**Uppsala U 2023**

# Data privacy: From centralized learning to federated learning

Vicenç Torra

March 2023

Computing Sciences Department, Umeå University, Sweden

# Index

1. Introduction
   - A context: Data-driven ML
   - Privacy for machine learning and statistics
   - Our research
2. Privacy for graphs
   - Problem
   - Graph addition
   - Extension to dynamic graphs
3. Smart grid
4. Federated Learning

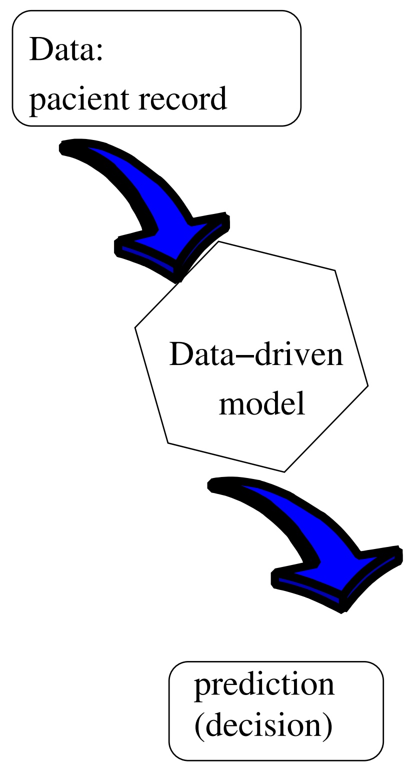
# Introduction

# A context:

**Data-driven machine learning/statistical models**

# Prediction using (machine learning/statistical) models

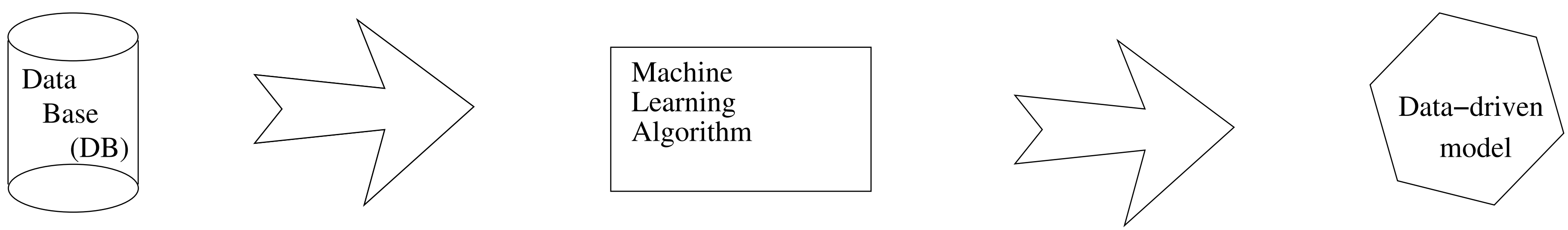
- Application of a model for decision making
  data ⇒ prediction/decision




- Example: predict the length-of-stay at admission

# Data-driven machine learning/statistical models

- From huge databases, build the "decision maker"
  - Use (logistic) regression, deep lerning, neural networks, . . .



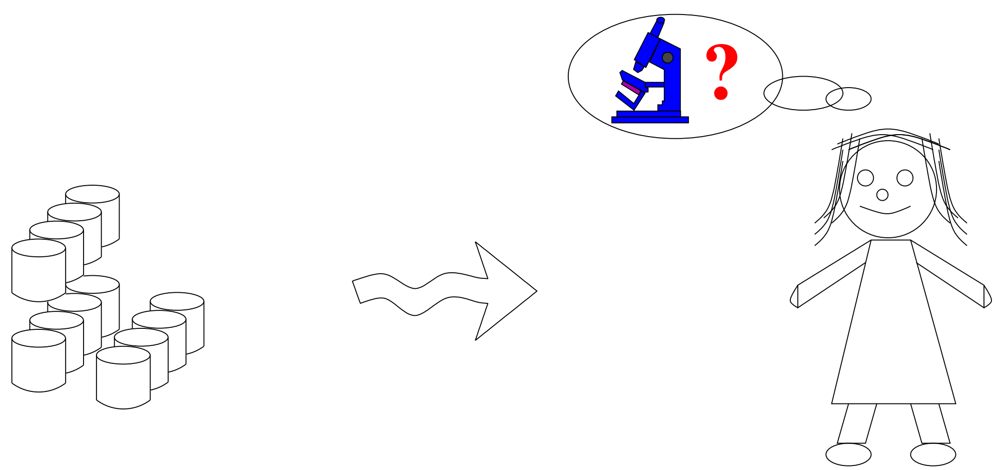

- Example: build a predictor from hospital historical data about length-of-stay at admission

# Privacy for machine learning and statistics:

**Data-driven machine learning/statistical models**

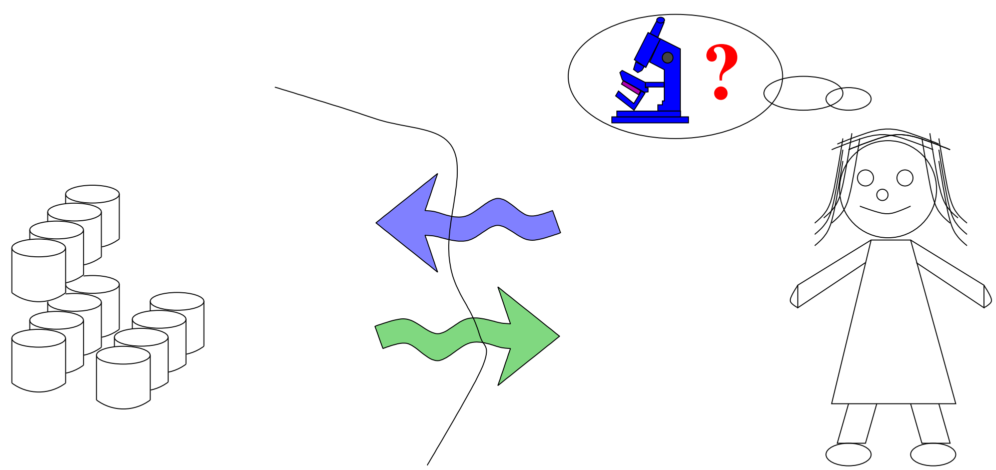
# Data is sensitive

- Who/how is going to create this model (this "decision maker")?
- Case #1. Sharing (part of the data)



