Machine Learning and Optimization Laboratory

Incentivized, Privacy-preserving, and Personalized Distributed Machine Learning Framework for Medical Data

Semester project

by Felix Grimberg

Mary-Anne Hartley
Supervisor

Martin Jaggi
Co-supervisor

Andres Colubri
Co-supervisor

18/05/2020

Abstract

Contents

1 Aim

This project is part of a larger endeavour to create a platform/framework for medical professionals around the globe, and espeicially in low-resource settings, to train machine learning models collaboratively. Main requirements to achieve this goal are identified and separated into necessary features and performance features.

1.1 Necessary features

Data privacy. First and foremost, the proposed framework must safeguard each participant from the leakage and re-identification of individual patients stored within their data set. This implies maintaining data at its original location and limiting transfers, which motivates the use of distributed machine learning techniques.

Incentives and intellectual property. Secondly, it must present incentives for the collection of high-quality health data and protect the intellectual property of their owner.

Resilience. Moreover, it must be designed to work with incomplete and non-identically distributed data, as well as with arbitrary connection/disconnection patterns from participants.

Personalisation. Finally, it is important that the selection of training data, features, and model architecture, is interpretable to the users (i.e., medical professionals) and subjected to their judgement. To this end, users must be given the tools to ensure that the trained machine learning model is adapted to the particularities of their local population.

1.2 Performance features

Data analysis. For better model personalisation, the proposed framework should facilitate the selection of appropriate training data by providing and visualizing measures of similarity between data sets.

Feedback. Additionally, it should give its users feedback on the quality of their data collection. For instance, it could point out physiological or epidemiological anomalies (e.g., breathing rates being consistently reported higher than usual). Likewise, it could suggest which additional features should be recorded to efficiently maximize future model performance and thus increase their collaborative scope.

Environmental impact. The framework should also be optimised to quantify, minimise and possibly compensate for the environmental impact associated with training each machine learning model.

Communication efficiency. The communication effort exerted by each participating device should be kept as low as possible, to facilitate participation in low-infrastructure settings. This simultaneously serves to reduce the environmental impact of each training run.

Robustness. Finally, the design should be robust to a small fraction of non-collaborative participants. Different robustness models exist for various types of non-collaboration, ranging from the inadvertent provision and use of false data to fully adversarial behaviour.

2 Background

Available tools for reproducible data science

1. Overall problem

2. What people have done to solve this problem → literature

- Record software versions, hardware and dates and times (e.g., using watermark[1]).

- Better yet, use virtual environments / containers. Google colab and Renku are convenient services that provide their own standardized virtual environment. Modifications to that environment can be recorded in the same notebook that also contains the code.

- Use version control! I.e., git.

- Maybe use a standard project structure (files/folders)

- Sumatra (probably not worth trying out in the scope of my semester project): "for each experiment that you conduct through Sumatra, this software will act like a 'save game state' often found in videogames."[2]

Available tools for (privacy-preserving) distributed machine learning

1. Overall problem

2. What people have done to solve this problem → literature

- In terms of methods(references in ERC grant application):
  - Local SGD
  - Secure multi-party computation (could be incorporated in all of my proposed methods, I believe)
  - Differential Privacy (Don't have time to learn how it works)
  - Homomorphic Encryption: Not in our focus.

- The open-source Python library PySyft[3] "using Federated Learning, Differential Privacy, and Encrypted Computation (like Multi-Party Computation (MPC) and Homomorphic Encryption (HE)) within the main Deep Learning frameworks like PyTorch and TensorFlow."

[1] https://pypi.org/project/watermark/

[2] https://datascience.stackexchange.com/questions/758/tools-and-protocol-for-reproducible-data-science-using-python

[3] https://github.com/OpenMined/PySyft

- `https://github.com/epfml/collaborative-learning-benchmark` Doesn't do privacy-preserving stuff, but they simulate various types of non-iid-ness for CV and NLP tasks.

Available tools for (privacy-preserving, distributed) data analysis

1. Overall problem

2. What people have done to solve this problem → literature

- [4]: SQL based data analysis platform that integrates differential privacy. Each user can make queries only within their allotted ϵ-privacy budget. Users need not have knowledge of any privacy-related topics. Similar work: [1]

- [5]: Presents methods and metrics for privacy-preserving ML (supervised and unsupervised) and KDD (knowledge discovery from data) at different stages of the data lifecycle: data collection, data publication and data output (i.e. after a model has been trained). Includes a discussion of SMC and HE as methods of distributed privacy.

