CS-E4740 - Federated Learning

Welcome and Introduction

Assoc. Prof. Alexander Jung

Spring 2025

At a Glance

- runs Feb. - May 2025, no mandatory attendance
- consists of lectures, assignments and project
- Aalto students register via `https://sisu.aalto.fi/`
- Nordic Five Tech students register via local admin
- everybody can subscribe to course via this form

Table of Contents

Pre-Requisites and Learning Goals

Positioning in ML Curricula

Course Logistics

Introduction to FL

ML Basics Refresher

Table of Contents

Pre-Requisites and Learning Goals

Positioning in ML Curricula

Course Logistics

Introduction to FL

ML Basics Refresher

Prerequisites

- **Linear Algebra:** vectors $\mathbf{w} \in \mathbb{R}^n$, matrices $\mathbf{Q} \in \mathbb{R}^{n \times n}$, norms $\|\mathbf{w}\|_2$
- **Multivariable Calculus:** smooth functions $f(\mathbf{w})$ and their gradient $\nabla f(\mathbf{w})$.
- **Basic Machine Learning:** empirical risk minimization (ERM), $\min_{h \in \mathcal{H}} (1/m) \sum_{r=1}^{m} L\left(\left(\mathbf{x}^{(r)}, y^{(r)}\right), h\right)$
- **Python:** basic coding skills and familiarity with libraries `numpy` and `scikit-learn`

Learning Goals

after completing this course, you

- can model FL applications using network models
- can formulate FL as an optimization problem
- can use opt. methods to construct FL algorithms
- can design and diagnose FL systems

Table of Contents

Pre-Requisites and Learning Goals

Positioning in ML Curricula

Course Logistics

Introduction to FL

ML Basics Refresher

Positioning of CS-E4740 in ML Curriculum

In what follows, we briefly explain how CS-E4740 relates to selected courses at Aalto University and University of Helsinki.

Related Courses - Bare Necessities

- **MS-A0001 - Matrix Algebra.** Introduction to linear algebra in $\mathbb{R}^d$. We will use $\mathbb{R}^d$ as a mathematical model for FL.
- **CS-C3240 - Machine Learning.** Teaches basic techniques for training a single ML model on a given dataset. FL extends this centralized setting to networks of devices, each having access to a local dataset and a local (personalized) model.
- **Data Analysis with Python.** Teaches how to implement basic ML methods in Python. The assignments of our course require to implement (parts of) FL algorithms in Python.

Related Courses - Nice to Have

- **MS-C2105 - Introduction to Optimization.** Teaches basic concepts for the design and analysis of optimization methods. Our course formulates FL as an optimization problem. FL algorithms are obtained, in turn, by applying optimization methods to solve this problem.

- **ELEC-E5424 - Convex Optimization.** Teaches advanced tools (such as convergence analysis of gradient methods) for the study and design of FL algorithms.

Related Courses - Follow Up

- **ELEC-E7120 - Wireless Systems.** Discusses the fundamentals of radio communications which can be used to implement FL algorithms.
- **ELEC-E8102 - Distributed and Intelligent Automation Systems.** Discusses automation systems consisting of interconnected sensors and actuators. We can use FL to train predictive models used by these devices.

Table of Contents

Pre-Requisites and Learning Goals

Positioning in ML Curricula

Course Logistics

Introduction to FL

ML Basics Refresher

Six Modules

- **ML Refresher.** Model training, validation and regularization.
- **FL Networks.** Use graphs to model FL networks.
- **FL Design Principle.** Formulate FL as optimization over graph.
- **FL Algorithms.** Solve FL problem with distributed algorithms.
- **FL Flavours.** Clustered, vertical, horizontal and more.
- **Trustworthy FL.** Explainability, robustness, privacy-protection.

each module (**M**) consists of lecture (**L**) and assignment (**A**).

