

CS-E4740 - Federated Learning

L1 - From ML to FL

Assoc. Prof. Alexander Jung

Spring 2026

Calendar



Glossary



Book



GitHub



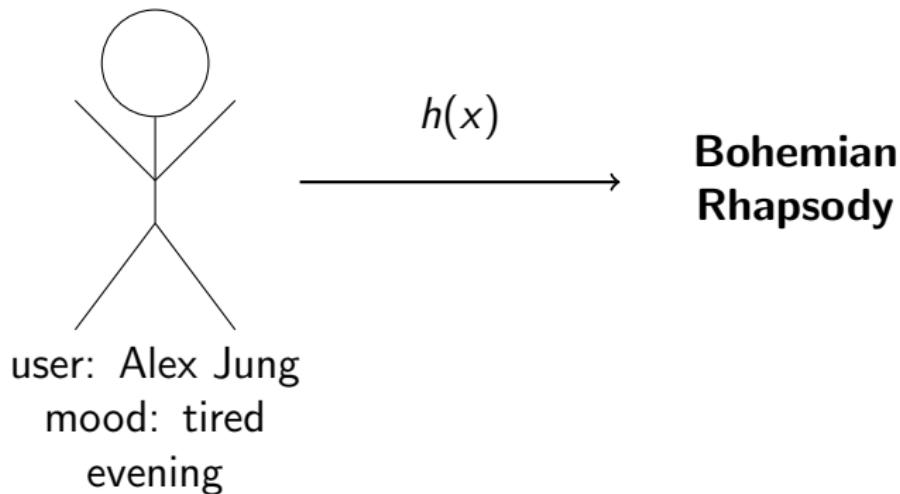
Table of Contents

machine learning (ML) Basics Refresher

Introduction to federated learning (FL)

From ML to FL

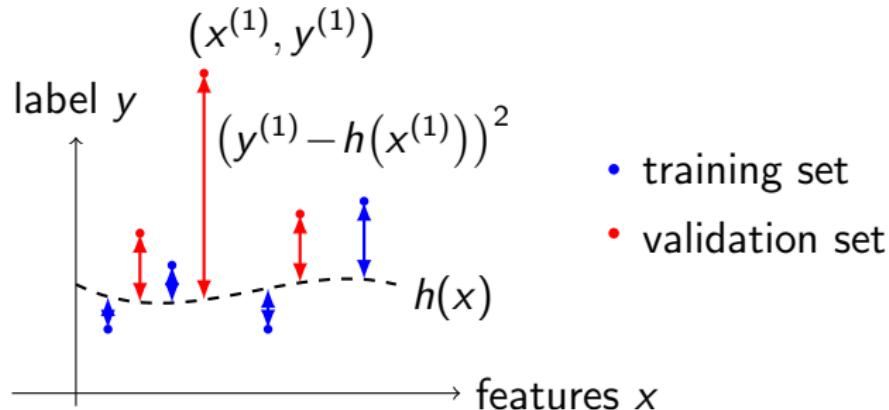
The Right Song Can Save the Day



How do we get a good hypothesis map $h(x)$?

Wang, M., Wu, J., Yan, H. (2023). "Effect of music therapy on older adults with depression: A systematic review and meta-analysis." *Complementary Therapies in Clinical Practice* <https://doi.org/10.1016/j.ctcp.2023.101809>

Empirical risk minimization (ERM)



