Learning theory from first principles

Chapter 2 : Introduction Decision theory

Data : (x_i, y_i) $i = 1, \dots, n$ training data

$x \in \mathcal{X}$ $y \in \mathcal{Y}$

response label output $y \subset \mathbb{R}$

Estimate $f : \mathcal{X} \to \mathcal{Y}$

Compute

Probability distribution over $(x, y) \in \mathcal{X} \times \mathcal{Y}$. P

test distribution = training data IID

Machine learning algorithm

A : data

$(x_1, y_1), \ldots, (x_n, y_n)$ ⟶ function $x \to y$

↓

Practical evaluation : no access to $p(x, y)$

testing data.

Decision theory

loss function : $\ell : Y \times Y \to \mathbb{R}$

$\ell(y, z)$ = loss of predicting z instead of y.

Classification: (y finite) $\ell(y,z) = 1_{y \neq z}$ 0-1 loss

$Y = \{1, \dots, k\}$ $\quad$ $Y = \{-1, 1\}$

Regression : ($Y = \mathbb{R}$) $\ell(y,z) = (y-z)^2$ square

($|y-z|$ —)

Examples: "surrogates"

Expected risk / test(ing) error / generalization performance

$$R(f) = \mathbb{E}\left[\ell(y, f(x))\right]$$

$f : X \to Y$

Expected risk / test(ing) error / generalization performance

$$R(f) = \mathbb{E}\left[\ell(y, f(x))\right]$$

fixed f

Empirical risk / training error

$$\hat{R}(f) = \frac{1}{n}\sum_{i=1}^{n} \ell(y_i, f(x_i))$$

$(x_i, y_i) \sim$ iid $p(x,y)$

property: if f is deterministic $\mathbb{E}\left[\hat{R}(f)\right] = R(f)$

$$\hat{R}(f) - R(f) \simeq O\left(\frac{1}{\sqrt{n}}\right)$$

Uniform deviations $\sup_{f \in \mathcal{F}} \left|\hat{R}(f) - R(f)\right| = O\left(\frac{1}{\sqrt{n}}\right)$

Examples: Classification $\ell(y,z) = 1_{y \neq z}$

$R(f) = \mathbb{E}\left(1_{f(x) \neq y}\right) = \mathbb{P}(f(x) \neq y)$ error rate

$\hat{R}(f)$ = empirical error rate
proportion of mistakes.

$f^*(x) = \operatorname{argmin} \mathbb{P}(y \neq z \mid x = x')$
$= \operatorname{argmax} \mathbb{P}(y = z \mid x = x')$

Regression: $\ell(y,z) = (y-z)^2$

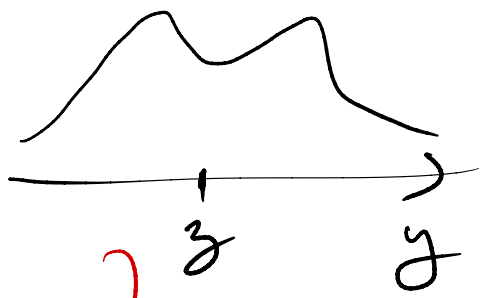
$R(f) = \mathbb{E}\left[(y - f(x))^2\right]$

$\hat{R}(f) = \frac{1}{n} \sum_{i=1}^{n} (y_i - f(x_i))^2$

$f^*(x') \in \operatorname{argmin}_{z \in \mathbb{R}} \mathbb{E}(\ell(y,z) \mid x = x')$

$(y-z)^2$

$= \mathbb{E}[y \mid x = x']$



Optimal predictor: $f^* = \operatorname{argmin}_{f \text{ measurable}} R(f)$

$= $ Bayes predictor

$R(f^*) = R^* =$ Bayes risk

Excess risk $R(f) - R^*$

$$R(f) = \int \ell(y, f(x))\, dp(x,y) = \mathbb{E}\left(\ell(y, f(x))\right)$$

$$= \mathbb{E}_{x'}\left[\underbrace{\mathbb{E}\left(\ell(y, f(x')) \mid x = x'\right)}_{\text{depends a } x'}\right]$$

$$f^*(x') \in \operatorname*{argmin}_{z \in y} \mathbb{E}\left(\ell(y, z) \mid x = x'\right).$$

"Optimal ML": given x' "access" $p(y \mid x = x')$.

$$\min_z \mathbb{E}\left(\ell(y, z) \mid x = x'\right)$$

Two main classes of ML algorithms

- local averaging (k-NN)
- ERM
- No free lunch
- Hoeffding.

IC 43 → 11h00.

