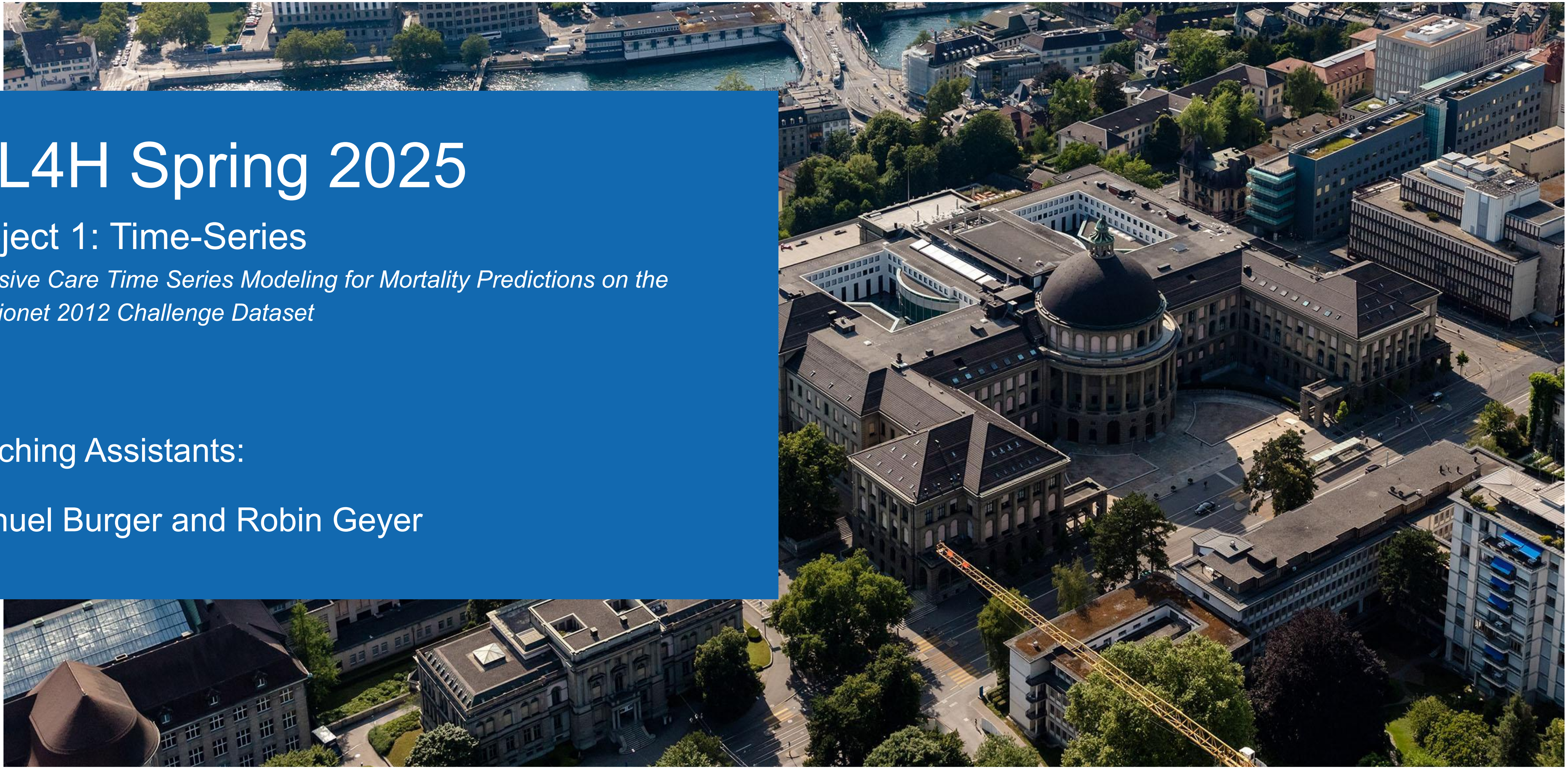
ETH zürich

BIOMEDICAL INFORMATICS



ML4H Spring 2025

Project 1: Time-Series

Intensive Care Time Series Modeling for Mortality Predictions on the Physionet 2012 Challenge Dataset

Teaching Assistants:

Manuel Burger and Robin Geyer

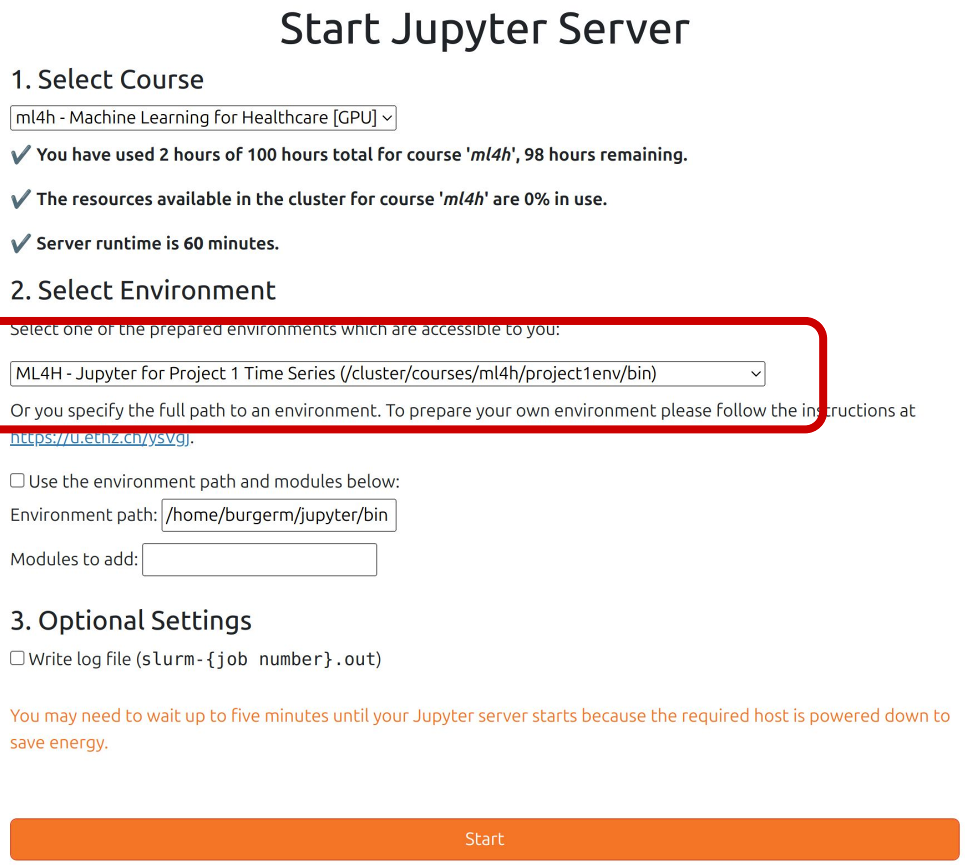
Cluster Usage

Login

- Connect to https://student-jupyter.inf.ethz.ch/ for notebooks only, this also gives shell access to a compute node via Jupyter Lab tabs
- For batch jobs and shell access to the login nodes use `ssh student-cluster`:

```
host student-cluster
  HostName student-cluster.inf.ethz.ch
  IdentityFile {Your-Key-File or remove line to use password}
  User {Your NETHZ user}
  ProxyJump eth # Easier if working from outside ETH network w/o VPN

host eth
    HostName jumphost.inf.ethz.ch
    User burgerm
    IdentityFile {Your-Key-File or remove line to use password}
```

Choose existing environment



Start Jupyter Server

1. Select Course

ml4h - Machine Learning for Healthcare [GPU]

✔ **You have used 2 hours of 100 hours total for course '*ml4h*', 98 hours remaining.**

✔ **The resources available in the cluster for course '*ml4h*' are 0% in use.**

✔ **Server runtime is 60 minutes.**

2. Select Environment

Select one of the prepared environments which are accessible to you:

ML4H - Jupyter for Project 1 Time Series (/cluster/courses/ml4h/project1env/bin)

Or you specify the full path to an environment. To prepare your own environment please follow the instructions at https://u.ethz.ch/ysVgj.

