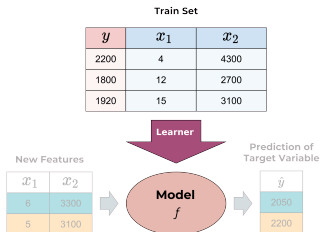


ML-Basics Learner



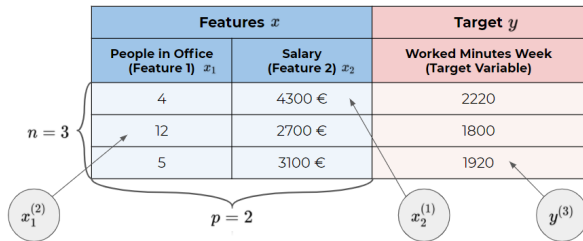
- Know formal definition of learner
- Understand that a learner receives training data and outputs the best model from \mathcal{H}

- Know formal definition of learner
- Understand that a learner receives training data and outputs the best model from \mathcal{H}

SUPERVISED LEARNING EXAMPLE

Imagine we want to investigate how working conditions affect productivity of employees.

- It is a **regression** task since the target *productivity* is continuous.
- We collect data about worked minutes per week (*productivity*), how many people work in the same office as the employee in question, and the employee's salary.



SUPERVISED LEARNING EXAMPLE / 2

How could we construct a model from these data?

We could investigate the data manually and come up with a simple, hand-crafted rule such as:

- The baseline productivity of an employee with salary 3000 and 7 people in the office is 1850 minutes.
- A decrease of 1 person in the office increases productivity by 30.
- An increase of the salary by 100 increases productivity by 10.

⇒ Obviously, this is neither feasible nor leads to a good model.



IDEA OF SUPERVISED LEARNING

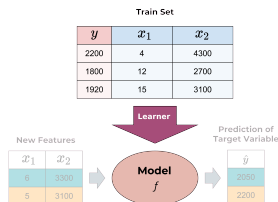
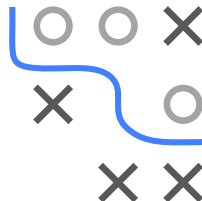
Goal: Identify the functional relationship that maps features to target.

- **Supervised** learning means we use *labeled* data to learn model f .
- Later, we use model f to predict y for new *unlabeled* data.



LEARNER DEFINITION

- Algorithm for finding f is called **learner / learning algorithm / inducer**.
- The learner is our means of picking the best element from the hypothesis space \mathcal{H} for given training data.
- Formally, it maps training data $\mathcal{D} \in \mathbb{D}$ (plus a vector of **hyperparameter** control settings $\lambda \in \Lambda$) to a model:



$$\mathcal{I} : \mathbb{D} \times \Lambda \rightarrow \mathcal{H}$$

- Practically, we often construct a mapping

$$\mathcal{I} : \mathbb{D} \times \Lambda \rightarrow \Theta$$

LEARNER DEFINITION / 2

In pseudo-code:

