Dime qué dicen los datos y no podré decirte quién eres

Cómo publicar información sensible sin comprometer la privacidad de las personas

Matías Toro

Doctor en Ciencias Mención Computación por la Universidad de Chile. Profesor Asistente del Departamento de Ciencias de la Computación de la misma Universidad e Investigador Joven del Instituto Milenio Fundamento de los Datos. Líneas de investigación: lenguajes de programación, sistemas de tipos y privacidad diferencial.

mtoro@dcc.uchile.cl

Resumen / Vivimos rodeados de datos sensibles, y publicarlos sin cuidado puede exponer información sensible, incluso después de procesos de "anonimización". Técnicas clásicas como el *k*-anonimato y la ℓ-diversidad ayudan a ocultar identidades agrupando registros, pero son vulnerables cuando existen datos externos capaces de recombinar o inferir información sensible.

La privacidad diferencial ofrece un enfoque más robusto: en lugar de proteger la tabla publicada, protege el mecanismo que genera los datos, garantizando que el resultado cambie muy poco si una persona participa o no. Esto se logra añadiendo ruido cuidadosamente calibrado, lo que permite mantener la utilidad estadística sin revelar información individual. Hoy es el estándar usado por Google, Apple, Meta y el Censo de Estados Unidos, y será clave para cumplir la nueva Ley de Protección de Datos en Chile.

Introducción

Vivimos rodeados de datos. Cada vez que navegamos por Internet, usamos una tarjeta de transporte, pagamos con el teléfono, vamos al médico o simplemente escuchamos música, generamos información. Estos datos pueden incluir aspectos profundamente personales: nuestra salud, patrones de movilidad, historial financiero, hábitos de citas, compras, gustos artísticos, registros académicos y mucho más. Y, como si fuera poco, los generamos en volúmenes crecientes día a día.

Toda esta información abre un mundo de oportunidades. Permite personalizar servicios, automatizar tareas mediante modelos computacionales, orientar políticas públicas con evidencia, mejorar la transparencia del Estado, e incluso entrenar modelos avanzados de inteligencia artificial. Sin embargo, junto con estas posibilidades aparece un riesgo ineludible: muchos de esos datos contienen información extremadamente sensible. Su uso inadecuado —o su publicación sin suficiente protección— puede exponer facetas íntimas de la vida de una persona.

Este riesgo no es teórico. Hay casos emblemáticos que lo evidencian con crudeza. En 2007, Netflix lanzó una competencia abierta para mejorar su sistema de recomendación y publicó un conjunto de datos "anonimizado". Bastó cruzarlo con información pública de IMDb para que participantes de la competencia lograran reidentificar a numerosas personas, reconstruyendo parte de su historial de visualización.

Un caso aún más icónico ocurrió en 1997, cuando el gobernador de Massachusetts publicó registros médicos de funcionarios públicos tras un proceso de anonimización. Dos días después, Latanya Sweeney —entonces estudiante de doctorado en el MIT— le envió al gobernador una carta con *sus propios* registros médicos. ¿Cómo lo hizo? Cruzó los datos anonimizados con el padrón electoral usando solo tres atributos: código postal, fecha de nacimiento y sexo. La anonimización falló al no considerar información auxiliar disponible para un atacante.

Estos ejemplos ilustran un punto crucial: *el problema no está solo en los datos publicados, sino en los datos externos que podrían combinarse con ellos.*

Estos desafíos adquieren hoy una urgencia especial. La nueva Ley de Protección de Datos Personales, que entrará en vigencia en diciembre de 2026, establece un marco mucho más estricto para el manejo y publicación de información, obligando a organismos públicos y privados a asegurar que cualquier dato liberado —incluso después de procesos de anonimización— no permita identificar a una persona. La ley introduce sanciones importantes y exige adoptar estándares modernos de privacidad. En este escenario, comprender las técnicas que podemos usar para alcanzar ese objetivo se vuelve clave no sólo para investigadores y desarrolladores, sino para cualquier institución que aspire a liberar datos de forma responsable y legal.