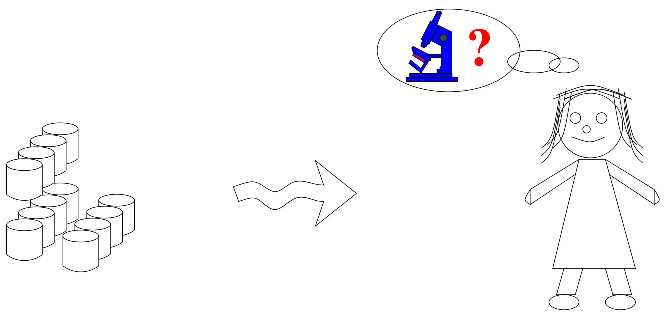
# Data is sensitive

- Who/how is going to create this model (this "decision maker")?
- Case #2. Not sharing data, only querying data



# Data is sensitive

- Case #1. Sharing (part of the data)
- Naive anonymization does not work[1]



- Predict length-of-stay, database with only (year-birth, town, illness/ICD-9 codes)
  1967, Umeå, circulatory system
  1957, Umeå, digestive system
  1964, Umeå, congenital anomalies
  1997, Umeå, injury and poisoning
  1986, Täfteå, injury and poisoning
  ...

  However:
  1984, Holmöns distrikt, xxx

[1] Folkmängd: 63 (https://sv.wikipedia.org/wiki/Holm%C3%B6ns_distrikt)

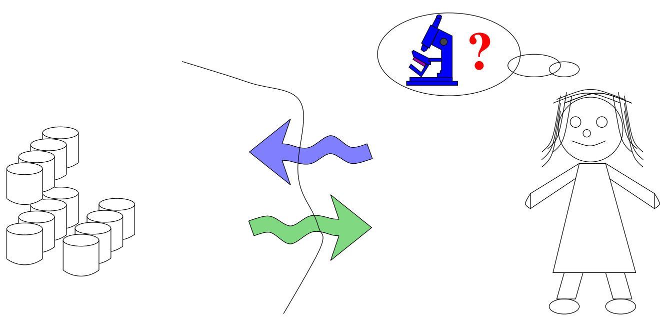
# Data is sensitive: How to make ML possible?

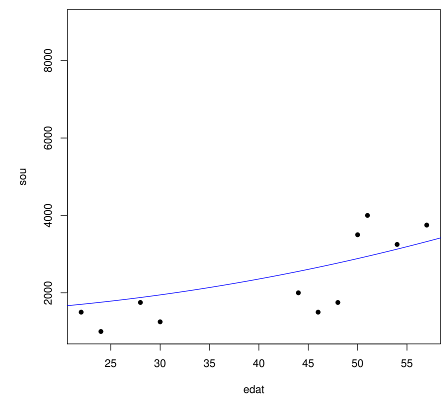
- Case #1. Sharing (part of the data)
- How ML is possible:
  - **Privacy models**. Computational definitions of privacy
  - **Data protection mechanisms**.
- Example:
  - Group a few people with similar characteristics,
  - provide **safe** summaries of these people.
- Example Sävar-Holmöns, combining Sävar, Täfteå and Holmöns (or combine Väddö Björkö Arholma in Norrtälje)

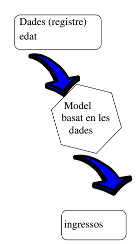
# Model is sensitive

- Case #2. Not sharing data, only querying data, sharing the model

- Models may reveal sensitive information
  - Income prediction vs. age for a town



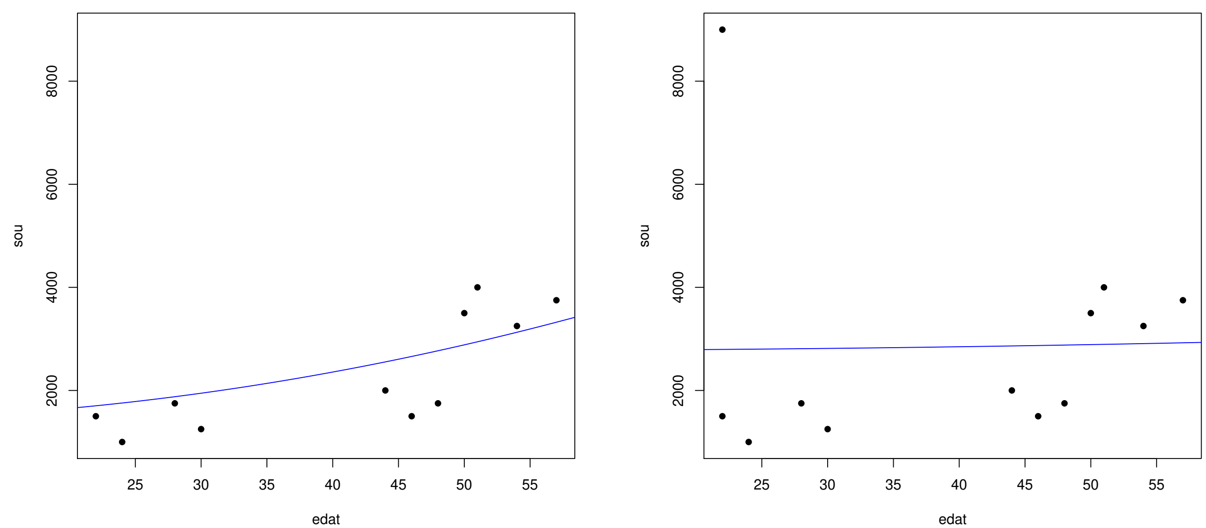



$$\text{income} = 1418.63 + 0.5864 * \text{age}^2$$

# Model is sensitive

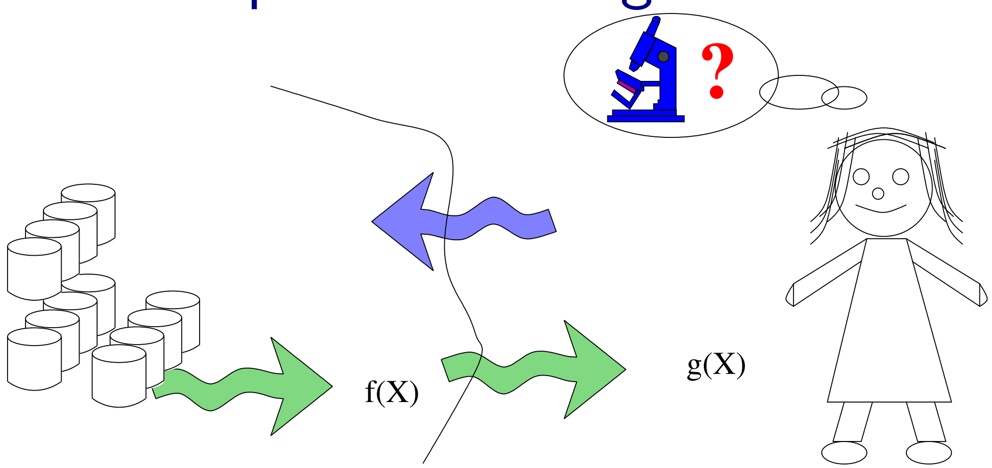
- Case #2. Not sharing data, only querying data, sharing the model
- Models may reveal sensitive information
  Did they use my data (without permission)??
  - Membership inference attacks:
    We add Dona Obdúlia (who is very very rich and young)




