CS-E4740 - Federated Learning

FL Design Principle

Assoc. Prof. Alexander Jung

Spring 2025

Playlist

Glossary

Course Site

Table of Contents

Formulating FL as Optimization

Computational Aspects

Statistical Aspects

Interpretations

Table of Contents

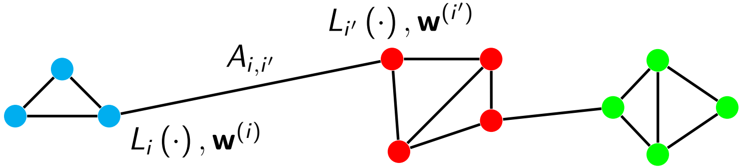
Formulating FL as Optimization

Computational Aspects

Statistical Aspects

Interpretations

An FL Network



- FL network consisting of devices $i = 1, \ldots, n$.
- Some i, i' connected by an edge with weight $A_{i,i'} > 0$.
- Device i learns model params. $\mathbf{w}^{(i)} \in \mathbb{R}^d$.
- Usefulness of $\mathbf{w}^{(i)}$ measured by some local loss, e.g.,

$$L_i\left(\mathbf{w}^{(i)}\right) := \frac{1}{m_i} \sum_{r=1}^{m_i} \left(y^{(i,r)} - \left(\mathbf{w}^{(i)}\right)^T \mathbf{x}^{(i,r)} \right)^2.$$

FL via Regularization

- Each node carries a linear model $h^{(\mathbf{w}^{(i)})}(\mathbf{x}) := \mathbf{x}^T \mathbf{w}^{(i)}$.
- Each node carries m_i labelled data points.
- Node-wise ML fails if $m_i \ll d$ (overfitting).

Idea:

Use the neighbours $\mathcal{N}^{(i)} := \{i' : \{i, i'\} \in \mathcal{E}\}$ to regularize!

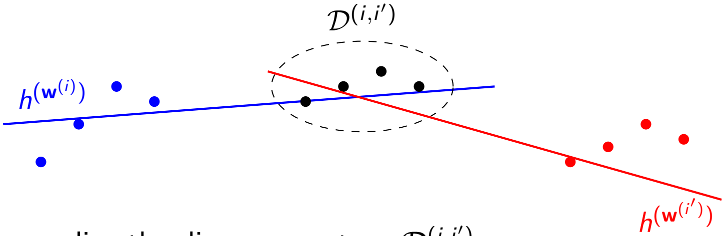
FL via Regularization (ctd.)

As for basic ML, regularization can be done either via

- **Data augmentation** using data from the neighbours.
- **Prune local models** by requiring them to agree across edges.
- **Add a penalty term** to the local loss function.

Building a Penalty Across Edges

- Consider two nodes i, i' with local datasets $\mathcal{D}^{(i)}, \mathcal{D}^{(i')}$.
- Assume there is a non-empty overlap $\mathcal{D}^{(i)} \cap \mathcal{D}^{(i')}$.



We penalize the disagreement on $\mathcal{D}^{(i,i')}$:

$$\sum_{\mathbf{x} \in \mathcal{D}^{(i,i')}} \left(h^{(\mathbf{w}^{(i)})}(\mathbf{x}) - h^{(\mathbf{w}^{(i')})}(\mathbf{x}) \right)^2 = \sum_{\mathbf{x} \in \mathcal{D}^{(i,i')}} \left(\mathbf{x}^T \mathbf{w}^{(i)} - \mathbf{x}^T \mathbf{w}^{(i')} \right)^2$$

$$= \left(\mathbf{w}^{(i)} - \mathbf{w}^{(i')} \right)^T \left[\sum_{\mathbf{x} \in \mathcal{D}^{(i,i')}} \mathbf{x}^T \mathbf{x} \right] \left(\mathbf{w}^{(i)} - \mathbf{w}^{(i')} \right).$$

Generalized TV Minimization (GTVMin)

Learn model params. $\widehat{\mathbf{w}}^{(i)}$ by balancing local loss and GTV

$$\min_{\mathbf{w}^{(1)},\ldots,\mathbf{w}^{(n)} \in \mathbb{R}^d} \sum_{i=1}^{n} \left[L_i\left(\mathbf{w}^{(i)}\right) + \alpha \sum_{\{i,i'\} \in \mathcal{E}} A_{i,i'} \phi\left(\mathbf{w}^{(i)} - \mathbf{w}^{(i')}\right) \right]$$