Entonces, ¿cómo podemos publicar —o incluso resumir— datos sin revelar información sensible sobre las personas? En la práctica, existen dos grandes enfoques de privacidad:

- **El modelo basado en ataques específicos.** Supone un conjunto definido de amenazas —como los cruces de bases de datos mencionados— y busca defenderse de ellos. Aquí surgen técnicas clásicas de anonimización, como el *k*-anonimato y la *l*-diversidad.

- **El modelo basado en la desinformación controlada.** Su principio es distinto: una publicación es privada si, al observarla, un atacante obtiene *muy poca* información adicional respecto de lo que ya sabía. Este es el fundamento de la *privacidad diferencial*, hoy estándar en proyectos de Google, Apple, Meta y el Censo de Estados Unidos.

Es importante notar que, en marcos legales como la ley de protección de datos, ambos enfoques caen bajo el paraguas de "anonimización", pues buscan impedir que alguien pueda vincular datos sensibles con una persona específica. Pero conceptualmente funcionan de formas muy distintas.

En las siguientes secciones exploraremos estas técnicas, cómo protegen (o no) frente a distintos tipos de ataques, y porqué la privacidad diferencial ha emergido como un cambio de paradigma en la publicación segura de datos.

Anonimización clásica

La anonimización tradicional clasifica los atributos de un conjunto de datos en cuatro categorías:

1. **Identificadores explícitos:** permiten identificar directamente a una persona (por ejemplo, el RUT).

2. **Cuasi-identificadores:** no identifican por sí solos, pero sí cuando se combinan con otros datos. Este fue justamente el caso del ZIP, fecha de nacimiento y género utilizado por Latanya Sweeney para reidentificar al gobernador de Massachusetts.

3. **Atributos sensibles:** contienen información particularmente delicada, como condiciones médicas, historial financiero, antecedentes criminales o nivel de ingresos.

4. **Atributos no sensibles:** todo atributo que no calza en las categorías anteriores.

Podemos analizar los distintos tipos de atributos considerando la Tabla 1, la cual presenta un conjunto de datos ilustrativos sobre personas y su estado COVID-19.

En este ejemplo, *Nombre* es un identificador explícito; *Sexo* y *Dirección* son cuasi-identificadores; y *COVID* es un atributo sensible.

Eliminar nombres no basta

Cuando se publican datos, la técnica más intuitiva consiste en eliminar los identificadores explícitos. Por ejemplo, el conjunto de datos mostrado en la Tabla 2 presenta una versión anonimizada únicamente mediante la supresión del atributo *Nombre*.

Sin embargo, como ya vimos en los casos emblemáticos, *esto no es suficiente* para proteger la privacidad. En particular, no protege frente a los dos tipos de ataques más comunes:

- **Ataques de asociación de registros:** reasocian una fila específica con una persona.

- **Ataques de asociación de atributos:** intentan inferir un atributo sensible sobre una persona, aunque no sepamos exactamente qué fila es.

En el ejemplo, si sabemos que Pedro está en el dataset, que es hombre y vive en *Los Tilos 61*, podemos concluir que le corresponde la fila 1 y, por lo tanto, que tuvo COVID. Esto es un ataque de asociación de registros.

El problema no está sólo en los datos publicados, sino en los datos externos que podrían combinarse con ellos.

N°	Nombre	Sexo	Dirección	COVID
1	Pedro	M	Los Tilos 61	+
2	José	M	Los Alerces 74	+
3	Karla	F	Pasaje Aurora 331	+
4	María	F	Gran Avenida 8585	-

Tabla 1 / Registro de personas y estado de resultado COVID-19.