☐ Use the environment path and modules below:

Environment path: /home/burgerm/jupyter/bin

Modules to add:

3. Optional Settings

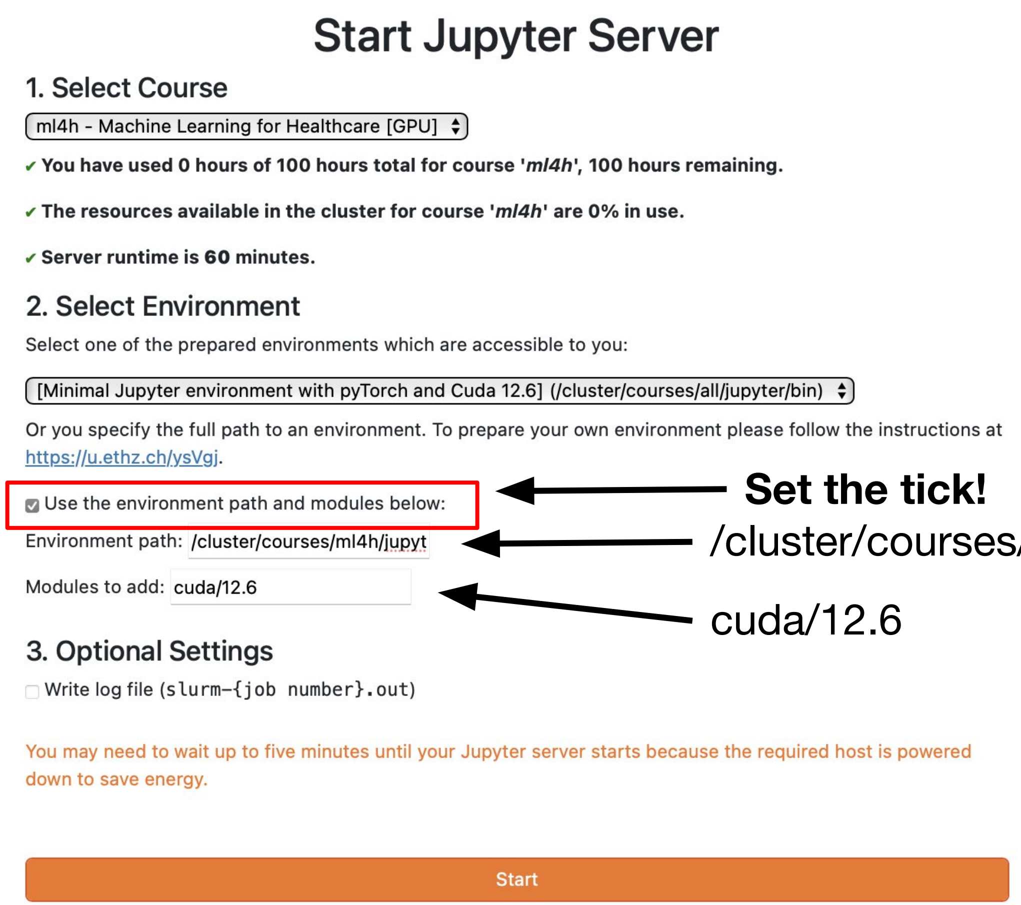
☐ Write log file (`slurm-{job number}.out`)

You may need to wait up to five minutes until your Jupyter server starts because the required host is powered down to save energy.

Start

If you miss a relevant package, let us know!

Create your own and launch the notebook using it



Set the tick!

/cluster/courses/ml4h/jupyter/bin

cuda/12.6

To create your own environment follow this [here]

Job Submissions

- You can also submit slurm jobs (more information here)
- More information on how to use Slurm here
- Custom python environment instructions here

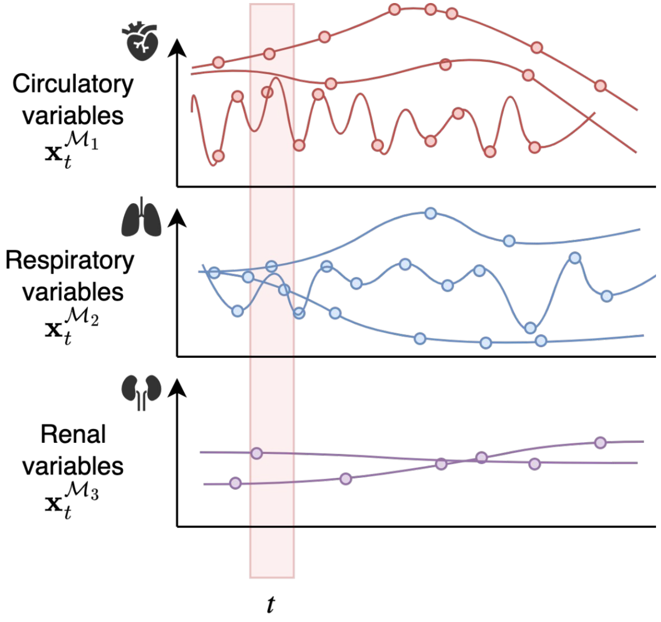
```
#!/bin/bash

#SBATCH --time=00:10
#SBATCH --account=ml4h
#SBATCH --output=nvidia-smi.out

module load cuda/12.6
source /cluster/courses/ml4h/jupyter/bin/activate
pip list | grep torch
nvidia-smi
```

Submit file to the cluster with:
`sbatch test.sh`

Project 1: Time Series

Intensive Care Units

Irregularly-sampled Multivariate Time-Series



- Target Dataset: [Physionet 2012 Challenge]
- Data from the first 48 hours of stay
- Goal: predict whether the patient gets discharged alive or dies inside the ICU
- 37 dynamic variables observed over time
- 4 static variables (Age, Gender, Height, Weight)

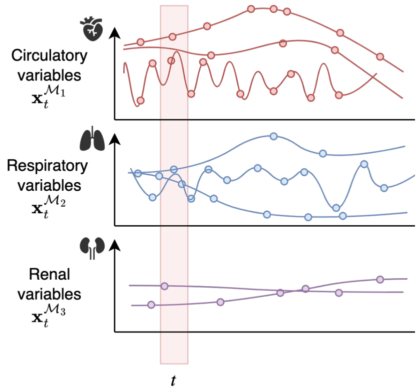
Deliverables

- Submit the report and code on Moodle: **07.04.25**
- Please make sure to state all group member names and Legi numbers on the report
- The **report** should be at most 4 pages (we encourage you to use the NeurIPS paper template here)
- The report has to be self-contained, i.e. no references to code.
- The report must be handed in as a PDF.
- Underlined sections within questions specify how many points can be achieved by solving that specific subquestion.
- You will also need to hand in your **code**. Please include a requirements.txt or similar for your Python environment and a README.md explaining how to run your code.
- Use train/validation (A/B) splits for training and tuning only. Report results on the test set (C). Note that the performance of the different methods can vary a lot.
- Using publicly available code is okay, but properly reference repositories when you use them. Of course, you are not allowed to use the code of other teams from the course.
- If not noted otherwise, report performances on the binary classification task using area under the curves (receiver-operator and precision-recall i.e. AuROC and AuPRC)

Outline

1. Data Processing and Exploration
2. Supervised Training
3. Representation Learning
4. Foundation Models
5. General Questions

1) Data Processing

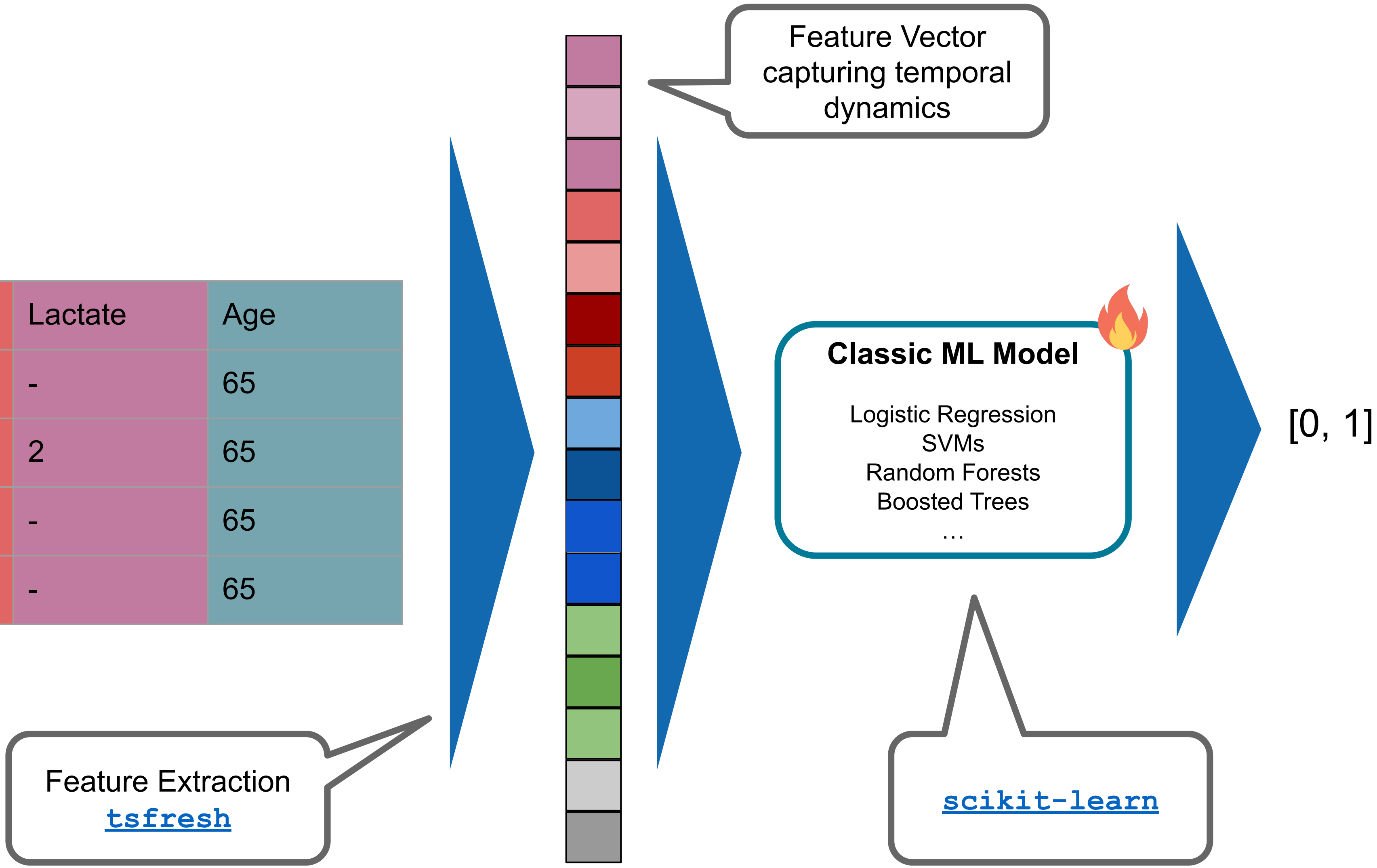


Time	Heart Rate	Lactate	Age
0h	120	-	65
1h	140	2	65
2h	-	-	65
3h	180	-	65

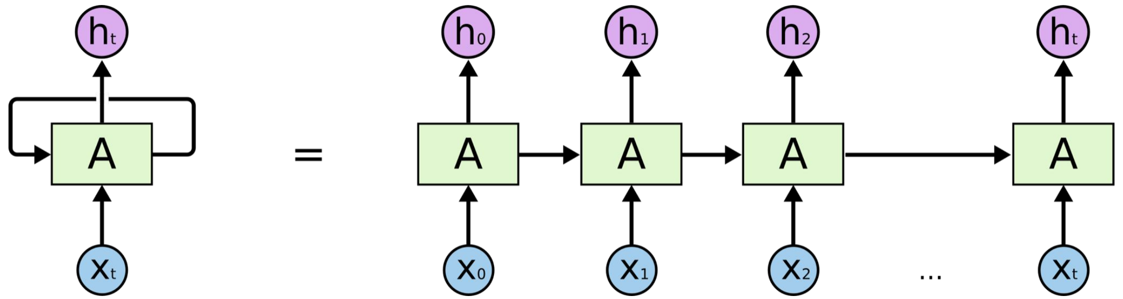
- Source data in home folder at `ml4h_data`
  - `Outcomes-{a,b,c}`
  - `set-{a,b,c}/{PatientID}.txt`
- Explore the dataset
- Prepare an hourly time-gridded data format
- Impute and scale the data
- **Careful:** always respect time

