UNIVERSITAT POLITÈCNICA DE VALÈNCIA | IN-RED 2023 IX Congreso de Innovación Educativa y Docencia en Red

Congreso In-Red 2023
UPV, 13 – 14 de julio de 2023
Doi:https://doi.org/10.4995/INRED2023.2023.16594

La k-anonimización para mantener la privacidad en el ámbito universitario

Alejandro Arbelaez[1] **y Laura Climent**[2]

[1] Departamento de Ingeniería Informática, Universidad Autónoma de Madrid
alejandro.arbelaez@uam.es
[2] Departamento de Ingeniería Informática, Universidad Autónoma de Madrid
laura.climent@uam.es

How to cite: Alejandro Arbelaez y Laura Climent. 2023. La k-anonimización para mantener la privacidad en el ámbito universitario. En libro de actas: *IX Congreso de Innovación Educativa y Docencia en Red*. Valencia, 13 – 14 de julio de 2023. https://doi.org/10.4995/INRED2023.2023.16594

Abstract

In this paper, we present a set of k-anonymization techniques for sharing the results of university examinations and provide an effective feedback in course evaluations. These techniques allow university lectures to share the examination results (with other universities or future students) in compliance with the General Data Protection Regulation (GDPR). Our empirical evaluation shows that k-anonymization allows a proper balance between the privacy of the informacion and the statistical properties of the original dataset. In particular, our simulations show that the k-anonymized dataset losses up to 5 % of the quality while significantly reducing the probability of re-identification. Therefore, our results suggest that k-anonimization is quite effective to ensure an adequate protection of the sensible data while preserving its utility.

Keywords*: k-anonimization, GDPR, privacy-preservation.*

Resumen

Este trabajo presenta un estudio del uso de técnicas de k-anonimización para preservar la privacidad al compartir los resultados de evaluaciones y proveer retro-alimentación en cursos universitarios. Estas técnicas de anonimización permiten a los profesores compartir los resultados con el curso (incluso con otras universidades o estudiantes de otros años) mientras se cumple con la regulación de protección de datos de la Unión Europea. Nuestros resultados muestran que la k-anonmización permite un balance entre la privacidad y la propiedades estadísticas de la información original. Especificamente, nuestras simulaciones indican que la k-anonimización pierde hasta un 5 % de información mientras reduce la considerablemente probabilidad de inferir información sensible de los estudiantes. De esta manera, nuestro resultados muestran que la k-anonimización es una técnica efectiva para proteger la privacidad de los datos sensibles mientras se preserva su utilidad.

Keywords*: k-anonimización, GDPR, preservación de privacidad.*

1 Introducción

La k-anonimidad es una técnica que busca proteger la privacidad de un conjunto de individuos en una base de datos. Para lograrlo, se elimina toda la información que pueda identificar a un conjunto de individuos en una base de datos, e.g., estudiantes, y se aplican métodos que impiden que de la información restante se pueda inferir información sensible o privada. Esta técnica se enfoca en asegurar que cada individuo en la base de datos sea indistinguible de al menos otros $k-1$ individuos, donde k determina el grado de anonimización en la base de datos anonimizada.

Para comprender la k-anonimización en detalle es necesario diferenciar entre tres tipos de datos: identificativos, cuasi-identificativos, y los sensibles. Como su nombre lo indica, los identificativos son aquellos que permiten identificar directamente a los individuos, por ejemplo, número de seguridad social, DNI, etc. Los datos cuasi-identificativos, por otro lado, no permiten identificar directamente a un individuo, e.g., la edad de una persona. Sin embargo, esta información se puede combinar con otras fuentes para identificar los individuos en la base de datos, como la fecha de nacimiento y el código postal. Finalmente, los datos sensibles son aquellos que se consideran privados o confidenciales, como información médica o financiera.

Es importante destacar que solo elimando los datos identificativos no es suficiente y puede vulnerar la privacidad de los individuos en la base de datos. (Kenthapadi, Mironov & Thakurta, 2019) describe varios casos documentados de violación de privacidad derivados de publicar datos no anonimizados o anonimizados incorrectamente. Cabe destacar, el caso Netflix, donde esta organización publicó un conjunto anonimizado de datos con clasificaciones de películas para más de 500,000 usuarios, en el contexto del *Netflix Challenge*, cuyo objetivo consistió en fomentar los algoritmos de recomendación de películas. Sin embargo, un grupo de investigación de la Universidad de Texas demostró que era posible identificar a ciertos usuarios combinando información de diferentes fuentes disponibles en la red. Los investigadores resaltan que algunos usuarios de Netflix pueden utilizar otras bases de datos, como IMDB, para clasificar y compartir información.

