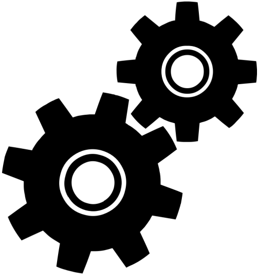
Introduction to Machine Learning

Chapter 1: Introduction

Bernd Bischl, Christoph Molnar, Daniel Schalk, Fabian Scheipl

Department of Statistics – LMU Munich

DATA SCIENCE AND MACHINE LEARNING



Machine learning is a branch of computer science and applied statistics covering algorithms that improve their performance at a given task based on sample data.

MACHINE LEARNING IS CHANGING OUR WORLD

- Search engines learn what you want
- Recommender systems learn your taste in books, music, movies,...
- Algorithms do automatic stock trading
- Google Translate learns how to translate text
- Siri learns to understand speech
- DeepMind beats humans at Go
- Cars drive themselves
- Smart-watches monitor your health
- Election campaigns use algorithmically targeted ads to influence voters
- Data-driven discoveries are made in Physics, Biology, Genetics, Astronomy, Chemistry, Neurology,...
- ...

MACHINE LEARNING AS BLACK-BOX MODELING

- Many concepts in ML can be explained without referring to the inner workings of a certain algorithm or model, especially things like model evaluation and hyperparameter tuning.
- ML consists of dozens (or hundreds?) of different modelling techniques. Not clear which of them are really needed (outside of pure research) and which are really best.
- Understanding basic concepts and model-agnostic techniques is really paramount and can be achieved in a limited amount of time.

ML AS BLACK-BOX MODELING

Studying to understand the inner workings of each and every ML model can take years. Do we even need to do this at all for some models?

- No: The useful ones are implemented in software. We can simply try them out, hopefully using a helpful program that iterates over them and optimizes them for us (spoiler alert: that's `mlr`).
- Yes: Some basic knowledge is required to make sensible choices. Actually knowing what it is you are doing is always good, also outside of science.
  And if things go wrong
  – and they usually do –
  then understanding things really
  does help a lot, too.

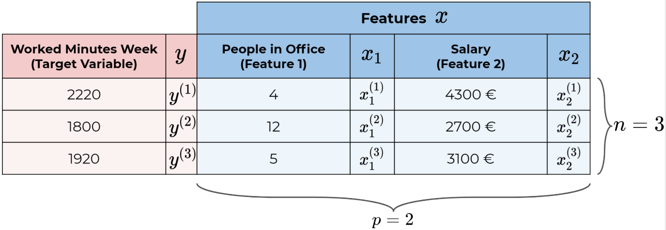
ML AS BLACK-BOX MODELING

- In the following slides we will go through the fundamental terminology and concepts in ML which are relevant for everything that comes next.
- We will also look at a couple of fairly simple ML models to obtain a basic understanding and look at some concrete examples.
- More complex stuff comes later.

DATA, TARGET AND INPUT FEATURES

Imagine you want to investigate how salary and workplace conditions affect productivity of employees. Therefore, you collect data about their worked minutes per week (productivity), how many people work in the same office as the employees in question and the employees' salary.

DATA, TARGET AND INPUT FEATURES



		Features x			
Worked Minutes Week (Target Variable)	y	**People in Office (Feature 1)**	x_1	**Salary (Feature 2)**	x_2
2220	$y^{(1)}$	4	$x_1^{(1)}$	4300 €	$x_2^{(1)}$
1800	$y^{(2)}$	12	$x_1^{(2)}$	2700 €	$x_2^{(2)}$
1920	$y^{(3)}$	5	$x_1^{(3)}$	3100 €	$x_2^{(3)}$

$n = 3$

$p = 2$

The entire **data set** is expressed by

$$\mathcal{D} = \left\{ \left(x^{(1)}, y^{(1)}\right), \ldots, \left(x^{(n)}, y^{(n)}\right) \right\}$$

with the i-th **observation** $\left(x^{(i)}, y^{(i)}\right) \in \mathcal{X} \times \mathcal{Y}$.
$\mathcal{X}$ is called **input space** and contains all possible values of the **features.**
$\mathcal{Y}$ is the **output space** or **target space** and contains all possible values of the **target variable**.