N°	Sexo	Dirección	COVID
1	M	Los Tilos 61	+
2	M	Los Alerces 74	+
3	F	Pasaje Aurora 331	+
4	F	Gran Avenida 8585	-

Tabla 2 / Conjunto de datos con identificadores explícitos removidos.

k-Anonimato: protegerse de la reidentificación por filas

Para mitigar este tipo de ataques, Samarati y Sweeney introdujeron la noción de *k-anonimato* [1]. Se trata de una propiedad matemática que, (in)formalmente, se puede describir así:

> Una tabla T satisface *k-anonimato* si, para cada combinación de valores de los cuasi-identificadores, existen al menos *k* filas en T que comparten exactamente esa misma combinación de valores.

Intuitivamente, si una tabla satisface *k*-anonimato con $k \geq 2$, entonces un atacante que conoce los cuasi-identificadores de un individuo (por ejemplo, sexo y dirección) no puede saber cuál de esas *k* filas corresponde a la persona. Mientras mayor sea *k*, más fuerte es la protección: el atacante queda "perdido" entre más candidatos posibles. En la tabla original, el grupo de

hombres que viven en *Los Tilos 61* aparece sólo una vez, por lo que la tabla es *1*-anónima: no ofrece anonimato efectivo.

¿Qué podemos hacer para lograr k ≥ 2?

Existen varias técnicas para transformar una tabla y hacer que satisfaga *k*-anonimato, entre ellas:

- Generalización de datos
  (abstraer valores, perdiendo precisión),
- Supresión de registros
  (eliminar filas completas),
- Supresión de celdas
  (ocultar valores puntuales),
- Anatomización,
- Permutación,
- Perturbación aleatoria, entre otras.

La más común y fácil de explicar es la *generalización*: en vez de publicar el valor exacto, se publica una versión más abstracta. Por ejemplo, podemos reemplazar la dirección exacta por la comuna, tal como se muestra en la Tabla 3.

Tras la generalización, cada combinación de cuasi-identificadores *(Sexo, Comuna)* aparece al menos dos veces. La tabla es *2*-anónima. Un atacante que sabe que Pedro es hombre y vive en Valdivia sólo puede acotar que es o la fila 1 o la 2, pero no distinguir entre ambas.

ℓ-Diversidad: evitar asociar personas a valores sensibles

El *k*-anonimato, sin embargo, no protege contra los ataques de asociación de atributos. En nuestro ejemplo, si sabemos que Pedro está en el dataset, que es hombre y vive en Valdivia, basta mirar la tabla *2*-anónima: todas las filas del grupo *(M, Valdivia)* tienen COVID. No sabemos exactamente cuál fila es Pedro, pero sí podemos concluir su atributo sensible.

Para mitigar este tipo de ataque, Machanavajjhala et al. [2] introdujeron la noción de *ℓ-diversidad*. De nuevo, se trata de una propiedad matemática, que se puede describir así:

> Una tabla satisface *ℓ-diversidad* si, para cada grupo de cuasi-identificadores, existen al menos ℓ valores sensibles distintos dentro de ese grupo.

Si una tabla satisface ℓ-diversidad para ℓ≥2, entonces, incluso si el atacante identifica el grupo al que pertenece una persona, *no puede estar seguro del valor sensible,* porque hay al menos dos posibilidades distintas. Al igual que con *k*-anonimato, valores mayores de ℓ entregan garantías de privacidad más fuertes.

N°	Sexo	Dirección	COVID
1	M	Valdivia	+
2	M	Valdivia	+
3	F	La Cisterna	+
4	F	La Cisterna	-

Tabla 3 / Conjunto de datos con direcciones generalizadas a nivel de comuna.

N°	Sexo	Dirección	COVID
1	-	Chile	+
2	-	Chile	+
3	-	Chile	+
4	-	Chile	-

Tabla 4 / Conjunto de datos con cuasi-identificadores altamente generalizados

Es interesante notar que ℓ-diversidad implica *k*-anonimato: si en un grupo hay al menos ℓ valores sensibles distintos, necesariamente ese grupo tiene al menos ℓ registros, por lo que también es al menos ℓ-anónimo.