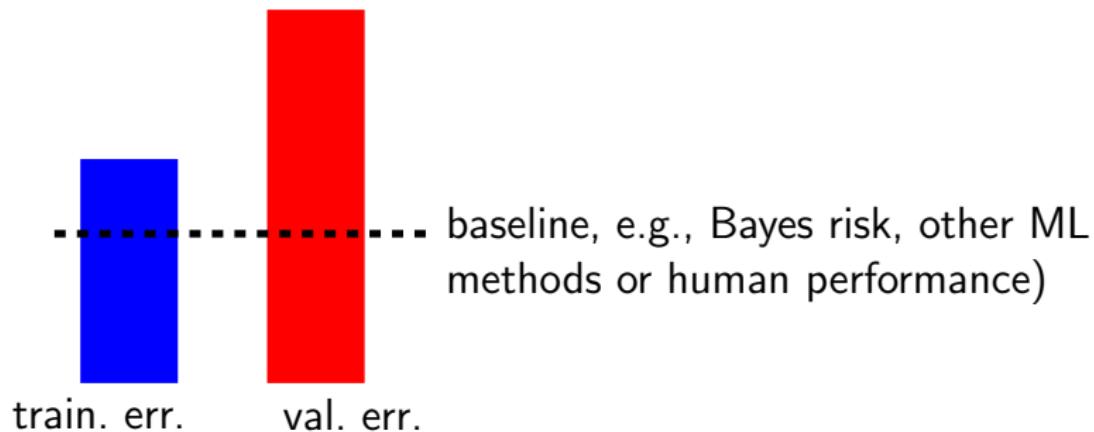
Learn $h \in \mathcal{H}$ by min. average loss (empirical risk),

$$\min_{h \in \mathcal{H}} \frac{1}{m} \sum_{r=1}^m L((\mathbf{x}, y), h).$$

Different choices for \mathcal{H} and loss L yield different ML methods.

see Chapters 3,4 of AJ, "Machine Learning: The Basics," Springer, 2022.
<https://mlbook.cs.aalto.fi>

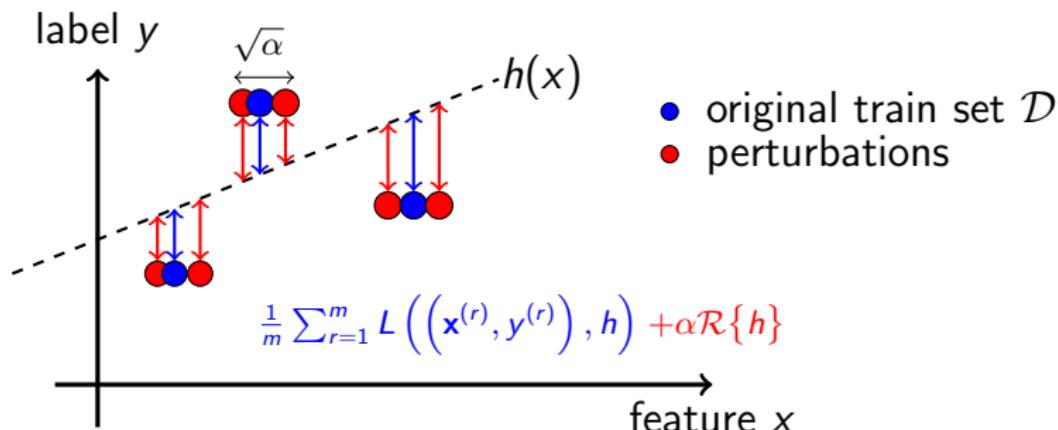
Applied ML - Trial and Error



ML diagnosis by comparing training error with validation error and a baseline.

see Chapter 6 of AJ, "Machine Learning: The Basics," Springer, 2022.
<https://mlbook.cs.aalto.fi>

Applied ML - Regularization



Start with large \mathcal{H} , then shrink it via

- ▶ data augmentation, e.g., $\mathbf{x} \mapsto \mathbf{x} + \mathcal{N}(0, \alpha)$, or
- ▶ adding penalty term to loss function, e.g., $\dots + \alpha \|\mathbf{w}\|_2^2$,
- ▶ **constraining** model parameters, e.g., $\|\mathbf{w}\|_2 \leq 1$.

see Chapter 7 of AJ, "Machine Learning: The Basics," Springer, 2022.
<https://mlbook.cs.aalto.fi>

Table of Contents

ML Basics Refresher

Introduction to FL

From ML to FL

What is FL?

FL trains ML models over a network of devices.

