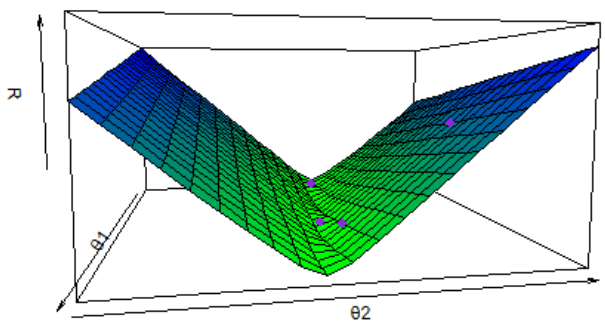
Introduction to Machine Learning

Introduction: Losses & Risk Minimization



Learning goals

- Know the concept of loss
- Understand the relationship between loss and risk
- Understand the relationship between risk minimization and finding the best model

HOW TO EVALUATE MODELS

In the training, we want to optimize θ. To score θ, we have to compare the actual output with the predicted output:

Features x		Target y	$\approx^{?}$	Prediction $\hat{y}$
People in Office (Feature 1) x_1	Salary (Feature 2) x_2	Worked Minutes Week (Target Variable)		Worked Minutes Week (Target Variable)
4	4300 €	2220		2588
12	2700 €	1800		1644
5	3100 €	1920		1870

$\mathcal{D}_{\text{train}}$

MOTIVATION

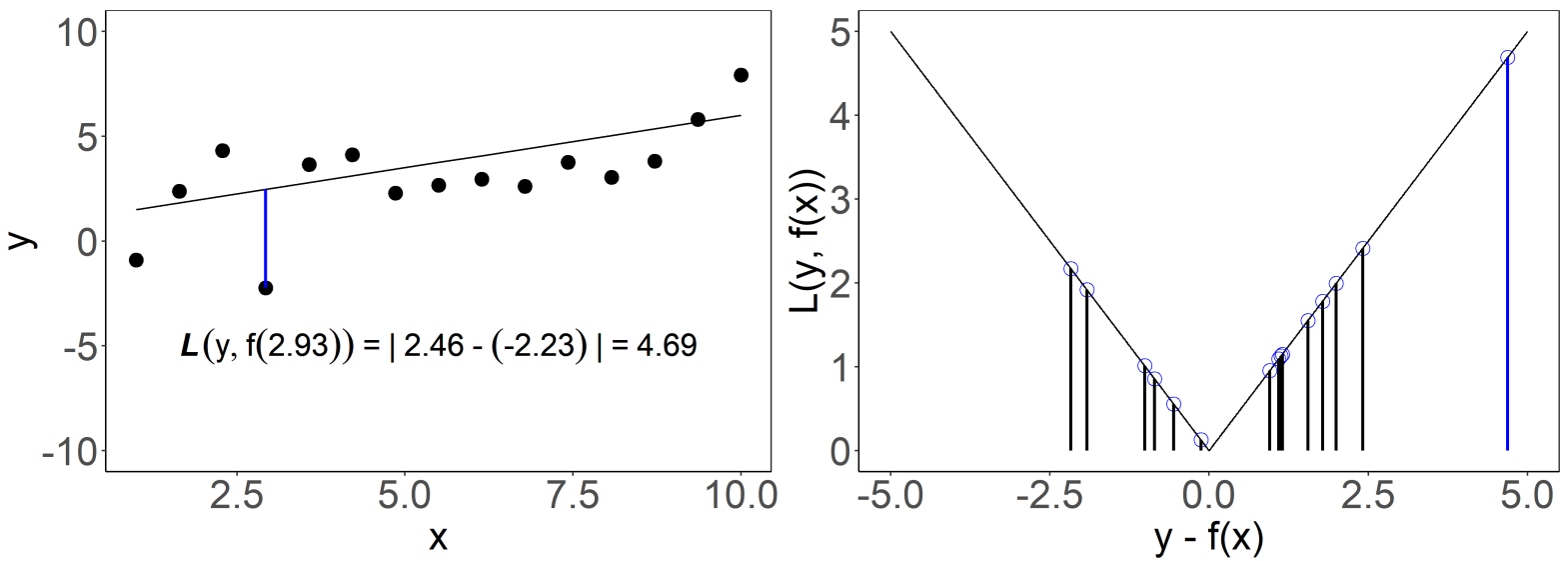
- Assume we trained a model to predict flat rent based on some features (size, location, age, ...).
- The real rent of a flat is EUR 1600, our model predicts EUR 1300.
- How do we measure the performance of our model?
- Need to define a suitable criterion, e.g.:
  - Absolute error $|1600 - 1300| = 300$
  - 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**.