El ámbito universitario no esta excepto de ataques cibernéticos. En febrero de 2023, una prestigiosa universidad irlandesa fue víctima de un fuerte ataque cibernético, dicha universidad se vio en la necesidad de cerrar temporalmente uno de sus campus para proteger la integridad de cierta información de los miembros la comunidad universitaria. En marzo de 2023, otra prestigiosa universidad canadiense se vio forzada a suspender el acceso a su email por un ataque cibernéticos. De igual manera, muchas otras instituciones universitarias han sido víctimas de ataques cibernéticos que pueden vulnerar la privacidad tanto estudiantes como profesores. Todo esto es de gran importancia para recordar la necesidad de anonimizar adecuadamente los datos en diferentes ámbitos para proteger la privacidad de los usuarios.

2 Objetivos de la k-anonimización

La GDPR (por sus siglas en inglés – *General Data Protection Regulation*) surgió como una respuesta para proteger la privacidad e información personal (o sensible) de los ciudadanos de la Unión Europea. La GDPR considera la k-anonimización como una medida adecuada para garantizar la protección de datos personales. Fortaleciendo y mejorando la ley de protección de datos de 1995, la cual se considera obsoleta en el mundo digital en el que vivimos.

La tabla 1 muestra una base de datos, ejemplo, con los resultados de una evaluación de un curso universitario. En particular, tenemos un identificador (DNI) y cinco cuasi-identificadores (prácti-

DNI	Participación	Examen	Prácticas	Año	Profesor
****123	6	8	5	2022	Jana Doe
****321	10	3	5	2021	Jana Doe
****456	5	6	8	2022	Jana Doe
****654	4	4	10	2020	John Doe
****678	5	7	9	2018	John Doe
****456	5	6	8	2022	John Doe
****789	4	4	10	2020	Richard Roe
****987	5	7	9	2018	Richard Roe
****891	5	6	6	2022	Richard Roe

Tabla 1: Base de datos con resultados de una evaluación

cas, evaluación final, participación en clase, año del curso académico, y nombre del profesor). Es importante destacar que el objetivo de compartir esta información es retroalimentar a los estudiantes con el desempeño global, mientras se protege la privacidad de cada uno de los estudiantes. Sin embargo, en este caso en concreto, cualquier estudiante del curso puede inferir fácilmente quien es el estudiante con la mejor nota de participación (dado que puede ser el estudiante que más interactúa en clase), de esta manera, también se pueden inferir información confidencial de dicho estudiante, como la calificación final y sus tres últimos dígitos del DNI. Es necesario señalar que en este ejemplo incluimos parte del DNI dado que es una practica usual en ciertos ámbitos universitarios, sin embargo, la asociación española de protección de datos (Agencia Española de Protección de Datos, 2016) recomienda suprimir o seudonimizar dicha información, por lo tanto de aquí en adelante suprimiremos el atributo DNI.

La k-anonimización se puede conseguir de varias maneras: generalización, supresión, anatomización y perturbación. Sin embargo, la generalización y la supresión son los enfoques más populares hoy en día. La generalización apunta a reemplazar los valores con información más general, por ejemplo, incrementando el dominio de las variables. Por ejemplo, si un estudiante obtiene un 6 en un examen, este valor se puede generalizar con el dominio [6-10], lo que quiere decir, que la calificación de dicho estudiante fue entre 6 y 10. La función de generalización puede ser global para todo el conjunto de datos, por ejemplo, todos los estudiantes que obtienen un 6 en la evaluación se deben mapear de la misma manera (e.g., [6-10]). Otra posibilidad, consiste en permitir funciones flexibles y dinámicas, es decir, cada estudiante puede tener su propia función de generalización. De esta manera, dos o más estudiantes con la misma calificación pueden usar diferentes funciones de generalización, y terminar de esta manera con dominios diferentes, e.g., [4-6] y [6-10].

La supresión tiene como objetivo eliminar (total o parcialmente) la información de los cuasi-identificadores de un estudiante. Por lo general, en este método se reemplazan el cuasi-identificador con un valor neutral (e.g., '*'). La perturbación adiciona ruido controlado al conjunto de datos para evitar la reidentificación de los estudiantes. La cantidad de ruido se determina dependiendo de condiciones específicas de cada problema. La anatomización elimina la relación entre los cuasi-identificadores y la información sensible, por medio de la creación de nuevas tablas con información independiente.