2) Supervised Learning - Classic ML Methods

Time	Heart Rate	Lactate	Age
0h	120	-	65
1h	140	2	65
2h	-	-	65
3h	180	-	65

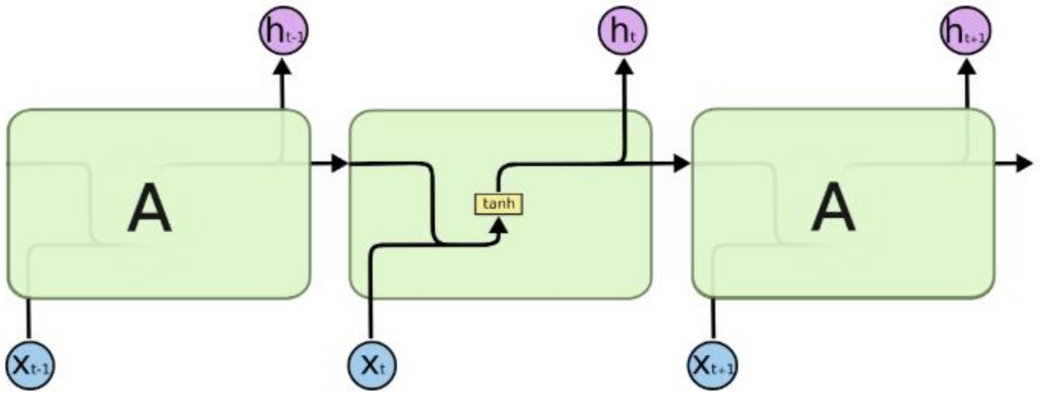


Recap on RNN/LSTM



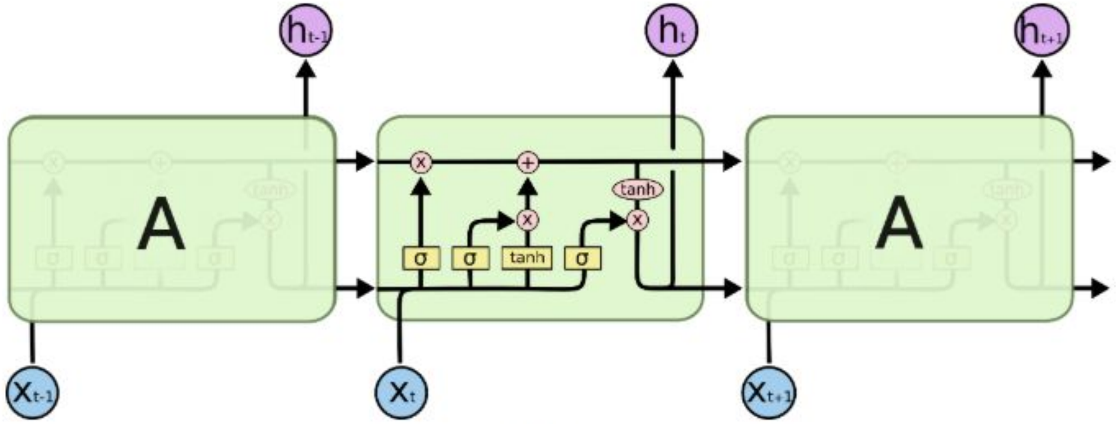
Recurrent network: the same network is applied to every time point. Memory is preserved by passing the hidden state to the successor.

RNN: vanishing gradient problem



The repeating module in a standard RNN contains a single layer.

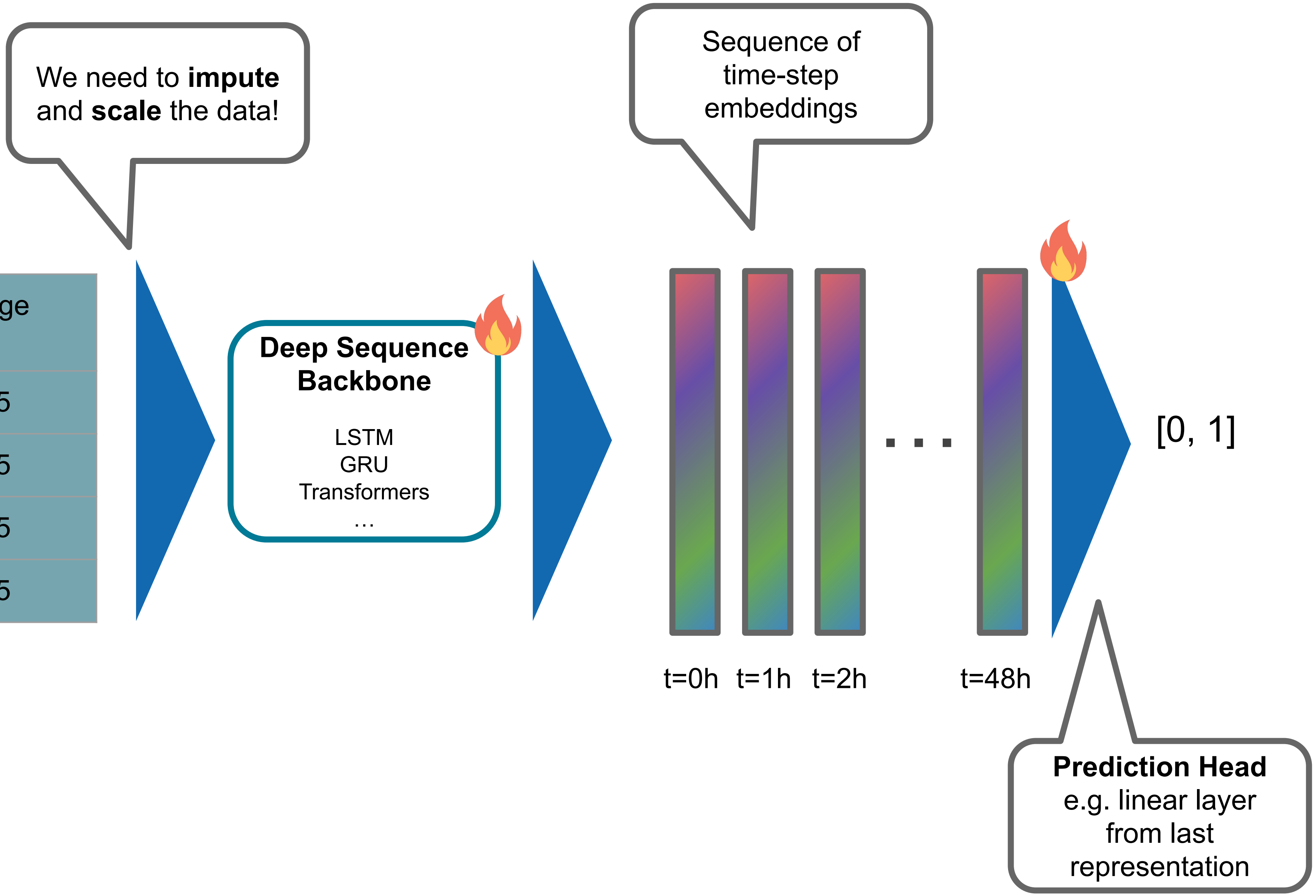
LSTM:



The repeating module in an LSTM contains four interacting layers.

2) Supervised Learning - Neural Networks

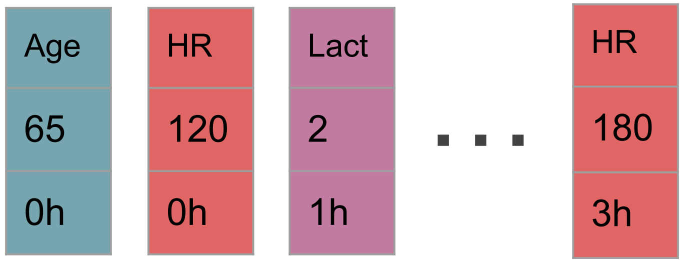
Time	Heart Rate	Lactate	Age
0h	120	-	65
1h	140	2	65
2h	-	-	65
3h	180	-	65



2) Supervised Learning - Tokenizing Time-Series

Time	Heart Rate	Lactate	Age
0h	120	-	65
1h	140	2	65
2h	-	-	65
3h	180	-	65

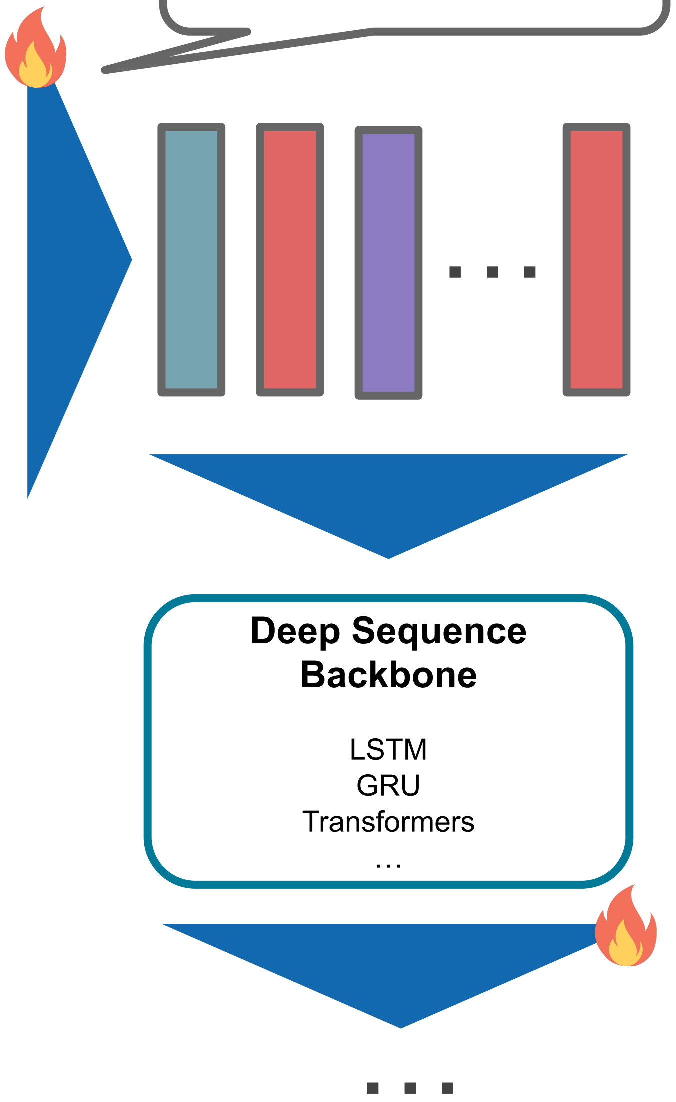
No imputation!



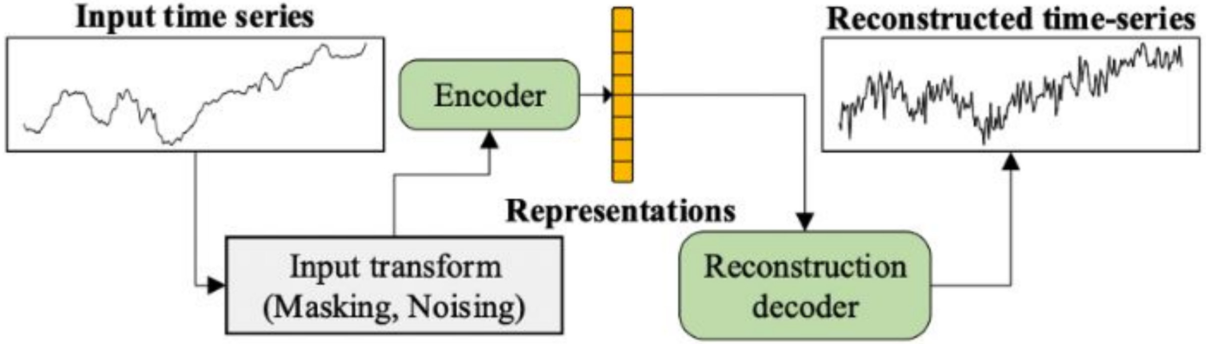
The goal: obtain a neural network friendly vector for each measurement:

- Horn et al.
- Gorishny et al.

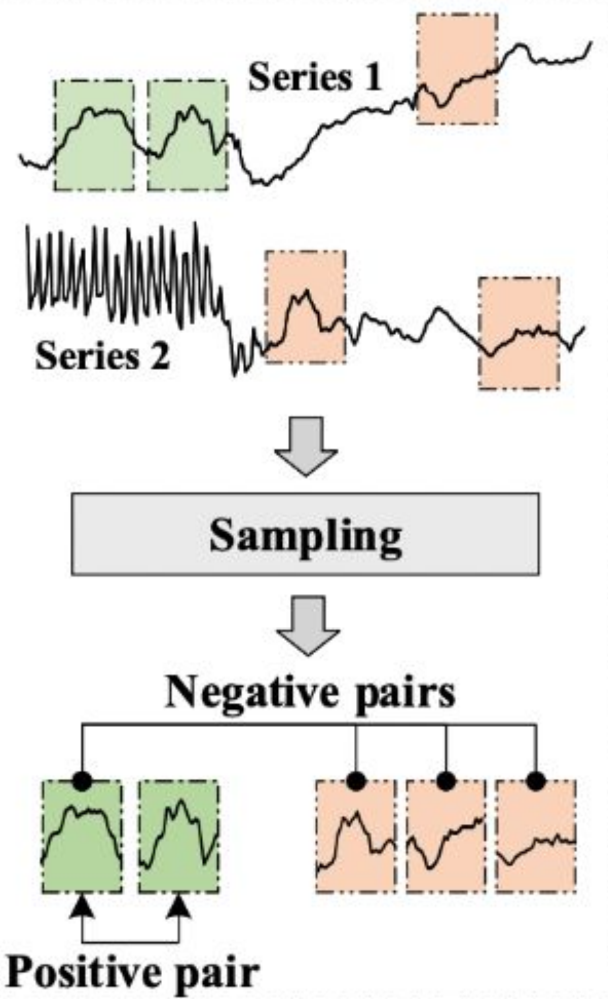
`torch.nn.Embedding` might be useful.



Representation Learning on Time-Series



Autoencoder-based approaches



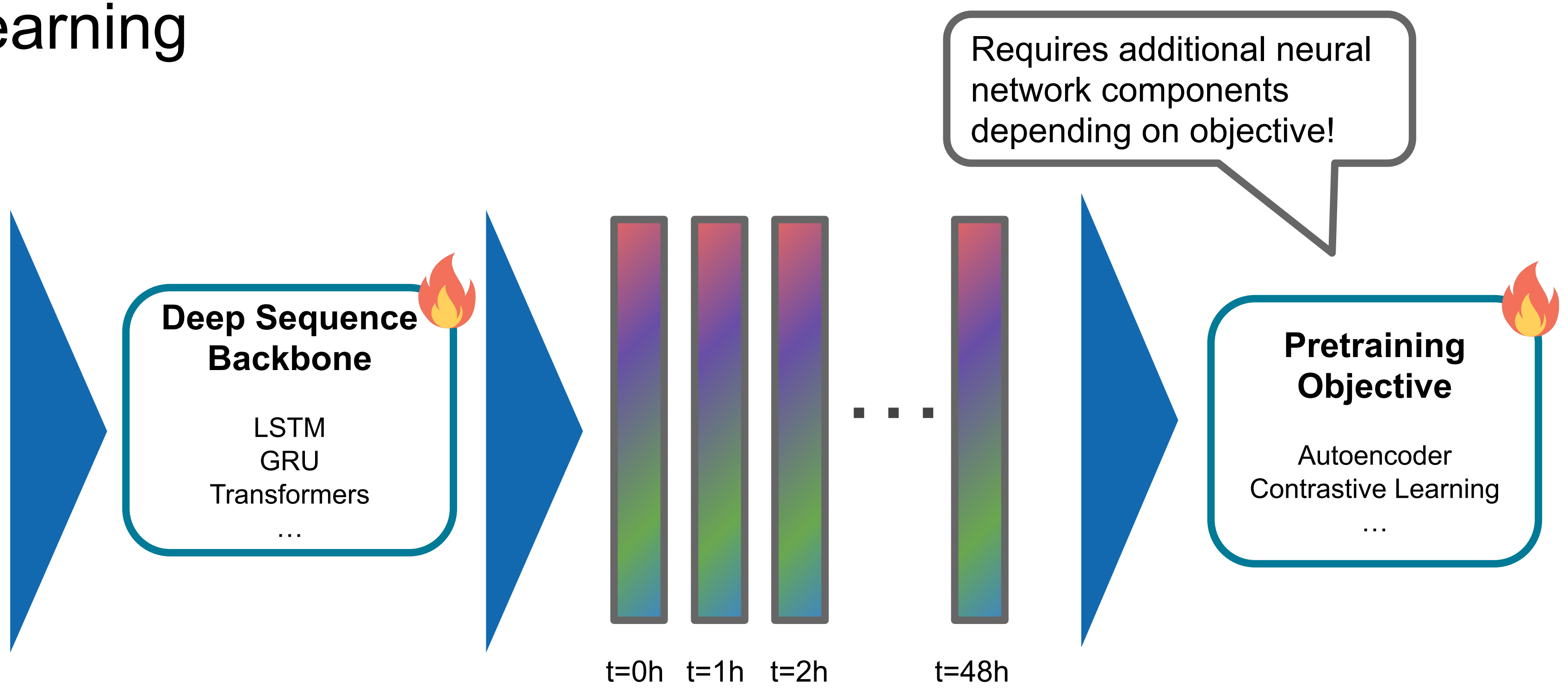
Contrastive approaches

Zhang et al., Self-Supervised Learning for Time Series Analysis: Taxonomy, Progress, and Prospects. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*. 2024.

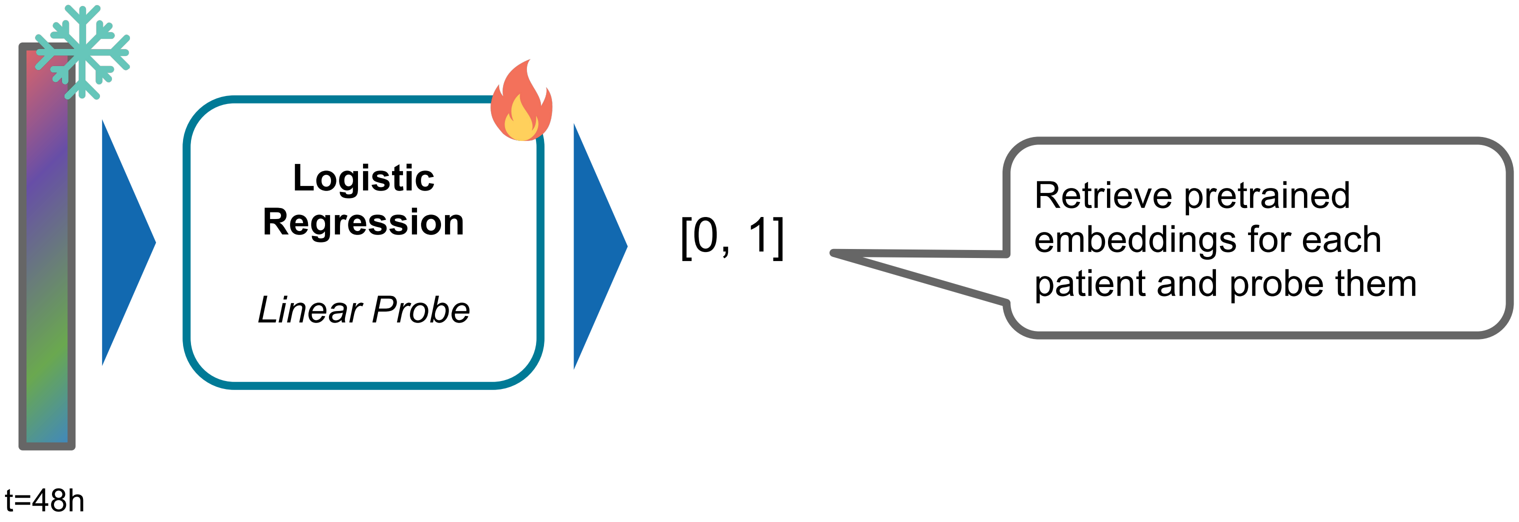
3) Representation Learning

Representation Learning

Time	Heart Rate	Lactate	Age
0h	120	-	65
1h	140	2	65
2h	-	-	65
3h	180	-	65

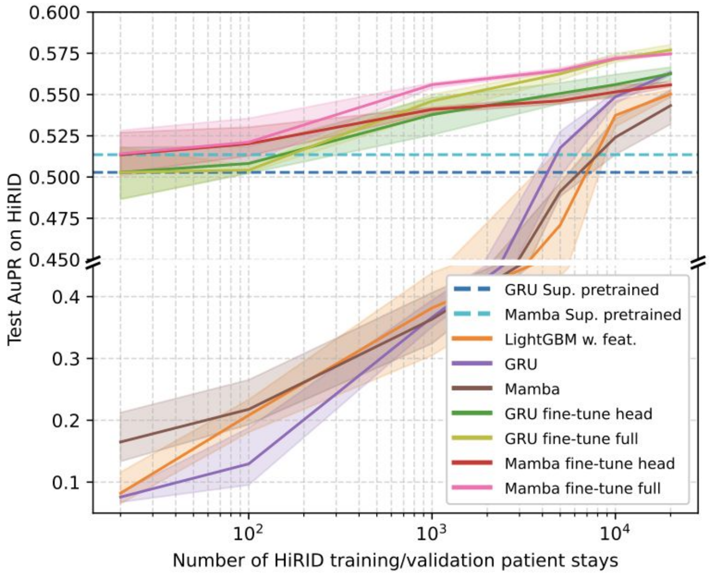


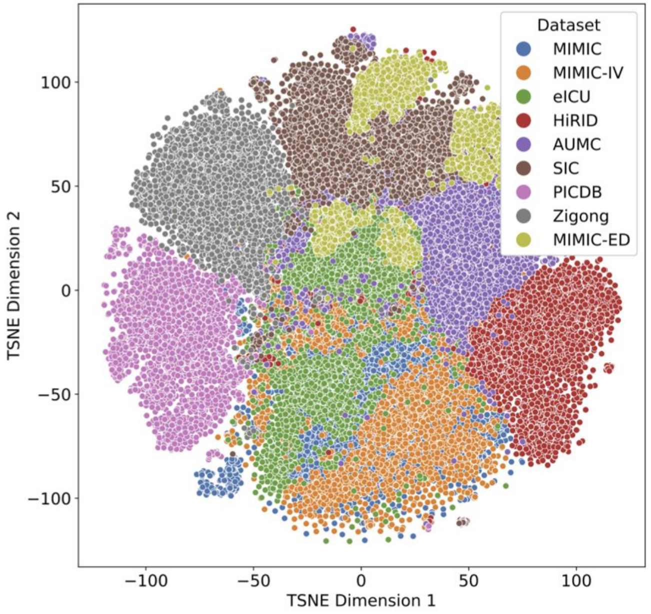
Evaluation



3) Representation Learning

- Simulate label scarcity
  - Train supervised models on fewer patients
  - Train linear probe on fewer patients
  - Plot both curves x=#{training patients} and y={full test set performance}
- Visualize latent space
  - Apply a dimensionality reduction and visualize the latent space
  - Compute a clustering metric to assess the clustering quality in reduced space





Plots: Burger et al. 2024

4) Foundation Models - LLMs and TS as text

Time	Heart Rate	Lactate	Age
0h	120	-	65
1h	140	2	65
2h	-	-	65
3h	180	-	65

“Will the given patient with age 65, max heart rate 180, … in the ICU be discharge alive, give a score from 1 to 10? ”

Pretrained LLM

[0, 1] / {0,1}

Feature extraction into natural language text space

Transform and parse the LLM output

Brief overview how to launch `ollama`

- Open a jupyter notebook on https://student-jupyter.inf.ethz.ch/
- Open a terminal tab and launch an ollama server:
  - `OLLAMA_MODELS=/cluster/courses/ml4h/llm/models /cluster/courses/ml4h/llm/bin/ollama serve`
  - `/cluster/courses/ml4h/llm/bin/ollama list` will show available models
- In a Jupyter Notebook you can now query LLMs

```
from ollama import chat

stream = chat(
    model='gemma2:2b',
    messages=[{'role': 'user', 'content': 'Why is the sky blue?'}],
    stream=True,
)

for chunk in stream:
  print(chunk['message']['content'], end='', flush=True)
```

4) Foundation Models - LLMs for embeddings

Foundation Model Representations

Time	Heart Rate	Lactate	Age
0h	120	-	65
1h	140	2	65
2h	-	-	65
3h	180	-	65

"Will the given patient with age 65, max heart rate 180, … in the ICU be discharge alive, give a score from 1 to 10? "

Pretrained LLM

Directly obtain the LLMs hidden state: `ollama.embed`

Evaluation

t=48h

Logistic Regression

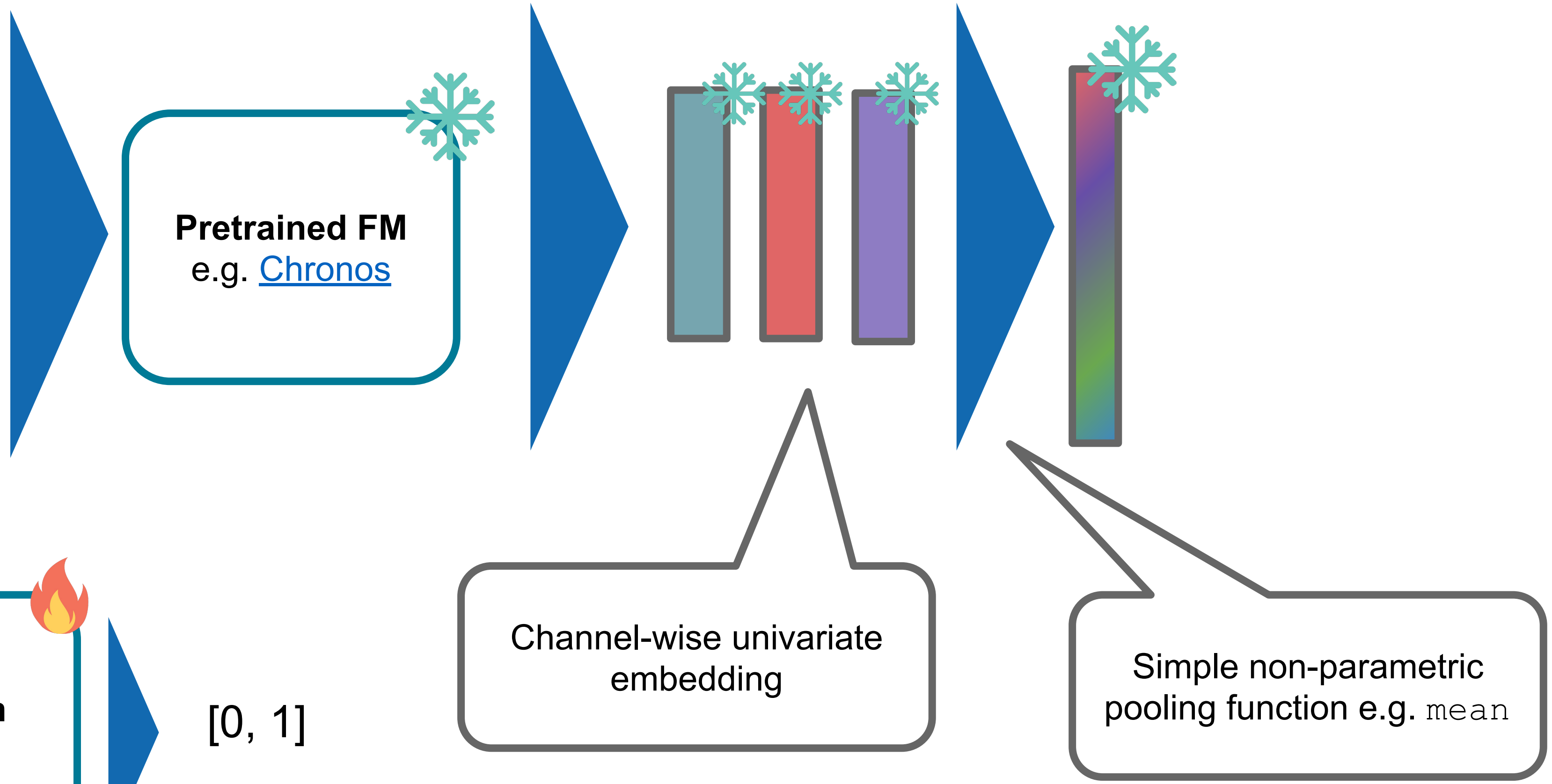
Linear Probe

[0, 1]

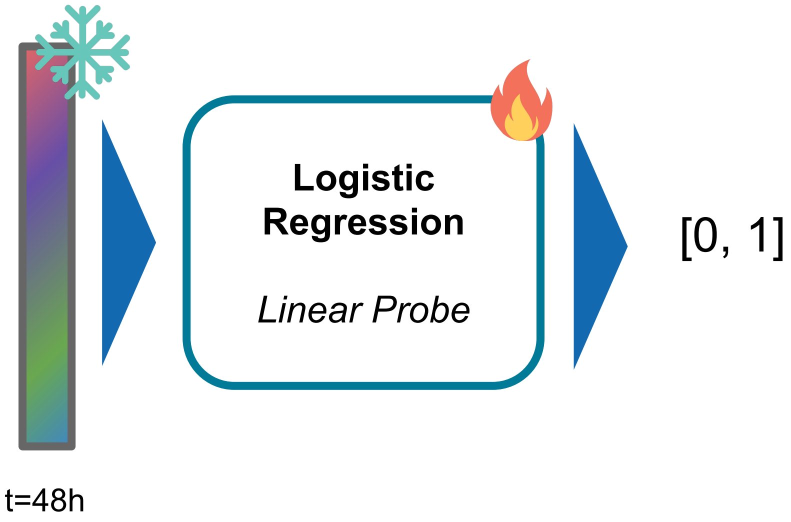
4) Foundation Models - Time-Series Models