LOSS

The **loss function** $L(y, f(\mathbf{x}))$ quantifies the "quality" of the prediction $f(\mathbf{x})$ of a single observation $\mathbf{x}$:

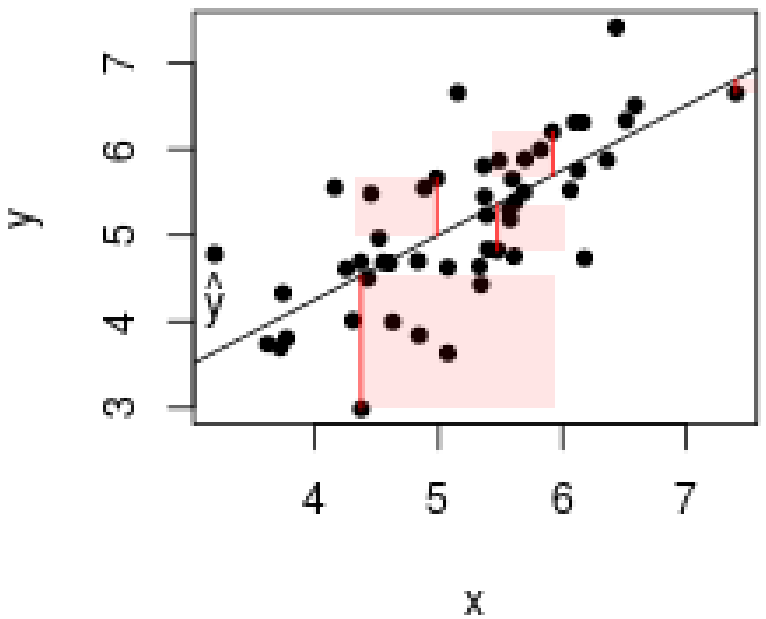
$$L : \mathcal{Y} \times \mathbb{R}^g \to \mathbb{R},$$

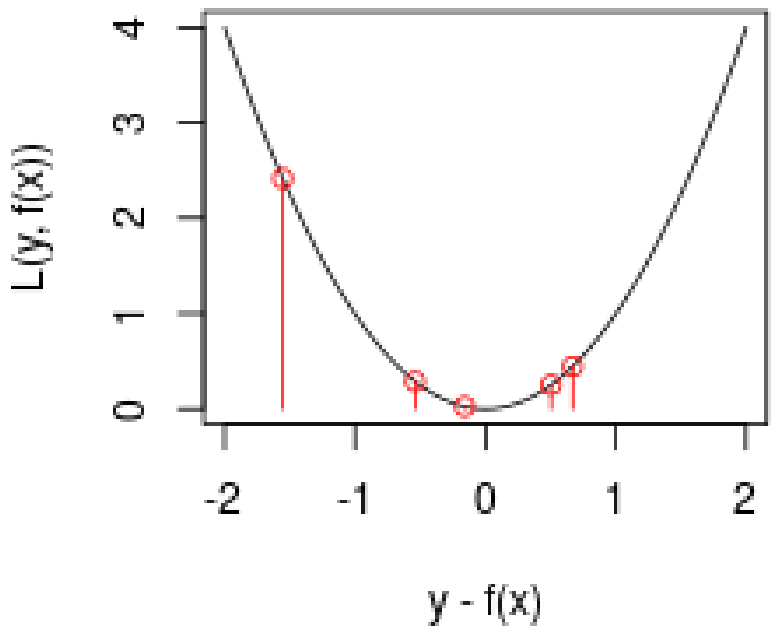
How "close" $f(\mathbf{x})$ is to y can be quantified e. g. by the absolute loss $L(y, f(\mathbf{x})) = |f(\mathbf{x}) - y|$.