En la tabla anterior, el grupo *(M, Valdivia)* tiene un solo valor sensible ("+"), por lo que la tabla sólo satisface *1*-diversidad, que es débil: permite ataques de asociación de atributos.

El costo de exigir más privacidad

Intentemos ahora lograr *2*-diversidad sólo mediante generalización. Una opción extrema sería generalizar tanto los cuasi-identificadores que prácticamente los hacemos desaparecer. Esto se observa en la Tabla 4, donde Sexo y Comuna han sido reemplazados por categorías tan amplias que se vuelven prácticamente inútiles.

En este punto, hemos perdido casi toda la estructura: ya no distinguimos ni sexo ni comuna. Es como si hubiéramos eliminado esas columnas. Esto ilustra un punto importante:

Figura 1 / Ejemplo ilustrativo de una imagen original y su versión con ruido pixel a pixel.

> *A medida que aumentamos k o ℓ para obtener más privacidad, los datos tienden a perder precisión y, con ello, utilidad.*

La anonimización clásica está atrapada en una tensión inevitable: no existe "privacidad perfecta" con "utilidad perfecta" al mismo tiempo. Proteger más implica, en general, publicar datos más agregados, menos detallados y menos útiles para ciertos análisis.

La gran limitación de la anonimización clásica

Para empeorar las cosas, incluso si logramos tablas que satisfacen *k*-anonimato y ℓ-diversidad, estas técnicas siguen teniendo una debilidad estructural: dependen fuertemente de la información auxiliar que podría tener un atacante.

Si alguien conoce suficiente contexto sobre una persona —por ejemplo, sabe que no pertenece a ciertos valores sensibles posibles, o tiene acceso a bases de datos externas muy informativas—, aún podría inferir su atributo sensible, pese a las transformaciones.

Este es el gran problema de la anonimización clásica: sus garantías de privacidad son relativas a lo que suponemos que el atacante *no* sabe.

Para mitigar esta dependencia de la información auxiliar, se ha desarrollado una técnica más moderna, privacidad diferencial, que ofrece garantías formales de privacidad incluso frente a atacantes con mucha información extra. En la siguiente sección describiremos en qué consiste y cómo se compara con estas técnicas clásicas.

Privacidad diferencial

La *privacidad diferencial* (DP) [3] representa un cambio de paradigma frente a las técnicas clásicas de anonimización como el *k*-anonimato y la ℓ-diversidad. A diferencia de ellas, la privacidad diferencial no es una propiedad de los datos, sino una propiedad formal —matemática— del mecanismo que genera o publica esos datos.

Su idea central es sorprendentemente intuitiva:

> *El resultado de un análisis debe ser prácticamente el mismo haya o no participado una persona en la base de datos.*

Esto calza con una noción muy natural de privacidad: *"Nada malo debería ocurrirme como resultado de participar en un estudio. Si algo malo pasa, habría pasado igual incluso si yo no hubiera participado"*.

Lo más importante es que la privacidad diferencial es independiente de la información auxiliar que pueda poseer un adversario. Esto incluye escenarios extremos, como cuando el atacante conoce toda la base de datos excepto la fila objetivo. En este sentido, la privacidad diferencial previene los ataques clásicos que afectan al *k*-anonimato y a la ℓ-diversidad, y se acerca —según lo que se conoce hoy— a la única forma de anonimización "verdadera" en sentido robusto.

Una definición (ligeramente) más formal

Sea un mecanismo *F* que recibe una base de datos y produce un resultado numérico. Decimos que *F* es ε-*diferencialmente*

privado (ε-DP) si, al aplicarlo a dos bases de datos idénticas excepto por un solo individuo, los resultados que entrega son indistinguibles. Esta indistinguibilidad se logra mediante ruido agregado de forma cuidadosamente calibrada. La belleza de DP está en su universalidad: no restringe qué sabe el adversario, no impone supuestos sobre el mundo externo y no depende de clasificar columnas como "cuasi-identificadores" o "sensibles". Es una propiedad matemática pura del mecanismo.