income $= 1418.63 + 0.5864$ * age$^2$ vs. income $= 2774 + 0.04639$ * age$^2$

# Model is sensitive: How to make ML possible?

- Case #2. Not sharing data, only querying data, sharing the model
- How ML is possible:
  - **Privacy models**. Computational definitions of privacy
  - **Privacy mechanisms for building models**.
- Example:
  - The model should not depend on a single individual



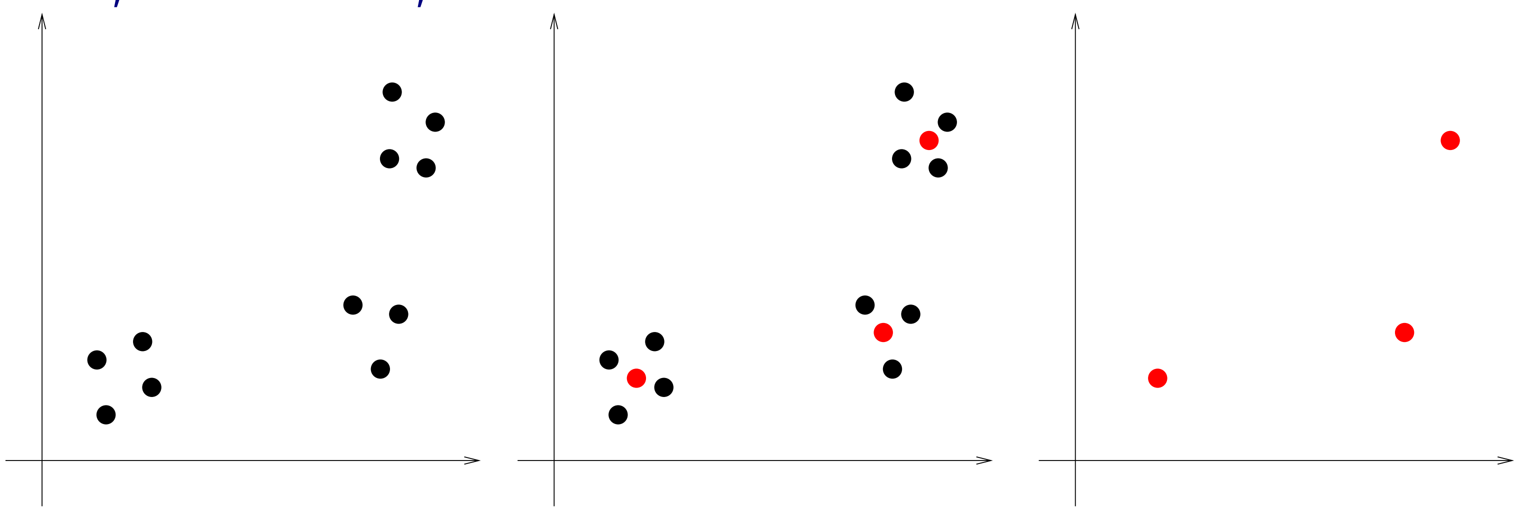
# Privacy models

**Privacy models.** A computational definition for privacy. Examples

- Privacy for data publishing
  - **Reidentification privacy.** Avoid finding a record in a database.
  - **k-Anonymity.** A record indistinguishable with $k-1$ other records.
- Privacy for queries/functions
  - **Differential privacy.** The output of a query to a database should not depend (much) on whether a record is in the database or not.
  - **Integral privacy.** The model should be recurrent. Different ways to reach to the same model.

# Privacy mechanisms: privacy for data

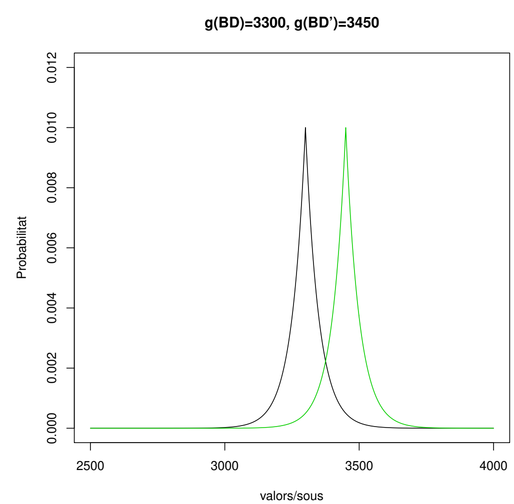
- Privacy for re-identification and/or $k$-anonymity
  - Noise addition: Gaussian (correlated, uncorrelated), Laplacian noise
  - PRAM (Post-randomization method) – Randomized response
  - Microaggregation (grouping)
    - ▷ MDAV, Mondrian, and variations



  - etc.

# Privacy mechanisms: privacy for computations

- Differential privacy
  - Replace query/program $q$ by $K_q(D)$, a randomized version of $q(D)$
    - Given neighbouring databases $D$, $D'$: $K_q(D)$ similar enough to $K_q(D')$
  - $q(X)$ numerical: add Laplacian noise
  - $q(X)$ nominal: apply randomized response (PRAM)
  - Example with $f(DB) = 3300$ and $f(DB') = 3450$, with Laplace distribution $L(0, 50)$



# Our research

# Our research

- Research questions:
  - How to protect data?
  - How to evaluate risk? (for models and data)
  - How to evaluate utility?
- for different types of data sets (centralized databases)
  - standard databases
  - graph and network data
  - electric grid data and time series
- Considering now federated learning