Local averaging

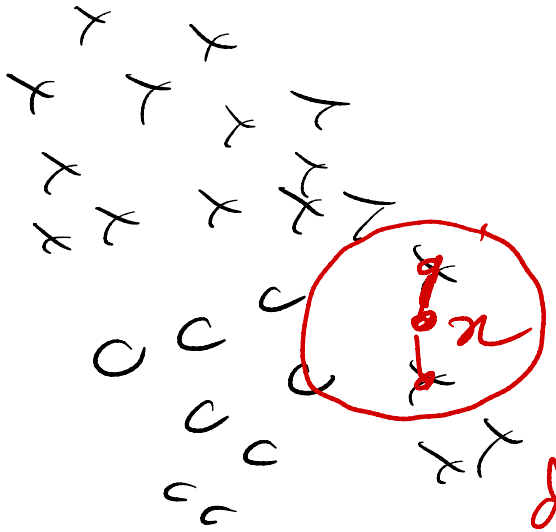
$$f^*(x) = \arg\min_{z \in \mathcal{Y}} \int \ell(y,z)\, dp(y|x)$$

↑ replace some estimate

$$d\hat{p}(y|x)$$

Curse of dimensionality

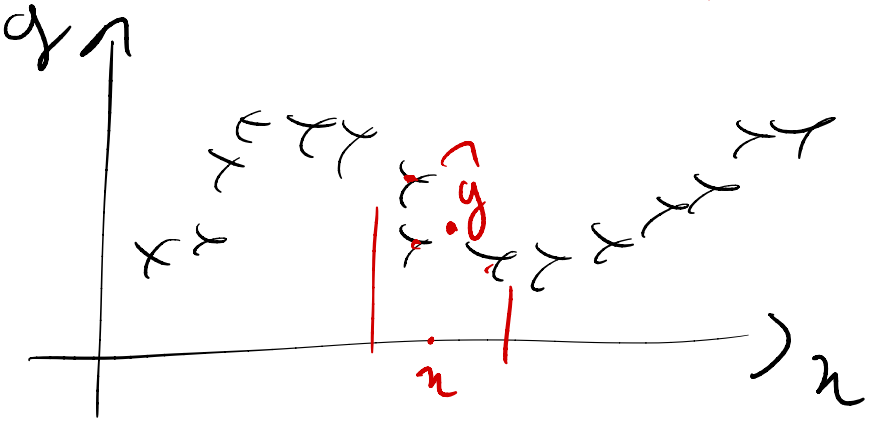
$R(\hat{f}) - R^* \leq O\left(\frac{1}{N^{1/d}}\right)$



$\mathcal{X} = \mathbb{R}^?$ · k-nearest neighbors

$\mathcal{Y} = \{x, c\}$ · kernel regression Nadaraya Watson

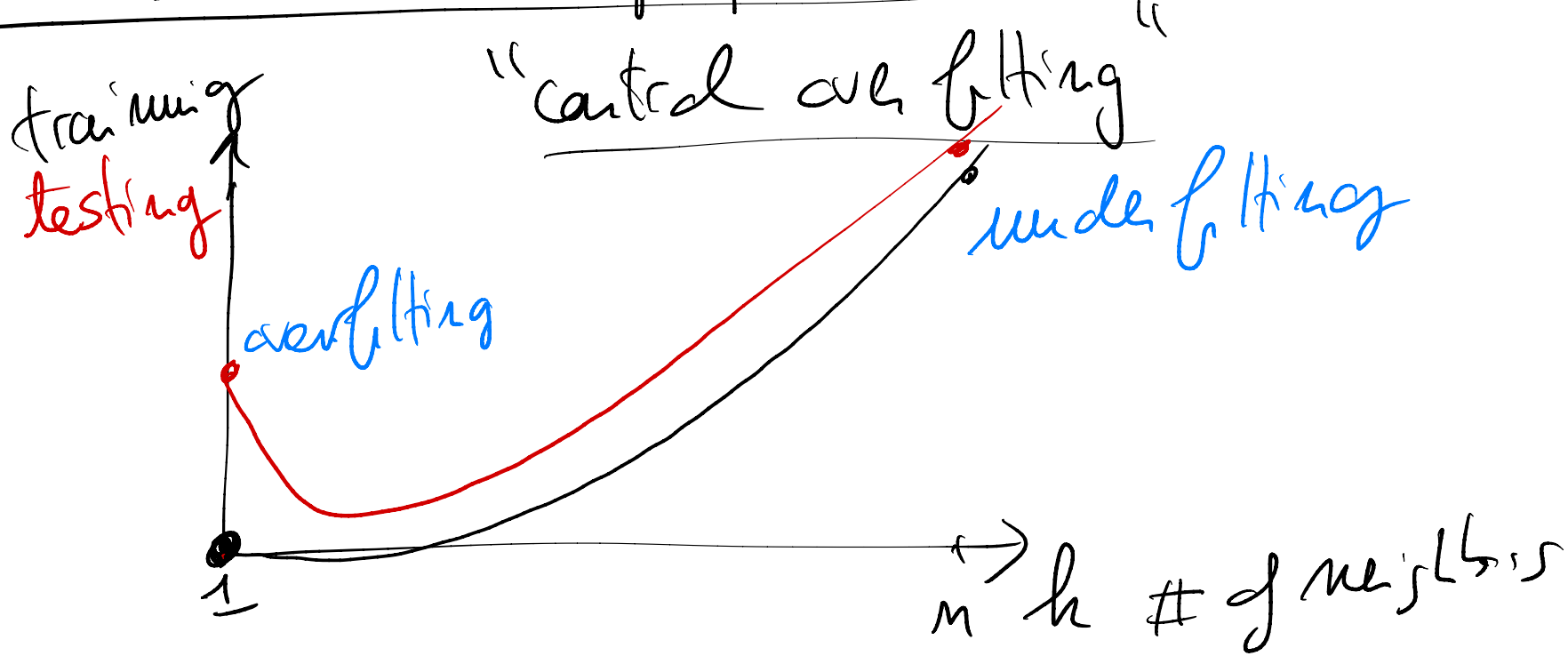
$$dp(y|x) = \sum_{i=1}^{n} w_i(x)\, \delta_{y=y_i}$$ Chap Ver 7