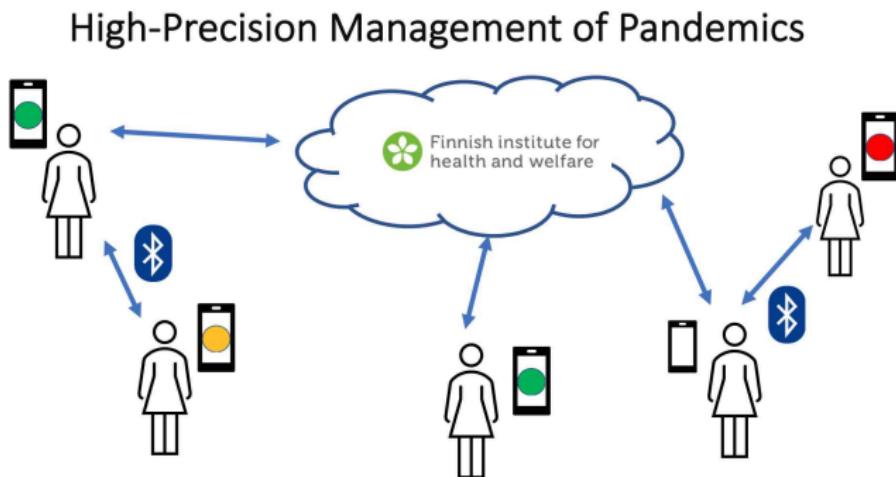


Figure: A hypothetical FL system for pandemic forecasting.

Smartphones train personalized models based on their observations (e.g., audio recordings of coughing) as well as public health-care data.

Devices

We use the term device broadly.

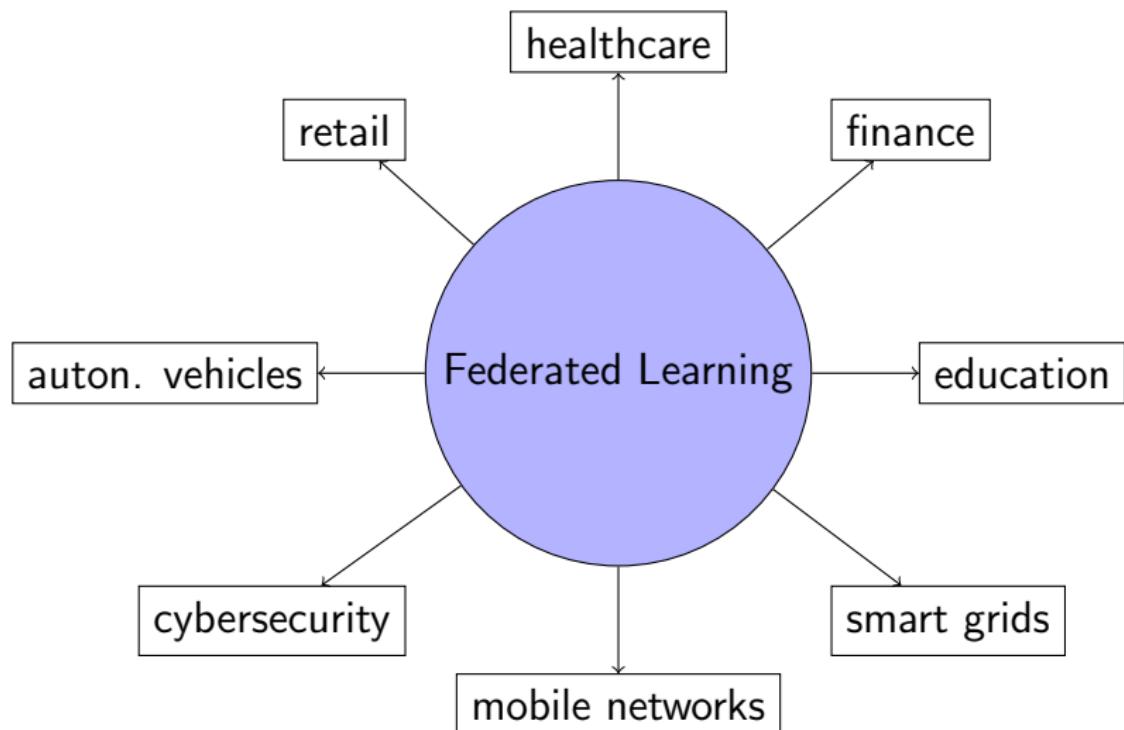
It is any computational system that is able to

- ▶ access data,
- ▶ train a model, and
- ▶ communicate with other devices.

Key Characteristics of FL

- ▶ No centralized data collection (no single point of failure)).
- ▶ Each device trains a tailored model (high-precision).
- ▶ Scalability: more devices yield more compute and data.
- ▶ No raw data is shared (privacy-friendly).

FL Applications



FL in Healthcare

- ▶ Turn smartphone into personal health-care advisor.
- ▶ Smartphone app uses FL to train personalized model.
- ▶ Combine personal data with public health-care data.

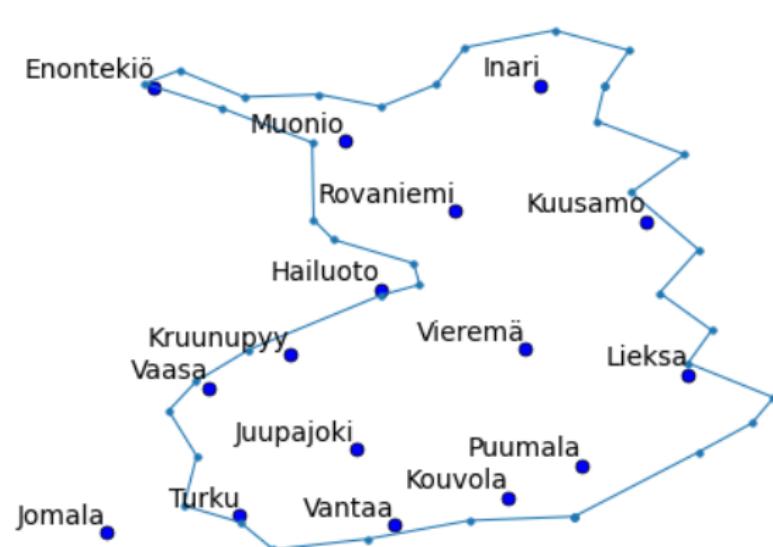
Key Reference: Rieke, N., et al. *The future of digital health with federated learning*. Nature Medicine, 2020.

FL in Finance

FL can help financial institutions to improve

- ▶ **Fraud detection.** N. F. Aurna, et.al., "Federated Learning-Based Credit Card Fraud Detection: Performance Analysis with Sampling Methods and Deep Learning Algorithms," 2023,
- ▶ **Risk assessment.** W. Li, et.al., "Personal Credit Evaluation Model Based on Federated Learning," 2024

FL at FMI

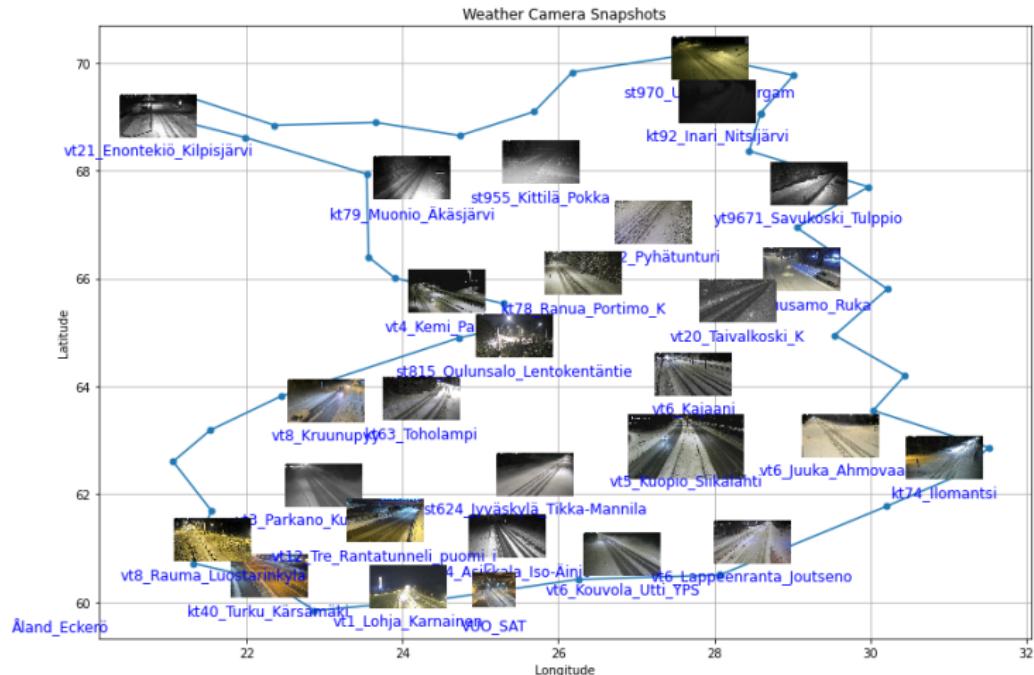


Train a separate model for each Finnish Meteorological Institute (FMI) weather station

Python script for reproducing the Fig.:



FL for Finnish Road Safety



Train separate model for each camera operated by FinTraffic

Python script for reproducing the Fig.:



The Internet of Things (IoT) is Growing

IoT connections (billion)

IoT	2023	2029	CAGR
Wide-area IoT	3.6	7.2	12%
Cellular IoT	3.4	6.7	12%
Short-range IoT	12.1	31.6	17%
Total	15.7	38.8	16%

Note: Based on rounded figures. Cellular IoT figures are also included in the figures for wide-area IoT.

Figure: Some IoT statistics from



The IoT - A Global FL System

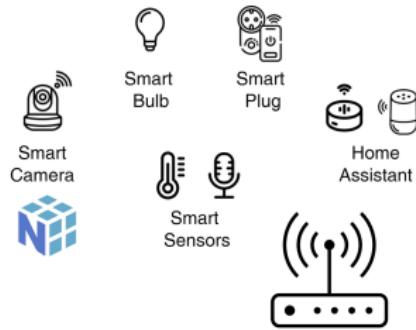


Table of Contents

ML Basics Refresher

Introduction to FL

From ML to FL

From ML to FL

- ▶ Basic ML: Train a single model \mathcal{H} by minimizing average loss on a single dataset
- ▶ FL: Train a separate model $\mathcal{H}^{(i)}$ for each node i of an interconnected FL system.

From ML to FL

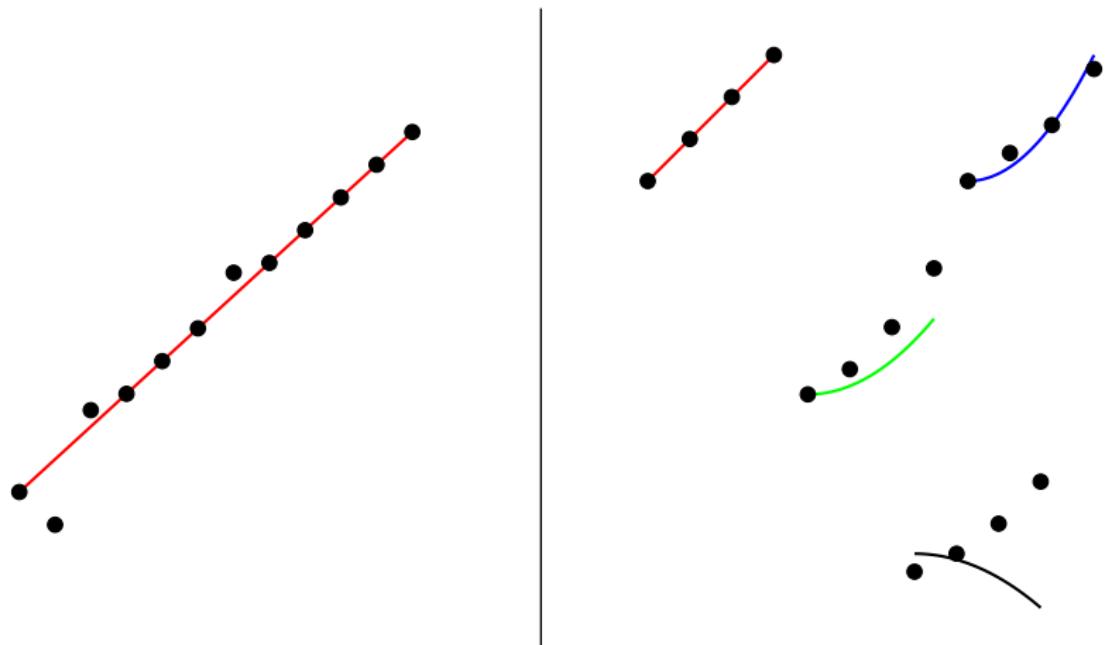


Figure: Left: A ML method uses a single dataset to train a single model. Right: FL methods train ML models from decentralized data.