# Privacy for graphs

# Problem

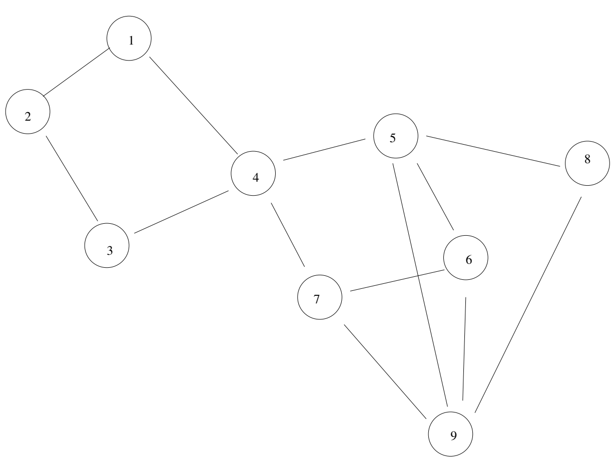
# Graphs

**Graph:** Representation of a large number of problems

**Representation:**

- $G(V, E)$
  with $V$ vertices / nodes
  with $E$ edges $E \subseteq V \times V$

$E$ represented by the adjacency matrix

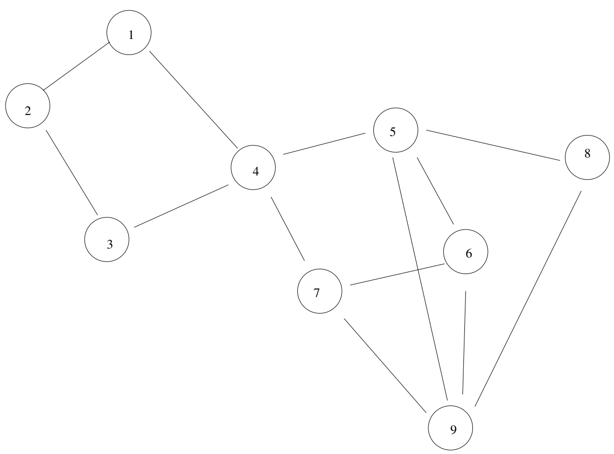


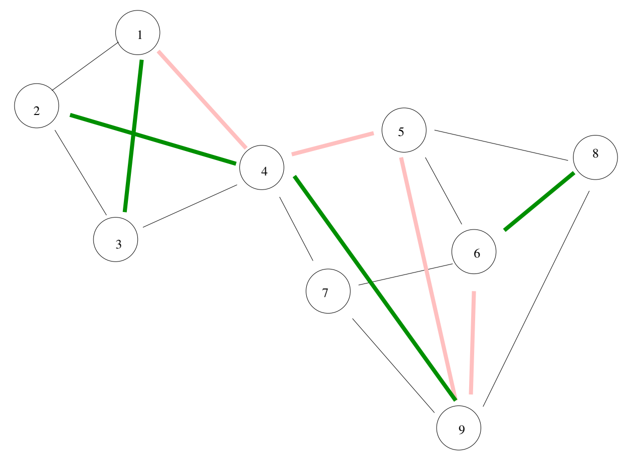

$$\begin{pmatrix} 010100000 \\ 101000000 \\ 010100000 \\ 101010100 \\ 000101011 \\ 000010101 \\ 000101001 \\ 000010001 \\ 000011110 \end{pmatrix}$$

# Problem

## Data protection for graphs:

- Given a graph $G$, produce a protected graph $G'$
- $G'$ ressembles $G$
- and avoids disclosure (e.g., do not find you)



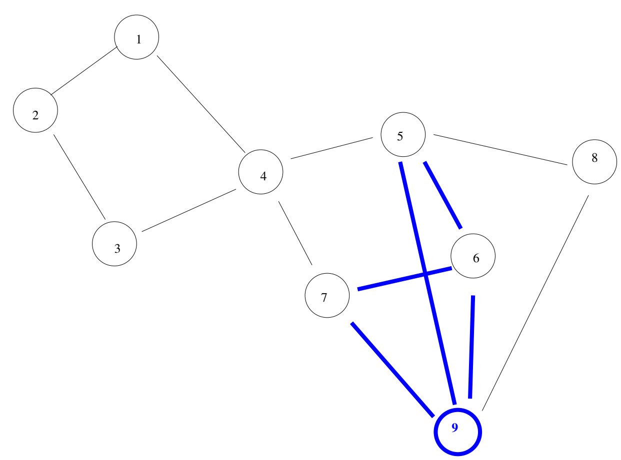
# Problem

**Data protection for graphs:** Avoids disclosure (definition)

- An intruder with some information $I$ on node $v$ of the graph
- is not able to identify the node.

**Example** of information $I$

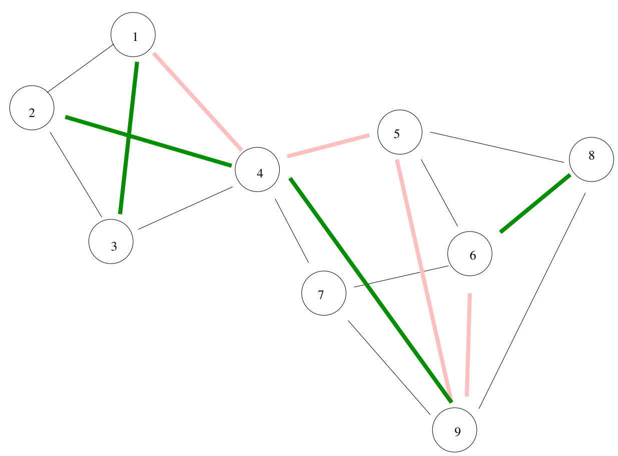
- The degree of a node (i.e., $|N(v)|$)
- The subgraph of neighbours (i.e., $\tilde{G}$ from $v$ and $N(v)$) (subgraph isomorphism problem // subgraph matching)



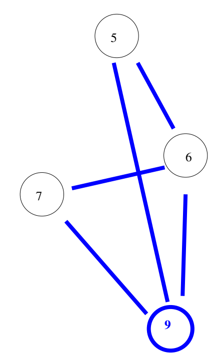
# Problem

**Data protection for graphs:** How to ?

- Adhoc protection: change structure
  - Random addition and deletion of nodes
  - Random addition and deletion of edges
  - Check how much addition / deletion is needed with some attacks




Find this now?