LOSS

Often, we use the L2-loss $L(y, f(\mathbf{x})) = (y - f(\mathbf{x}))^2$:



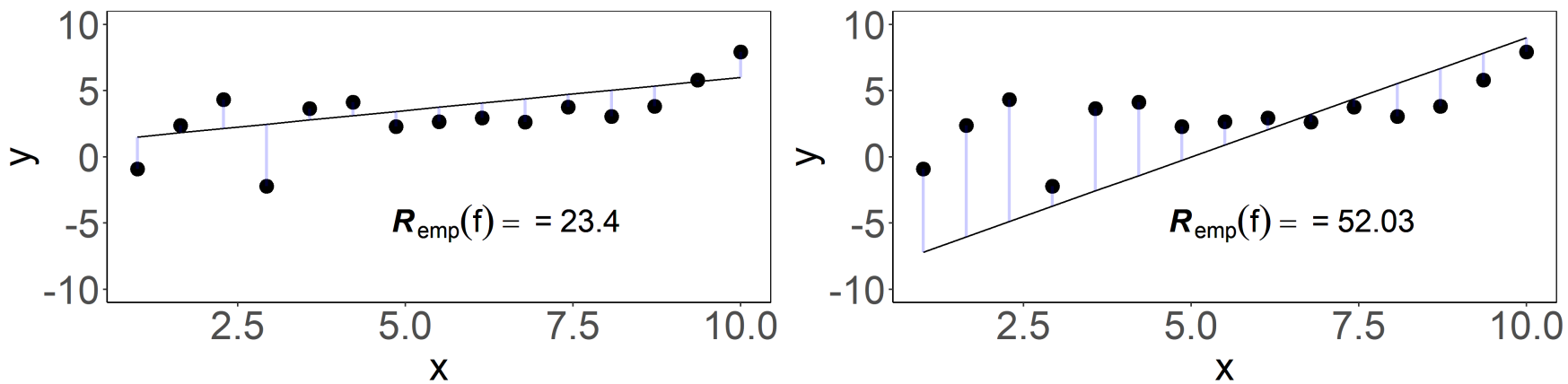


RISK

The **risk function** quantifies the "quality" of the whole model.

The ability of a model f to reproduce the association between $\mathbf{x}$ and y that is present in the data $\mathcal{D}$ can be measured by the **summed loss**, also called **"empirical risk"**:

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



RISK

Note:
The risk is often denoted as empirical mean over $L\left(y, f(\mathbf{x})\right)$

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

The factor $\frac{1}{n}$ does not make a difference in optimization, so we will consider $\mathcal{R}_{\text{emp}}(f)$ most of the time.

????

The best model is the model with the smallest risk.

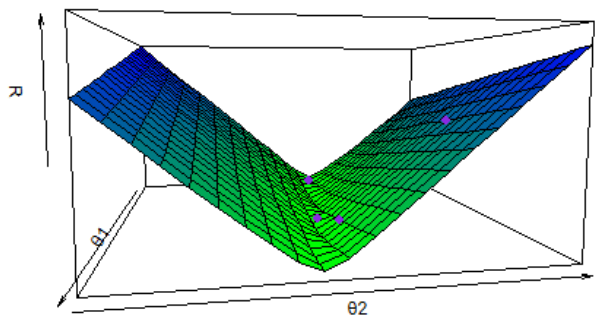
If we have a finite number of models f, we can compare the risk $\mathcal{R}_{\text{emp}}(\boldsymbol{\theta})$ of all models:

Model	$\boldsymbol{\theta}_{intercept}$	$\boldsymbol{\theta}_{slope}$	$\mathcal{R}_{\text{emp}}(\boldsymbol{\theta})$
f_1	4	1	96.37
f_2	3	7	576.37
f_3	1	0.5	1.56
f_4	-9	1.8	52.03

???