Una intuición visual: ruido a nivel de píxeles

Para construir intuición, imaginemos que queremos publicar una imagen. En la Figura 1, la imagen original está a la izquierda, mientras que a la derecha, mostramos la misma imagen, pero con ruido añadido pixel a pixel.

Si hacemos zoom sobre un pixel particular, el ruido hace que sea imposible saber si el color original era más claro o más oscuro. Este fenómeno recibe el nombre de *denegación plausible*: el pixel puede "negar" haber tenido su valor real porque el ruido lo enmascara.

Pero si miramos la imagen completa, podemos seguir distinguiendo su estructura global: contornos, colores predominantes, patrones. Esto es exactamente lo que permite la privacidad diferencial: perder precisión en lo individual, manteniendo utilidad estadística a nivel agregado.

Si añadimos muy poco ruido, la privacidad se pierde; si añadimos demasiado, como en la Figura 2, la utilidad desaparece y obtenemos un manchón irreconocible. El hilo conductor de DP es encontrar el punto intermedio óptimo.

¿Cómo se decide cuánto ruido agregar?

Todo depende de la sensibilidad de la consulta: cuánto puede cambiar el resultado si cambia un solo individuo.

Por ejemplo, para la consulta *f(x) = #(personas con COVID)*, la sensibilidad es 1, porque agregar o quitar a alguien cambia el resultado como máximo en 1.

Con esto, una forma más básica de crear un mecanismo diferencialmente privado es:

$$F(x) = f(x) + Laplace(s/\varepsilon)\,,$$

donde *s* es la sensibilidad de la consulta, ε el *presupuesto de privacidad*, y Laplace es la distribución de ruido utilizada. Así, *F* es un mecanismo ε-DP.

Eliminar nombres no basta [...] para proteger la privacidad.

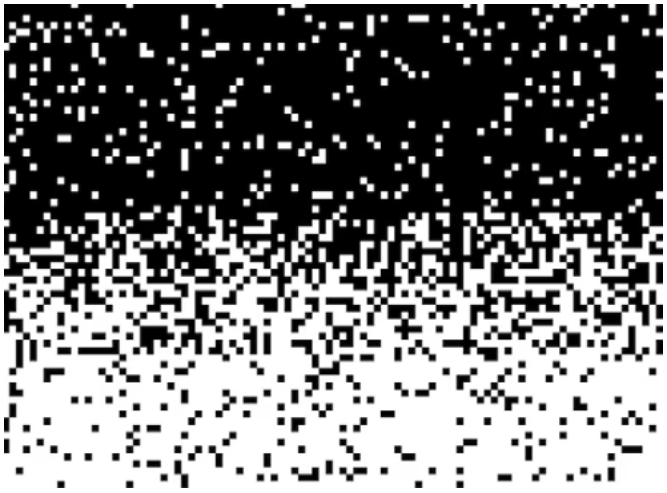

Figura 2 / Ejemplo de imagen con ruido extremo donde se pierde completamente la utilidad.

A mayor *ε*: menos ruido → más utilidad, pero menos privacidad.
A menor *ε*: más ruido → más privacidad, pero menos precisión.

Elegir ε es difícil y requiere experimentación: es uno de los temas más activos en la literatura actual.

Ejemplo numérico

Si el número real de personas con COVID es 14.237, una versión 2-DP podría devolver valores como 14.211, 14.260 o 14.237 (casualmente), valores cercanos, plausibles, y estadísticamente útiles.

Es importante mencionar que aquí usamos la *propiedad de post-procesamiento*: cualquier modificación posterior al resultado (por ejemplo, truncarlo a enteros) no disminuye la garantía ε-DP.