# Graph addition

# Noise addition (for numerical data)

**Our proposal:**

- Inspired in noise addition for numerical data
- Add noise to hide e.g. age and salary

**Noise addition:** Data protection via noise addition

$$X' = X + \epsilon$$

with $\epsilon \sim N(0, kVar)$

- This definition permits to deduce properties for $X'$ (e.g., mean of $X'$ = mean of $X$, variance of $X'$, etc.) Related definitions with correlated noise in multivariate $X$

# Noise addition for graphs

**Noise addition for graphs:** Similar idea but with graphs

$$G' = G \oplus g$$

- $G \oplus g$ for $G = (V, E)$ and $g = (V_g, E_g)$ as follows
  - align nodes of both graphs
  - edges in terms of exclusive-or of edges, or symmetric difference.

$$\begin{aligned} E_1 \Delta E_2 \quad &:= \quad (E_1 \setminus E_2) \cup (E_2 \setminus E_1) \\ & \quad\;\; \{e | e \in E_1 \wedge e \notin E_2\} \cup \{e | e \notin E_1 \wedge e \in E_2\} \end{aligned}$$

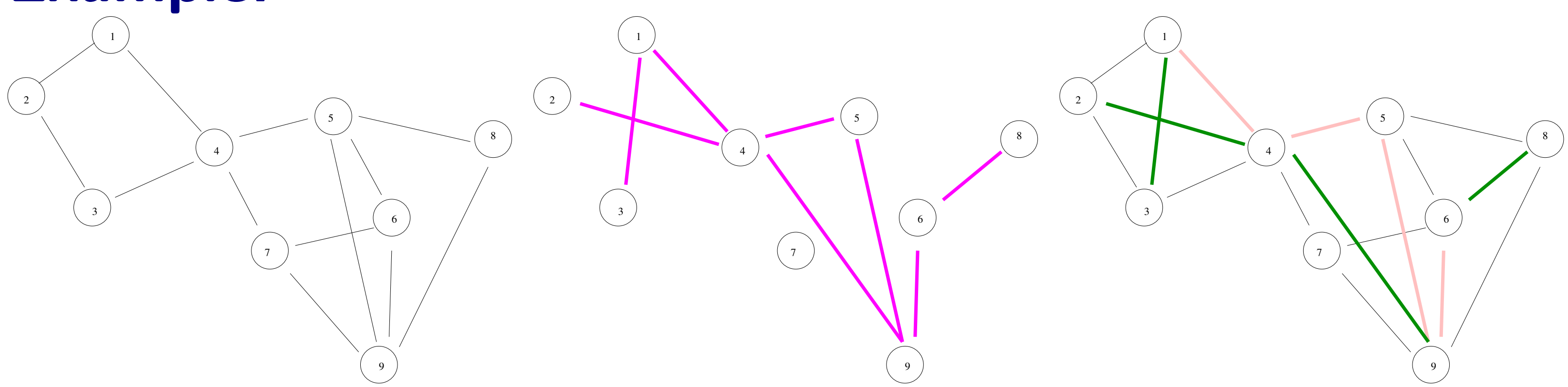
$\rightarrow G' = (V', E')$ with $E' = E \Delta E_g$

# Noise addition for graphs: Example

**Noise addition for graphs:** Similar idea but with graphs

$$G' = G \oplus g$$

- **Example:**



# Noise addition: random graphs

**Noise addition for graphs:** Similar idea but with graphs

$$G' = G \oplus g$$

- $g$ is a random graph[2]

[2] VT, JS, Graph Perturbation as Noise Graph Addition: A New Perspective for Graph Anonymization. Proc. DPM 2019; JS, VT, Differentially Private Graph Publishing and Randomized Response for Collaborative Filtering. Proc. SECRYPT 2020

# Noise addition: Graphs to add

**Graphs.** Examples of random graphs

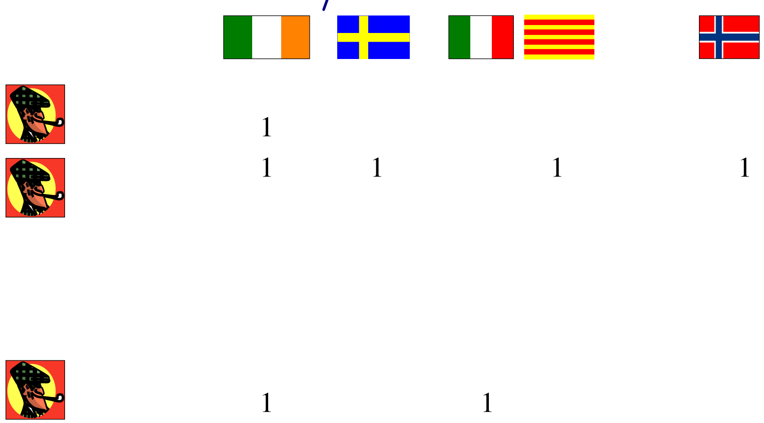
- Gilbert model $\mathcal{G}(n, p)$
  - $n$: number of nodes
  - $p$: each edge is chosen with probability $p$
- That is, $E = \{e_{ij}\}_{ij}$, $e_{ij} \in \{0, 1\}$ and $e_{ij} = 1$ with probability $p$

# Noise addition: Graphs to add

**Graphs.** For bipartite graphs

- Gilbert model $\mathcal{G}(n, m, p)$
  - $n, m$: number of nodes each part $U$, $V$
  - $p$: each edge $(U - V)$ is chosen with probability $p$

Preferences/likes: travellers vs. countries; customers vs. products



# Differential privacy

**Definition.** For $0 < p < 1/2$, we define the noise-graph protection mechanism as:

$$\mathcal{A}_{n,p}(G) = E(G \oplus g)$$

with $g \in \mathcal{G}(n, p)$ (Gilbert model)

**Theorem.** This mechanism provides $ln((1-p)/p)$-differential privacy

- This is for edge-differential privacy: Presence/absence of an edge does not make a difference: hiding individual edges

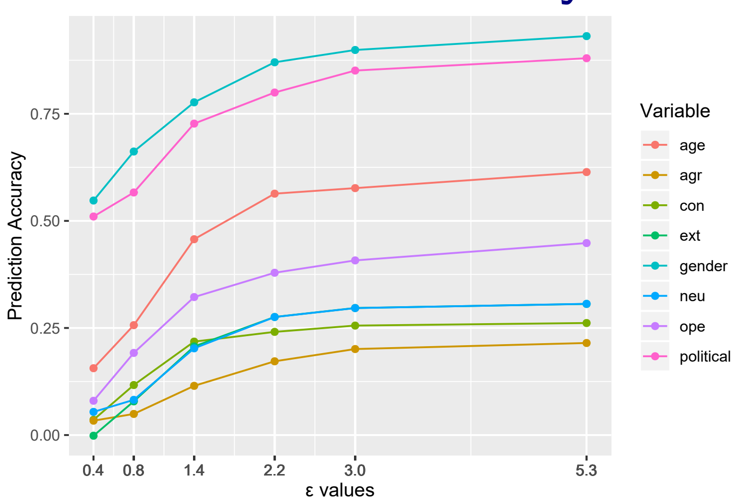
# Differential privacy

**Example.** Facebook likes data (after trimming, min 50 likes, 150 users/like) (19,724 users, 8,523 likes, 3,817,840 user-like pairs)

