Private AI via Federated Learning

GHOST PII

Capnion, Inc. April 2022

Who is the speaker?

30 second resume:

- an ancient metro Saint Louis townie
- B.A. Washington University in Saint Louis 2007
- Ph.D. University of Michigan 2013 (math)
- a few years of data science
- founded data privacy tech company Capnion, Inc.

Slides URL:

`https://github.com/capnion/random/blob/master/acm_capnion_fedlearn.pdf`

Agenda

overview

- informal definition: code to data, not data to code
- stylized example and setting
- important subcategories

what federated learning isn’t

- no new models, just different information flows
- overlap with other privacy-enhancing tech

practical matters

- software tools
- performance
- business value - compliance, cybersecurity, bizdev

Goals

Main target question:

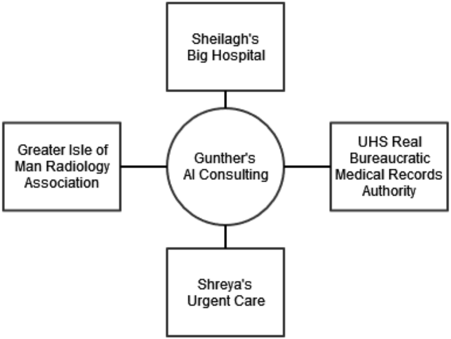
What is federated learning and why might it be useful to me?

Advanced target question:

How does federated learning work? How do I start using it?

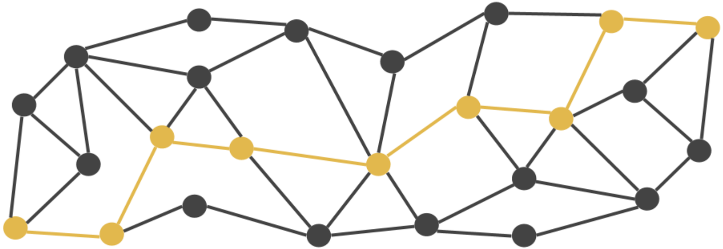
The big idea is “**code to data, not data to code**”

Federated Learning Example Setting



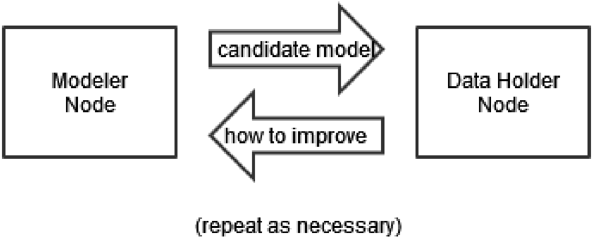
Gunther needs data from a number of different, possibly less-than-trusting, sources to be successful. Is there a way to build the model without being given the data? Yes!

Some Common Language and Generalities



Literature about federated learning tends to borrow from graph theory. A “node” is a silo with sensitive data its owner doesn’t want to share, but by transmitting privatized data (more coming on just how) along the edges in a special way, someone elsewhere at the graph can still build a model using data from all the nodes. How to privatize is the “code” sent to the “data”...

A Common Type of Approach



Modeler Node

candidate model

how to improve

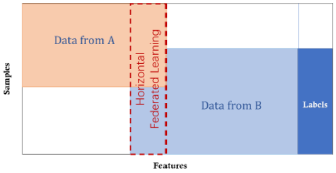
Data Holder Node

(repeat as necessary)

The modeler has a candidate model they share with the data holder. The data holder provides feedback, based on their data, to improve the model. This is done in a careful, previously agreed way (“code to data”) involving various privacy-enhancing technologies to minimize information leakage.

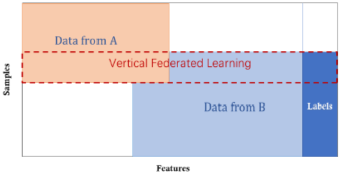
Some enlightening subcategories

Horizontal FML



- Large overlap of features of the two data sets

Vertical FML



- Large overlap of sample IDs (users) of the two data sets

How is the data split up across nodes?

- horizontal - same schema, different observations (homo)
- vertical - different attributes, same observations (hetero)

Some techniques presume a central modeler (Gunther) while others presume more decentralized peer-to-peer networks.

Overlap with other privacy technologies

Federated learning is not a type of model, it is a collection of techniques for training familiar models in a decentralized way...

The information moving node-to-node might be protected by

- homomorphic encryption
- differential privacy

and underneath it is likely a less-sensitive bit of information, like a gradient.

Linear and logistic regression models can of course be trained in a federated way...

Regression as Optimization

You can get your regression coefficients directly, but they are really someone else's solution to an optimization problem where you want to find an

$$f(\overline{x}) = \theta_0 + \sum_{i=1}^{p} \theta_i x_i$$

that minimizes a loss function

$$\text{MSE} = \frac{1}{n} \sum_{j=1}^{n} (f(\overline{x}_j) - y_j)^2$$

You do this by looking at the derivatives (a *gradient* all taken together) of this loss function relative to the coefficients θ_i.

Regression, Linear and Logistic

You can train your favorite regression model via gradient descent with update rules that look like

$$\theta_i \to \theta_i - \alpha \frac{\partial}{\partial \theta_i} J(\bar{\theta})$$

where α is a learning rate, J a “nice” loss function, and the θ_i are the coefficients you want to compute. It is fairly straightforward to compute this update rule in pieces at each node and add them together to get the new coefficient.

As we have discussed, there is a deep toolbox of tools for protecting the pieces computed at each node as they travel around the network.