TARGET AND FEATURES RELATIONSHIP

- For our observed data we know which outcome is produced:

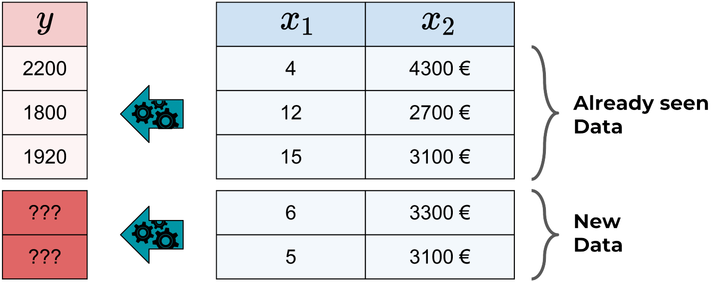
y
2200
1800
1920

x_1	x_2
4	4300 €
12	2700 €
15	3100 €

Already seen Data

TARGET AND FEATURES RELATIONSHIP

- For new employees we can only observe the features, but not the target:



y	x_1	x_2
2200	4	4300 €
1800	12	2700 €
1920	15	3100 €
???	6	3300 €
???	5	3100 €

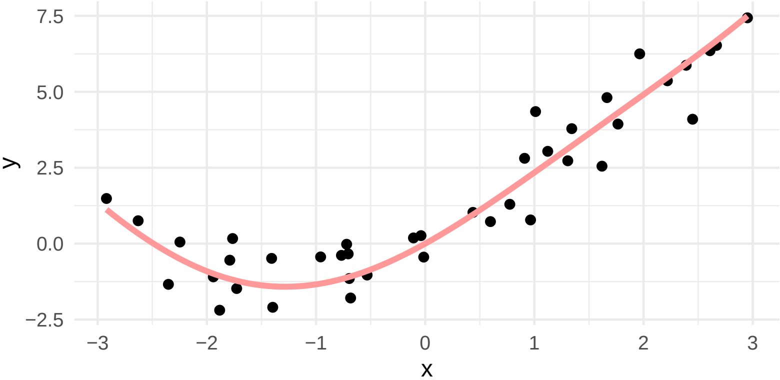
$\Longrightarrow$ The goal is to predict the target variable for **unseen new data** by using a **model** trained on the already seen **training data**.
This is a **supervised learning** task. In this course, we will only deal with ML problems of this kind.

SUPERVISED LEARNING TASKS

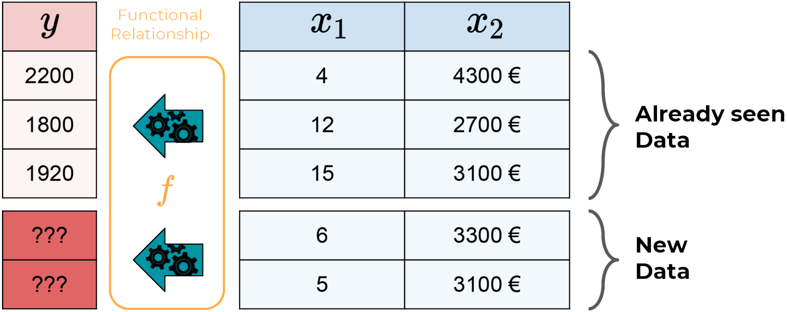
- **Regression**: Given features x, predict corresponding output from $\mathcal{Y} \in \mathbb{R}^m, 1 \leq m < \infty$.
- **Classification**: Assigning an observation with features x to one class of a finite set of classes $\mathcal{Y} = \{C_1, ..., C_g\}, 2 \leq g < \infty$.
- **Density estimation**: Given an input x, predict the probability distribution $p(y|x)$ on $\mathcal{Y}$.

REGRESSION TASK

- **Goal**: Predict a continuous output
- y is a metric variable (with values in $\mathbb{R}$)
- Regression model can be constructed by different methods, e.g., linear regression, trees or splines



TARGET AND FEATURES RELATIONSHIP



y	x_1	x_2
2200	4	4300 €
1800	12	2700 €
1920	15	3100 €
???	6	3300 €
???	5	3100 €

Functional Relationship

f

Already seen Data

New Data

TARGET AND FEATURES RELATIONSHIP

- In ML, we want to be “lazy”. We do not want to specify *f* manually.
- We want to learn *f* **automatically from labeled data**.
- Mathematically, we face a problem of function approximation: search for an *f*, such that, for all points in the training data and also all newly observed points

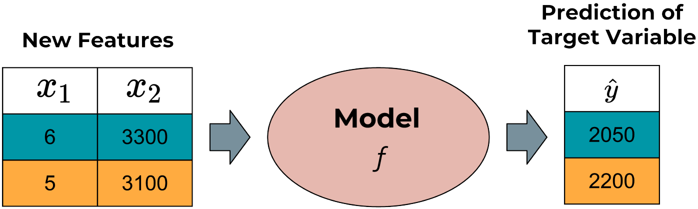
$$\mathrm{x} \longrightarrow \boxed{\; y \approx f(x) \;} \longrightarrow \mathrm{y}$$

- We call this **supervised learning**.
- (Later we will see that we need to specify at least *f*’s general structure and how to quantify the difference between y and $f(x)$ in order to make this problem feasible.)

