Sistemas de optimización para el balance entre resiliencia a ataques a la privacidad y rendimiento en Aprendizaje Federado

Autor: Julio Pérez Cabeza

Tutores: Javier Merí de la Maza, Nuria Rodríguez Barroso

Departamento de Análisis Matemático, Departamento de Ciencias de la Computación e Inteligencia Artificial

20 de julio de 2025

Doble Grado en Ingeniería Informática y Matemáticas, UGR

UNIVERSIDAD DE GRANADA

Contenido

1. Introducción
   - Privacidad en los datos
   - Leyes de privacidad
   - Solución al problema
2. Optimización
   - Marco teórico
   - Problemas de Programación Lineal
   - El método Símplex
   - Optimización convexa
3. Aprendizaje Federado y Privacidad Diferencial
   - Aprendizaje Federado
   - Privacidad Diferencial
   - Optimización dinámica de la PD
4. Experimentación
5. Conclusiones y trabajos futuros

Privacidad en los datos

En la actualidad, los datos son uno de los recursos más valiosos para grandes empresas multinacionales.

Privacidad en los datos

En la actualidad, los datos son uno de los recursos más valiosos para grandes empresas multinacionales.
La recopilación masiva de datos plantea **riesgos significativos**:

- Pérdida de anonimato.
- Filtraciones y accesos no autorizados.
- Uso indebido con fines comerciales o de vigilancia.
- Corrupción de los datos con fines de manipulación.

Privacidad en los datos

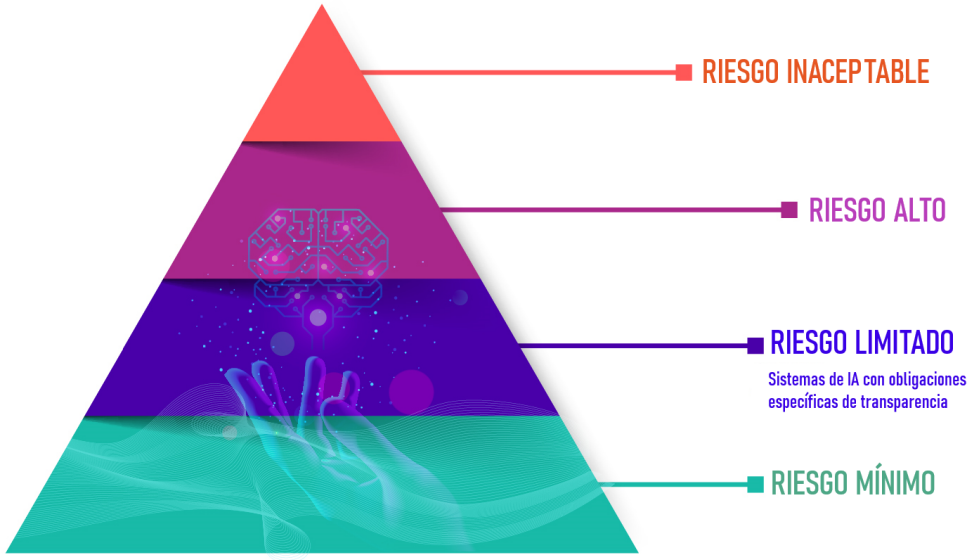
En la actualidad, los datos son uno de los recursos más valiosos para grandes empresas multinacionales.
La recopilación masiva de datos plantea **riesgos significativos**:

- Pérdida de anonimato.
- Filtraciones y accesos no autorizados.
- Uso indebido con fines comerciales o de vigilancia.
- Corrupción de los datos con fines de manipulación.

Reto: Proteger los datos sin comprometer su utilidad

Leyes de privacidad

Con el paso de los años, se han establecido diversas leyes y regulaciones para proteger la privacidad de los datos.



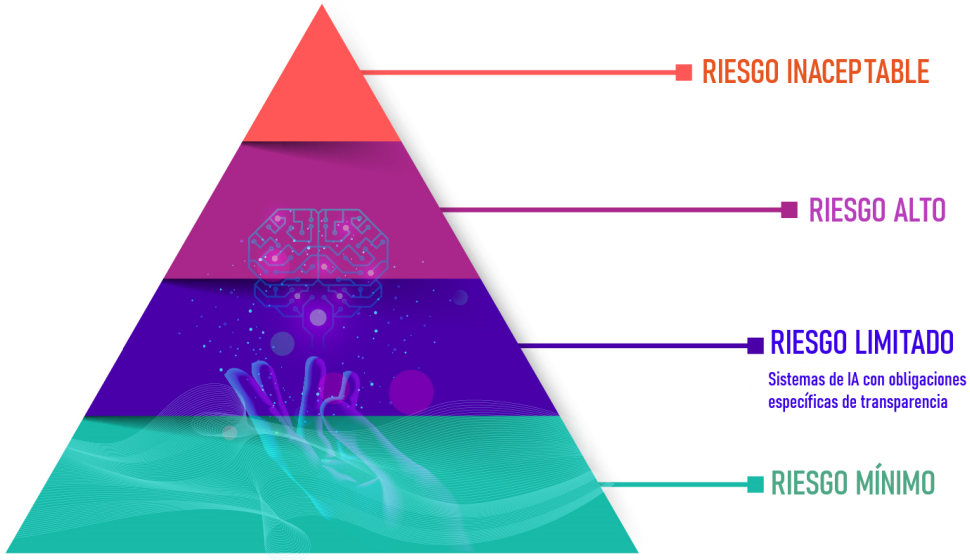
Pirámide con los niveles de riesgo para sistemas de IA, según la Unión Europea.

Fuente de la imagen: European Commission. "Regulatory Framework Proposal on Artificial Intelligence". Accessed: 2025-05-03. 2024. URL: https://digital-strategy.ec.europa.eu/en/policies/regulatory-framework-ai.

Leyes de privacidad

Con el paso de los años, se han establecido diversas leyes y regulaciones para proteger la privacidad de los datos.

- Reglamento General de Protección de Datos (GDPR), 2018.



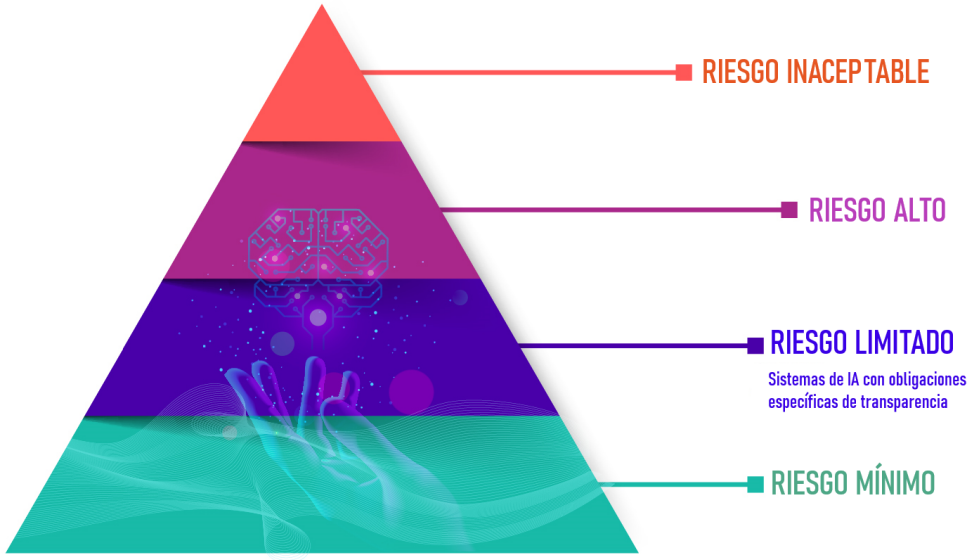
Pirámide con los niveles de riesgo para sistemas de IA, según la Unión Europea.

Fuente de la imagen: European Commission. "Regulatory Framework Proposal on Artificial Intelligence". Accessed: 2025-05-03. 2024. URL: https://digital-strategy.ec.europa.eu/en/policies/regulatory-framework-ai.

Leyes de privacidad

Con el paso de los años, se han establecido diversas leyes y regulaciones para proteger la privacidad de los datos.

- Reglamento General de Protección de Datos (GDPR), 2018.
- Ley o Acto de Inteligencia Artificial de la UE (AI Act), 2021.

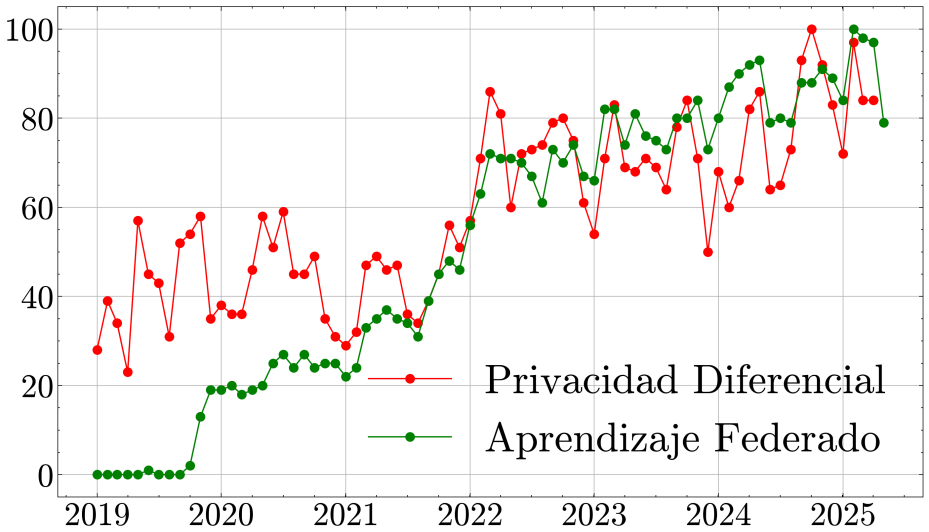


Pirámide con los niveles de riesgo para sistemas de IA, según la Unión Europea.

Fuente de la imagen: European Commission. "Regulatory Framework Proposal on Artificial Intelligence". Accessed: 2025-05-03. 2024. URL: https://digital-strategy.ec.europa.eu/en/policies/regulatory-framework-ai.

Privacidad Diferencial en Aprendizaje Federado

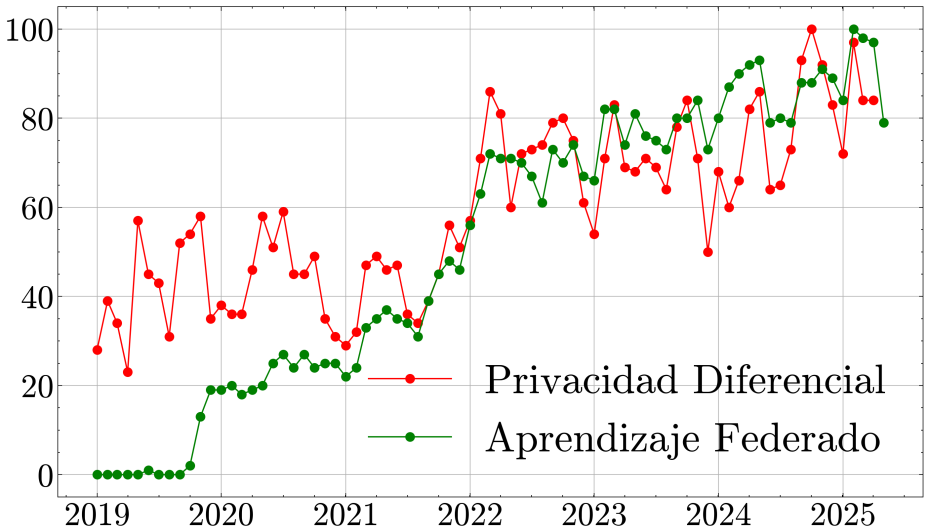
Para abordar el problema de la privacidad en el aprendizaje automático y profundo, la combinación de dos herramientas ha demostrado ser efectivas:



Interés en la combinación de Privacidad Diferencial y Aprendizaje Federado.

Privacidad Diferencial en Aprendizaje Federado

Para abordar el problema de la privacidad en el aprendizaje automático y profundo, la combinación de dos herramientas ha demostrado ser efectivas:



Interés en la combinación de Privacidad Diferencial y Aprendizaje Federado.

- **Privacidad Diferencial**: Mecanismo que ha demostrado servir como defensa ante numerosos ataques en *Deep Learning*.
- **Aprendizaje Federado**: Paradigma cuyo propósito principal es evitar que los datos deban ser centralizados, evitando riesgos de privacidad.

Trade-off entre privacidad y rendimiento

Objetivo

Encontrar un **equilibrio** entre la privacidad de los datos y el rendimiento del modelo.

Solución

Optimización de la aplicación de Privacidad Diferencial en el contexto del Aprendizaje Federado mediante herramientas de optimización lineal.

En resumen, se busca:

- **Aumentar la privacidad** de los datos durante el entrenamiento.
- **Mantener el rendimiento** del modelo entrenado.

Fundamentos

Fundamentos teóricos

Se hace un repaso de importantes resultados sobre:

- Álgebra Lineal.
- Afinidad y convexidad.
- Análisis Funcional (2º Teorema de Hahn-Banach, Teorema de representación de Riesz).
- Prueba del Lema de Farkas.

Serán de gran utilidad para entender los resultados que se presentan a lo largo del trabajo.

Puntos extremos

Punto extremo

Sea $C \subseteq X$ un conjunto convexo. Un punto $x \in C$ se dice **punto extremal o extremo** de C si no puede escribirse como combinación convexa no trivial de otros dos puntos de C. Es decir, si

$$x = \lambda y + (1 - \lambda)z \quad \text{con } y, z \in C,\ \lambda \in (0, 1),$$

entonces se tiene que $y = z = x$.

Idea fundamental: Los puntos extremos de un conjunto convexo juegan un papel crucial en la teoría de la optimización.

Teorema de Minkowski-Carathéodory

Teorema de Carathéodory

Sea A un conjunto contenido en un conjunto afín de dimensión k. Entonces, cualquier vector $x \in \text{conv}(A)$ puede representarse como una combinación convexa de $k+1$ o menos elementos de A.

Teorema de Minkowski-Carathéodory

Sea K un subconjunto compacto y convexo de $\mathbb{R}^n$. Entonces, todo $x \in K$ se puede escribir como combinación convexa de, a lo sumo, $n+1$, puntos extremos de K. En particular, $K = conv(ex(K))$.

Problemas de Programación Lineal

Dado un subconjunto $A \subset \mathbb{R}^n$, y una función $f : A \to \mathbb{R}$, se pretende encontrar un punto a_* de manera que $f(a_*) \geq f(a)$ para todo $a \in A$.

Caso particular: Programación Lineal

Sea $X_0 = \mathbb{R}^n$ y sean las funciones $f : X_0 \longrightarrow \mathbb{R}$ y $g : X_0 \longrightarrow X_0$ lineales tales que

$$f(x) = \langle -b, x \rangle, \quad g(x) = \begin{pmatrix} Ax - c \\ -x \end{pmatrix}$$

con A una matriz $m \times n$, $m, n \in \mathbb{N}$ y c una constante real. Sea también

$$X = \{x \in X_0 : g(x) \leq 0\}.$$

Problema: Maximización de f en X.

Presentaciones de PPL

1. **Forma Estándar (PPLE):**

   Maximizar $f(x) = \langle b, x \rangle$ *sujeto a* $Ax \leq c$, *con* $x \geq 0$.

