Experiments on Federated Data Synthesis

CLAIRE LITTLE, MARK ELLIOT, RICHARD ALLMENDINGER
UNIVERSITY OF MANCHESTER
Questions?

https://tinyurl.com/QuestionsUoM
Federated Learning (FL)

FL (McMahan et al., 2017) is a decentralized approach to training statistical models

- Multiple clients can produce one global model
- Clients do not share or exchange their own data
- Can reduce privacy and security risks (compared to methods that combine multiple data sources)
- Allows models to train on data that is more representative of the whole distribution
- Useful where clients do not possess enough data to generate the required statistical power
Federated Learning (FL)

Central server controls the process (but does not access any client data)
- Initialise model, sends to each client
  - Typically, neural network type models are used

Each client trains the model on their own data
- Send updates (parameters or model weights) back to server

Server aggregates the client updates
- Sends updated model back to clients

Iterative process
- Training usually terminated when specific criterion is met:
  - E.g., maximum number of iterations
Federated Synthesis

Using FL to generate synthetic data
  • Emerging research field
  • Small body of research focusing mostly on image data
  • Less research on tabular data
  • Methods predominantly use GANs (Generative Adversarial Networks, Goodfellow et al. 2014)

Is it possible to produce useful synthetic microdata in a federated way?
  • Proof of concept using Genetic Algorithm (GA)
Genetic Algorithms (GAs)

GAs (Holland, 1992) perform iterative optimisation, training over multiple generations
- Three main biologically inspired operators:
  - Selection, Crossover, Mutation

- Initial population of candidate solutions (candidate solution = synthetic dataset)
- Fitness (similarity to original data) of each candidate calculated
- Select fitter candidates (parents) to reproduce for new population
- Crossover – combines parents to produce new candidates (children)
- Mutation – randomly change some of the candidates features
- Next generation – children, or combination of best (fittest) parents and children (elitism)
- Repeat process multiple times (generations) using fitness to guide
Study Design - Data

A (very) simple binary dataset, randomly sampled from UK 1991 Census microdata (University of Manchester, 2023)

- Small dataset to enable understanding
- 10 rows, 5 binary variables
  - “Original” dataset
- Randomly split into two five-row datasets
  - representing two clients (A and B)

<table>
<thead>
<tr>
<th>AGE</th>
<th>MSTATUS</th>
<th>SEX</th>
<th>LTILL</th>
<th>TENURE</th>
<th>client</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>A</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>A</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>A</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>A</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>A</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>B</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>B</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>B</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>B</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>B</td>
</tr>
</tbody>
</table>
Study Design - Parameters

Huge potential range of variation in the simulation

Three types of parameters:

- **Model**: changeable settings for the GA (e.g., mutation rate)
- **Simulation**: variations in the scenario being presented (e.g., number of clients)
- **Experimental**: elements that are not part of the simulation itself (e.g., data choice, number of runs)

Model complexity is kept low to aid with interpreting the results

- Focus only on utility (not risk)
- Small dataset
- GA uses mutation but not crossover
- Two clients for FL
### Study Design - Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Type</th>
<th>Value chosen</th>
<th>Further details</th>
</tr>
</thead>
<tbody>
<tr>
<td>No. of clients</td>
<td>Simulation</td>
<td>VARIES</td>
<td>-</td>
</tr>
<tr>
<td>Initial Metadata sent by clients</td>
<td>Simulation</td>
<td>Univariate</td>
<td>-</td>
</tr>
<tr>
<td>Combination of client scores</td>
<td>Simulation</td>
<td>VARIES</td>
<td>-</td>
</tr>
<tr>
<td>No. of objectives for GA</td>
<td>Simulation</td>
<td>1</td>
<td>Similarity (utility)</td>
</tr>
<tr>
<td>SDC applied to the output sent to server</td>
<td>Simulation</td>
<td>None</td>
<td>-</td>
</tr>
<tr>
<td>Output passed to client by server</td>
<td>Simulation</td>
<td>VARIES</td>
<td>-</td>
</tr>
<tr>
<td>Population size</td>
<td>Model</td>
<td>50</td>
<td>-</td>
</tr>
<tr>
<td>Parental selection</td>
<td>Model</td>
<td>Binary Tournament</td>
<td>k=2</td>
</tr>
<tr>
<td>Mutation rate</td>
<td>Model</td>
<td>0.05</td>
<td>-</td>
</tr>
<tr>
<td>Crossover Operator</td>
<td>Model</td>
<td>None</td>
<td>-</td>
</tr>
<tr>
<td>Environmental selection</td>
<td>Model</td>
<td>Elitism</td>
<td>-</td>
</tr>
<tr>
<td>No. of generations</td>
<td>Experiment</td>
<td>150</td>
<td>-</td>
</tr>
<tr>
<td>Choice of Dataset</td>
<td>Experiment</td>
<td>UK Census microdata</td>
<td>1991</td>
</tr>
<tr>
<td>No. of rows (per client)</td>
<td>Experiment</td>
<td>VARIES</td>
<td>-</td>
</tr>
<tr>
<td>No. of variables</td>
<td>Experiment</td>
<td>5</td>
<td>-</td>
</tr>
<tr>
<td>Type of variables</td>
<td>Experiment</td>
<td>Binary</td>
<td>-</td>
</tr>
<tr>
<td>No. of runs</td>
<td>Experiment</td>
<td>5</td>
<td>-</td>
</tr>
</tbody>
</table>
Initialisation

1. Clients send metadata to the server.
2. The server creates an initial population and sends it to the clients.
3. Clients score the candidates against their own data and send scores back to the server.

Main repeating process

- **Parental selection**: Collate clients' fitness scores, use collated scores to select parents.
- **Mutation**: Mutate children.
- Send children to clients to get fitness scores.
- **Environmental selection**: Choose the next generation.

Final output: Optimised synthetic datasets.
Results – Experiment 1

Running GA on original dataset (10 rows)
• All five randomly initialised runs converged
• i.e., they reproduced the original dataset

![Graph showing mean similarity over generations for Experiment 1]
Results – Experiment 2

Running GA separately on client A and B datasets (5 rows each)

- For each, all five randomly initialised runs converged and reproduced the original dataset
Results
Experiment 3

FL with two clients (A and B)

• All but one of the randomly initialised runs converged and reproduced the original datasets
• Panel 4 would not be available in reality – used for evaluation
• Convergence achieved despite the evaluations from clients, and the server aggregated score indicating suboptimality
Discussion

Experiment 3 demonstrates proof of concept
• Analytically useful datasets were synthesised across distributed datasets

It was not clear on the server that the original data had been reproduced
• Might be useful in terms of disclosure risk
• Means we cannot rely on server-side restraint to minimise risk
Caveats and future work

Experiments conducted on small sample of binary Census microdata
• May not scale to larger, more complex data
• Very large datasets may be computationally impractical

Would need to consider different parameters
• More than 2 clients

Single-objective focus on utility
• In a real-life scenario, the goal would not be to reproduce the original data
• Risk would need to be factored in
  ◦ A multi-objective approach within the GA could be used
  ◦ Deep learning methods also a possibility
Questions?

https://tinyurl.com/QuestionsUoM

Email: claire.little@manchester.ac.uk
References