Existing methods for model personalisation / training data selection (TDS)

and varying degrees of inequality between data sets.What people have done to solve this problem → literature

1• [**2019training˙data˙selection**]: A selection distribution generator (SDG) and a predictor are optimized via reinforcement learning (RL) by taking each other's outputs as input.

  - Self-optimizing system for training data selection → no manual thresholding needed, but also no immediately obvious way to involve users.
  - Proven method (achieved competitive performance in their paper).
  - Effective TDS can reduce the environmental impact of model training.
  - Designed for a neural network consisting of a feature extractor followed by a classifier (Where the SDG is also a neural net). The selected training data and the guidance set are transformed into a feature representation by the feature extractor. The SDG is rewarded according to the improvement of one of the following distribution discrepancy measures (refer to their cited sources):
    - Jensen-Shannon divergence (= arithmetic mean of the Kulback-Leibler divergence of each distribution with the arithmetic mean of the distributions)
    - Maximum mean discrepancy
    - Symmetric Rényi divergence
    - Guidance loss…I don't exactly understand what it punishes/rewards.
  - Not immediately obvious how to use their contribution in a way that makes individual decisions explainable (model predictions and selection of specific training data), e.g., for medical applications.

- [6]: This Google patent proposes adapting a machine learning model based on information collected locally on the device, such as use patterns or voice samples of the user. It suggests:
  - "selecting a Subset of a machine learning model to load into memory"
  - "adjusting a classification threshold value of the machine learning model"
  - "normalizing a feature output of the machine learning model"
- [7]: This paper proposes an online active learning (OAL) approach for human activity recognition (HAR) where the user is asked to annotate their own tasks (parsimoniously, because they will not comply otherwise). It addresses a different problem of personalisation: instead of having labeled data from several sources and selecting a subset thereof, HAR is based on data from only one source, for which only a very restricted number of ground-truth labels can be generated.
- Model *personalisation* is different from model *selection* because we wish to select *rows*, not *columns* of the feature matrix.

Existing incentive schemes

1. Overall problem
2. What people have done to solve this problem → literature

- [8]: Designs a scheme for the efficient approximation of the influence of each user's provided data (i.e., how much their data has contributed to minimising the loss function), which could serve as a basis for rewarding users. This requires that the payer has a private test set on which to compute the loss / the influence. Under this assumption, their scheme incentivises truthful data collection and reporting.

OPTIONAL – Existing methods for model compression

1. Overall problem
2. What people have done to solve this problem → literature

- Gradient compression with error feedback (reference in ERC grant application)
- Model compression (references in ERC grant application):
  - model sparsification
  - weight quantization

3 Objectives

Within the scope of this project, our main focus will be on developing and evaluating methods for model personalisation. Throughout the project, we will be mindful of the other requirements identified in section 1 and seek methods that are compatible with, or even beneficial for, as many of them as possible. We will strive to achieve objectives 3.3 and 3.4 in a fully reproducible manner, while maintaining the confidentiality of the provided data sets.

3.1 Landscape analysis - 3 weeks

We will begin by exploring available tools for reproducible data science, privacy-preserving distributed machine learning, and data analysis, and investigating how these tools can be leveraged to efficiently realise the other objectives listed herein. This landscape analysis will also research existing incentive schemes, as well as methods for model personalisation and communication compression. Our findings will be documented in the background section of this report.

3.2 Model personalisation and data selection - 4 weeks

Often, each user collects data on a different population. For instance, a hospital in Freetown and an Ebola treatment center in rural Sierra Leone would see fairly different patients in general, which would again differ from the patients admitted in Ebola treatment facilities in the Democratic Republic of the Congo. While it is also highly useful to make global inference across populations, clinicians in the Freetown hospital may be more interested in being able to accurately predict outcomes for new patients admitted *to their hospital*, than in predicting outcomes for new patients admitted *to any facility in general.* They would still need to use each other's data collaboratively, simply because the amount of data collected by each user is limited.

We will propose a variety of ways to help users in optimising the selected model for their local population. Some of the proposed methods will involve ranking the data sets of other users, exploiting various notions of similarity to the local data set. Ideas that we will pursue are:

- Computing a (potentially adapted) Ndoye factor for each available user before training.
- Using unsupervised machine learning techniques, such as Gaussian mixture model (GMM) clustering or principal component analysis (PCA), to analyze how individual data sets differ.
- Using a large portion of the local data set as a test set, and then selecting other users according to how much they contribute to reducing the test loss. As an additional advantage, this would guarantee that the evaluation metrics of the resulting model measure its ability to make predictions on the local population (as opposed to the global population or some aggregated subset thereof).
- Training a global model, where the mini-batch gradients computed by each user are used to define a notion of similarity. This concept is based on the idea that similar data sets will produce similar gradients in expectation over a mini-batch SGD step.

- Training a global model with all users and subsequently performing a small number of training steps on the local data set.

These ideas will be evaluated with respect to the following criteria:

- Freedom of choice left to the users and interpretability of the information presented to them.
- Compatibility with the distributed machine learning setting, resilience to incomplete data and to unreliably connected users.
- Potential to provide incentives and/or protect intellectual property.
- Potential to generate feedback on data quality and feature collection.
- Compatibility with existing methods that ensure data privacy on the one hand, and robustness on the other hand.
- Computation and communication requirements.

We will then develop suitable ideas and formalize them into concrete methods to be described in the corresponding section of this report.

3.3 Application to non-iid medical data - 3 weeks

Next, we will implement these methods and apply them to a medical data set collected on 577 patients in Sierra Leone during the 2013 - 2015 West African epidemic of Ebola Virus Disease (EVD) [2]. When a larger or more complete data set is needed, we will use the publicly available Titanic dataset [3]. The distributed machine learning problem will be simulated on a single machine. We will split the available data across simulated devices in a variety of ways to emulate unequal data set size, unequal distribution of features, unequal distribution of labels, unequal distribution of missing entries, and varying degrees of inequality between data sets.

We will evaluate how well each method performs for each split by using it to train and evaluate a set of models using cross-validation. Two baseline methods will also be evaluated, where models are trained once using only the local data set, and once using all available data. Evaluation metrics, such as root mean squared error or classification accuracy, will be computed on the local test set for each of the trained models. We will visualize these metrics graphically per method for the various splits and models using grouped bar charts, where each bar is a stacked bar chart showing the results obtained using the method under consideration versus either of the baseline methods. An example of such a grouped stacked bar chart is shown in Figure 1. Similar grouped stacked bar charts could also be made per data split to facilitate method selection for a given use case. Pairs of methods could be compared and contrasted by aligning their individual grouped-stacked bar charts in the fashion of a population pyramid (cf. Figure 2). Finally, instead of showing the improvement over the baseline, each stacked bar chart within the grouped stacked bar chart could be used to display the evaluation metric on the test set reached within a certain number of SGD steps, a certain communication traffic per participant, a certain collaborative scope, etc.

We will also evaluate the methods qualitatively by comparing which data sets are deemed similar to each other for any given task, and analyzing how these patterns arise.

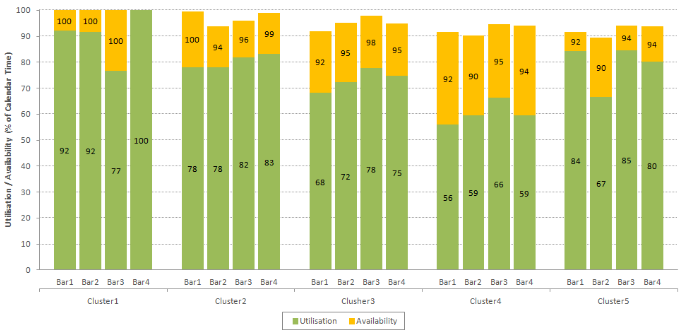

Figure 1: Example of a grouped stacked bar chart. Taken (unchanged) from MLD under CC BY-SA 4.0.

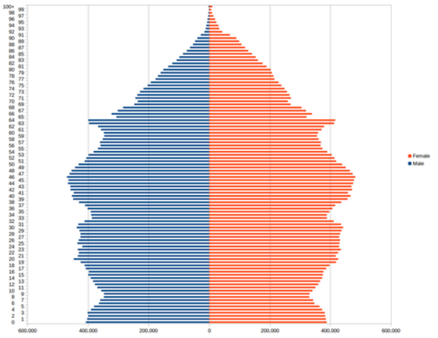

Figure 2: Example of a population pyramid. Taken (unchanged) from SkateTier under CC BY-SA 3.0.

