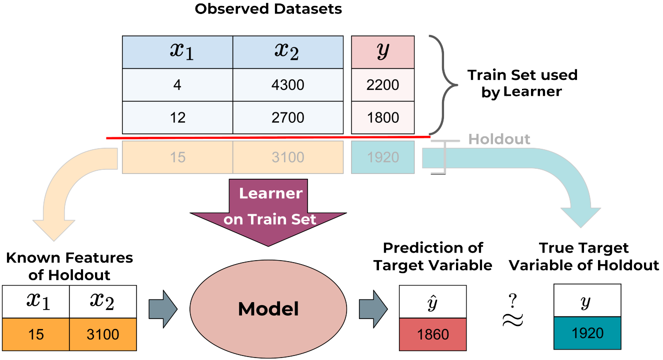
Introduction to Machine Learning

Introduction: Losses & Risk Minimization

compstat-lmu.github.io/lecture_i2ml

HOW TO EVALUATE MODELS

Compare predictions from a model with observed target values:



MOTIVATION

- Assume we trained a model to predict flat rent based on some features (size, location, age, ...).
- The real rent of a flat is EUR 1600, our model predicts EUR 1300.
- How do we measure the performance of our model?
- Need to define a suitable criterion, e.g.:
  - Absolute error $|1600 - 1300| = 300$
  - Squared error: $(1600 - 1300)^2 = 90000$
    (puts more emphasis on predictions that are far off the mark)
- The choice of this metric has a major influence on the final model, because it determines what constitutes a *good* model: it will determine the ranking of the different models $f \in \mathcal{H}$.
- the metric we use is called the **loss function**.

RISK MINIMIZATION

- The "quality" of a prediction $f(\mathbf{x})$ is measured by a **loss function** $L(y, f(\mathbf{x}))$ that quantifies how "close" $f(\mathbf{x})$ is to y.
  For example, $L(y, f(\mathbf{x})) = |f(\mathbf{x}) - y|$.
- The ability of a model f to reproduce the association between x and y that is present in the data $\mathcal{D}$ can be measured by the average loss: the **empirical risk**

$$\mathcal{R}_{\text{emp}}(f) = \frac{1}{n} \sum_{i=1}^{n} L\left(y^{(i)}, f\left(\mathbf{x}^{(i)}\right)\right).$$

- Learning – finding the "best" model – then amounts to **empirical risk minimization**:
  figuring out which model f has the smallest average loss:

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

RISK MINIMIZATION

Since the model f is usually defined by **parameters** θ in a parameter space Θ, this becomes:

$$\begin{aligned}
\mathcal{R}_{\text{emp}}(\theta) &= \frac{1}{n}\sum_{i=1}^{n} L\left(y^{(i)}, f\left(\mathbf{x}^{(i)} \mid \boldsymbol{\theta}\right)\right) \\
\hat{\theta} &= \operatorname*{arg\,min}_{\theta \in \Theta} \mathcal{R}_{\text{emp}}(\theta)
\end{aligned}$$

Most learners in ML try to solve the above *optimization problem*, which implies a tight connection between ML and optimization.

RISK MINIMIZATION

- For regression tasks, the loss often only depends on the residual $L\left(y, f(\mathbf{x})\right) = L(y - f(\mathbf{x})) = L(\epsilon)$.
- The choice of loss implies which kinds of errors are important or not – requires *domain knowledge*!
- For learners that correspond to probabilistic models, the loss determines / is equivalent to distributional assumptions.
- Since learning can be re-phrased as minimizing the loss, the choice of loss strongly affects the computational difficulty of learning:
  - How smooth is $\mathcal{R}_{\text{emp}}(\theta)$ in θ?
  - Is $\mathcal{R}_{\text{emp}}(\theta)$ differentiable so that we can use gradient-based methods?
  - Does $\mathcal{R}_{\text{emp}}(\theta)$ have multiple local minima or saddlepoints over Θ?