Applied Machine Learning

Benchmarking:
Large-Scale Benchmarking & OpenML

OpenML

Learning goals

- Why Benchmarking?
- What is OpenML?

WHY BENCHMARKING

It is hard to evaluate methods using theoretical/mathematical analysis alone.

- Theoretical analysis typically requires strong assumptions (e.g., infinite sample size, no noise, ideal model class).
- Implementation details cannot be formally compared through purely theoretical/mathematical analysis (e.g., runtime, memory use).
- Effect of different data properties on algorithms cannot be explained theoretically, e.g., does XGBoost outperform an LM on data with only categorical features?
- Hyperparameter settings are often not part of the theoretical analysis.

THE STEPS OF BENCHMARKING

1. Formulate a hypothesis (about algorithm performance or behavior)
2. Define an appropriate performance metric (if not part of the hypothesis)
3. Select or collect datasets to evaluate the hypothesis
4. Define a resampling strategy
5. Define hyperparameter search space and tuning strategy
6. Execute the full cross-product of learners, datasets, and tuning configurations
7. Analyze results and draw conclusions

THE PROBLEM WITH DATASETS

- No universally adopted standard dataset format
- No universally adopted dataset sharing platform
- Few platforms offer programmatic (API-based) access to data

THE IDEAL WORLD - FRICTIONLESS ML

Datasets

- Unified access to all datasets via unified API
- Possibility to share own data (mostly relevant in academic context)
- Discover and search datasets via their meta-data

Associated Objects

- **Tasks**: Predefined train/test splits for standardized evaluation
- **Runs**: Publicly shared results (algorithm, settings, performance)

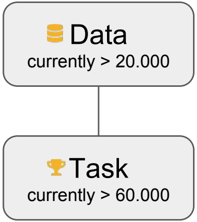
OPENML - THE PROJECT ▸ *OpenML.org* n.d.

▸ *OpenML.org* n.d. is not only a data repository, it is a collaborative ML platform for sharing individual **components** involved in benchmark experiments.

Benchmarking: compare algorithms w.r.t. performance/runtime on datasets.

Goal: Reproducibility, transparency, and large-scale collaboration in ML.

OpenML relies on 4 **basic components** (just like benchmark experiments):



Supervised ML **tasks** contain:

- data + target feature
- performance measure
- resampling technique (e.g., 10-fold CV) and train-test splits

OPENML - THE PROJECT ▸ *OpenML.org* n.d.

▸ *OpenML.org* n.d. is not only a data repository, it is a collaborative ML platform for sharing individual **components** involved in benchmark experiments.

Benchmarking: compare algorithms w.r.t. performance/runtime on datasets.

Goal: Reproducibility, transparency, and large-scale collaboration in ML.

OpenML relies on 4 **basic components** (just like benchmark experiments):

Data
currently > 20.000

Task
currently > 60.000

Supervised ML **tasks** contain:

- data + target feature
- performance measure
- resampling technique (e.g., 10-fold CV) and train-test splits

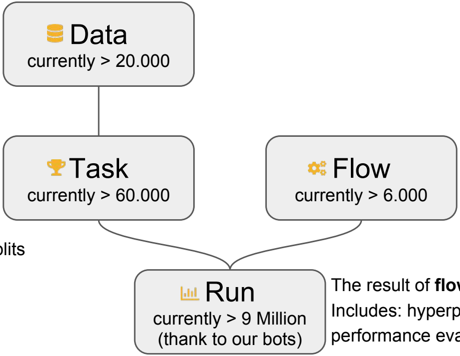
Flow
currently > 6.000

A **flow** can be a ML algorithm or a complex ML pipeline that solves a **task**.

OPENML - THE PROJECT ▸ *OpenML.org* n.d.

