Statistical Foundations of Learning

Debarghya Ghoshdastidar

School of Computation, Information and Technology
Technical University of Munich

Ensemble methods

Context

- We have so far focussed on analysis of single classifiers
  - k-NN / plugin estimators:

$$\mathbb{E}[L_{\mathcal{D}}(\widehat{h}_S)] \to L^*_{\mathcal{D}} \text{ as } m \to \infty, \qquad \text{if conditions of Stone's theorem hold}$$

  - ERM over $\mathcal{H}$ with $\text{VCdim}(\mathcal{H}) = d < \infty$ (linear classifiers, neural networks)

$$L_{\mathcal{D}}(\widehat{h}) \;\leq\; \underbrace{\inf_{h \in \mathcal{H}} L_{\mathcal{D}}(h)}_{\text{inductive bias}} \;+\; \underbrace{O\left(\sqrt{\frac{d \ln m}{m}}\right)}_{\text{estimation error}}$$

- Simple classifiers (1-NN, ERM over decision stumps)
  - are computationally faster, have low estimation error,
  - **but** inductive bias can be high, may not be close to Bayer error

Ensemble methods: Definition from scikit-learn

- The goal of ensemble methods is to combine the predictions of several base estimators built with a given learning algorithm in order to improve generalizability / robustness over a single estimator.

- In **averaging methods**, the driving principle is to build several estimators independently and then to average their predictions.
  - Examples: Voting classifiers, Bagging, Random Forests

- In **boosting methods**, base estimators are built sequentially and one tries to reduce the bias of the combined estimator. The motivation is to combine several weak models to produce a powerful ensemble.
  - Examples: AdaBoost, Gradient Boosting (XGBoost)

Outline

- Voting and Averaged predictors
  - Generalisation error bound for majority vote
  - If individual classifiers are consistent, then averaged classifier is also consistent
- Bagging
  - Ensemble of 1-NNs is universally consistent
  - Bagging leads to classifiers with optimal sample complexity (so better than ERM)
- Boosting
  - Weak learning: Performing better than random guessing (error $< \frac{1}{2}$ for binary case)
  - Adaboost: Combines weak learners to build a PAC learner; Achieves low training error

Average / Majority vote classifier

- Let $S = \{(x_1, y_1), \ldots, (x_m, y_m)\} \subset \mathcal{X} \times \{\pm 1\}$ be training sample
- Let $h_{S,1}, \ldots, h_{S,T}$ be T binary classifiers learned from S (or some sub-sample of S)
  - $h_{S,1}, \ldots, h_{S,T}$ are called base classifiers
  - Example: $h_{S,t}$ = classifier learned using ERM on $\mathcal{H}_t \subset \{\pm 1\}^{\mathcal{X}}$

    ($\mathcal{H}_t$ is a base hypothesis class)
- Average / majority vote classifier:

$$h_S^{(T)}(x) = \text{sign}\left(\frac{1}{T}\sum_{t=1}^{T} h_{S,t}(x)\right)$$

Hypothesis class of majority vote, and generalisation error

- Let $h_{S,t}$ = classifier learned using ERM on $\mathcal{H}_t \subset \{\pm 1\}^{\mathcal{X}}$, where $\text{VCdim}(\mathcal{H}_t) = d < \infty$
- Hypothesis class of majority vote classifier:

$$M_{\mathcal{H}_1,\ldots,\mathcal{H}_T} = \left\{ \text{sign}\left(\frac{1}{T} \sum_{t=1}^{T} h_t(\cdot) \right) \;:\; h_t \in \mathcal{H}_t \;\forall t \right\} \quad \subset \{\pm 1\}^{\mathcal{X}}$$

- **Exercise:** Show that $\text{VCdim}(M_{\mathcal{H}_1,\ldots,\mathcal{H}_T}) = O\big(dT \cdot \log_2(dT)\big)$
- Generalisation error bound: With probability $1-\delta$ (over $S \sim \mathcal{D}^m$),

$$L_{\mathcal{D}}\left(h_S^{(T)}\right) \;\leq\; L_S\left(h_S^{(T)}\right) + O\left(\sqrt{\frac{dT \cdot \log_2(dT)\ln(2em) + \ln(\frac{1}{\delta})}{m}} \right)$$

- Can we state a bound on $L_{\mathcal{D}}\left(h_S^{(T)}\right)$ in terms of $\inf_{h \in M_{\mathcal{H}_1,\ldots,\mathcal{H}_T}} L_{\mathcal{D}}(h)$?