   Donde $x \in \mathbb{R}^n$, A es una matriz $m \times n$, $b \neq 0$ un vector de $\mathbb{R}^n$ y c es un vector de $\mathbb{R}^m$.

Presentaciones de PPL

1. **Forma Estándar (PPLE):**

   Maximizar $f(x) = \langle b, x \rangle$ *sujeto a* $Ax \leq c$, *con* $x \geq 0$.

   Donde $x \in \mathbb{R}^n$, A es una matriz $m \times n$, $b \neq 0$ un vector de $\mathbb{R}^n$ y c es un vector de $\mathbb{R}^m$.

2. **Forma Normal (PPLN):** Se introduce la restricción de que todas las variables deben ser no negativas. Se convierte en

   Maximizar $f(x) = \langle b, x \rangle$ *sujeto a* $\widetilde{A}x \leq \widetilde{c}$.

Presentaciones de PPL

1. **Forma Estándar (PPLE):**

   Maximizar $f(x) = \langle b, x \rangle$ *sujeto a* $Ax \leq c$, *con* $x \geq 0$.

   Donde $x \in \mathbb{R}^n$, A es una matriz $m \times n$, $b \neq 0$ un vector de $\mathbb{R}^n$ y c es un vector de $\mathbb{R}^m$.

2. **Forma Normal (PPLN):** Se introduce la restricción de que todas las variables deben ser no negativas. Se convierte en

   Maximizar $f(x) = \langle b, x \rangle$ *sujeto a* $\widetilde{A}x \leq \widetilde{c}$.

3. **Forma Canónica (PPLC):** Las restricciones de desigualdad se convierten en restricciones de igualdad, introduciendo un vector no negativo cuyas componentes se conocen como **variables de holgura**.

Motivación: La eficiencia

La eficiencia en la optimización de problemas es crucial, así como siempre lo ha sido en los problemas de *Machine Learning* y *Deep Learning*.

Teorema

Sea un PPLC, con solución x_0. Entonces existe un punto extremo z del conjunto factible que también es solución al problema.

Idea

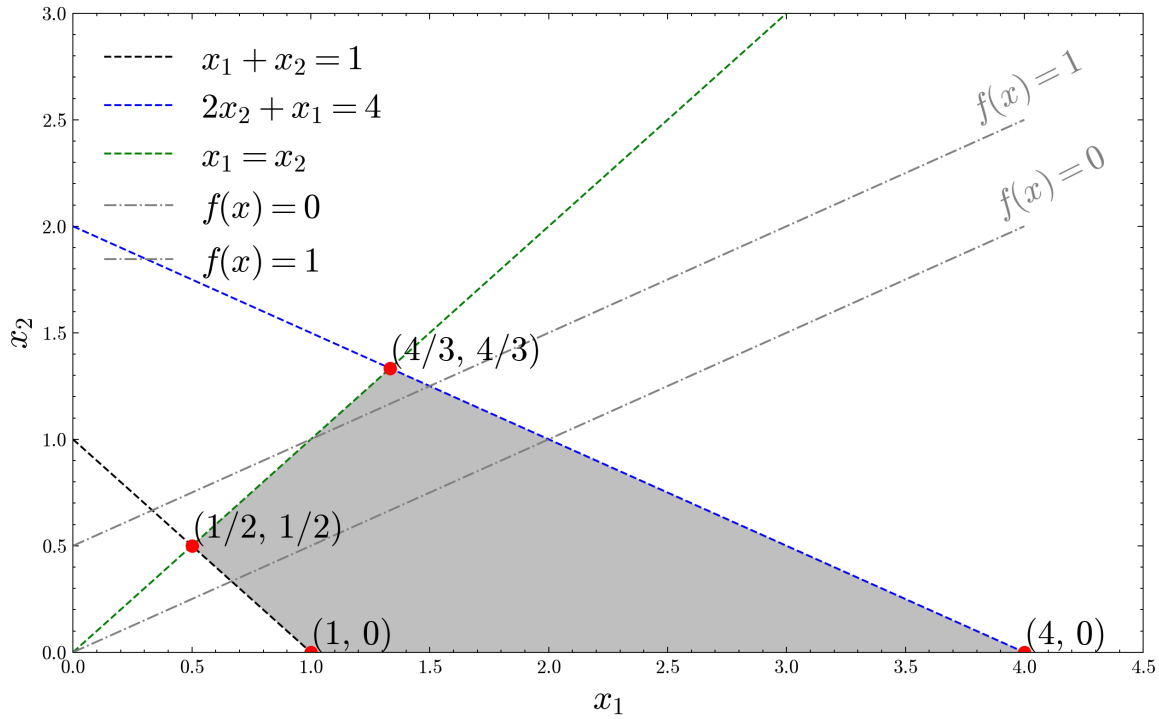
Dado un PPLC, podemos encontrar soluciones óptimas en los puntos extremos del conjunto factible, reduciendo la complejidad de la búsqueda al **acotar** el análisis a estos puntos.

Motivación: La eficiencia II

$$f(x) = -x_1 + 2x_2$$

Restricciones:

$$\begin{aligned} -x_1 - x_2 &\leq -1, \\ -x_1 + x_2 &\leq 0, \\ x_1 + 2x_2 &\leq 4, \\ x_1 \geq 0, \quad x_2 &\geq 0 \end{aligned}$$

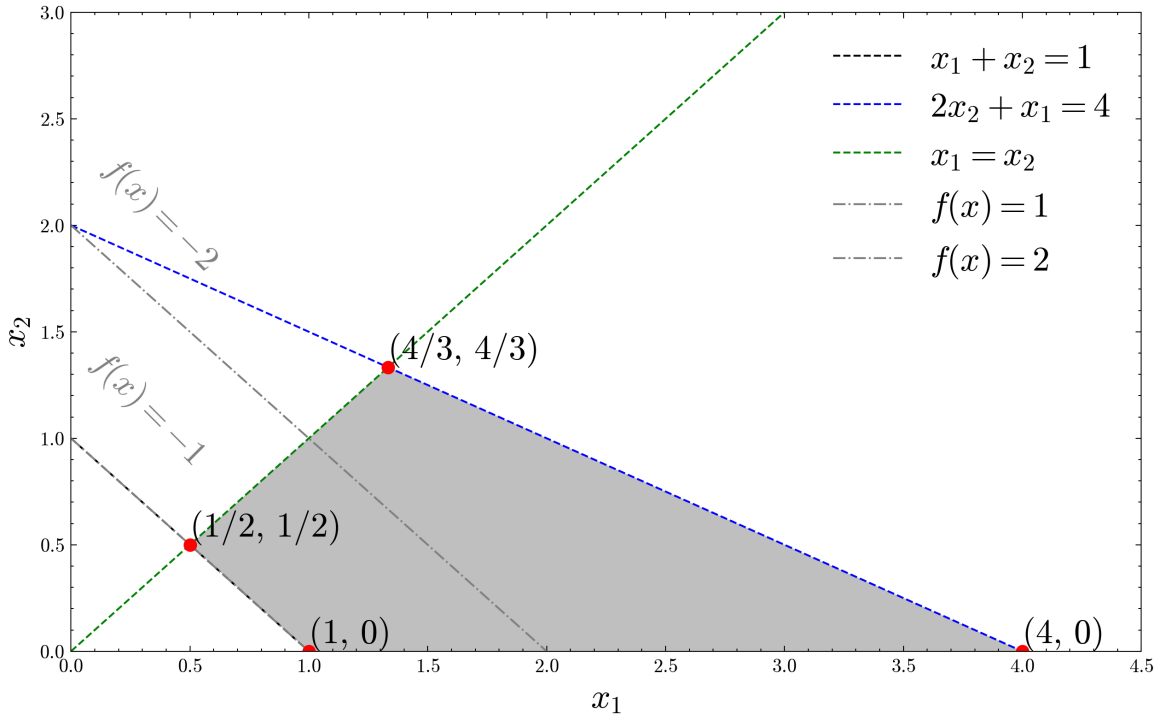


Motivación: La eficiencia II

$$f(x) = -x_1 - x_2$$

Restricciones:

$$\begin{aligned} -x_1 - x_2 &\leq -1, \\ -x_1 + x_2 &\leq 0, \\ x_1 + 2x_2 &\leq 4, \\ x_1 \geq 0, \quad x_2 &\geq 0 \end{aligned}$$



