Data Science Workflow

Lecture 4: HPO and Evaluation

M.Sc. Teodor Chiaburu

FB 6, Data Science, Berliner Hochschule für Technik

26. Juni 2025

Monitoring Tools

W&B

We will learn to monitor our model training online using **Weights & Biases (WandB)**.

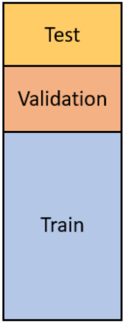
Monitoring

Task 1: Create an account on WandB. It's easiest to sign up with your GitHub account.

Task 2: Launch a pod with GPU on the cluster, connect to it and open a remote VSCode window. Install *wandb* in your Python environment in your PV: *pip install wandb*

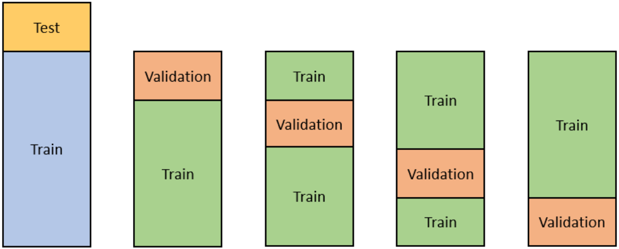
Task 3: Under *Lecture_4* check *main_wandb.py*, which logs the training and validation IoU live on WandB while your model is training. Run it and watch the iterations live on WandB. Look into this tutorial for more customization.

Simple CV



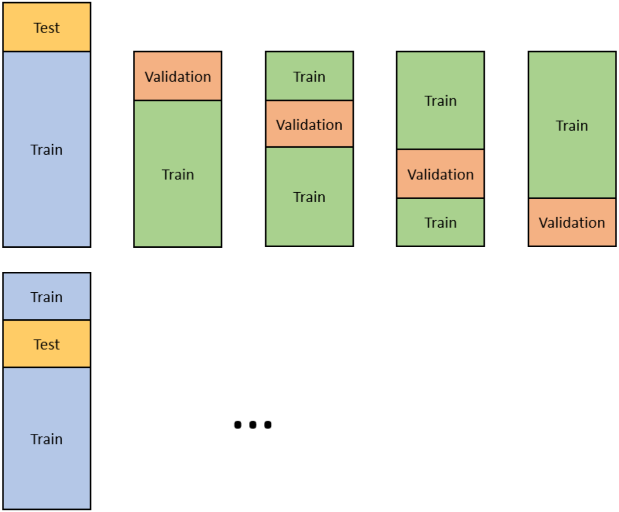
For the project: minimal requirement! All teams need to split their data into train/validation/test sets.

K-fold CV



For the project: you get extra points if you do this.

K-fold Nested CV



For the project: let's not exaggerate :-) Invest time in other aspects instead of nesting your cv.

We will learn to fine-tune hyperparameters using **Optuna**.

HPO

Task 1: In the remote VSCode window: Install *optuna* in your Python environment in your PV: *pip install optuna*

Task 2: Under *Lecture_4* check *main_wand_optuna.py* which optimizes the learning rate and weight decay with Optuna. Run the script and again watch the live updates on WandB. Check the Optuna docs for more examples.

Note (also relevant for the project): Once you have found your best hyperparameters, train the best model again on the concatenated training + validation set and then evaluate on the test set.

HPO Strategies

Samplers

- Grid Search
- Random Search
- Bayesian
- Check more here

Pruners

- Median
- HyperBandit
- Check more here

Speeding up Your Workflow in VSCode

You may have noticed that you have to type in some of your credentials every time you connect to a new pod e.g. git username or wandb token. You can speed this up by either:

1. Copying the credentials in the Dockerfile and rebuilding the image.
2. Writing yourself a shell script with the credentials commands and running it every time you're in a new pod (which is much faster than typing everything again by hand).

Bonus task: You have a template shell script in Moodle called *additional_configs.sh*. Replace the placeholders with your credentials. Don't push this file to GitHub, since it has sensitive information. You should add it instead to *.gitignore* and it's going to persist only in your PV. Also note that you can't execute the file immediately after creating it (check *ls -la*). You need to mark it as executable first. How do you do this?

Extras to Complete Your ML Workflow

- **User Study**: Cognition
  - A free online platform to conduct human-in-the-loop (HIL) experiments
  - Uses *jsPsych*, a framework in JavaScript
  - Check out an example on the bee dataset. Here, we wanted to test whether explanations generated by various XAI methods had a didactic effect on laypersons. Spoiler: turned out, they didn't. Check our paper.

- **Simple App**: Streamlit
  - An easy way to deploy apps in Python online

- **Professional Docs**: Sphinx & Read the Docs
  - A tool for building source code documentation like the pros
  - See an example here from the KInsecta project.