Las tabla 2 y 3 muestran dos versiones k-anonimizadas de la tabla 1 con k=3. Como se puede ver estos ejemplos, cada estudiante es indistinguible de k-1(i.e., 2) estudiantes. De esta manera, la probabilidad de reidentificar el estudiante con mejor participación en clase es 1/3. Esta probabilidad se puede aumentar o reducir con el valor de k. En estos ejemplos, y de manera ilustrativa, la base

Participación	Examen	Prácticas	Año	Profesor
5-10	3-8	5-8	2021-2022	Jana Doe
5-10	3-8	5-8	2021-2022	Jana Doe
5-10	3-8	5-8	2021-2022	Jana Doe
4-5	4-7	8-10	2022-2020	John Doe
4-5	4-7	8-10	2022-2020	John Doe
4-5	4-7	8-10	2022-2020	John Doe
4-5	4-7	6-10	2022-2020	Richard Roe
4-5	4-7	6-10	2022-2020	Richard Roe
4-5	4-7	6-10	2022-2020	Richard Roe

Tabla 2: Base de datos anonimizada (k=3)

Participación	Examen	Prácticas	Año	Profesor
5-10	3-8	5-6	2021-2022	Jana Doe / Richard Roe
5-10	3-8	5-6	2021-2022	Jana Doe / Richard Roe
4-5	4-7	8-10	2018-2022	Jana Doe / Richard Roe
4-5	6-4	8-10	2018-2022	John Doe
4-5	6-4	8-10	2018-2022	John Doe
4-5	6-4	8-10	2018-2022	John Doe
4-5	4-7	8-10	2018-2022	Jana Doe / Richard Roe
4-5	4-7	8-10	2018-2022	Jana Doe / Richard Roe
5-10	3-8	5-6	2021-2022	Jana Doe / Richard Roe

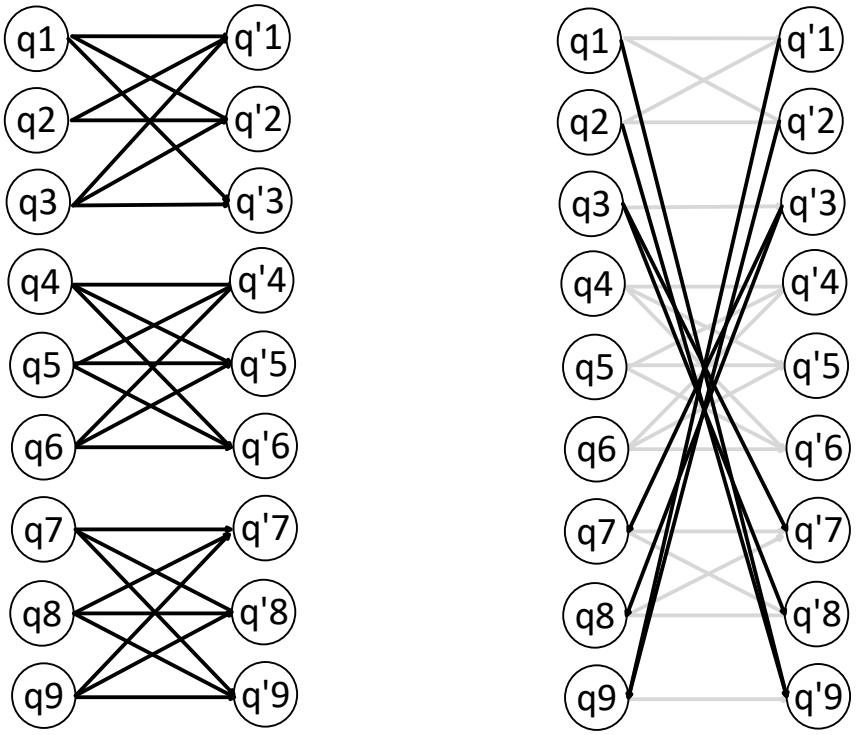
Tabla 3: Base de datos anonimizada (k=3)

de datos original y la anonimizada mantiene el mismo orden. Sin embargo, en situaciones reales, es necesario reordenar aleatoriamente la base de datos anonimizada.

3 Pérdida de información

Esta sección presenta una descripción formal de la k-anonimización como un grafo bipartito G=(Q, Q', E). Q representa la base de datos original, Q' representa la base de datos anonimizada, y $\langle q_i, q'_j \rangle \in$ E representa el conjunto de aristas de Q a Q' con generalizaciones entre la información original y la anonimizada. En la figura 1 se muestran los grafos de las tablas 2 y 3. En estas figuras cada grupo de k individuos corresponde a un clique, es decir, todos los vértices de cada subconjunto están completamente conectado entre sí. También es importante señalar que las aristas en gris (figura 1(b)) corresponden a las aristas comunes en los dos grafos.

Asegurar la privacidad en una base de datos, a través de la k-anonimización, conduce a un cierto grado de pérdida de información. Por lo tanto, el objetivo es encontrar un equilibrio entre la cantidad de información que perdemos y el nivel de privacidad (i.e., el valor de k).



(a) Grafo generado utilizando la tabla

(b) Grafo generado utilizando la tabla

Fig. 1: Grafos de anonimización con k=3

Para calcular la perdida de información es necesario diferenciar entre datos numéricos y categóricos. Para los datos numéricos es necesario calcular el valor máximo (max.) y mínimo (min.) de todas las tuplas del cuasi-identificador q_i y el valor máximo y mínimo del grupo o clique de q_i. Luego

utilizamos la ecuación 1 para cuantificar la información perdida después de anonimizar la tupla de la base de datos. En esta ecuación $\rho'_{i,j}$ representa el cuasi-identificador j de la fila i en la base de datos anonimizada. Por ejemplo, para la segunda tupla y el cuasi-identificador prácticas ($\rho'_{2,prácticas}$) de la tabla 2 perdemos el 60 % de la información, es decir NPC = (8-5)/(10-5) = 0.6. Sin embargo, en la tabla 3 para la misma tupla/cuasi-identificador tenemos NCP=(6-5)/(10-5) = 0.2. De esta manera, podemos concluir que para esta variable en concreto hemos perdido menos información con la generalización de la tabla 3.

$$NCP_{num}(\rho'_{i,j}) = \frac{max_{i,j} - min_{i,j}}{max^j - min^j} \tag{1}$$

Para las variables categóricas es necesario codificar la variable como un vector binario donde todas las posiciones son cero, excepto la que corresponde a la categoría en cuestión, que se establece en uno. Por ejemplo, supongamos que tenemos el cuasi-identificar profesor de la tabla 1, esta variable tiene 3 posibles valores, Jane Doe, John Doe y Richard Roe. La codificación de esta variable se representa de la siguiente manera:

- Jana Doe: [1, 0, 0]
- John Doe: [0, 1, 0]
- Richard Roe: [0, 0, 1]

Como se puede observar, cada posición del vector representa un posible valor de la categoría, y la posición correspondiente a la categoría en cuestión se establece en uno, mientras que todas las demás se establecen en cero. De esta manera, utilizamos la ecuación 2 para obtener la cantidad de información que se pierde al anonimizar una variable de tipo categoría. En esta ecuación $\tau'_{i,j}$ representa cuasi-identificador j de la fila i en la base de datos anonimizada y recordemos que esta variable es un vector de valores binarias. En nuestro ejemplo, tenemos que para $\tau'_{2,profesor}$ en la tabla 3 perdemos la mitad de la información, es decir NCP=(2-1)/(3-1) = 0.5.

$$NCP_{cat}(\tau'_{i,j}) = \frac{\sum_{v \in \tau_{i,j}} \tau^v_{i,j} - 1}{|\tau_{i,j}| - 1} \tag{2}$$

La ecuación 3 muestra como combinar todos cuasi-identificadores de una tupla.

$$NCP(q'_i) = \sum_{j \in qn_i} NCP_{num}(\rho'_{i,j}) + \sum_{j \in qc_i} NCP_{cat}(\tau'_{i,j}) \tag{3}$$

Ahora es necesario calcular la cantidad de información perdida en toda la versión anonimizada de la base de datos. Para ello, utilizamos la ecuación 4, donde d representa el número de cuasi-identificadores y |Q| representa el tamaño de la base de datos. Esta ecuación normaliza el valor NCP de todas las tuplas y siempre estará entre [0, 1]. Donde 0 representa que la base de datos no se ha visto afectada y 1 representa una pérdida total de la información.

$$GCP(Q') = \frac{\sum_{i \in Q'} NCP(q'_i)}{d \cdot |Q'|} \tag{4}$$

4 Desarrollo de la innovación mediante algoritmos voraces

Los algoritmos voraces son heurísticas para resolver problemas computacionalmente difíciles. Estos algoritmos toman una serie de decisiones en cada iteración basándose en la información que está disponible hasta el momento, sin tener en cuenta las consecuencias a largo plazo. Este enfoque suele ser muy eficiente, sin embargo es importante destacar que los algoritmos voraces no siempre garantizan que la solución encontrada es la óptima. En este artículo, nos enfocaremos en los algoritmos voraces, considerados hoy en día como el estado del arte para k-anonimizar bases de datos.

4.1 k-members

El algoritmo de k-members (Byun, Kamra, Bertino & Li, 2007) es un referente en k-anonimización. El algoritmo 1 ilustra, con un pseudocódigo, la heurística del algoritmo voraz para resolver incrementalmente el problema. El primer paso consiste en identificar el primer individuo de cada grupo (linea 7), para ello se utiliza el individuo que se encuentra más alejado del ultimo individuo (r) explorado por el algoritmo. En la primera iteración se utiliza como referencia un individuo seleccionado al azar (linea 5). Al igual que (Byun y col., 2007), en este artículos se utiliza la distancia Euclideana normalizada para calcular la distancia entre dos individuos.

El siguiente bucle (lineas 10-14) selecciona los mejores k-1 individuos que formaran parte del grupo, para ello se calcula la perdida de información que conlleva formar parte del grupo, posteriormente se utiliza la función MejorTupla para escoger el mejor individuo. Los individuos que no fueron

asignados a ningún grupo se distribuyen en el último bucle (lineas 17-21), aquí se insertan los elementos en el grupo que conllevan a una menor perdida de información (MejorGrupo – linea 20).

Algoritmo 1: k-members

```
1  if |S| ≤ k then
2  |  return S;
3  end
4  grupos := ∅;
5  r := TuplaAleatoria(S);
6  while |S| ≥ k do
7  |  r := TuplaMasLejana(S, r);
8  |  S := S - r;
9  |  c := { };
10 |  while |c| < k do
11 |  |  r := MejorTupla(S, c);
12 |  |  S := S - {r};
13 |  |  c := c ∪ {r};
14 |  end
15 |  grupos := grupos ∪ {c}
16 end
17 while S ≠ ∅ do
18 |  r := TuplaAleatoria(S);
19 |  S := S - {r};
20 |  c := MejorGrupo(result, r);
21 |  c := c ∪ r;
22 end
23 return grupos
```

4.2 l-greedy

El algoritmo l-greedy fue propuesto por (Liang & Samavi, 2020) y es una poderosa alternativa al k-members. El algoritmo 2 ilustra el pseudocódigo de esta heurística. Este algoritmo primero ordena la base de datos, en orden ascendente, dando prioridad a los cuasi-identificadores con menor varianza (lineas 1-2). (Liang & Samavi, 2020) sugiere que los individuos con menor varianza tienden a perder menos información durante el proceso de anonimización que los que tienen una mayor varianza.

Posteriormente, el primer bucle (lineas 5-19), inicializa cada grupo con el primer individuo disponible en la base de datos. El segundo bucle (lineas 9-17) iterativamente inserta k-1 individuos en el grupo, para ello el tercer bucle evalúa y almacena la cantidad de información derivada de insertar cada uno de los individuos disponibles en la base de datos. Las lineas 13-16 identifican e insertan el mejor individuo (s_j) en el grupo, finalmente el algoritmo marca s_j como anonimizado y lo elimina como candidato para futuras iteraciones del algoritmo.

Algoritmo 2: l-greedy

```
1  VAR := [VAR]_j ∈ J // Varianza de los cuasi-identificadores
2  S̃ := Ordenar(S, VAR);
3  grupos := ∅;
4  f := 0;
5  for x_i ∈ S̃ do
6  |  g_i := {s_i};
7  |  f_i := 0;
8  |  s_i := s_i - S̃;
9  |  for l = 1 to k -1 do
10 |  |  for s_j ∈ S̃ do
11 |  |  |  f_i^j := f(s_j ∪ g_i)
12 |  |  end
13 |  |  s_j' := arg mín_j(f_i^j);
14 |  |  g_i := g_i ∪ s'_j;
15 |  |  f_i := actualizar(f_i, f_i^j');
16 |  |  S̃ := S̃ - s_j';
17 |  end
18 |  f := actualizar(f, f_i);
19 |  grupos := grupos ∪ g_i;
20 end
21 return grupos;
```

4.3 Trabajo Relacionado

Es importante destacar que otros autores han utilizando una función de ordenamiento para anonimizar una base de datos. (Ghinita, Karras, Kalnis & Mamoulis, 2009), propone utilizar la curva de Hilbert para ordenar la base de datos. Este trabajo asume que individuos que están ubicados cercanos en la curvan deben estar en el mismo grupo. (Sánchez, Martínez & Domingo-Ferrer, 2016) propone ordenar la base de datos, utilizando la distancia de cada punto a un punto de referencia (típicamente utilizando el valor cero para todos los cuasi-identificadores). Este tipo de algoritmos, basados únicamente en funciones de ordenamiento, tienden a ser muy rápidos para anonimizar base de datos, pero a su vez, pierden mucha más información que los algoritmos de k-members y l-greedy, que utilizan una búsqueda más guiada.

Otras meta-heurísticas se han propuesto para la k-anonimización de base de datos. (Iyengar, 2002) utiliza algoritmos genéticos y la codificación de Michigan resolver el problema. Este trabajo codifica la solución con una sola cadena de números binarios y propone un conjunto de operaciones para aplicar los métodos de mutación y recombinación de genes. La principal desventaja de este método radica en la necesidad de predefinir una jerarquías de generalización para cada cuasi-identificador. Por ejemplo, para el cuasi-identificador profesor, de nuestro ejemplo anterior, se pueden incluir tres niveles: informática, estadística, y matemáticas; de igual manera para el cuasi-identificador examen se pueden incluir cuatro categorías: suspenso, aprobado, notable, sobresaliente.

ARX (Prasser, Eicher, Spengler, Bild & Kuhn, 2020) es una biblioteca abierta, eficiente, y compatible con una variedad de algoritmos para anonimizar bases de datos. Además de la k-anonoimización, soporta l-diversidad, y (ϵ, γ) privacidad diferencial. Sin embargo, al igual que los algoritmos genéticos es necesario definir jerarquías de generalización para poder anonimizar la base de datos. También, cabe destacar que (Prasser y col., 2020) demostró empíricamente que métodos como (Sánchez y col., 2016) son más eficientes que ARX. Además k-members y l-greedy son, en general, mejores que ARX.

El clustering jerárquico divisivo (o de arriba hacia abajo) también se ha utilizado para la k-anonimización de bases de datos. Este tipo de métodos, busca dividir la base de datos en grupos pequeños y homogéneos. La idea es construir un grupo inicial que sea demasiado grande y poco específico (con mucha perdida de información), posteriormente este grupo se divide en grupos pequeños mejorando de esta manera la información que se perdía con los grupos grandes. Este proceso se repite hasta alcanzar el nivel de deseado de privacidad (valor de k). La principal desventaja de este método es la necesidad de las jerarquías de generalización.

Por lo que sabemos hasta el momento, (Doka, Xue, Tsoumakos & Karras, 2015) es el único método que k-anonimiza una base de datos con menos perdida de información que k-members y l-greedy. Sin embargo, este método no utiliza la visión clásica donde todos los individuos de un grupo se generalización con los mismos valores. En este caso, la anonimización es no homogénea, es decir, el hecho de que la tupla q_i anonimize la tupla q_j no implica que la tupla q_j tenga que anonimizar la tupla q_i. Esta pequeña variación tiene implicaciones en el nivel de privacidad del algoritmo, como resultado, la base de datos anonimizada es más vulnerable que con el método tradicional. Para más detalles, remitimos al lector a (Choromanski, Jebara & Tang, 2013) con un análisis teórico y detallado de las vulnerabilidades de la k-anonimización heterogénea. También es importante destacar que la solución propuesta en (Doka y col., 2015) es computacionalmente muy costosa y el tiempo de computación es considerablemente más alto que en el caso de l-greedy y k-member.

Muchas otras soluciones se han propuesto para la k-anonimización, e.g., (Abbasi & Mohammadi, 2022; Li, Yuan, Yuan, Chen & Yu, 2022; Onesimu, Karthikeyan, Eunice, Pomplun & Dang, 2022;

Yan, Herman, Mahmood, Feng & Xie, 2021). Sin embargo, estos trabajos son pequeñas variaciones del algoritmo de k-members o no existe evidencia estadística que sugiera que estos algoritmos son mejores que k-members o l-greedy.

5 Resultados

En este trabajo evaluamos los algoritmos de k-members y l-greedy con una base de datos con información pseudoaleatoría de 500 estudiantes utilizando los cinco cuasi-identificadores de nuestro ejemplo inicial, i.e., participación, examen, prácticas, año, y profesor. Para los cuasi-identificadores relacionados con calificaciones (i.e., prácticas, examen y participación) usamos una distribución normal, con media = 6 (aprobado) y una desviación típica de 2, es decir, alrededor del 68 % de los estudiantes tiene calificación entre [5, 7]; para el cuasi-identificador profesor, usamos una distribución uniforme con 10 profesores; y para cuasi-identificador año también usamos una distribución uniforme con el dominio [2018, 2022].

Toda nuestra evaluación la realizamos en un ordenador MacBook Pro (M1 2020) con el sistema operativo macOS Big Sur y 16GB de memoria. Los dos algoritmos los hemos implementado en C++. Cabe destacar que nuestra implementación es, por lo menos, 30 veces más rápida que la implementación original en Python del algoritmo l-greedy, disponible en el GitHub de los autores del mismo.

La figura 2 ilustra visualmente la cantidad de información que perdemos al incremental la privacidad (o el valor de k) al anonimizar la información. Cabe destacar que los dos algoritmos anonimizan la base de datos en menos de 1 segundo y la perdida de información va desde 0.52 % (k=2) hasta 5.09 % (k=40) en ambos casos con utilizando el algoritmo l-greedy. Como es de esperar, la cantidad de información que perdemos se incrementa al incremental el valor de k. Es importante recordar que con k=2 la probabilidad de reidentificar a un estudiante es 50 %, mientras que con k=40 esta probabilidad se reduce a un 2.5 %.

De esta evaluación se puede concluir que ambos algoritmos se pueden utilizar de una manera muy efectiva. Sin embargo, el algoritmo l-greedy es un poco mejor con una perdida promedio (de todos los experimentos de este trabajo) de información de 2.70 % comparado con k-members que pierde en promedio 2.71 %.

6 Conclusiones y trabajo futuro

En conclusión, este trabajo muestra la efectividad de dos técnicas de k-anonimización (k-members y l-greedy) para anonimizar los resultados de los estudiantes a nivel universitario mientras se cumple con la regulación europea de protección de datos (GDPR). Al aplicar estas técnicas podemos proteger la privacidad de los estudiantes y proveer una retroalimentación general sin violar la privacidad de ningún otro estudiante. De esta manera, cada estudiante se puede auto-evaluar al revisar su propio desempeño contra el resto del curso (o incluso con respecto a cursos anteriores).

También dado que nuestra implementación es, por lo menos, 30 veces más rápida que el estado del arte, es posible anonimizar las calificaciones a nivel global de toda la universidad. Cabe destacar que el comité europeo de protección de datos no considera la información anonimizada como información personal. Sin embargo, siempre es necesario mantener un balance entre el nivel de seguridad (por medio del valor de k) y la cantidad de información que se pierde al anonimizar

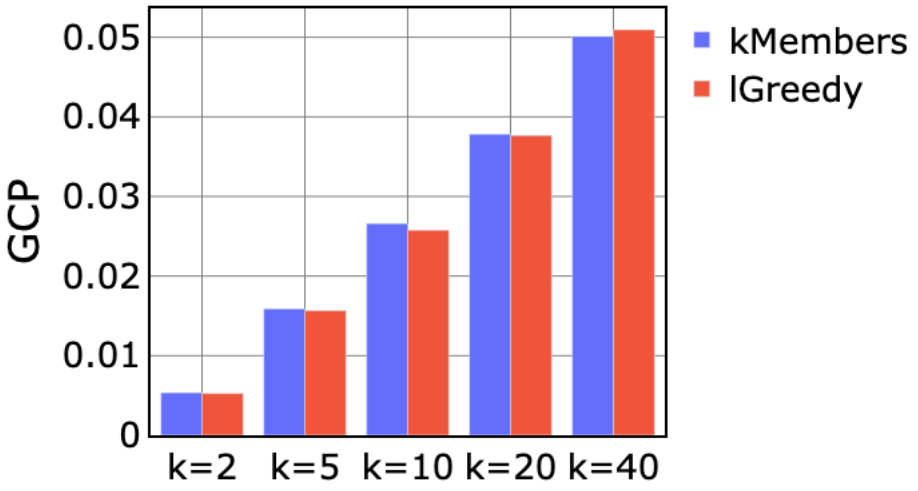


Fig. 2: GCP incrementado el valor de k y 500 estudiantes

los datos. En este trabajo, con datos pseudoaleatorios, se puede concluir que los algoritmos l-members y l-greedy pierden hasta un 5 % de información mientras se asegura que la probabilidad de reidentificar a un estudiante es tan solo un 2.5 %.

En trabajos futuros, planeamos explorar la efectividad de modelos de anonimización más avanzados, como l-diversidad (Yao, Chen, Hu, Wu & Wu, 2021) y t-cercanía (Soria-Comas, Domingo-Ferrer, Sánchez & Martínez, 2015), que abordan algunas de las limitaciones de la k-anonimización. Por ejemplo, la k-anonimización no garantiza una diversidad de la información sensible y esta técnica también asume que todos los cuasi-identificadores sensibles son igualmente importantes.

Referencias bibliográficas

Abbasi, A. & Mohammadi, B. (2022). A clustering-based anonymization approach for privacy-preserving in the healthcare cloud. *Concurrency and Computation: Practice and Experience*, *34* (1), e6487.

Agencia Española de Protección de Datos. (2016). Orientaciones y Garantías en los Procedimientos de Anonimización de Datos Personales.

Byun, J., Kamra, A., Bertino, E. & Li, N. (2007). Efficient k-Anonymization Using Clustering Techniques. En *Advances in Databases: Concepts, Systems and Applications, 12th International Conference on Database Systems for Advanced Applications, DASFAA 2007, Bangkok, Thailand, April 9-12, 2007, Proceedings* (Vol. 4443, pp. 188-200). Lecture Notes in Computer Science. Springer.

Choromanski, K., Jebara, T. & Tang, K. (2013). Adaptive Anonymity via b-Matching. En *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States* (pp. 3192-3200).

Doka, K., Xue, M., Tsoumakos, D. & Karras, P. (2015). k-Anonymization by Freeform Generalization. En *Proceedings of the 10th ACM Symposium on Information, Computer and Communications Security, ASIA CCS '15, Singapore, April 14-17, 2015* (pp. 519-530). ACM.

Ghinita, G., Karras, P., Kalnis, P. & Mamoulis, N. (2009). A framework for efficient data anonymization under privacy and accuracy constraints. *ACM Trans. Database Syst. 34*(2), 9:1-9:47.

Iyengar, V. S. (2002). Transforming data to satisfy privacy constraints. En *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, July 23-26, 2002, Edmonton, Alberta, Canada* (pp. 279-288). ACM.

Kenthapadi, K., Mironov, I. & Thakurta, A. G. (2019). Privacy-Preserving Data Mining in Industry. En *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining* (pp. 840-841). WSDM '19. Melbourne VIC, Australia: Association for Computing Machinery.

Li, Y., Yuan, S., Yuan, Y., Chen, C. & Yu, J. (2022). Anonymization of Quasi-Sensitive Attribute Sets in Aggregated Dataset. *Security and Communication Networks*, *2022*.

Liang, Y. & Samavi, R. (2020). Optimization-based k-anonymity algorithms. *Comput. Secur. 93*, 101753.

Onesimu, J. A., Karthikeyan, J., Eunice, J., Pomplun, M. & Dang, H. (2022). Privacy Preserving Attribute-Focused Anonymization Scheme for Healthcare Data Publishing. *IEEE Access*, *10*, 86979-86997.

Prasser, F., Eicher, J., Spengler, H., Bild, R. & Kuhn, K. A. (2020). Flexible data anonymization using ARX - Current status and challenges ahead. *Softw. Pract. Exp. 50*(7), 1277-1304.

Sánchez, D., Martínez, S. & Domingo-Ferrer, J. (2016). Comment on "Unique in the shopping mall: On the reidentifiability of credit card metadata". *Science*, *351*(6279), 1274-1274.

Soria-Comas, J., Domingo-Ferrer, J., Sánchez, D. & Martínez, S. (2015). t-Closeness through Microaggregation: Strict Privacy with Enhanced Utility Preservation. *IEEE Trans. Knowl. Data Eng. 27*(11), 3098-3110.

Yan, Y., Herman, E. A., Mahmood, A., Feng, T. & Xie, P. (2021). A weighted K-member clustering algorithm for K-anonymization. *Computing*, *103*(10), 2251-2273.

Yao, L., Chen, Z., Hu, H., Wu, G. & Wu, B. (2021). Sensitive attribute privacy preservation of trajectory data publishing based on l-diversity. *Distributed Parallel Databases*, *39*(3), 785-811.