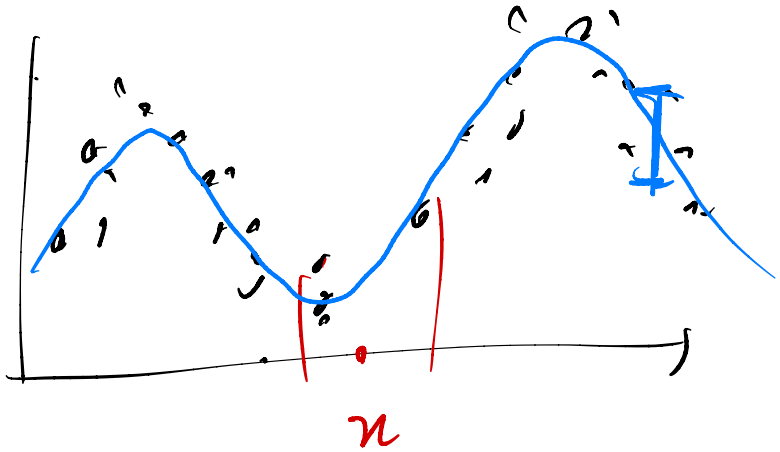


$\mathcal{X} = \mathbb{R}$

$\mathcal{Y} = \mathbb{R}$

All methods have an hyperparameter





Empirical risk minimization ERM

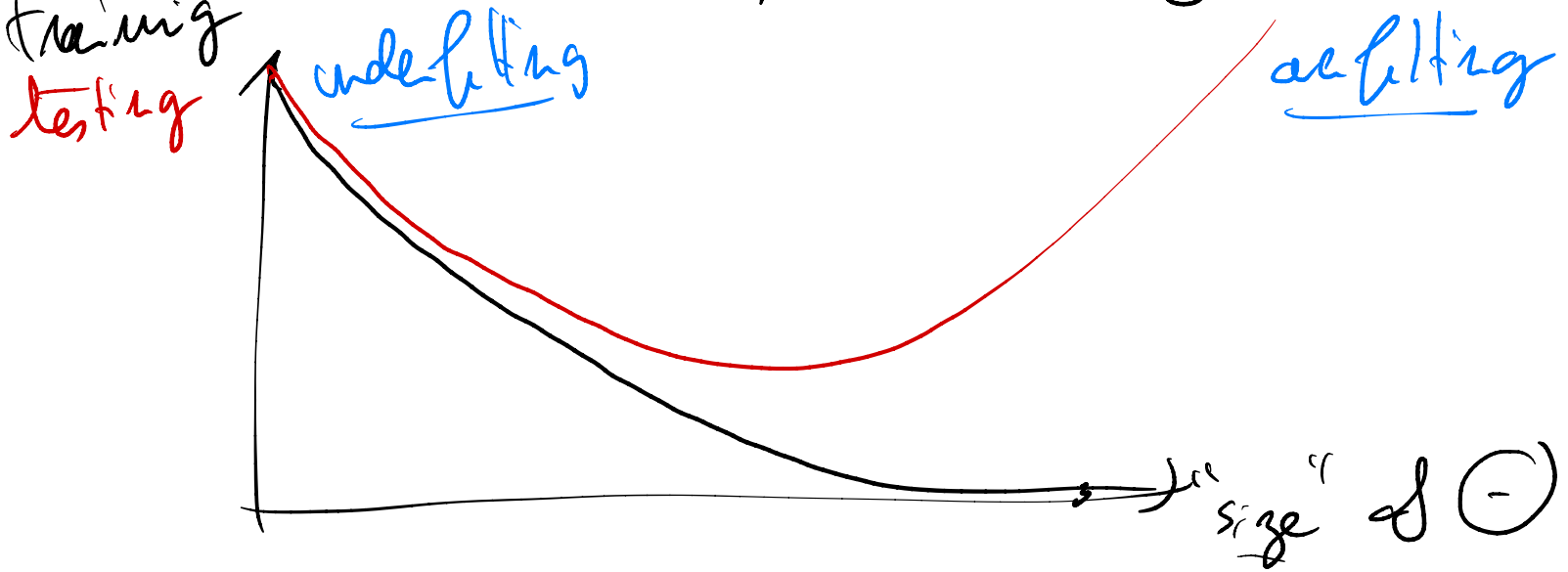
Estimator : model $f_\theta : X \to Y$

θ parameter $\in \Theta$

$$\hat{\theta} = \arg\min_{\theta \in \Theta} \hat{R}(f_\theta) = \frac{1}{n} \sum_{i=1}^{n} \ell(y_i, f_\theta(x_i))$$

$$\hat{f} = f_{\hat{\theta}}$$

- two practical issues $\leftarrow$ $Y = \{-1, 1\} \Rightarrow$ convex surrogates "logistic loss"
  non-convexity.
-



Risk decomposition

model $f_\sigma : x \to y$, $\sigma \in \Sigma$

$\hat{\sigma} = \underset{\sigma \in \Sigma}{\operatorname{argmin}} \hat{R}(f_\sigma)$

σ^*

expected risk ↑

$$R(f_{\hat{\sigma}}) - R^* = R(f_{\hat{\sigma}}) - \inf_{\sigma' \in \Sigma} R(f_{\sigma'}) + \inf_{\sigma' \in \Sigma} R(f_{\sigma'}) - R^*$$

≥ 0

σ' minimize of R

estimation error

random ≥ 0

depends on n

approximation error

≥ 0

goes down with size of Σ

deterministic

$$= R(f_{\hat{\sigma}}) - R(f_{\sigma'}) + R(f_{\sigma'}) - R^* \quad \leq 0 \text{ optim error}$$

$$= R(f_{\hat{\sigma}}) - \hat{R}(f_{\hat{\sigma}}) + \hat{R}(f_{\hat{\sigma}}) - \hat{R}(f_{\sigma'}) + \hat{R}(f_{\sigma'}) - R(f_{\sigma'}) + R(f_{\sigma'}) - R^*$$

$\hat{R}(f_{\sigma'}) - R(f_{\sigma'})$: $\sim \frac{1}{\sqrt{n}}$

$R(f_{\sigma'}) - R^*$: approx error

$\leq \sup_{\sigma \in \Sigma} |R(f_\sigma) - \hat{R}(f_\sigma)|$ uniform deviation