- Penalty function $\phi(\mathbf{u})$ is a design choice.
- Previous slide used $\phi(\mathbf{u}) = \mathbf{u}^T \mathbf{Q} \mathbf{u}$ with $\mathbf{Q} := \sum_{\mathbf{x} \in \mathcal{D}^{(i,i')}} \mathbf{x}\mathbf{x}^T$.
- Our focus is on the choice $\phi(\mathbf{u}) := \|\mathbf{u}\|_2^2$.
- Another popular choice is $\phi(\mathbf{u}) := \|\mathbf{u}\|$.[1]

[1] Y. SarcheshmehPour, et.al, "Clustered Federated Learning via Generalized Total Variation Minimization," in IEEE Trans. Sig. Proc, 2023, doi: 10.1109/TSP.2023.3322848.

Model-Agnostic GTVMin

Replacing $\phi\big(\mathbf{w}^{(i)} - \mathbf{w}^{(i')}\big)$ with the disagreement measure

$$D\big(h^{(i)}, h^{(i')}\big) := \sum_{\mathbf{x} \in \mathcal{D}^{(i,i')}} \big(h^{(i)}(\mathbf{x}) - h^{(i')}(\mathbf{x})\big)^2$$

yields a model-agnostic generalization of GTVMin

$$\min_{\substack{h^{(i)} \in \mathcal{H}^{(i)} \\ i \in \mathcal{V}}} \sum_{i \in \mathcal{V}} \bigg[L_i \left(h^{(i)}\right) + \alpha \sum_{\{i,i'\} \in \mathcal{E}} A_{i,i'} D\big(h^{(i)}, h^{(i')}\big) \bigg].$$

This allows for VERY heterogeneous FL networks, e.g.,
$\mathcal{H}^{(1)} =$ lin.model, $\mathcal{H}^{(2)} =$ LLM, $\mathcal{H}^{(3)} =$ decision tree.

Table of Contents

Formulating FL as Optimization

Computational Aspects

Statistical Aspects

Interpretations

Computational Aspects

$$\min_{\mathbf{w}^{(1)},\ldots,\mathbf{w}^{(n)}\in\mathbb{R}^d} \sum_{i=1}^{n} \left[L_i\left(\mathbf{w}^{(i)}\right) + \alpha \sum_{\{i,i'\}\in\mathcal{E}} A_{i,i'}\phi\big(\mathbf{w}^{(i)} - \mathbf{w}^{(i')}\big) \right]$$

- How can we solve it efficiently over an FL network?
- How much compute/comm. is needed at least?
- What is the effect of different choices for the edges $\mathcal{E}$, loss funcs. $L_i(\cdot)$, and GTV penalty ϕ?

Computational Aspects - Smooth GTVmin

Consider a GTVMin instance

$$\min_{\mathbf{w}^{(1)},\ldots,\mathbf{w}^{(n)} \in \mathbb{R}^d} \sum_{i=1}^{n} \left[L_i\left(\mathbf{w}^{(i)}\right) + \alpha \sum_{\{i,i'\} \in \mathcal{E}} A_{i,i'} \left\| \mathbf{w}^{(i)} - \mathbf{w}^{(i')} \right\|_2^2 \right]$$

with a smooth (differentiable) $L_i(\cdot)$.

If we use (distributed) gradient descent to solve GTVMin:

- How many iterations should we run?
- What is a good choice for the learning rate?
- How to communicate gradients over comm. links?

Characterizing GTVMin Solutions

- Consider GTVMin solution $\widehat{\mathbf{w}}^{(i)} \in \mathbb{R}^d$, for $i = 1, \ldots, n$.
- We stack them into a long vector

$$\widehat{\mathbf{w}} := \left(\widehat{\mathbf{w}}^{(1)}, \ldots, \widehat{\mathbf{w}}^{(n)}\right)^T \in \mathbb{R}^{dn}.$$

- We characterize the solutions as a fixed-point of some $\mathcal{F}$,

$$\boxed{\widehat{\mathbf{w}} \text{ solves GTVMin } \Leftrightarrow \widehat{\mathbf{w}} = \mathcal{F}\widehat{\mathbf{w}}}$$

- The operator $\mathcal{F}$ is not unique (design choice!).