Consultas maliciosas y la importancia del ruido

Consideremos una consulta maliciosa: *g(x) = #(personas llamadas "Alan Brito" con COVID)*. Supongamos que en la base real el valor es 1. Una versión *2*-DP podría dar: 18, 47, –0,68. Estos resultados *no son útiles*, pero está bien: la consulta en sí misma era peligrosa. La privacidad diferencial está diseñada precisamente para impedir que consultas hiperespecíficas o maliciosas revelen secretos individuales.

¿Y si repetimos la consulta muchas veces?

El lector atento notará que si consultamos repetidamente, con ruido distinto cada vez, podríamos aproximar el valor original a través de un histograma, como se ilustra en la Figura 3.

Para evitarlo existe el *presupuesto de privacidad* ε: cada consulta "consume" parte del presupuesto y el sistema debe dividirlo entre las consultas.

Esto se formaliza mediante la propiedad de *composición secuencial*:

Si un mecanismo publica dos resultados, uno ε_1-DP y otro ε_2-DP, entonces el mecanismo completo es $(\varepsilon_1 + \varepsilon_2)$-DP.

Por ejemplo, si queremos publicar cuatro veces una consulta con presupuesto total ε, cada una debe usar $\varepsilon/4$, lo que produce resultados más ruidosos e impide reconstrucciones maliciosas.

Pero aquí aparece una dificultad importante: si hacemos muchas consultas, el ε disponible para cada una se vuelve muy pequeño, y el ruido agregado puede crecer hasta un punto en que la utilidad disminuye drásticamente.

Composición paralela: cuando los datos no se solapan

Afortunadamente, existe otra propiedad fundamental de la privacidad diferencial: la *composición paralela*.

Si múltiples mecanismos ε-DP se aplican a conjuntos disjuntos de individuos, el mecanismo conjunto sigue siendo ε-DP.

Esto tiene una consecuencia práctica muy valiosa: si las consultas se aplican sobre partes distintas de la población, no es necesario pagar el costo de la composición secuencial, es decir, no se "consume" más presupuesto de privacidad. En la práctica, podemos aplicar varias consultas ruidosas "gratis" en términos de ε.

Esto es especialmente útil para publicar histogramas, donde cada celda (o *bin*) corresponde a un grupo distinto de personas. Por ejemplo, si queremos contar casos de COVID separados por sexo y comuna, podemos aplicar ruido laplaciano a cada celda sin dividir ε, porque cada celda involucra individuos diferentes y, por lo tanto, cumple las condiciones de la composición paralela. Por ejemplo, podemos partir de la Tabla 5, que contiene los conteos exactos de casos para cada combinación de sexo y comuna.

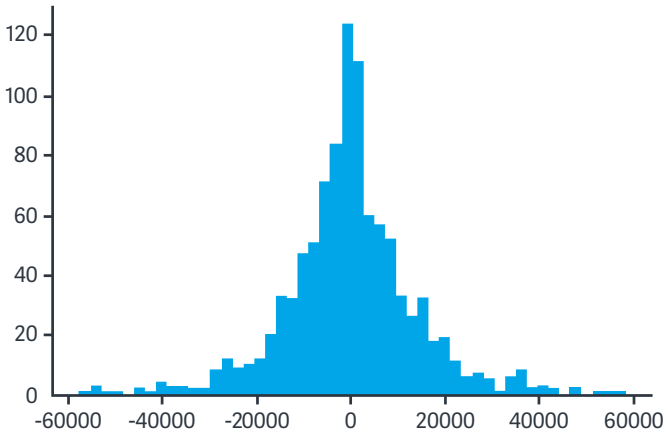

Figura 3 / Histograma obtenido al repetir una consulta con ruido Laplace.

Sexo	Comuna	COVID
M	Valdivia	295
F	Valdivia	743
M	La Cisterna	289
F	La Cisterna	122

Tabla 5 / Conteo real de casos de COVID por sexo y comuna.

Sexo	Comuna	COVID
M	Valdivia	296
F	Valdivia	740
M	La Cisterna	286
F	La Cisterna	126

Tabla 6 / Conteo publicado con ruido laplaciano y redondeo.