WHAT IS A MODEL?

A **model** (or hypothesis) $f : \mathcal{X} \rightarrow \mathcal{Y}$ maps inputs (or input features) to outputs (or targets).

A hypothesis class H is a set of such functions.



New Features

x_1	x_2
6	3300
5	3100

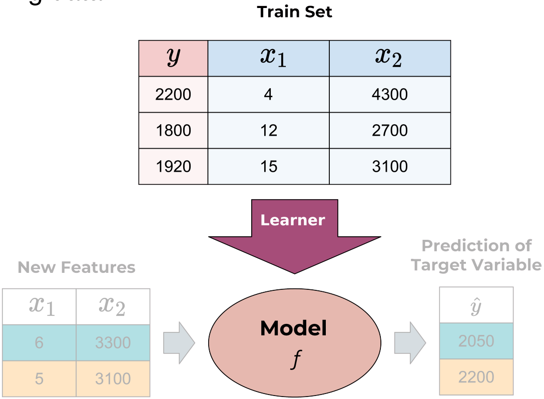
Model f

Prediction of Target Variable

$\hat{y}$
2050
2200

WHAT IS A LEARNER?

An **learner** (inducer, algorithm) takes a data set with features and outputs (**training set**, $\in \mathcal{X} \times \mathcal{Y}$) and produces a **model** (which is a function $f : \mathcal{X} \to \mathcal{Y}$):
So: Applying a learning algorithm means coming up with a model based on training data.



Train Set

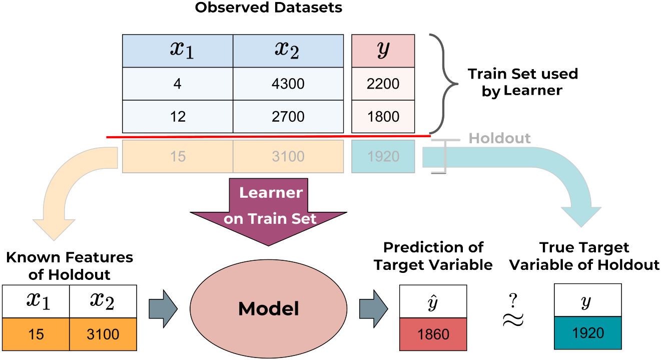
y	x_1	x_2
2200	4	4300
1800	12	2700
1920	15	3100

x_1	x_2
6	3300
5	3100

$\hat{y}$
2050
2200

HOW TO EVALUATE MODELS

- Simply compare predictions from model with truth:



Observed Datasets

x_1	x_2	y
4	4300	2200
12	2700	1800
15	3100	1920

Train Set used by Learner

Holdout

Learner on Train Set

Known Features of Holdout

x_1	x_2
15	3100

Model

Prediction of Target Variable

$\hat{y}$
1860

$\overset{?}{\approx}$

True Target Variable of Holdout

y
1920

COMPONENTS OF A LEARNER

Nearly all ML supervised learning training algorithms can be described by three components:

Learning = Representation + Evaluation + Optimization

- **Representation / Hypothesis Space:** Defines which kind of model structure of f can be learned from the data.
- **Evaluation:** A metric that quantifies how well a specific model performs on a given data set. 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 evaluation metric.
- All of these components represent important choices in ML which can have drastic effects:
  By making smart choices here, we can tailor learners to specific problems - but that usually requires quite a lot of experience and deeper insights into ML.

COMPONENTS OF AN INDUCER

Representation :

- Neighbors
- Linear functions
- Decision trees
- Sets of rules
- Neural networks
- Graphical models
- ...

Evaluation :

- Squared error
- Misclassification
- Likelihood
- Information gain
- ...

Optimization :

- Gradient descent
- Quadratic programming
- Combinatorial optimization
- Genetic algorithms
- ...