Motivación: La eficiencia III

Corolario

El número de puntos extremos de un conjunto factible es menor o igual a

$$\binom{n}{m} = \frac{n!}{m!(n-m)!}$$

Ejemplo

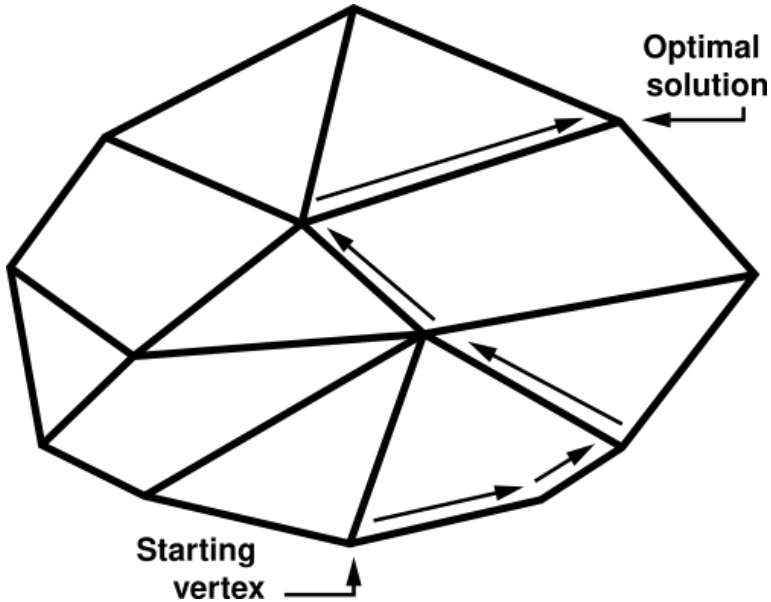
Para un sistema no muy grande, por ejemplo, de 5 ecuaciones de restricción y 25 variables, puede haber hasta

$$\binom{25}{5} = \frac{25!}{5!20!} = 53,130$$

puntos extremos.

Método Símplex

- Un sistema de desigualdades lineales define un poliedro como una región factible.
- Comienza en un punto extremo y se mueve a **otros puntos extremos adyacentes** del poliedro hasta que alcanza la solución óptima.



Representación del poliedro de soluciones factibles de un PPLC.

Fuente de la imagen: Wikipedia contributors. “Algoritmo símplex — Wikipedia, La enciclopedia libre”. 2024. URL: https://es.wikipedia.org/wiki/Algoritmo_s%C3%ADmplex.

Fases del método Símplex

Fase I

Dos opciones:

- Si el PPLC tiene solución, se busca una solución básica factible inicial.
- Si no tiene solución, se busca una solución básica factible inicial para un PPLC asociado, que es un PPLC sin solución.

Pasos de la fase II

1. **Función objetivo:** Expresión de la misma en términos de las variables básicas.
2. **Selección de la variable entrante:** Se elige la variable que aumentará su valor.
3. **Selección de la variable saliente:** Se elige la variable que disminuirá su valor.
4. **Actualización de la base**
5. **Iteración:** Se repiten los pasos hasta llegar a una solución óptima.

Optimización Convexa

Ahora, la función objetivo y las restricciones del problema son convexas: se trata de **una generalización de la programación lineal**.

Redefinición del problema

Sea $f : \mathbb{R}^n \to \mathbb{R}$ diferenciable y continua y sea $X \subset \mathbb{R}^n$ un conjunto convexo y compacto. Se plantea el problema

$$\begin{aligned} \text{mín} \quad & f(x) \\ \text{sujeto a} \quad & x \in X. \end{aligned}$$

Teorema

En un problema de minimización convexa, todo óptimo local es un óptimo global.

Métodos de gradiente descendente

En problemas con restricciones, los métodos deben asegurar que las soluciones **permanecen dentro del conjunto factible**.

Proyección de un punto

Dado $z \notin X$, la **proyección** de z sobre X es el $x^* \in X$ tal que

$$\|z - x^*\| = \min_{x \in X} \|z - x\|.$$

Método del gradiente proyectado:

1. Se calcula el gradiente de la función objetivo.
2. Se realiza un paso en la dirección opuesta al gradiente.
3. Se proyecta el punto resultante sobre el conjunto factible X.

Generalización a dimensión infinita

Teorema de Krein-Milman

Sea X un espacio normado. Entonces:

1. Si $A \subset X$ es un subconjunto débilmente compacto, entonces

$$A \subseteq \overline{conv(ex(A))},$$

y, si A es convexo, entonces se tiene la igualdad.

2. Si A es un subconjunto débilmente-$*$ compacto de X^*, entonces

$$A \subseteq \overline{conv^*(ex(A))},$$

y, si A es convexo, entonces se da la igualdad.

Generalización a dimensión infinita

Principio del máximo de Bauer

Sea X un espacio normado, A un subconjunto no vacío convexo y débilmente compacto de X. Sea f una función semicontinua superiormente y casi-convexa. Entonces f alcanza su máximo en un punto extremo de A, es decir, existe $x_0 \in ex(A)$ tal que $f(x) \leq f(x_0), \forall x \in A$.

Generalización a dimensión infinita

Principio del máximo de Bauer

Sea X un espacio normado, A un subconjunto no vacío convexo y débilmente compacto de X. Sea f una función semicontinua superiormente y casi-convexa. Entonces f alcanza su máximo en un punto extremo de A, es decir, existe $x_0 \in ex(A)$ tal que $f(x) \leq f(x_0), \forall x \in A$.

Principio del máximo de Bauer para funciones convexas

Toda función convexa y continua definida sobre un conjunto convexo y compacto alcanza su máximo en algún punto extremo del conjunto.

Generalización a dimensión infinita

Principio del máximo de Bauer

Sea X un espacio normado, A un subconjunto no vacío convexo y débilmente compacto de X. Sea f una función semicontinua superiormente y casi-convexa. Entonces f alcanza su máximo en un punto extremo de A, es decir, existe $x_0 \in ex(A)$ tal que $f(x) \leq f(x_0), \forall x \in A$.

Principio del máximo de Bauer para funciones convexas

Toda función convexa y continua definida sobre un conjunto convexo y compacto alcanza su máximo en algún punto extremo del conjunto.

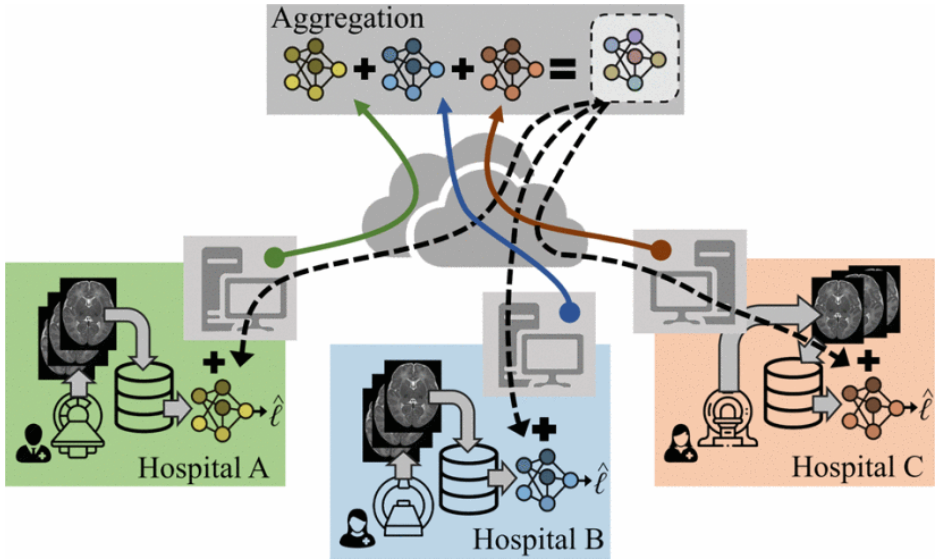
Utilidad

Abordar problemas de optimización en espacios de dimensión infinita, como por ejemplo, espacios de sucesiones o de funciones ($\ell_1, c_0, ...$).