Si aplicamos ruido Laplaciano según una distribución *Laplace(½)* a cada celda del histograma, luego, gracias a la propiedad de postprocesamiento de la privacidad diferencial, podemos truncar los valores al entero más cercano sin perder la garantía de privacidad. El resultado final puede observarse en la Tabla 6, que muestra el histograma publicado después de añadir ruido y redondear los valores. En este caso, la utilidad estadística se preserva y la garantía ε-DP permanece totalmente vigente.

Más allá de consultas agregadas: datos sintéticos y modelos de IA

Como la privacidad diferencial es una propiedad de mecanismos, no de tablas, su aplicación va mucho más allá de consultas individuales. Hoy en día se usa para: generar datos sintéticos con ruido estadístico controlado, o para entrenar modelos de aprendizaje automático con privacidad garantizada.

En este último caso, la garantía es potente: si un modelo fue entrenado con tus datos de forma diferencialmente privada, entonces su comportamiento sería prácticamente el mismo aunque tus datos no hubieran estado en el entrenamiento. En otras palabras: Participar o no en el entrenamiento no cambia lo que el modelo puede revelar sobre ti. Esto es extremadamente relevante en la era de los modelos generativos, donde fragmentos de datos de entrenamiento pueden "colarse" en outputs inesperados.

Conclusión: ¿Cómo se comparan la anonimización clásica y la privacidad diferencial?

Las técnicas de anonimización clásica —como el *k*-anonimato y la ℓ-diversidad— buscan reducir el riesgo de reidentificación eliminando identificadores y agrupando datos. Son fáciles de aplicar y ampliamente usadas, pero dependen fuertemente de cómo se seleccionan y transforman los cuasi-identificadores. Además, son vulnerables a la información auxiliar: un adversario bien informado puede, en muchos casos, reconstruir o inferir atributos sensibles incluso cuando la tabla cumple las propiedades requeridas.

La belleza de DP está en su universalidad: no restringe qué sabe el adversario, no impone supuestos sobre el mundo externo y no depende de clasificar columnas.

La privacidad diferencial, en cambio, sigue un enfoque completamente distinto. En vez de proteger los datos publicados, es una propiedad del mecanismo que produce esos datos, garantizando que el resultado sea prácticamente indistinguible haya o no participado un individuo. Su fortaleza clave es que la garantía no depende del conocimiento del adversario, incluso si este conoce todo menos un registro. Por eso se considera la técnica más robusta disponible hoy.

En resumen:

- **Anonimización clásica**: útil, intuitiva, pero frágil ante datos auxiliares.
- **Privacidad diferencial**: formal, robusta y conceptualmente alineada con una noción moderna de privacidad.

La entrada en vigencia de la nueva Ley de Protección de Datos Personales en diciembre de 2026 hará que estas distinciones sean especialmente importantes. La ley exige anonimización robusta y evaluaciones de riesgo considerando posibles ataques de reidentificación. En este escenario, la privacidad diferencial emerge como una herramienta clave para cumplir con las nuevas obligaciones y proteger adecuadamente a las personas al publicar o compartir datos. **B**

Agradecimientos

Gracias a Arturo Kullmer por proporcionar los ejemplos usados aquí para explicar anonimización y privacidad diferencial.

Referencias

[1] Pierangela Samarati y Latanya Sweeney. 1998. *Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression.*

[2] Ashwin Machanavajjhala, Johannes Gehrke, Daniel Kifer, y Muthuramakrishnan Venkitasubramaniam. 2006. *L-diversity: Privacy beyond k-anonymity.* 22nd International Conference on Data Engineering (ICDE'06), 24–24.

[3] Cynthia Dwork. 2006. *Differential Privacy.* En Automata, Languages and Programming, 33rd International Colloquium, ICALP 2006, Proceedings, Part II (LNCS, Vol. 4052). Springer, 1–12.