Graded Activities

- **assignments:** implement concepts in Python
- **project:** study application of your wish
- instead of (in addition to) assignments you can do a **paper review:**
  - choose a paper from a curated list
  - interpret it using the concepts taught in the lectures
  - present your review in a slide talk
  - deliverable: slides and recording (max. 10 minutes)

Grading

grading based on points collected via

- 6 assignments, max. $6 \cdot 7 = 42$ points (Assgts)
- peer grading of projects, max. 7 points (PG)
- final project report, max. 51 points $(\mathrm{Project})$
- paper review, max. 49 points (Paper)

total number of points $= \max(\mathrm{Assgts}, \mathrm{Paper}) + \mathrm{Project} + \mathrm{PG}$

From Points to Grade

- **grade 1** for 50-59 points
- **grade 2** for 60-69 points
- **grade 3** for 70-79 points
- **grade 4** for 80-89 points
- **top grade 5** for at least 90 points

Assignments

Six assignments **A1**,. . .,**A6**, one for each module, require you to implement in Python . . .

- (**A1**) basic ML methods.
- (**A2**) a FL network.
- (**A3**) FL as optimization.
- (**A4**) basic FL algorithms via message passing.
- (**A5**) clustered and vertical FL.
- (**A6**) a privacy attack and subjective explainability.

Student Project

- choose an FL application of your choice
- design and study FL algorithms from the course
- write project report (template: tex, pdf)
- submit your report by end of April
- peer-review other reports until May-15
- submit revised report and response letter by end of May

Schedule

- (**L**) each Mo. 16:15, starting 24-Feb.
- (**A**) presentation each Wed. 16:15, starting 26-Feb.
- **A1**, **A2** must be submitted by 17-Mar-2025
- **A3**, **A4** by 31-Mar-2025
- **A5**, **A6** by 14-April-2025
- first project report submission by 30-April-2025
- peer grading until 15-May-2025
- final project submission by 31-May-2025

Ground Rules

As a student following this course, you must act according to the Code of Conduct of Aalto University (see here).

Two main ground rules for this course are . . .

Rule I - Be Honest

- course includes a lot of independent work:
  - completing assignments
  - preparing student project report
  - peer grading others projects
  - completing paper review
- do not steal (plagiarize) others work !
- you must indicate (cite) any sources used !
- randomly select students who need to explain their work

Rule II - Be Respectful

My personal wish is that this course provides a safe space for an enjoyable learning experience.

Any form of disrespectful behaviour, including any course-related communication platforms, will be sanctioned rigorously (including reporting to university authorities).

Table of Contents

Pre-Requisites and Learning Goals

Positioning in ML Curricula

Course Logistics

Introduction to FL

ML Basics Refresher

What is Federated Learning?

Federated Learning (FL) trains machine learning (ML) models in a distributed fashion over a network of devices