▸ *OpenML.org* n.d. is not only a data repository, it is a collaborative ML platform for sharing individual **components** involved in benchmark experiments.
Benchmarking: compare algorithms w.r.t. performance/runtime on datasets.
Goal: Reproducibility, transparency, and large-scale collaboration in ML.
OpenML relies on 4 **basic components** (just like benchmark experiments):

Supervised ML **tasks** contain:
- data + target feature
- performance measure
- resampling technique (e.g., 10-fold CV) and train-test splits



A **flow** can be a ML algorithm or a complex ML pipeline that solves a **task**.

The result of **flows** if applied to **tasks**. Includes: hyperparameters, predictions, performance evaluation, and runtime.

Note: *OpenML also supports **collections** of tasks and runs*

OPENML DATASETS

- **Raw data** stored in standardized formats:
  - `.arff` (Attribute-Relation File Format): plain-text format with inline schema
  - `.parquet`: efficient binary columnar storage (preferred for large data)
- **Meta-data** annotations include:
  - Dataset description (source, intended use, licensing)
  - Feature names and types (numeric, categorical, ordinal, date, string)
  - Designated roles: input features, target variable, ignored columns
- **Automatically extracted information:**
  - Meta-features (e.g., number of instances, number of features)
  - Statistical summaries (missing values, sparsity, imbalance)
  - Baseline performance of simple models (e.g., majority class classifier)

OPENML TASKS

- A **task** defines a concrete ML problem on a dataset, e.g., in supervised ML:
  - Target feature (to be predicted)
  - Resampling strategy (e.g., 10-fold CV, holdout)
  - Evaluation metric (e.g., accuracy, RMSE)
- Tasks ensure reproducibility by fixing all relevant components
- **Flows** (i.e., ML pipelines or learners) are executed on *tasks* and produce *runs*

OPEN QUESTION: WHAT DATASETS/TASKS TO USE?

- Hard to compare results across papers
- Benchmarking is often done on small set of datasets
  - Question about generalization to other datasets
  - Cherry picking or arbitrary selection
  - Different versions, different train-test setups
- Publication bias
  - Published papers report good results
  - Interesting to know WHEN an algorithm works (and when it doesn’t)

OPENML BENCHMARKING SUITES

- **Benchmarking suite:** curated collection of standardized OpenML tasks
- Provide unified, shareable task definitions with fixed resampling and target
- Accessible via APIs in `R`, `Python`, and `Java`
- Enable reproducible and comparable experiments toward shared research goals

EXAMPLE I: OPENML-CC18 (CLASSIFICATION)

▸ “Bischl et al.” 2021

Standardized benchmark suite for evaluating supervised classification algorithms.

- 72 classification tasks
- Medium size: 500–100,000 observations, $\leq$ 5000 features (post encoding)
- Contains missing values and categorical features
- Excludes:
  - Strong class imbalance
  - Time series or group structure
  - Sparse data and free-form text
- Follows objective and subjective selection criteria (see paper)

EXAMPLE II: OPENML-CTR23 (REGRESSION)

▸ "Fischer et al." 2023

Extension of CC18 selection principles to regression tasks.

- Uses same structural filters as CC18
- Includes only datasets with numeric targets and ≥ 5 distinct values
- Excludes:
  - Datasets trivially solved by linear models
  - Datasets with ethical/legal concerns
  - Datasets restricted from public benchmarking

EXAMPLE III: AUTOML BENCHMARK SUITE (AMLB)

▸ **“Gijsbers et al.” 2024**

Designed to evaluate modern AutoML systems on realistic and diverse problems.

- 71 classification tasks and 33 regression tasks
- Stricter inclusion criteria for difficulty and real-world complexity
- Covers multiple domains; avoids trivial, overly cleaned, or synthetic data
- Excludes:
  - Free-text features
  - Time dependencies or metadata leaks

WHY PARALLELIZATION MATTERS

Benchmark Scenario (AutoML Study)

- 71 datasets $\times$ 10-fold cross-validation
- 9 AutoML systems evaluated
- 1 hour per system-dataset–fold combination

$\Rightarrow$ Total runtime: **6390 hours** = **266+ days** sequentially

Conclusion: *Benchmarking at scale is infeasible without parallelization.*