Randomised classifiers

- Given training sample $S = \{(x_1, y_1), \ldots, (x_m, y_m)\} \subset \mathcal{X} \times \mathcal{Y}$

- Randomised classifier:
  $\tilde{h}_S(x) = h_S(x, Z) =$ predictor learned from S,
  where learning involves a random variable/vector Z

  - Example (bagging): $Z = (z_1, \ldots, z_m) \in \{0, 1\}^m$
    $z_i =$ Bernoulli variable indicating i-th sample in S is used for training

- Error of a randomised classifier $\tilde{h}_S$:
  $$L_{\mathcal{D}}(\tilde{h}_S) = \mathbb{E}_{(x,y),Z}\left[\ell\big(h_S(x, Z), y\big)\right] = \mathbb{E}_{(x,y),Z}\left[\ell\big(h_S(x, Z), y\big)\big|S\right]$$

  - For 0-1 loss, $L_{\mathcal{D}}(\tilde{h}_S) = \mathbb{P}_{(x,y),Z}\big(h_S(x, Z) \neq y\big)$

Voting / Average of Ramdomised Classifiers

- It helps to let class labels be $\{0, 1\}$
  Given training sample $S = \{(x_1, y_1), \ldots, (x_m, y_m)\} \subset \mathcal{X} \times \{0, 1\}$

- $\tilde{h}_S$ is a randomised classifier:
  - Let $Z_1, \ldots, Z_T$ be i.i.d. random variables/vectors
  - For every $t = 1, \ldots, T$, we learn a classifier $h_S(\cdot, Z_t) : \mathcal{X} \to \{0, 1\}$

- Voting classifier:
  $$h_S^{(T)}(x) = \mathbf{1}\left\{\frac{1}{T}\sum_{t=1}^{T} h_S(x, Z_t) \;\geq\; \frac{1}{2}\right\}$$

Averaged (expected) classifier:

$$\bar{h}_S(x) = \mathbf{1}\left\{\mathbb{E}_Z[h_S(x, Z)] \;\geq\; \frac{1}{2}\right\}$$

(similar to $T \to \infty$)

(Universal) consistency of voting classifier

- We say randomised classifier $\tilde{h}_S$ is consistent with respect to $\mathcal{D}$ if

$$\mathbb{E}_S\left[L_{\mathcal{D}}(\tilde{h}_S)\right] \to L^*_{\mathcal{D}} \text{ as } m \to \infty$$

  $\tilde{h}_S$ is universally consistent if above holds for every distribution $\mathcal{D}$

Theorem En.1 (Voting/averaging of consistent randomised classifiers is consistent)

Assume 0-1 loss, and let $\tilde{h}_S$ be a randomised classifier, consistent with respect to $\mathcal{D}$. Then the voting classifier $h_S^{(T)}$ and the averaged classifier $\bar{h}_S$ are also consistent w.r.t. $\mathcal{D}$.

If $\tilde{h}_S$ is universally consistent, then $h_S^{(T)}$ and $\bar{h}_S$ are also universally consistent.

Proof left as exercise. Use hints/steps provided in next slides (exercises marked in red).

Proof sketch: Part-1 (Alternative formulation of consistency)

- Let $g : \mathcal{X} \to \{0, 1\}$ be a (possibly randomised) classifier such that, conditioned on x, prediction $g(x)$ and true label $y \sim \eta(x)$ are independent.

  (1.1) Show that $\mathbb{P}(g(x) \neq y|x) \geq \min\{\eta(x), 1 - \eta(x)\} = \mathbb{P}(h^*(x) \neq y|x)$, where h^* is the Bayes classifier.

- **FACT (you can assume this):**
  (1.1) implies classifier $\tilde{h}_S$ is consistent w.r.t $\mathcal{D}$, that is $\lim_{m\to\infty} \mathbb{E}_S[L_{\mathcal{D}}(\tilde{h}_S)] = L^*_{\mathcal{D}}$, if and only if,

  with probability 1 w.r.t $\mathcal{D}_{\mathcal{X}}$, $\lim_{m\to\infty} \mathbb{P}_{S,Z,y}(h_S(x, Z) \neq y|x) = \min\{\eta(x), 1 - \eta(x)\}$.