- Analysis:

| p | $\epsilon$ | $\lvert E(g)\rvert$ | $\lvert E(G \oplus g)\rvert$ |
|---|---|---|---|
| 0.005 | 5.29 | 840,162 | 4,619,770 |
| 0.05 | 2.94 | 8,408,449 | 11,844,981 |
| 0.1 | 2.19 | 16,824,538 | 19,878,770 |
| 0.2 | 1.38 | 33,657,261 | 35,949,261 |
| 0.4 | 0.40 | 67,302,556 | 68,070,070 |

- Prediction accuracy for gender, age, political views, ...



# Analysis of communities

Analysis of communities[3]

- Community detection using singular value decomposition + clustering

Approach:

- Use signless Laplacian matrix
$$|L| = D + A$$
where $D$: diagonal matrix with node degrees, $A$: adjacency matrix
- Matrix factorization of $|L|$ using SVD. Nodes as vectors in terms of orthogonal bases and singular values.
- Reduced dimensional approximation $|L|'$
- Similarity between pairs of vertices using dot products of vectors
- Clustering of vertices
(fuzzy clustering to permit multiple memberships to communities)

[3]VT, Graph addition: properties for its use for graph protection, ILAS 2020 (hold in Galway 2022 :) )

# Analysis of communities

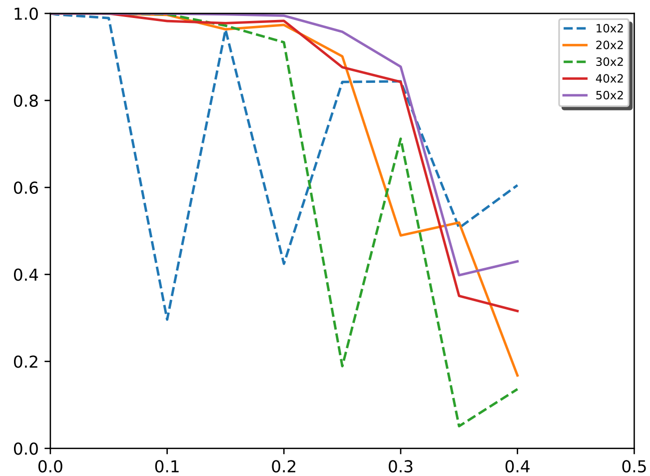
Example.

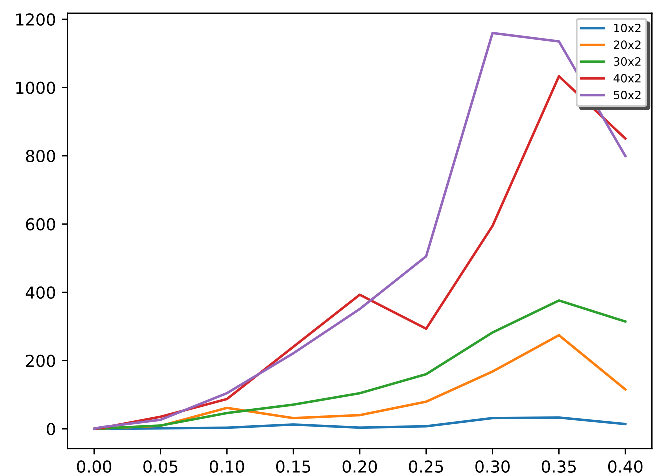
- Two communities. Gilbert model $G \sim \mathcal{G}(n, m, p_n, p_m, p_{nm})$
- Community detection for graph addition

$$G_p = G \oplus g_p$$

with $g_p \sim \mathcal{G}(n+m, p)$ and
$p \in \{0, 0.005, 0.05, 0.1, 0.15, 0.2, 0.25, 0.3, 0.35, 0.4\}$

- Membership correlation between $G$ and $G_p$



# Extension to dynamic graphs

# Analysis of communities

- Graph evolves with time. Snapshots of graphs.
- Edge-local differential privacy for dynamic graphs
  - $\mathcal{A}$ satisfies $\varepsilon$-edge local DP if for all nodes $u, v$, times stamps $t$ and edge values $i, j, k$:

$$Pr[\mathcal{A}(u, v, t; i) = k] \leq e^{\varepsilon} Pr[\mathcal{A}(u, v, t; j) = k], \tag{1}$$

- Parallel protection mechanism: $\mathcal{A}^{||}_{p_0,p_1}(G)$
  - $G = G_0, G_1, \ldots, G_T$ a dynamic graph, $\mathcal{A}_{p_0,p_1}$ a noise-graph mechanism, produce

$$\tilde{G} = \tilde{G}_0, \tilde{G}_1, \ldots, \tilde{G}_T$$

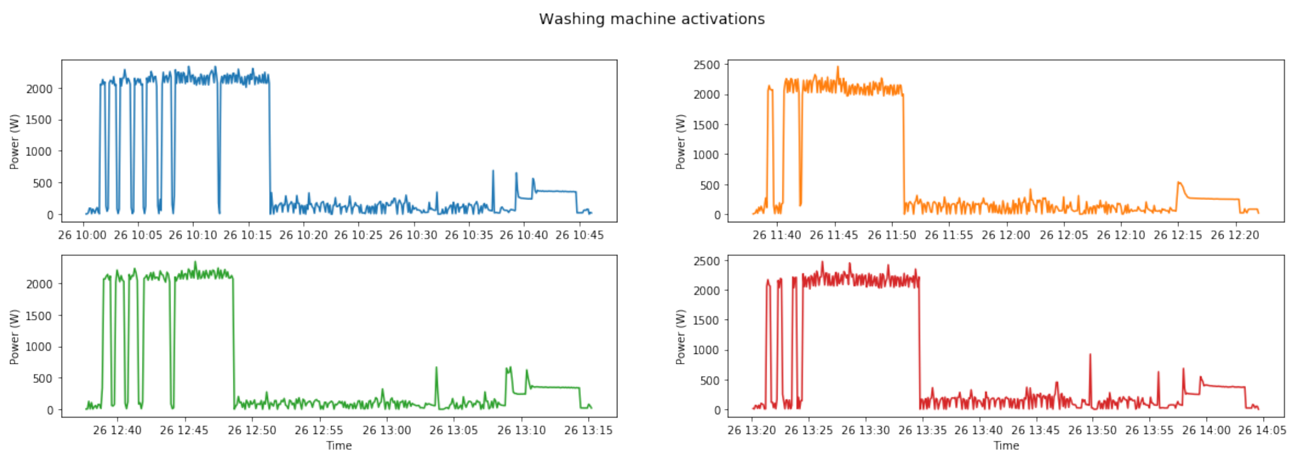
with $\tilde{G}_i = \mathcal{A}_{p_0,p_1}(G_i)$ for $i = 0, \ldots, T$.