Aprendizaje Federado

La utilidad de los datos reside en su capacidad para entrenar modelos de ML. Sin embargo, **la privacidad de los datos** es una preocupación importante.

- Los datos permanecen en los **dispositivos locales**.
- Se **envían actualizaciones** del modelo al servidor central.
- El servidor agrega las actualizaciones para mejorar el **modelo global**.

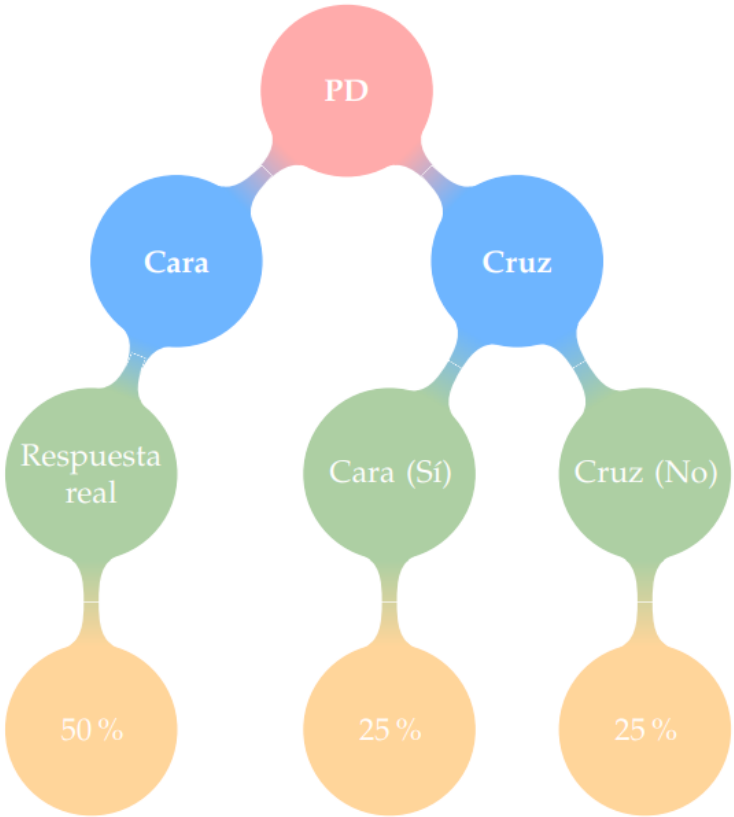


Esquema del Aprendizaje Federado.

Fuente de la imagen: M. Victoria Luzón et al. “A Tutorial on Federated Learning from Theory to Practice: Foundations, Software Frameworks, Exemplary Use Cases, and Selected Trends”. En: **IEEE/CAA Journal of Automatica Sinica** (2024).

Privacidad Diferencial

Pregunta: ¿Tomas café por la mañana?



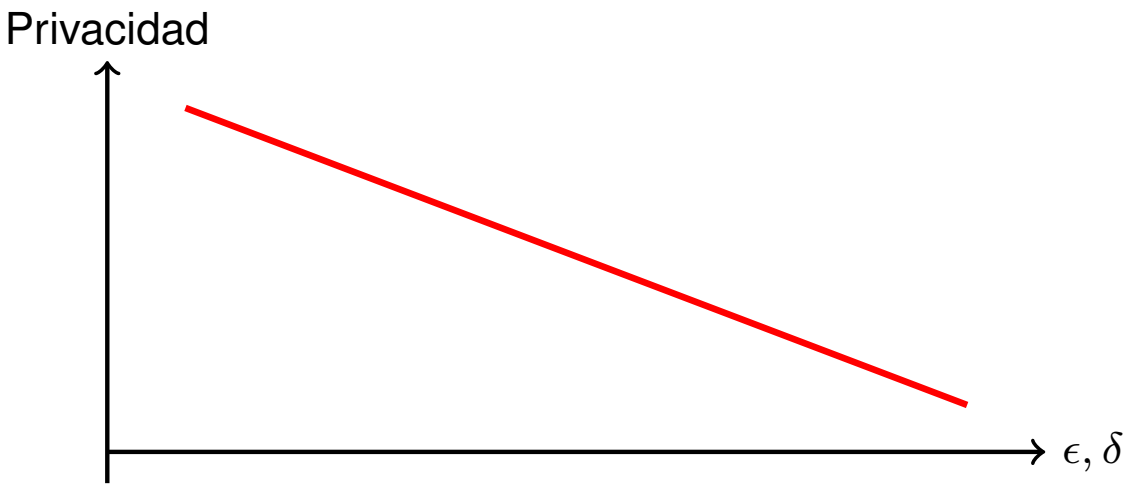
Ejemplo de Privacidad Diferencial.

Privacidad Diferencial

Definición

Se dice que un mecanismo de acceso a bases de datos, $\mathcal{M}$, preserva la (ϵ, δ)-PD si para todas las bases de datos vecinas x e y, para $0 < \delta < 1$, y para cada posible salida de $\mathcal{M}$, representada por $\mathcal{S}$, se cumple que:

$$P[\mathcal{M}(x) \in \mathcal{S}] \leq e^{\epsilon} P[\mathcal{M}(y) \in \mathcal{S}] + \delta$$



Relación inversa entre ϵ, δ y privacidad

Tipos de ataques

Ataques a la **privacidad**:

- Ataques basados en gradiente
- Ataques de inferencia de membresía
- Ataques de reconstrucción de datos

Ataques al **modelo**:

- Ataques de envenenamiento de datos
- Ataques *backdoor*
- Ataques de *label flipping*

Solución propuesta

Para abordar el problema en cada ronda de entrenamiento, se define el problema de optimización lineal:

Problema de minimización

$$\begin{aligned} \min_{\Delta\epsilon,\ \Delta\delta} \quad & c_1 \cdot \Delta\epsilon + c_2 \cdot \Delta\delta \\ \textit{sujeto a:} \quad & \Delta\epsilon + \Delta\delta \leq \text{presupuesto}_{\text{total}} - \text{presupuesto}_{\text{usado}} \\ & \epsilon \in [0{,}5, 1{,}5] \\ & \delta \in \left[10^{-4}, 10^{-7}\right] \end{aligned}$$

Entorno experimental

Conjuntos de datos

- **MNIST**: imágenes en blanco y negro de dígitos escritos a mano.
- **Fashion-MNIST**: imágenes en blanco y negro de ropa y accesorios.
- **CIFAR-10**: imágenes en color de animales y objetos.

Conjuntos	Tipo	Train	Test	Etiquetas
MNIST	Escala de grises	60,000	10,000	10
FashionMNIST	Escala de grises	60,000	10,000	10
CIFAR-10	RGB	50,000	10,000	10

Tabla: Resumen de los tres conjuntos de datos usados en la experimentación.

Entorno experimental

Conjuntos de datos

- **MNIST**: imágenes en blanco y negro de dígitos escritos a mano.
- **Fashion-MNIST**: imágenes en blanco y negro de ropa y accesorios.
- **CIFAR-10**: imágenes en color de animales y objetos.

