010;1(13):17–21. Published By Foundation of Computer Science.
[65] Aggarwal Charu C, Yu Philip S. On static and dynamic methods for condensation-based privacy-preserving data mining. ACM Trans Database Syst 2008;33(1):1–39.
[66] Brucker P. On the complexity of clustering problems. Optim Oper Res 1977:45–54.
[67] Pferschy Ulrich, Rudolf Rudiger, Woeginger Gerhard J. Some geometric clustering problems. Nordic J Comput 1994;1:246–63.
[68] MacQueen JB. Some methods for classification and analysis of multivariate observations. In: Le Cam LM, Neyman J, editors. Proc. of the fifth Berkeley symposium on mathematical statistics and probability, vol. 1. University of California Press; 1967. p. 281–97.
[69] Manning Christopher D, Raghavan Prabhakar, Schtze Hinrich. Introduction to information retrieval. New York, NY, USA: Cambridge University Press; 2008.
[70] Sirovich L, Kirby M. Low-dimensional procedure for the characterization of human faces. J Opt Soc Am A 1987;4(3):519–24.
[71] Turk Matthew, Pentland Alex. Eigenfaces for recognition. J Cogn Neurosci 1991;3(1):71–86.
[72] O'Toole Alice, Abdi Herve, Deffenbacher Kenneth A, Valentin Dominique. Low-dimensional representation of faces in higher dimensions of the face space; 1993.
[73] Agrawal D, Aggarwal CC. On the design and quantification of privacy preserving data mining algorithms. In: 20th ACM PODS, Santa Barbara, CA; 2001. p. 247–55.