# Smart grid

# Temporal data: smart grid

- Smart grid: electric grid data
  - Data from households

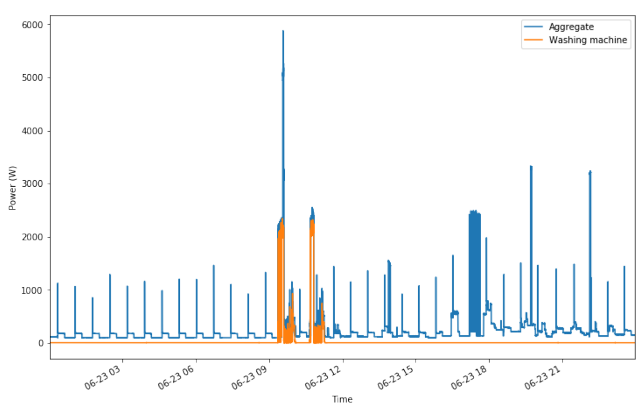
- Sensitive data:
  - consumer habits,
  - Non-intrusive load monitoring (NILM): deduce types of appliances from aggregated energy consumption.

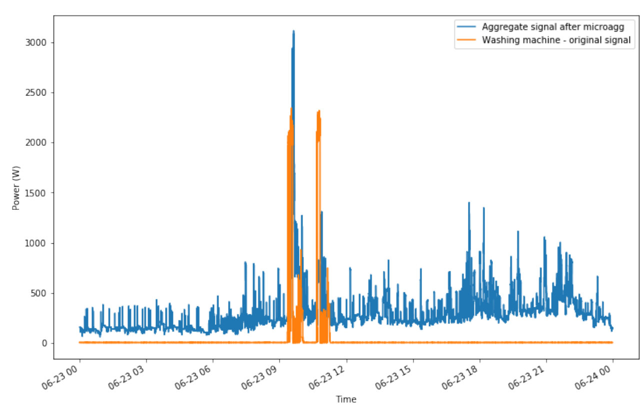


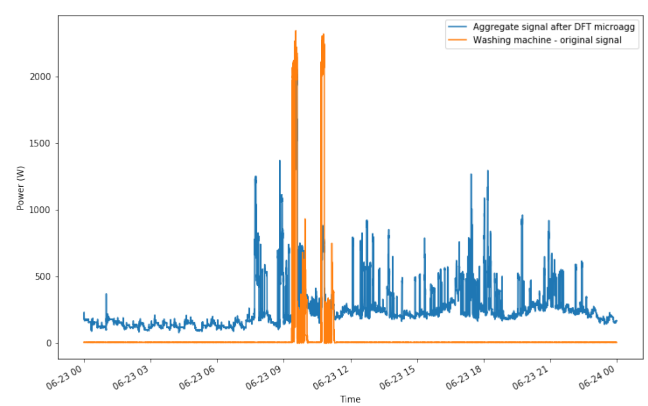
# Temporal data: smart grid

- Our approach:
  - Data is centralized by the service provider
  - Data needs to be shared without disclosure

- Protection through microaggregation and DFT



# Temporal data: smart grid

- Data utility based on data mining tasks[4]:
  - clustering: k-means
  - classification (type of consumer): kNN
  - forecasting: mean hourly load forecasting using SARIMAX model (seasonal ARIMA)

- Adversarial model:
  - Re-identification (based on record linkage)
  - Interval disclosure (is the masked value too similar?)
  - Non-intrusive load monitoring (NILM) detection.

[4] K. Adewole, V. Torra, DGTMicroagg: a dual-level anonymization algorithm for smart grid data, Int. J. of Inf. Systems 2022; K. Adewole, V. Torra, On the application of microaggregation and discrete Fourier transform for energy disaggregation risk reduction, submitted.