Parámetros experimentales

- **Número de rondas de entrenamiento**: 100
- **Número de clientes**: 10 y 2
- **Número de épocas por ronda**: 1
- **Defensa**: DP-local y CDP

Entorno experimental

Conjuntos de datos

- **MNIST**: imágenes en blanco y negro de dígitos escritos a mano.
- **Fashion-MNIST**: imágenes en blanco y negro de ropa y accesorios.
- **CIFAR-10**: imágenes en color de animales y objetos.

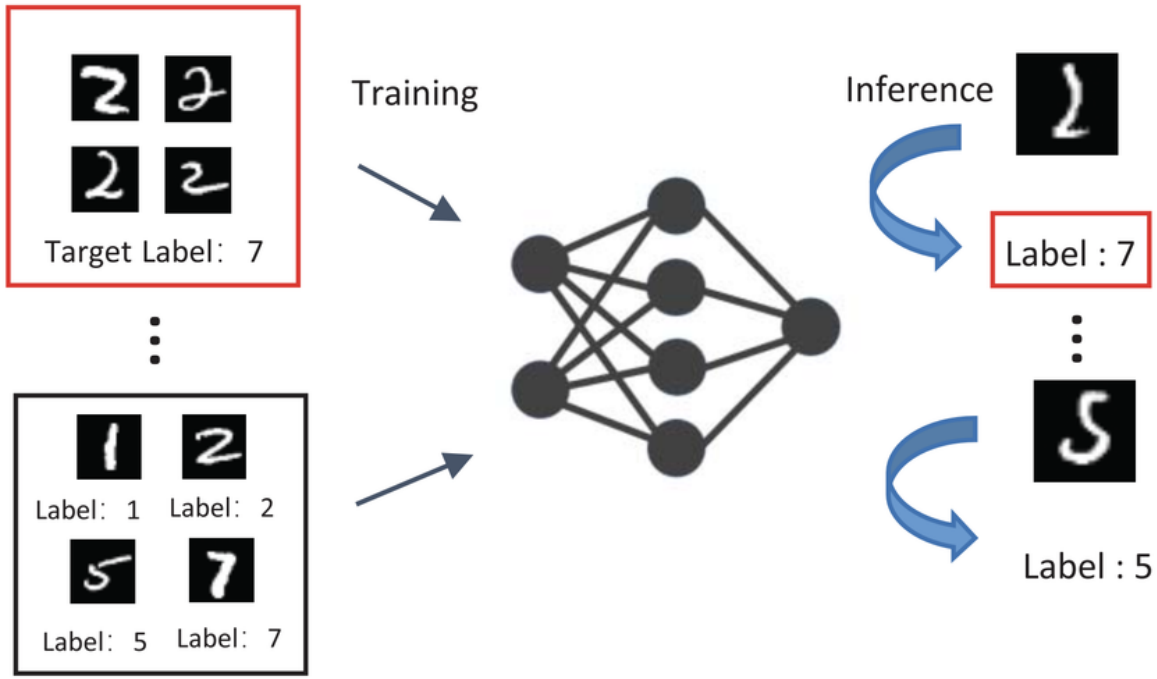
Parámetros experimentales

- **Número de rondas de entrenamiento**: 100
- **Número de clientes**: 10 y 2
- **Número de épocas por ronda**: 1
- **Defensa**: DP-local y CDP

Se compara en todo momento: Aplicación estática de la PD y la optimización dinámica.

Ataques simulados

- **Label Flipping**: Se cambian las etiquetas de las imágenes.



Ejemplo de ataque de Label Flipping.

Fuente de la imagen: Miguel A Ramirez et al. “New data poison attacks on machine learning classifiers for mobile exfiltration”. En: **arXiv preprint arXiv:2210.11592** (2022).

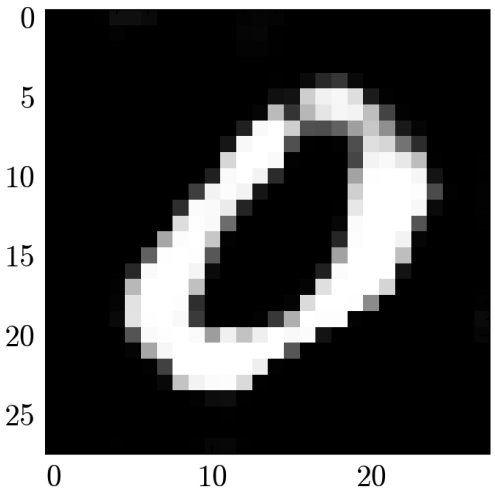
Ataques simulados

- **Label Flipping**: Se cambian las etiquetas de las imágenes.
- **Backdoor**: Se inyecta un patrón específico en las imágenes.

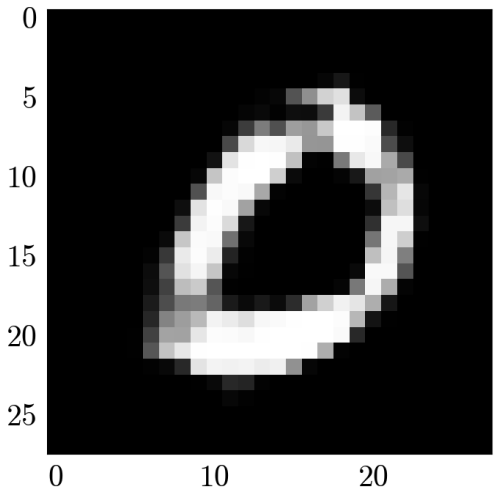
Ejemplo de ataque backdoor, inyección de un patrón específico en las imágenes.

Ataques simulados

- **Label Flipping**: Se cambian las etiquetas de las imágenes.
- **Backdoor**: Se inyecta un patrón específico en las imágenes.
- **DMUG**: Reconstrucción de muestras mediante una GAN.



Reconstrucción final con PD estática.



Reconstrucción final con PD optimizada.

Ataques simulados

- **Label Flipping**: Se cambian las etiquetas de las imágenes.
- **Backdoor**: Se inyecta un patrón específico en las imágenes.
- **DMUG**: Reconstrucción de muestras mediante una GAN.
- **Ataque gaussiano**: Se mandan actualizaciones del modelo con ruido gaussiano.

Métricas de evaluación

Se hace una monitorización de:

- **Rendimiento del modelo**: Porcentaje de predicciones correctas.
- **Epsilon acumulado**: Privacidad consumida del presupuesto total.
- **Éxito de los ataques**:
  - Degradamiento del rendimiento del modelo.
  - *Attack success rate (ASR)*.
  - Distancia mínima entre las muestras originales y las reconstruidas.

Se comparan los resultados obtenidos con PD estática y optimizada en cada métrica.

Herramientas para el desarrollo

Framework FLEXible-FL.

Framework Opacus.

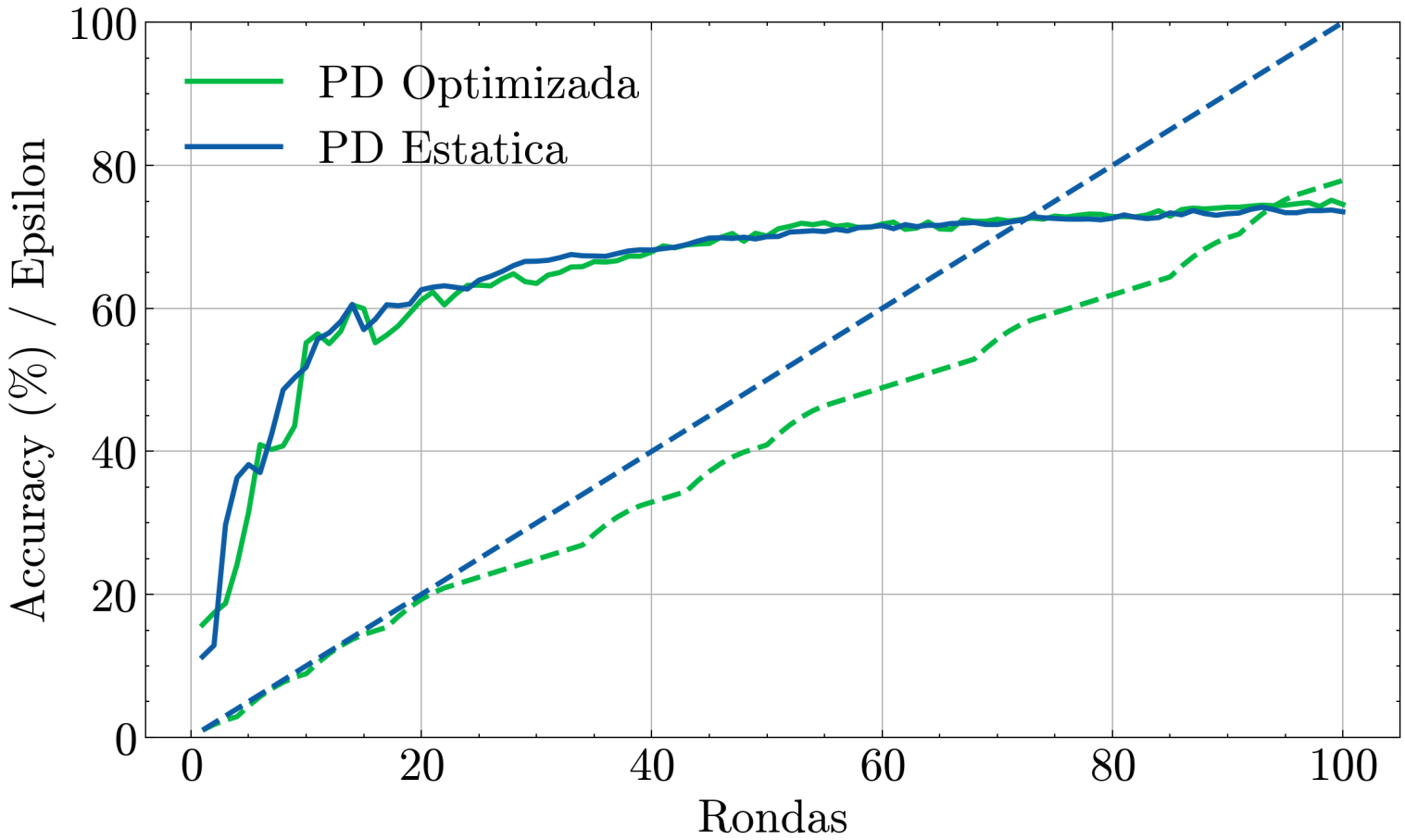
SciPy

LinProg de Scipy.

PyTorch

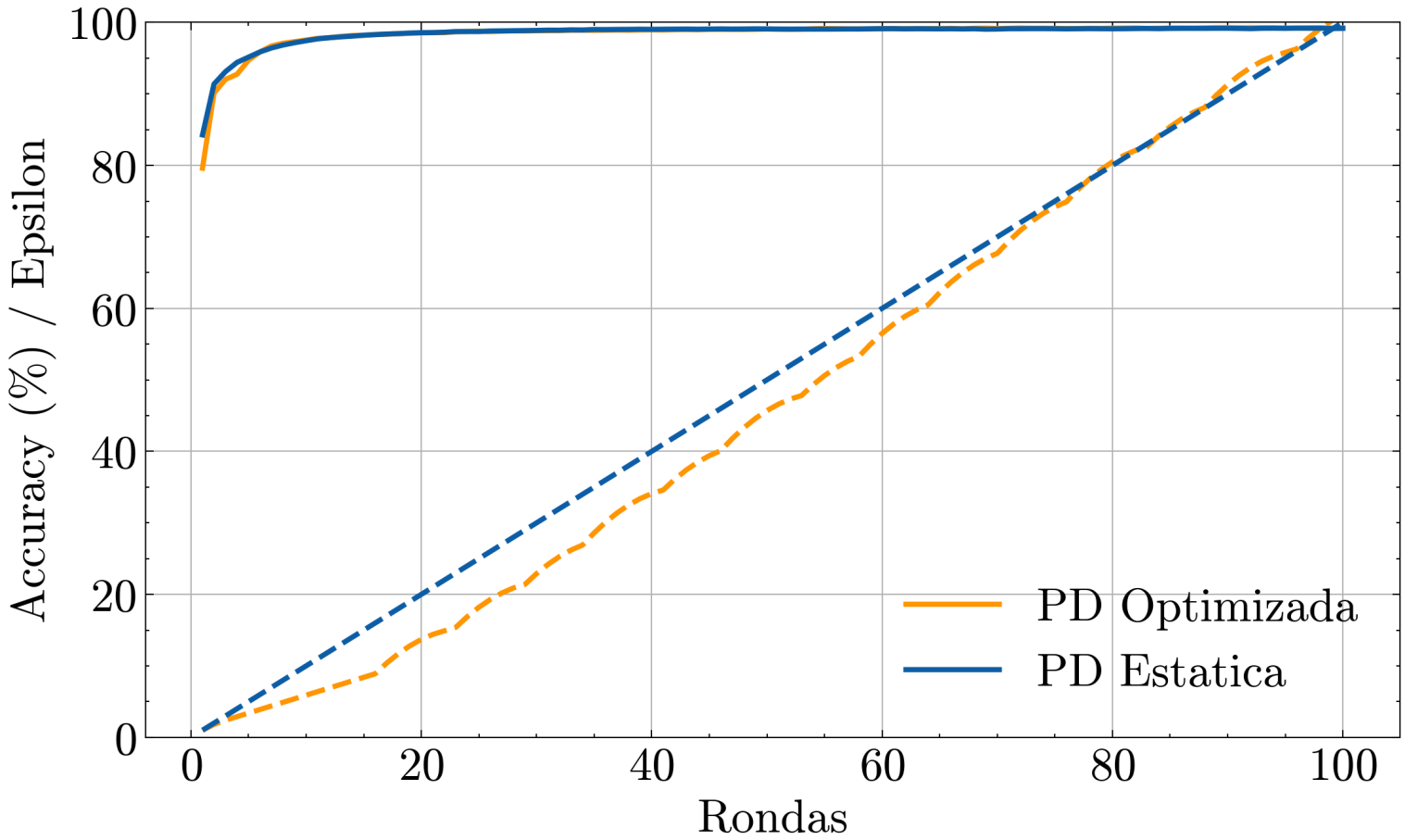
Pytorch para DL en Python.

Resultados



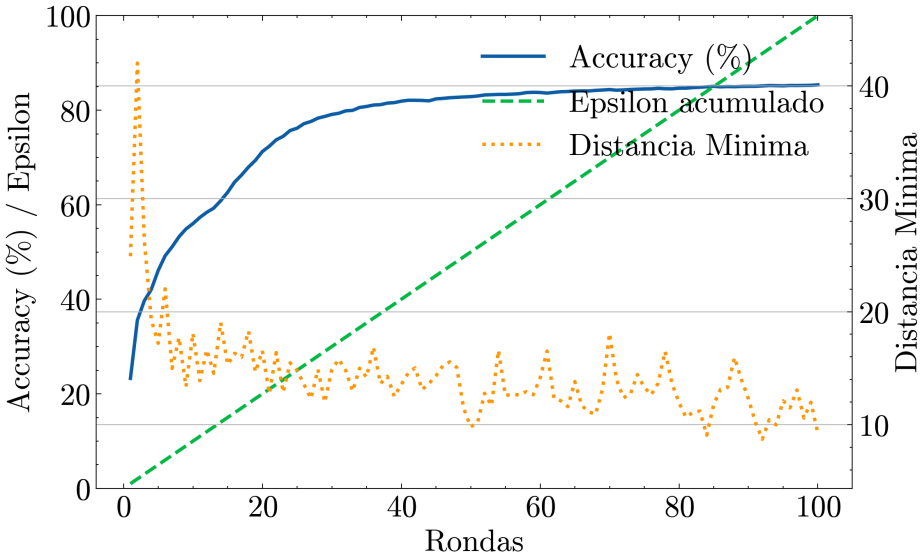
Resultados para Label Flipping con FashionMNIST.