Foundation Model Representations

Time	Heart Rate	Lactate	Age
0h	120	-	65
1h	140	2	65
2h	-	-	65
3h	180	-	65



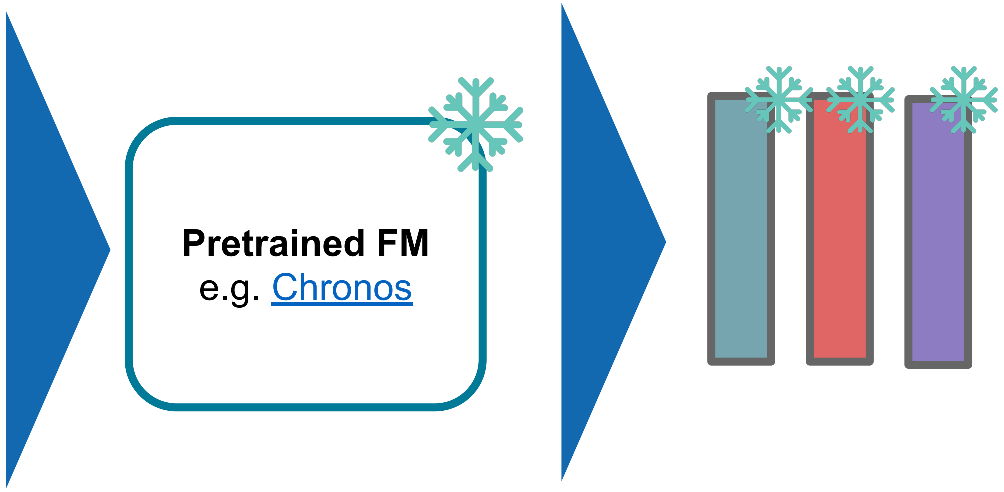
Evaluation



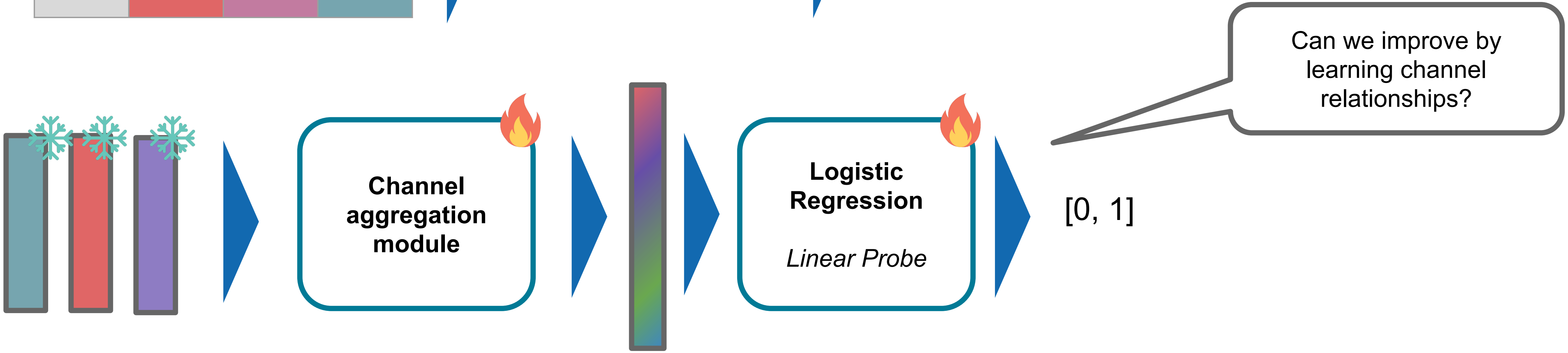
4) Foundation Models - Time-Series Models

Foundation Model Representations

Time	Heart Rate	Lactat e	Age
0h	120	-	65
1h	140	2	65
2h	-	-	65
3h	180	-	65



Evaluation



t=48h

5) General Questions (6 Pts)

- **Q5.1:** There are many machine learning settings where classic methods are still competitive with deep learning architectures. Have you observed this in this project? Why is this (not) the case? (2 pts)
- **Q5.2:** Can you think of an attention-related bottleneck regarding very (very) long time series? Conceptually, which deep methods from above are more suitable for such long time series? (2 pt)
- **Q5.3:** What are some challenges in using self-supervised representation learning? What difficulties have you observed in your approach? Can you think of additional ones? (2 pt)

ETH zürich

BIOMEDICAL INFORMATICS

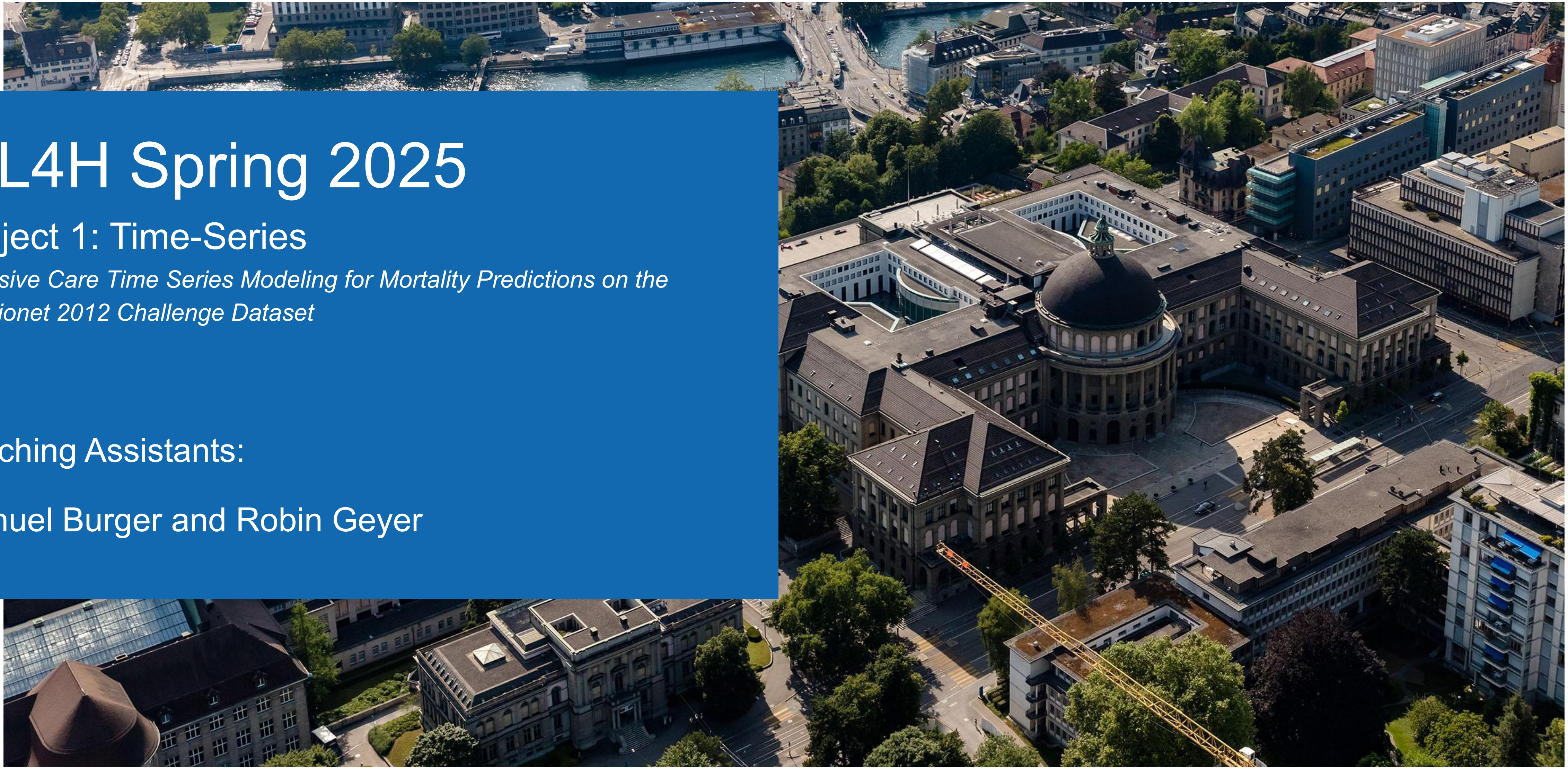
ML4H Spring 2025

Project 1: Time-Series

Intensive Care Time Series Modeling for Mortality Predictions on the Physionet 2012 Challenge Dataset

Teaching Assistants:

Manuel Burger and Robin Geyer



References

- Used a bit of flaticons: https://www.flaticon.com/