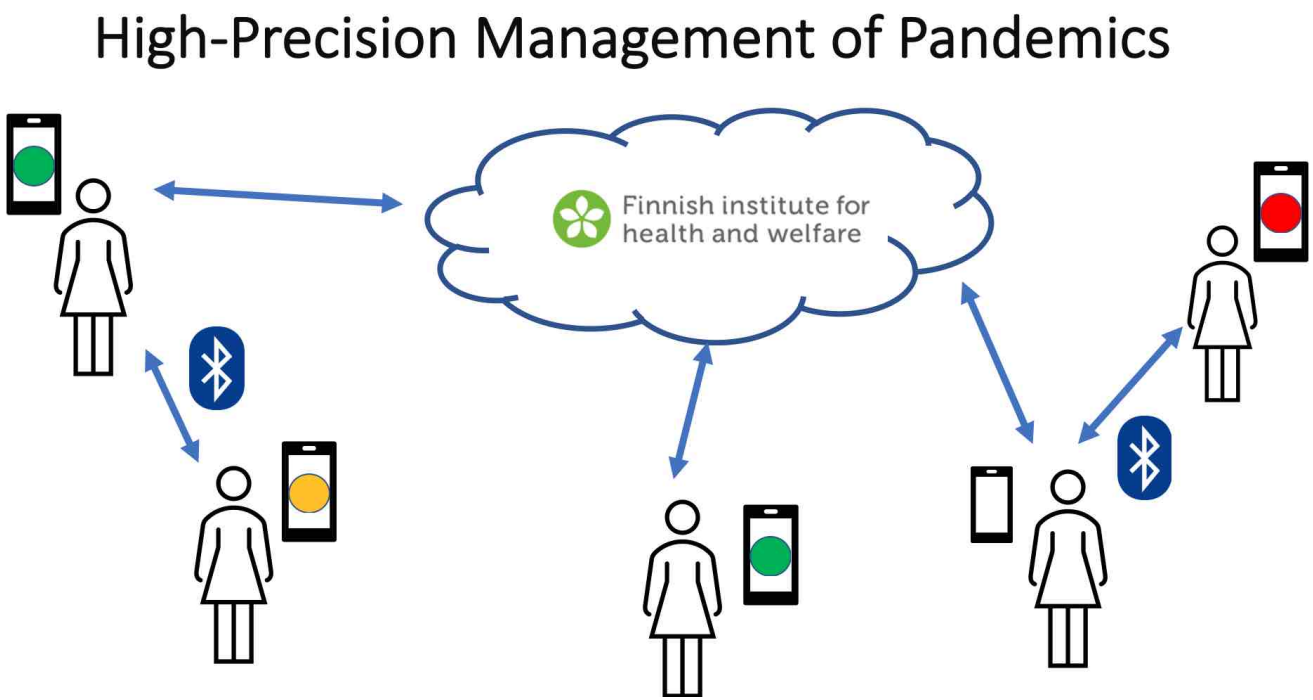
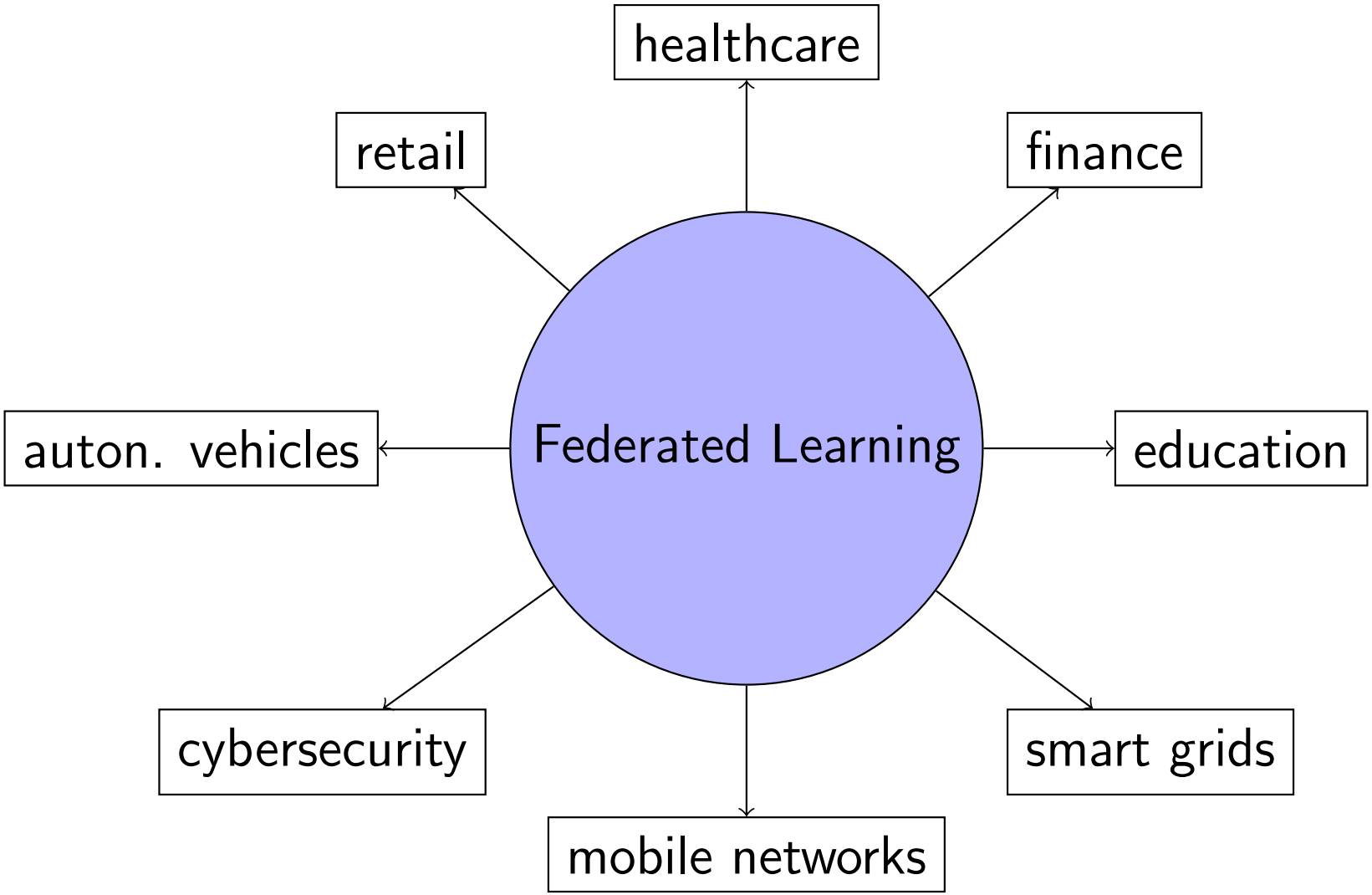