Convex and Smooth GTVMin

Consider GTVMin with a smooth and convex $L_i(\cdot)$,

$$\min_{\mathbf{w}^{(1)},\ldots,\mathbf{w}^{(n)}\in\mathbb{R}^d} \sum_{i=1}^{n} \left[L_i\left(\mathbf{w}^{(i)}\right) + \alpha \sum_{\{i,i'\}\in\mathcal{E}} A_{i,i'} \left\| \mathbf{w}^{(i)} - \mathbf{w}^{(i')} \right\|_2^2 \right] \quad (1)$$

$$\boxed{\widehat{\mathbf{w}} \text{ solves } (1) \Leftrightarrow \widehat{\mathbf{w}} = \mathcal{F}^{(\eta)}\widehat{\mathbf{w}}}$$

$\mathcal{F}^{(\eta)}$ maps $\mathbf{u} = \left(\mathbf{u}^{(1)},\ldots,\mathbf{u}^{(n)}\right)^T$ to $\mathbf{v} = \left(\mathbf{v}^{(1)},\ldots,\mathbf{v}^{(n)}\right)^T$,

$$\mathbf{v}^{(i)} = \mathbf{u}^{(i)} - \eta \left[\nabla L_i\left(\mathbf{u}^{(i)}\right) + 2\alpha \sum_{i'\in\mathcal{N}^{(i)}} A_{i,i'}\left(\mathbf{u}^{(i)} - \mathbf{u}^{(i')}\right) \right].$$

Different choices for "step-size" $\eta > 0$ yield different $\mathcal{F}$.

Fixed-Point Iterations

Q: How to compute a fixed point $\widehat{\mathbf{w}}$ of $\mathcal{F}$?

A: Start with initial guess $\widehat{\mathbf{w}}^{(0)}$ and iterate

$$\widehat{\mathbf{w}}^{(k)} = \mathcal{F}\widehat{\mathbf{w}}^{(k-1)}, \text{ for } k = 1, 2, \ldots.$$

If $\mathcal{F}$ is **firmly non-expansive** $\lim_{k\to\infty} \widehat{\mathbf{w}}^{(k)} = \widehat{\mathbf{w}}$.[2]

If $\mathcal{F}$ is even **contractive** with constant $\kappa < 1$,

$$\left\|\widehat{\mathbf{w}}^{(k)} - \widehat{\mathbf{w}}\right\|_2 \leq \kappa^k \left\|\widehat{\mathbf{w}}^{(0)} - \widehat{\mathbf{w}}\right\|_2.$$

[2] H. Bauschke, P. Combettes, "Convex Analysis and Monotone Operator Theory in Hilbert Spaces," Springer, 2017.

Gradient Descent as Fixed-Point Iteration

GD for smooth and convex objective function $f(\mathbf{w})$,

$$\mathbf{w}^{(k)} = \mathbf{w}^{(k-1)} - \eta \nabla f\left(\mathbf{w}^{(k-1)}\right)$$

is a fixed-point iteration with $\mathcal{F}^{(\eta)} : \mathbf{w} \mapsto \mathbf{w} - \eta \nabla f(\mathbf{w})$.

- In general, $\mathcal{F}^{(\eta)}$ is neither firmly non-exp. nor contractive.
- Convergence can still be ensured if η is sufficiently small.
- E.g., using learning rate $\eta_k = 1/k$ for smooth $f\left(\mathbf{w}\right)$.

Table of Contents

Formulating FL as Optimization

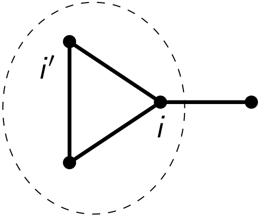
Computational Aspects

Statistical Aspects

Interpretations

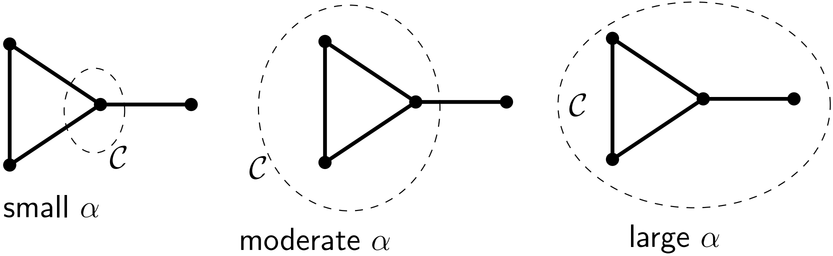
Statistical Aspects

- GTVMin solution yields model params. $\widehat{\mathbf{w}}^{(i)}$, $i = 1, \ldots, n$
- How useful are these model params. ?
- The local loss $L_i\left(\widehat{\mathbf{w}}^{(i)}\right)$ can be misleading (why?)
- Better to use aggregate $\sum_{i \in \mathcal{C}^{(i)}} L_i\left(\widehat{\mathbf{w}}^{(i)}\right)$, with cluster



$$\mathcal{C}^{(i)} := \left\{ i' : \widehat{\mathbf{w}}^{(i)} \approx \widehat{\mathbf{w}}^{(i')} \right\}.$$