Study of $\hat{\mathcal{R}}(f_{\hat{\theta}}) - \hat{\mathcal{R}}(f_{\theta^*}) \leq \hat{\mathcal{R}}(f_{\hat{\theta}}) - \min_{\theta \in \Theta} \hat{\mathcal{R}}(f_\theta)$

$\hat{\mathcal{R}}(f_{\hat{\theta}}) - \hat{\mathcal{R}}(f_{\theta^*})$ — approximate ERM

$\min_{\theta \in \Theta} \mathcal{R}(f_\theta)$

$= 0$ if $\hat{\theta} =$ ERM

≥ 0 otherwise

→ optimization error

Goals of statistical learning theory

Notion of consistency

Training data (x_i, y_i), $i = 1, \dots, n$ IID from distribution $p(x,y)$ / $p(x)\,p(y|x)$

$\Rightarrow D_n(\rho)$

ML Algo: $A : D_n(\rho) \to$ function $x \to y$

Good $R(A(D_n(\rho))) - R_p^*$ small

- $R(A(D_n(\rho)))$: testing error
- $\hat{R}$
- R_p^*: Bayes risk for p

two types of bounds:
- in high probability: PAC
- in expectation

non-asymptotic vs. asymptotic

probably approximately correct

(*) $\mathbb{E}\left(R(A(D_n(P))) - R_P^*\right)$

- Algorithm A is consistent for problem P if $(*) \longrightarrow 0$ $n \to \infty$.
- Algorithm A is universally consistent, if $\forall P$ it is consistent
- Consistency on a class $\mathcal{P}$ of problems
  - (**) $\sup_{P \in \mathcal{P}} \mathbb{E}\left(R(A(D_n(P))) - R_P^*\right) \leq \frac{C}{n^\alpha}$
  - No free lunch: if $\mathcal{P}$ = all problems
    - $\sup$ (**) arbitrarily slow

- Least-squares regression
- Uniform deviation → all linear models
- optimizers < GD / SGD
- kernel methods
- neural networks

1350