Figure: A Federated Learning system for prediction of infections. Smartphones train personalized models based on their observations (audio recordings of cough) as well as public health-care data.

Key Characteristics of FL

- no centralized data collection (robustness)
- each device trains personalized model (high-precision)
- share information/compute among devices (scalability)
- 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.

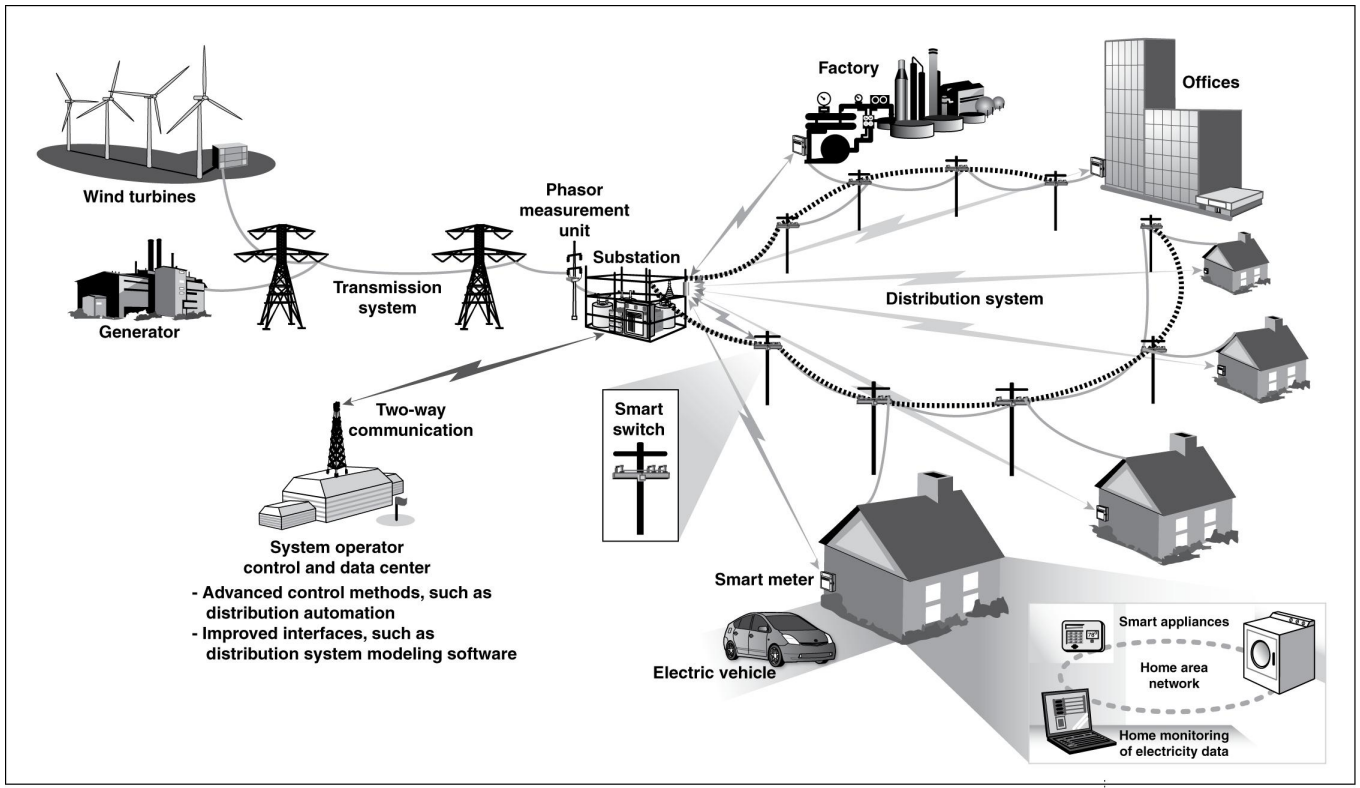
FL in Finance

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 in Smart Grids

use FL for high-precision demand/supply forecasting



Source: GAO analysis.

https://commons.wikimedia.org/w/index.php?curid=72267762

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 device i using decentralized data