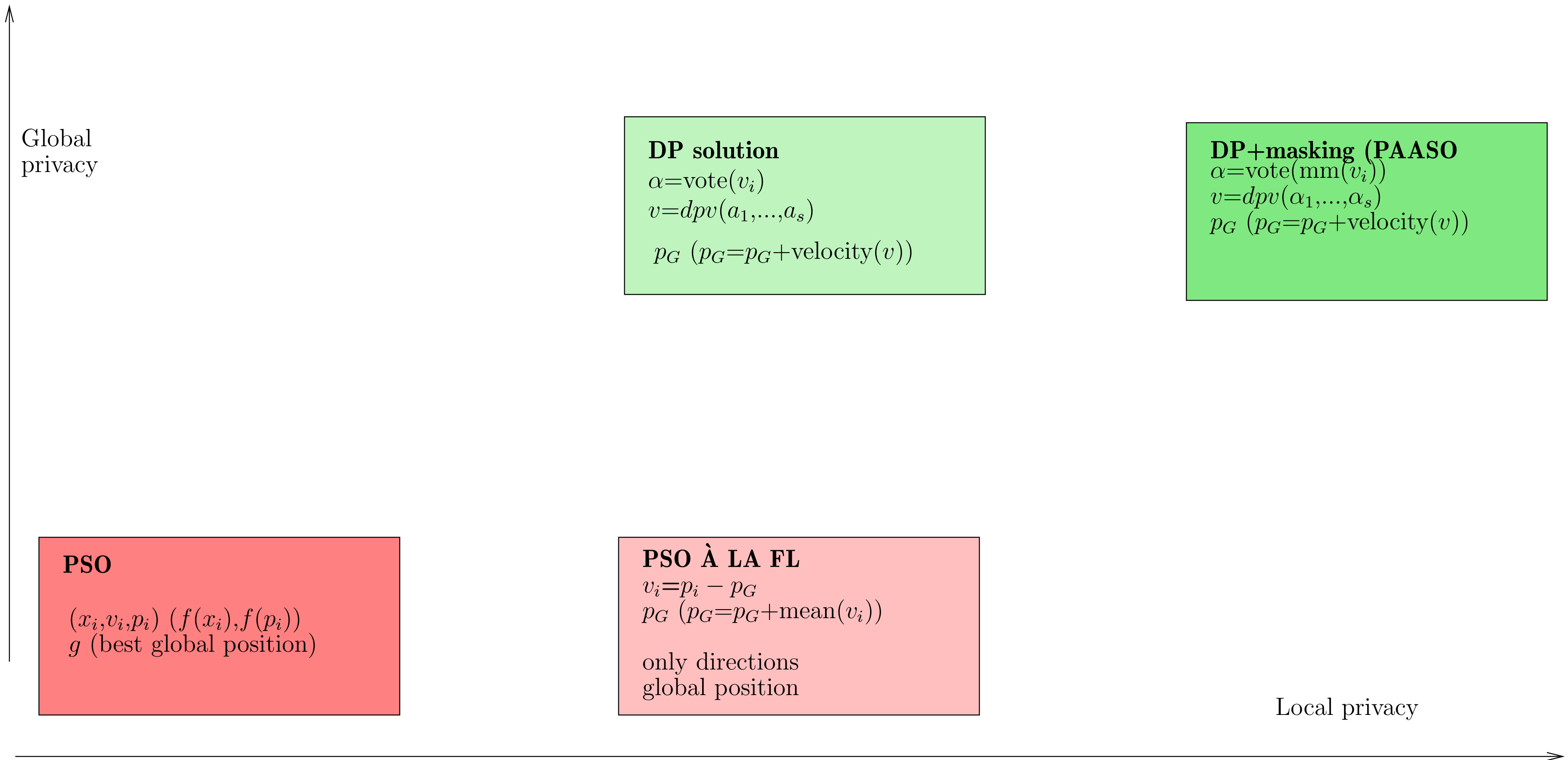
# Federated Learning

# Federated learning

- FL models
  - initial research on trying to reduce membership inference, model reconstruction and backdoor attacks.
- Symbolic models (decision trees, gradient boosting decision trees)
  - Local vs. global privacy: k-anonymity vs differential privacy.
  - Some work uses LSH to find similar instances from different devices. Data reconstruction attacks.

# Federated learning

- PSO + FL = PAASO: Privacy-aware agent swarm optimization



# Federated learning

- General comments PAASO[5]
  - In general, privacy mechanisms do not avoid convergence.
    It is slower. (this can be a concern, of course, rounds=information)
  - In terms of convergence, PSO and FL are best.
  - Local protection (PRAM) does not have strong effect.

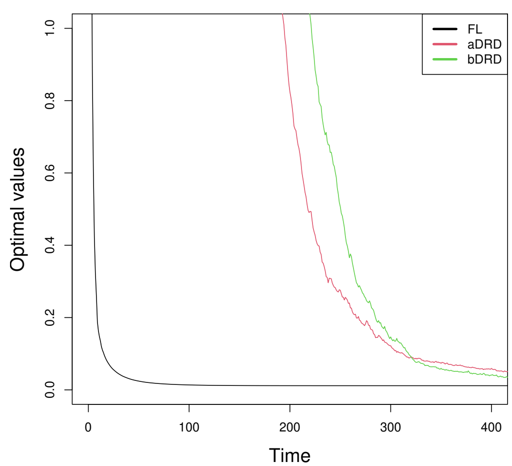
- On the parameters
  - Number of options in voting, low effect
  - Number of agents, key factor
  - Particular parameters depend on the problem + privacy strategy

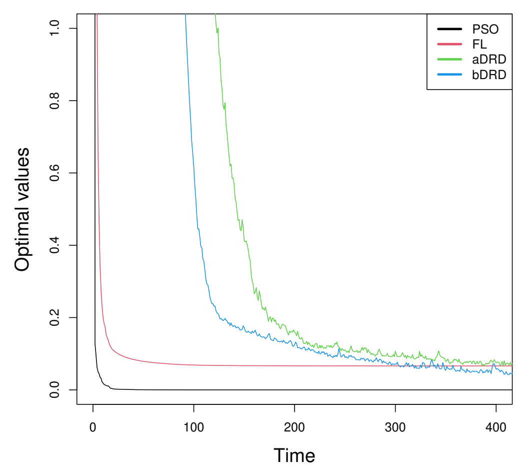
[5] VT, EG, GN, PSO + FL = PAASO: particle swarm optimization + federated learning = privacy-aware agent swarm optimization. Int. J. Inf. Sec. (2022)

# Federated learning

- An example:
  - Mean objective function for 20 executions for FL, aDRD, and bDRD. Function $f_4$, number of voting alternatives $k_\alpha = 8$, 50 agents, $\phi_p = \phi_g = 2.00$. $p_c = 1.0$.
  - (left) $\omega = 4.00$, $\omega_G = 0.005$; (right) $\omega = 0.005$, $\omega_G = 0.01$
  - Generalized Rosenbrock's function $(x_1, x_2 \in [-2.0, 2.0])$:

$$f_4(x_1, x_2) = 100 * (x_2 - x_1 * x_1)^2 + (x_1 - 1)^2$$



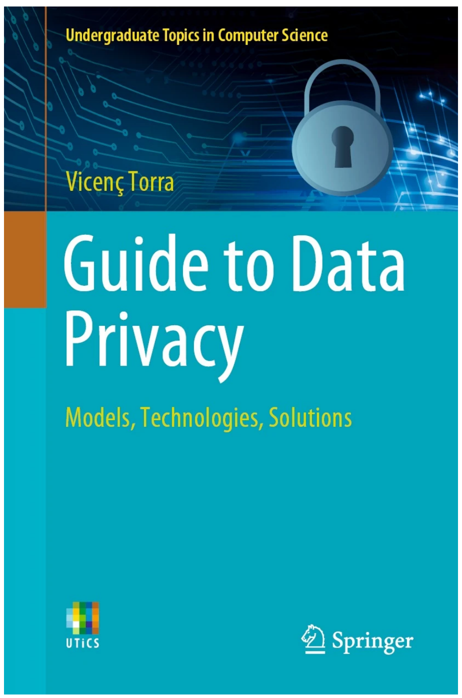
# Summary

# Summary

- Graphs
- Smart grid
- Federated learning

# Thank you

# Ads


Undergraduate Topics in Computer Science
Vicenç Torra
Guide to Data Privacy
Models, Technologies, Solutions
UTICS
Springer