Introduction to Machine Learning

Introduction: Learners

compstat-lmu.github.io/lecture_i2ml

COMPONENTS OF A LEARNER

Summarizing what we have seen before, nearly all supervised learning algorithms can be described in terms of three components:

Learning = Hypothesis Space + Risk + Optimization

- **Hypothesis Space:** Defines (and restricts!) what kind of model f can be learned from the data.
- **Risk:** Quantifies how well a specific model performs on a given data set. This defines how to compare observed values to predictions and allows us to rank candidate models in order to choose the best one.
- **Optimization:** Defines how to search for the best model in the **hypothesis space**, typically guided by the metric used for the **risk**.

SUPERVISED LEARNING, FORMALIZED

A **learner** (or **inducer**) is a program which

- receives a **training set** $\mathcal{D} \in \mathcal{X} \times \mathcal{Y}$
- and uses an **optimization** procedure to find

$$\hat{f} = \arg\min_{f \in \mathcal{H}} \mathcal{R}_{\text{emp}}(f)$$

- for a given **hypothesis class** $\mathcal{H}$ of **models** $f : \mathcal{X} \to \mathbb{R}^g$,
- based on a **risk** function $\mathcal{R}_{\text{emp}}(f)$ that quantifies the performance of $f \in \mathcal{H}$ on $\mathcal{D}$.

It is important to understand that learners and models are not the same thing: rather, a learner *returns* a model.

(This does not cover all special cases, but it's a useful framework for most supervised ML problems.)

LEARNING AS EMPIRICAL RISK MINIMIZATION

By decomposing learners into these components,

- we have a framework to understand how they work,
- we can more easily evaluate in which settings they may be more or less suitable, and
- we can tailor learners to specific problems by clever choice of each of the three components.

EXAMPLES FOR LEARNING COMPONENTS

Hypothesis Space :
- Step functions
- Linear functions
- Sets of rules
- Neural networks
- Voronoi tesselations
- ...

Risk :
- Mean squared error
- Misclassification rate
- Negative log-likelihood
- Information gain
- ...

Optimization :
- Analytical solution
- Gradient descent
- Combinatorial optimization
- Genetic algorithms
- ...