ML with Python

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

FL with Python

IP: 192.168.0.1

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

IP: 192.168.0.2

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

IP: 192.168.0.3

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

FL $=$ ML $\otimes \ldots \otimes$ ML

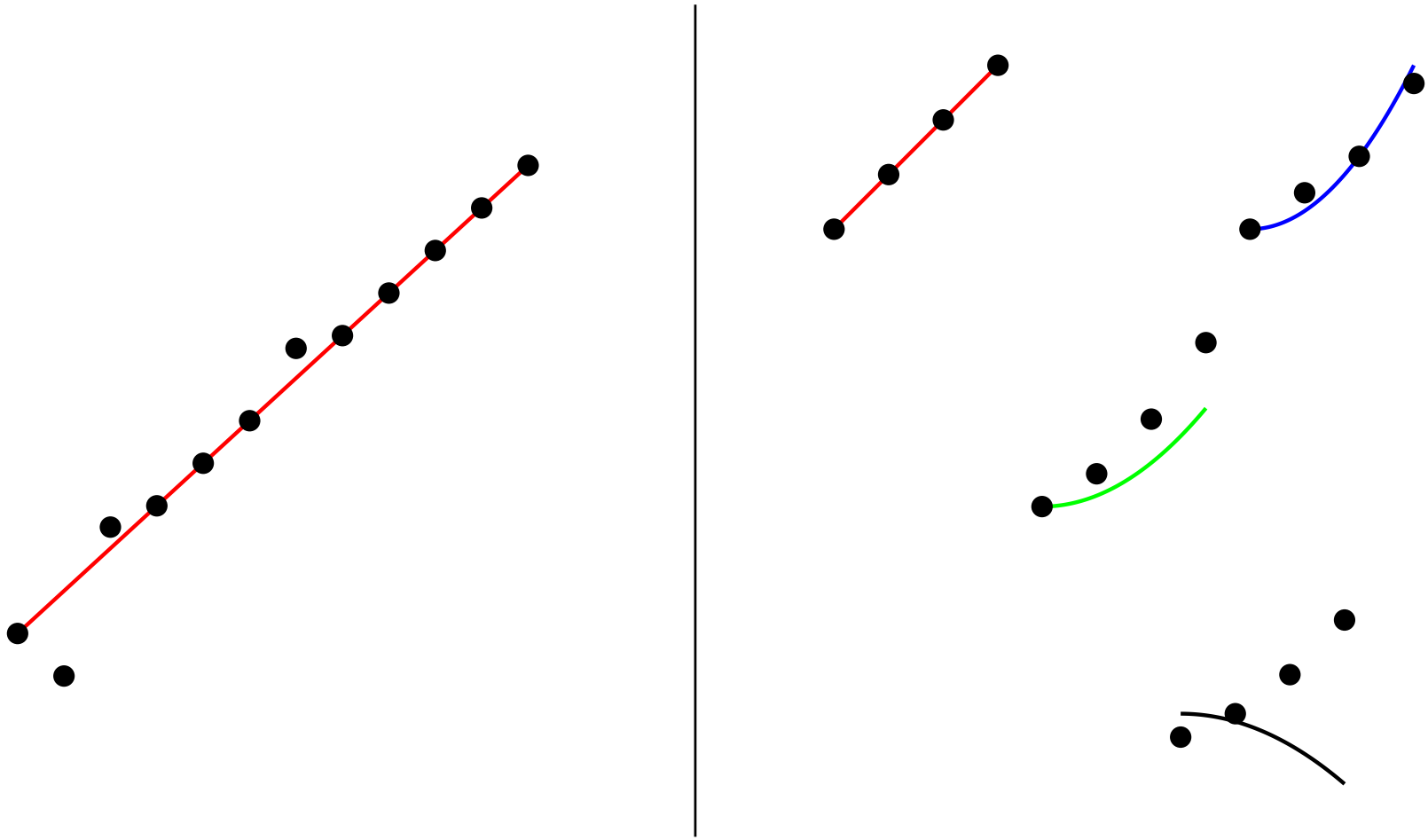


Figure: Left: A basic ML method uses a single dataset to train a single model. Right: FL methods train personalized models from decentralized datasets.

Table of Contents

Pre-Requisites and Learning Goals

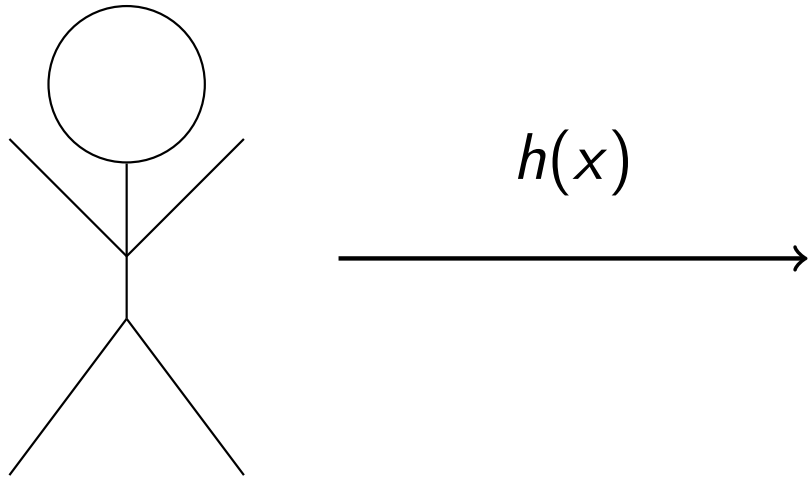
Positioning in ML Curricula

Course Logistics

Introduction to FL

ML Basics Refresher

The Right Song Can Save a Day



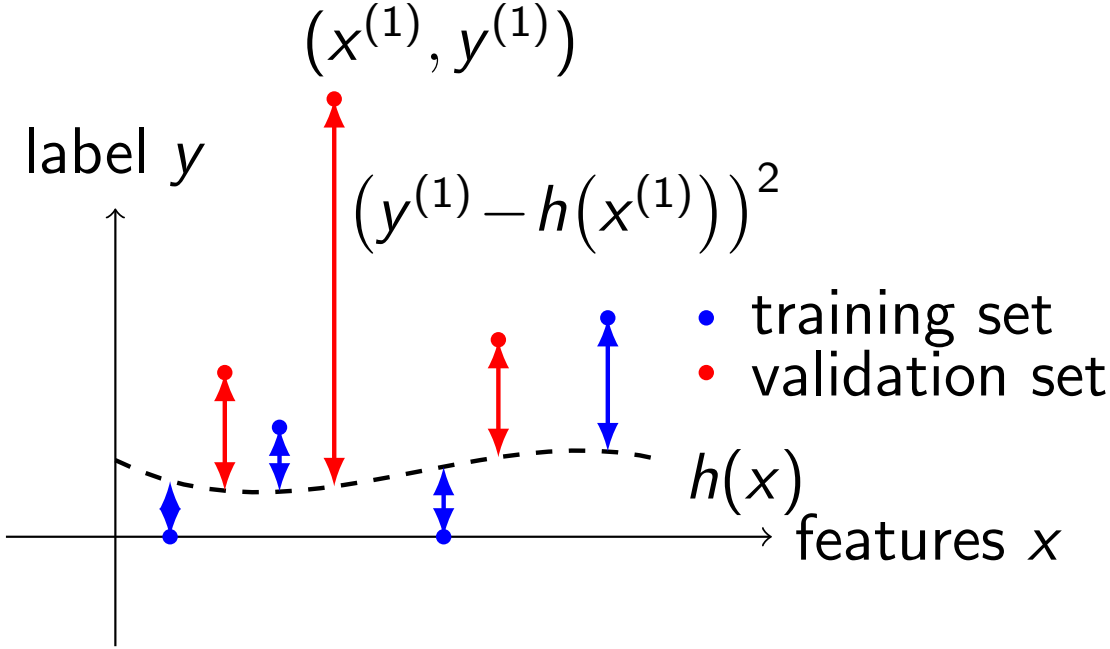
user: Alex Jung
mood: tired
evening

Bohemian Rhapsody

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)



ERM learns h from model $\mathcal{H}$ by min. average loss (empirical risk),

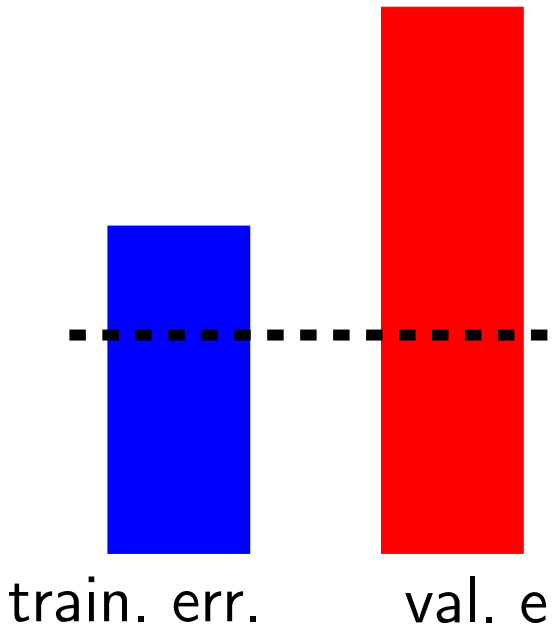
$$\min_{h \in \mathcal{H}} (1/m) \sum_{r=1}^{m} L\left(\left(\mathbf{x}, y\right), h\right)$$

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



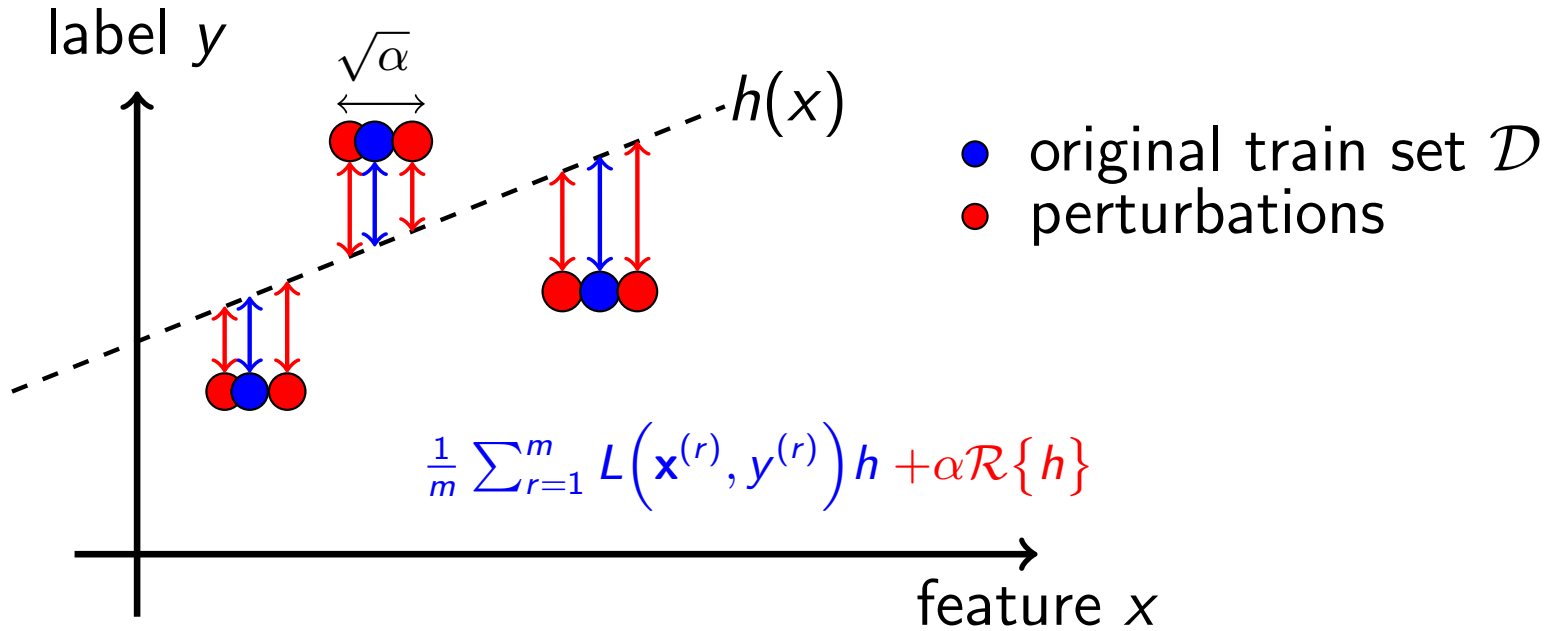
baseline or benchmark (e.g., Bayes' risk, other ML methods or human performance)

diagnose and select ML models by train. and val. err; if both are on same level as baseline (or benchmark) you are done !

---

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

Applied ML - Regularization



start with large model $\mathcal{H}$ and then reduce its effective size by

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

---

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

What's Next?

The next module introduces FL networks as our main mathematical model for FL applications.

Later modules use FL networks for the design and analysis of FL systems.