But: Normally, the hypothesis space $\mathcal{H}$ is infinitely large.

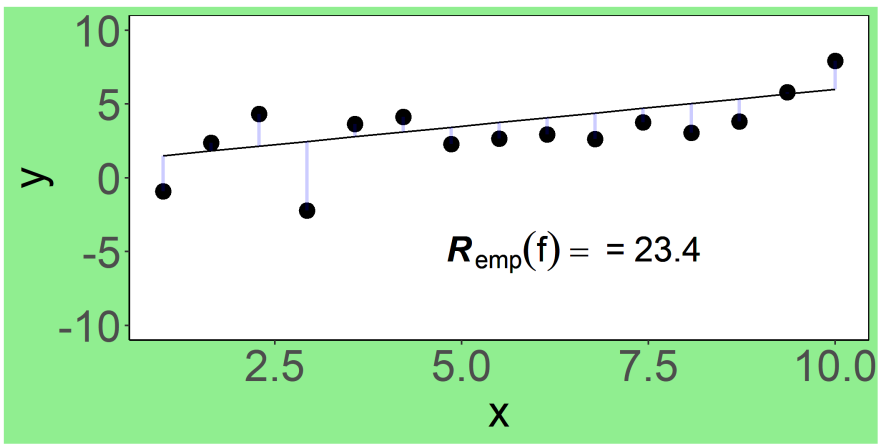
As the the mapping of the hypothesis space to its parameters is bijective, we can consider the error surface depending on the parameters:

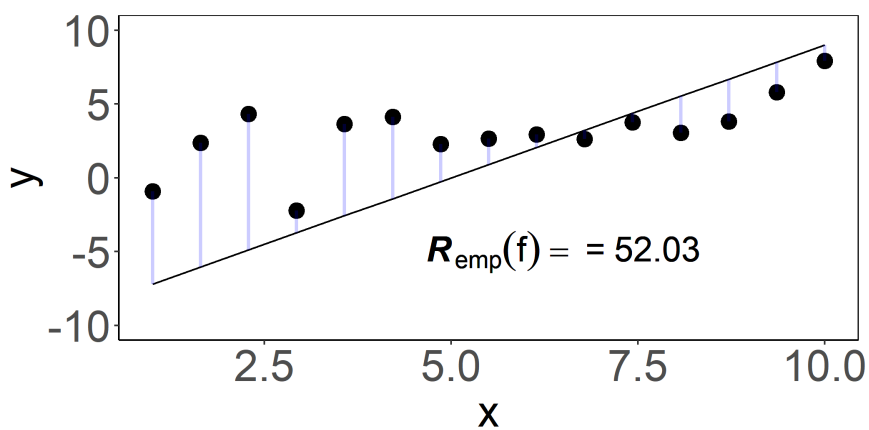


RISK MINIMIZATION

The process of finding the best model is called **empirical risk minimization** (ERM).

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





RISK MINIMIZATION

Since the model f is usually defined by **parameters** $\boldsymbol{\theta}$ in a parameter space Θ, this becomes:

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

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

FURTHER REMARKS

- For regression tasks, the loss often only depends on the residual $L(y, f(\mathbf{x})) = L(y - f(\mathbf{x})) = L(\epsilon)$.
- 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 $\mathcal{R}_{\text{emp}}(\boldsymbol{\theta})$ in $\boldsymbol{\theta}$?
  - Is $\mathcal{R}_{\text{emp}}(\boldsymbol{\theta})$ differentiable so that we can use gradient-based methods?
  - Does $\mathcal{R}_{\text{emp}}(\boldsymbol{\theta})$ have multiple local minima or saddlepoints over Θ?