Resultados

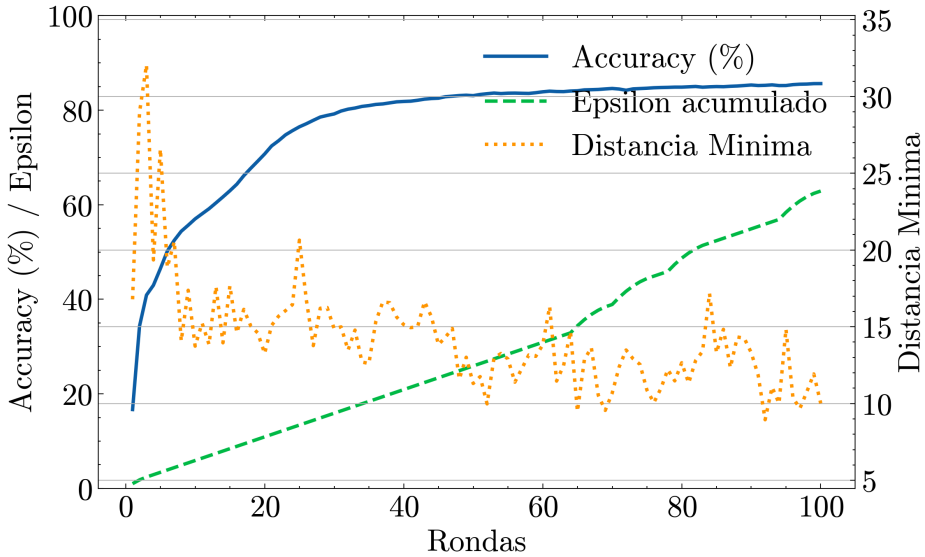


Resultados para el ataque gaussiano con MNIST.

Resultados II

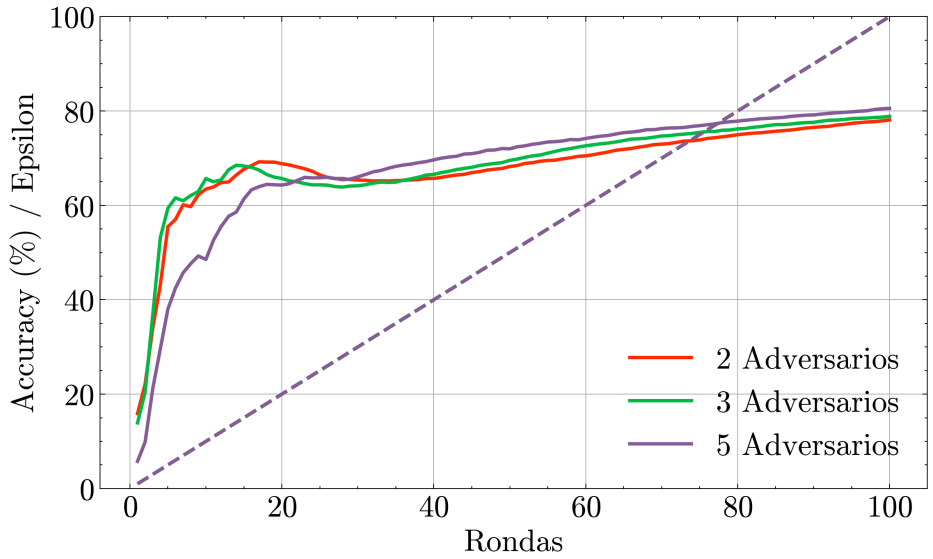


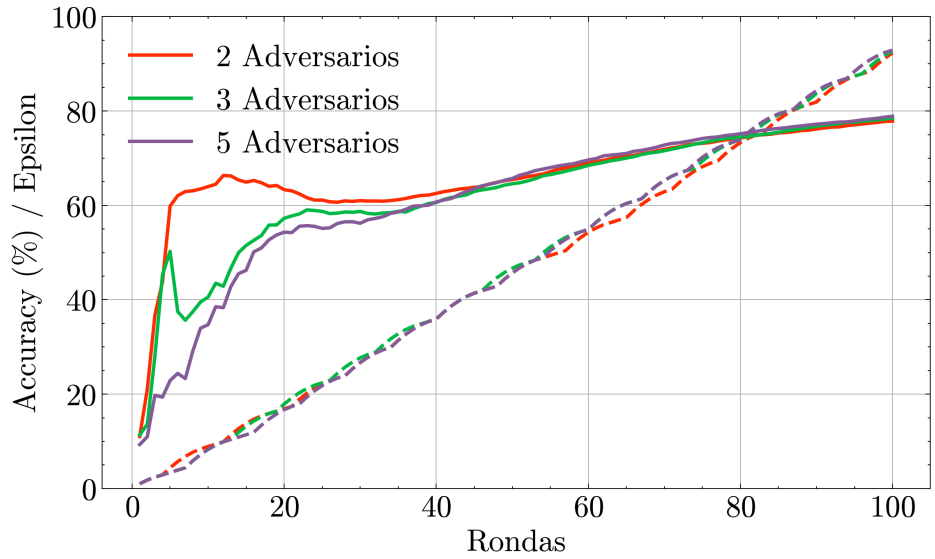
Aplicación estática de privacidad.



Optimización de la privacidad por rondas.

Experimentación adicional





Comparativa según el número de adversarios.

Conlusiones y trabajos futuros

- **Rendimiento:** La optimización dinámica de la PD mantiene el rendimiento del modelo.
- **Optimización:** Se consigue un ahorro del presupuesto de PD.
- **Adaptatividad:** El enfoque dinámico se adapta a las necesidades para cada ronda.
- **Resiliencia ante adversarios**.

Posibles mejoras

Uso de optimización convexa, importación a sistemas reales de FL y exploración de combinación con otras técnicas de defensa.

La investigación abre nuevas vías para mejorar la privacidad en sistemas de FL haciendo uso de algoritmos clásicos de optimización.

Sistemas de optimización para el balance entre resiliencia a ataques a la privacidad y rendimiento en Aprendizaje Federado

Autor: Julio Pérez Cabeza

Tutores: Javier Merí de la Maza, Nuria Rodríguez Barroso

Departamento de Análisis Matemático, Departamento de Ciencias de la Computación e Inteligencia Artificial

20 de julio de 2025

Doble Grado en Ingeniería Informática y Matemáticas, UGR

UNIVERSIDAD DE GRANADA

[1] European Commission. “Regulatory Framework Proposal on Artificial Intelligence”. Accessed: 2025-05-03. 2024. URL: https://digital-strategy.ec.europa.eu/en/policies/regulatory-framework-ai.

[2] Wikipedia contributors. “Algoritmo símplex — Wikipedia, La enciclopedia libre”. 2024. URL: https://es.wikipedia.org/wiki/Algoritmo_s%C3%ADmplex.

[3] M. Victoria Luzón et al. “A Tutorial on Federated Learning from Theory to Practice: Foundations, Software Frameworks, Exemplary Use Cases, and Selected Trends”. En: **IEEE/CAA Journal of Automatica Sinica** (2024).

[4] Miguel A Ramirez et al. “New data poison attacks on machine learning classifiers for mobile exfiltration”. En: **arXiv preprint arXiv:2210.11592** (2022).