ML with Python

```
X, y = read_data()  
model = SGDRegressor()  
model.fit(X, y)
```

FL with Python

IP: 192.168.0.1

```
model = SGDRegressor()  
y_hat =recv_preds(192.168.0.3)  
X, y = read_data()  
Xa,ya=augment_data(X, y, y_hat)  
model.fit(Xa,ya)
```

IP: 192.168.0.2

```
X,y = read_data()  
model=LinearRegression()  
model.fit(X, y)
```

IP: 192.168.0.3

```
model=DecisionTree()  
y_hat =recv_preds(192.168.0.2)  
X, y = read_data()  
Xa,ya=augment_data(X, y, y_hat)  
model.fit(Xa,ya)
```

Federated Learning Network (FL network)

We represent a FL system by a weighted undirected graph

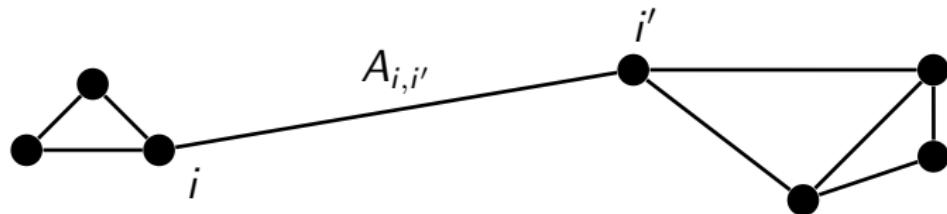
$$\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{A}),$$

where

- ▶ nodes $i \in \mathcal{V}$ represent participating devices
- ▶ edges $\{i, i'\} \in \mathcal{E}$ indicate possible collaboration,
- ▶ edge weight $A_{i,i'} \geq 0$ quantifies amount of collaboration

Unless stated otherwise,

- ▶ $A_{i,i}=0$ for all $i \in \mathcal{V}$ (no self-loops)
- ▶ and $\mathcal{V}=\{1, 2, \dots, n\}$ with some $n \in \mathbb{N}$ (finiteness).

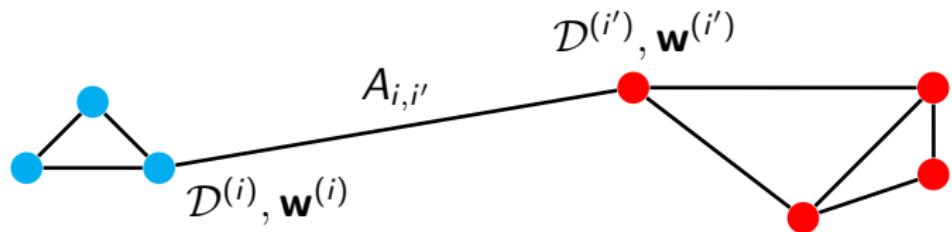


From FL network to FL system

Each node $i \in \mathcal{V}$,

- ▶ can access local dataset $\mathcal{D}^{(i)}$,
- ▶ maintains model parameters $\mathbf{w}^{(i)}$
- ▶ sends/receives messages from neighbors $\mathcal{N}^{(i)}$.

An FL algorithm specifies *when* and *how* these model parameters are updated.



FL Algorithms

Each node i uses current model parameters $\mathbf{w}^{(1,t)}, \dots, \mathbf{w}^{(n,t)}$ to compute new model parameters $\mathbf{w}^{(i,t+1)}$,

$$\mathbf{w}^{(i,t+1)} = \mathcal{F}^{(i)}(\mathbf{w}^{(1,t)}, \dots, \mathbf{w}^{(n,t)}) \text{ at time instants } t = 0, 1, \dots$$

The node-wise operator $\mathcal{F}^{(i)}$ involves

- ▶ local model updates (e.g., via gradient steps)
- ▶ sharing model parameters across edges of FL network.

What's Next?

The next module introduces generalized total variation minimization (GTVMin) as our main design principle for FL algorithms.