3.4 Resilience test - If time allows

Additionally, we will emulate a set of participants who are spread across time zones and experience various levels of connection bandwidth and reliability by specifying individual (non-arbitrary) random connection/disconnection patterns for each simulated device. We will use these to test the resilience of the proposed methods.

4 Data selection methods

We adopt the perspective of a user u^* who is in possession of N^* datum points $\{\mathbf{x}_n^*, y_n^*\}_{n=1,\ldots,N^*}$ collected independently from the unknown distribution $\mathcal{D}^*$, which is characterized by the (equally unknown) probability density/mass functions $p^*(\mathbf{x})$ and $p^*(y|\mathbf{x})$:

$$\{\mathbf{x}_n^*, y_n^*\} \overset{i.i.d.}{\sim} \mathcal{D}^*$$

$$\mathbb{P}_{\mathcal{D}^*}[\mathbf{x}, y] = p^*(\mathbf{x})\, p^*(y|\mathbf{x})$$

This user periodically draws a new feature vector $\mathbf{x}^{new}$ from the same distribution $\mathcal{D}^*$, without observing its label y^{new}, and wishes to accurately predict this label as often as possible. To this end, user u^* wishes to approximate $p^*(y|\mathbf{x})$ with a machine learning model $\hat{p}^{ML}(y|\mathbf{x})$:

$$\hat{y}(\mathbf{x}^{new}) = \underset{y}{\mathbf{argmax}}\ \hat{p}^{ML}(y|\mathbf{x}^{new})$$

User u^* assesses the performance in the machine learning model $\hat{p}^{ML}$ using a loss function $\mathcal{L}(y, \hat{y})$, by approximating the models expected loss $\mathcal{R}^*\left(\hat{p}^{ML}\right)$:

$$\mathcal{R}^*\left(\hat{p}^{ML}\right) = \mathbb{E}_{\{\mathbf{x}^{new}, y^{new}\}\sim\mathcal{D}^*}\left[\mathcal{L}\left(y^{new}, \hat{y}\left(\mathbf{x}^{new}\right)\right)\right]$$

In addition to training on their own data set, user u^* can communicate with other users $\{u^i\}_{i=1,\ldots,U}$, who are in possession of data sets $\{\mathbf{x}_n^i, y_n^i\}_{n=1,\ldots,N^i}$ collected from separate and unknown distributions $\mathcal{D}^i$, which exhibit varying degrees of similarity with $\mathcal{D}^*$:

$$\left\{\mathbf{x}_n^i, y_n^i\right\} \overset{i.i.d.}{\sim} \mathcal{D}^i$$

$$\mathbb{P}_{\mathcal{D}^i}\left[\mathbf{x}, y\right] = p^i\left(\mathbf{x}\right) p^i\left(y|\mathbf{x}\right)$$

The adapted Ndoye factor.

Mention EDP application. E.g., when talking about the datasets used.

References

[1] Moritz Hardt and Guy N Rothblum. “A multiplicative weights mechanism for privacy-preserving data analysis”. In: *2010 IEEE 51st Annual Symposium on Foundations of Computer Science*. IEEE. 2010, pp. 61–70.

[2] Mary-Anne Hartley et al. “Predicting Ebola infection: A malaria-sensitive triage score for Ebola virus disease”. In: *PLoS neglected tropical diseases* 11.2 (2017).

[3] Miaofeng Liu et al. *Reinforced Training Data Selection for Domain Adaptation*. 2019. DOI: `10.18653/v1/P19-1189`. URL: `https://www.aclweb.org/anthology/P19-1189` (visited on 03/27/2020).

[4] Frank D McSherry. “Privacy integrated queries: an extensible platform for privacy-preserving data analysis”. In: *Proceedings of the 2009 ACM SIGMOD International Conference on Management of data*. 2009, pp. 19–30.

[5] Ricardo Mendes and João P Vilela. “Privacy-preserving data mining: methods, metrics, and applications”. In: *IEEE Access* 5 (2017), pp. 10562–10582.

[6] Xu Miao. *Personalized machine learning models*. US Patent App. 14/105,650. June 2015.

[7] Tudor Miu, Paolo Missier, and Thomas Plötz. “Bootstrapping personalised human activity recognition models using online active learning”. In: *2015 IEEE International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing*. IEEE. 2015, pp. 1138–1147.

[8] Adam Richardson, Aris Filos-Ratsikas, and Boi Faltings. “Rewarding High-Quality Data via Influence Functions”. In: *arXiv preprint arXiv:1908.11598* (2019).