Some Notable Software Tools

Notable software tools (mostly Python ecosystem)...

- Ghost PII - good for manipulations, classical stat, and trees
- NVFlare - private neural nets on top of Torch
- OpenMined - an agglomeration of open source projects
- FATE AI - many implementations from in academic papers

Some tools are more about giving you an interface used directly for modeling (as in something like scikit-learn) while others provide a toolbox (things like homomorphic encryption and differential privacy) to streamline your work on your own implementations.

Performance Concerns

Federated learning is inevitably slower than putting all the data in one place and using scikit-learn. There are many different techniques, so we'll talk about why rather than how much.

- Network latency
- Every privacy trick (HE, DPriv) used costs time
- Possibly different computation route (SGD vs. direct computation)

Even for linear regression alone, there are many different approaches to federated learning that vary in level of privacy, network structure, etc.

Business value - Why use this technique?

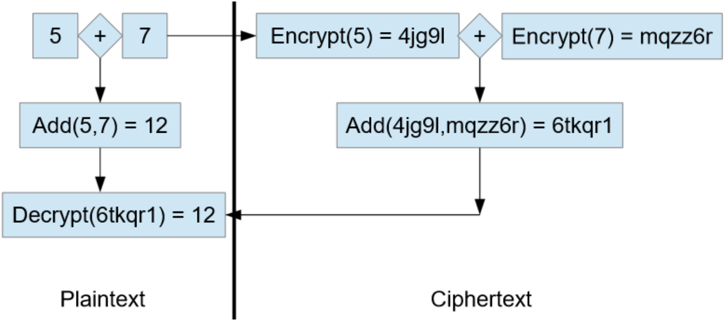
Using federated learning to build your model might please be good for business on a number of fronts...

Cybersecurity: Obvious, an attacker must be much more informed and sophisticated to access plaintext data.

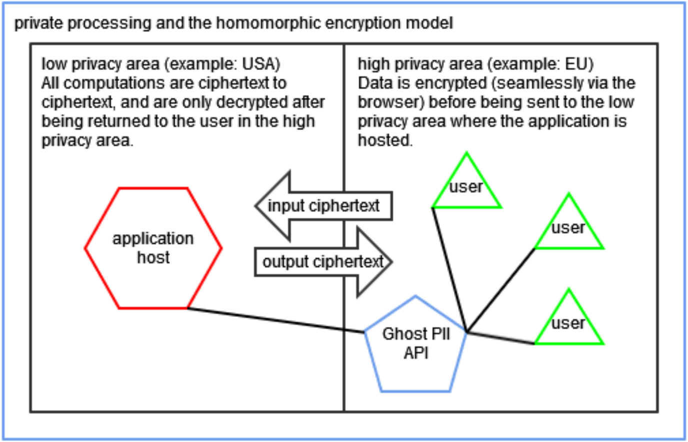
Compliance: Build the model in a dystopian spyzone like the United States without doing wrong by European privacy rules.

Business development: This is derivative of the two above but important. Your deal might be in trouble because of concerns around the security and privacy of data. It might be back on the tracks if you can offer ideas about how to do more with less risk.

What is homomorphic encryption?



HE and Privacy Compliance



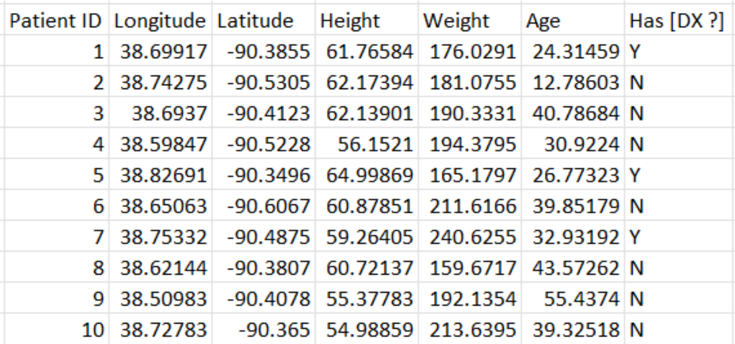
private processing and the homomorphic encryption model

low privacy area (example: USA)
All computations are ciphertext to ciphertext, and are only decrypted after being returned to the user in the high privacy area.

high privacy area (example: EU)
Data is encrypted (seamlessly via the browser) before being sent to the low privacy area where the application is hosted.

HE and Privacy Compliance

Differential privacy is a mathematical theory of how much noise to add to ensure everyone can blend in.



Patient ID	Longitude	Latitude	Height	Weight	Age	Has [DX ?]
1	38.69917	-90.3855	61.76584	176.0291	24.31459	Y
2	38.74275	-90.5305	62.17394	181.0755	12.78603	N
3	38.6937	-90.4123	62.13901	190.3331	40.78684	N
4	38.59847	-90.5228	56.1521	194.3795	30.9224	N
5	38.82691	-90.3496	64.99869	165.1797	26.77323	Y
6	38.65063	-90.6067	60.87851	211.6166	39.85179	N
7	38.75332	-90.4875	59.26405	240.6255	32.93192	Y
8	38.62144	-90.3807	60.72137	159.6717	43.57262	N
9	38.50983	-90.4078	55.37783	192.1354	55.4374	N
10	38.72783	-90.365	54.98859	213.6395	39.32518	N

Which patient is Alex Mueller?

Questions and Conversation

Any questions?

acmueller@capnion.com

https://twitter.com/capnion

https://www.linkedin.com/in/alexander-c-mueller-phd-0272a6108/