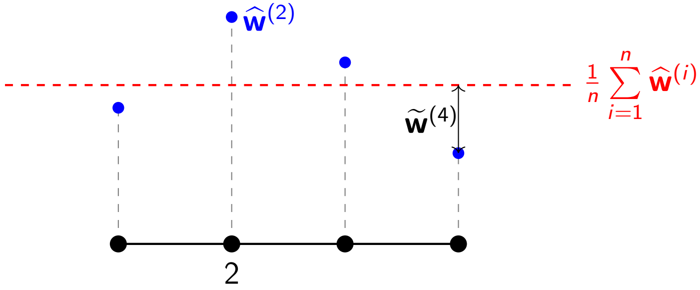
Clustering of GTVMin[3]



[3] Y. SarcheshmehPour, Y. Tian, L. Zhang and A. Jung, "Clustered Federated Learning via Generalized Total Variation Minimization," in IEEE Transactions on Signal Processing, 2023,

Analysis of Clustering - Assumptions

- Consider a connected FL network $\mathcal{G}$ with $\lambda_2 > 0$.
- Assume loss funcs. satisfy $\min_{\mathbf{v} \in \mathbb{R}^d} \sum_{i=1}^{n} L_i(\mathbf{v}) \leq \varepsilon$
- Use GTVMin to learn local params. $\widehat{\mathbf{w}}^{(i)}$.
- Define the variation $\widetilde{\mathbf{w}}^{(i)} := \widehat{\mathbf{w}}^{(i)} - \frac{1}{n} \sum_{i=1}^{n} \widehat{\mathbf{w}}^{(i)}$.



Analysis of Clustering - Upper Bound

The variation $\widetilde{\mathbf{w}}^{(i)}$ is upper bounded as

$$\sum_{i=1}^{n} \left\| \widetilde{\mathbf{w}}^{(i)} \right\|_2^2 \leq \frac{\varepsilon}{\alpha \lambda_2}.$$

This bound involves the

- connectivity of FL network (via λ_2),
- the properties of local loss functions (via ε), and
- the GTVMin parameter α.

A large $\alpha\lambda_2$ results in nearly identical local params. $\widetilde{\mathbf{w}}^{(i)} \approx \mathbf{0}$.

Table of Contents

Formulating FL as Optimization

Computational Aspects

Statistical Aspects

Interpretations

Interpretations

We next discuss some interpretations of GTVMin

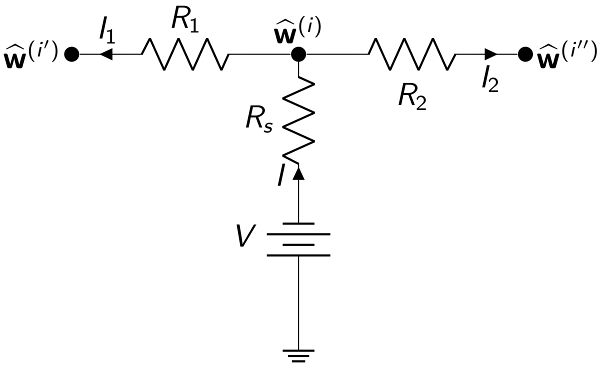
$$\min_{\mathbf{w}^{(1)},\ldots,\mathbf{w}^{(n)} \in \mathbb{R}^d} \sum_{i=1}^{n} \left[L_i\left(\mathbf{w}^{(i)}\right) + \alpha \sum_{\{i,i'\} \in \mathcal{E}} A_{i,i'} \left\| \mathbf{w}^{(i)} - \mathbf{w}^{(i')} \right\|_2^2 \right]$$

for some FL network with weighted undirected graph $\mathcal{G}$ and smooth and convex loss func. $L_i\left(\mathbf{w}^{(i)}\right)$.

We assume that there exists a solution $\widehat{\mathbf{w}}^{(1)}, \ldots, \widehat{\mathbf{w}}^{(n)}$. (Do we really need to make this assumption?)

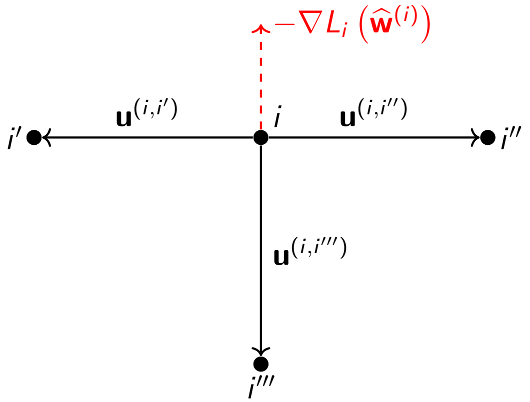
Electronic Circuit

Consider a node i with neighbours $\mathcal{N}^{(i)} = \{i', i''\}$.



$$\underbrace{-\nabla L_i\left(\widehat{\mathbf{w}}^{(i)}\right)}_{I} = \left[\underbrace{A_{i,i'}\left(\widehat{\mathbf{w}}^{(i)} - \widehat{\mathbf{w}}^{(i')}\right)}_{I_1} + \underbrace{A_{i,i''}\left(\widehat{\mathbf{w}}^{(i)} - \widehat{\mathbf{w}}^{(i'')}\right)}_{I_2}\right]$$

Vector-Valued Flows[4]

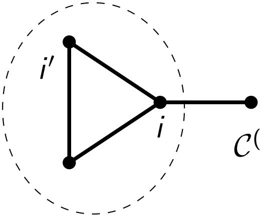


Vector-valued flow $\mathbf{u}^{(i,i')} := \nabla \phi\big(\mathbf{u}\big)\Big|_{\mathbf{u}=\widehat{\mathbf{w}}^{(i)}-\widehat{\mathbf{w}}^{(i')}}$.

[4] AJ, "On the Duality Between Network Flows and Network Lasso," in IEEE Signal Processing Letters, 2020.

Locally Weighted Learning

GTVMin delivers local params. $\widehat{\mathbf{w}}^{(i)}$ that are clustered.



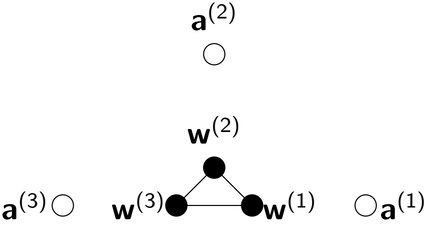
$$\mathcal{C}^{(i)} := \left\{ i' \in \mathcal{V} : \widehat{\mathbf{w}}^{(i)} \approx \widehat{\mathbf{w}}^{(i')} \right\}.$$

For node i, GTVmin is the same as locally weighted learning[5]

$$\min_{\mathbf{w}^{(i)} \in \mathbb{R}^d} \sum_{i'=1}^{n} L_{i'}\left(\mathbf{w}^{(i)}\right) \rho_{i'} \text{ with } \rho_{i'} = \begin{cases} 1 & \text{if } i' \in \mathcal{C}^{(i)} \\ 0 & \text{, otherwise.} \end{cases}$$

[5]C. G. Atkeson, S. A. Schaal and Andrew W, Moore, Locally Weighted Learning, AI Review,Volume 11, Pages 11-73 (Kluwer Publishers) 1997.

Generalized Convex Clustering[6]



$$\min_{\mathbf{w}^{(1)},\ldots,\mathbf{w}^{(n)}} \sum_{i=1}^{n} \left\|\mathbf{w}^{(i)} - \mathbf{a}^{(i)}\right\|_2^2 + \alpha \sum_{i,i' \in \mathcal{V}} \left\|\mathbf{w}^{(i)} - \mathbf{w}^{(i')}\right\|_2 .$$

[6]D. Sun, et.al, Convex Clustering: Model, Theoretical Guarantee and Efficient Algorithm, JMLR, 2021.

What's Next?

The next module applies optimization methods to solve GTVMin.

We can implement these methods as message passing over the edges of an FL network.

Further Resources

- **YouTube:** @alexjung111
- **LinkedIn:** Alexander Jung
- **GitHub:** alexjungaalto