- Informally, above statement says that proving consistency is equivalent to showing $\mathbb{P}_{S,Z,y}(h_S(x, Z) \neq y|x) \to \min\{\eta(x), 1 - \eta(x)\}$ for "every" x.

Proof sketch: Part-1 (Alternative formulation of consistency)

- Let $g : \mathcal{X} \to \{0, 1\}$ be a (possibly randomised) classifier that depends on m with $m \to \infty$.

(1.2) If $\eta(x) > \frac{1}{2}$, show that $\mathbb{P}(g(x) \neq y|x) \to \min\{\eta(x), 1 - \eta(x)\}$ if and only if $\mathbb{P}(g(x) = 0|x) \to 0$.
Similarly, for $\eta(x) < \frac{1}{2}$, show that $\mathbb{P}(g(x) \neq y|x) \to \min\{\eta(x), 1 - \eta(x)\}$ if and only if $\mathbb{P}(g(x) = 1|x) \to 0$.

- Combined with FACT, (1.2) implies that consistency of $\tilde{h}_S$ is equivalent to $\mathbb{P}_{S,Z}(h_S(x, Z) = 0|x) \to 0$ if $\eta(x) > \frac{1}{2}$ and $\mathbb{P}_{S,Z}(h_S(x, Z) = 1|x) \to 0$ if $\eta(x) < \frac{1}{2}$.

Proof sketch: Consistency of $\bar{h}_S$ and $h_S^{(T)}$

- Part-2: Consistency of averaged classifier $\bar{h}_S$

(2.1) Show that $\mathbf{1}\left\{\bar{h}_S(x) = a\right\} \leq 2\mathbb{P}_Z(h_S(x, Z) = a \,|\, x, S)$ for $a \in \{0, 1\}$.

(2.2) Use above to show that $\mathbb{P}_S(\bar{h}_S(x) = 0|x) \to 0$ if $\eta(x) > \frac{1}{2}$ and $\mathbb{P}_S(\bar{h}_S(x) = 1|x) \to 0$ if $\eta(x) < \frac{1}{2}$. Hence, argue that $\bar{h}_S$ is consistent.

- Part-3: Consistency of voting classifier $h_S^{(T)}$

(3.1) Observe that, for $a \in \{0, 1\}, h_S^{(T)}(x) = a$ if and only if $\frac{1}{T}\sum_{t=1}^{T} \mathbf{1}\{h_S(x, Z_t) = a\} \geq \frac{1}{2}$.
Use the observation and Markov's inequality to show that

$$\mathbb{P}_{Z_1,\ldots,Z_t}\left(h_S^{(T)}(x) = a \,|\, x, S\right) \leq 2\mathbb{P}_Z(h_S(x, Z) = a \,|\, x, S).$$

(3.2) Complete the proof following the same arguments as part-2.

Bagging 1-Nearest neighbours

- Bagging:
  - Given $S = \{(x_1, y_1), \ldots, (x_m, y_m)\} \subset \mathbb{R}^d \times \{0, 1\}$
  - Random vectors $Z_1, \ldots, Z_T \in \{0, 1\}^m$
  - The t-th classifier $h_S(\cdot, Z_t)$ is trained using a subset of samples $S_t = \{(x_i, y_i)\}_{i:Z_t^i=1}$
- Bagging 1-NN:
  - Recall in 1-NN, for every test point x, $x_{\pi_1(x)}$ is NN of x from training sample S
    ... but $x_{\pi_1(x)}$ may not be present in S_t
  - Let $x_{\pi_1(x,Z_t)}$ = NN of x from subsample S_t (which depends on random Z_t)
  - Voting 1-NN: $h_S^{NN,(T)}(x) = \mathbf{1}\left\{\frac{1}{T}\sum_{t=1}^{T} y_{\pi_1(x,Z_t)} \geq \frac{1}{2}\right\}$

Universal consistency of Averaged 1-NN

- Independent random subsampling:

  Random vector $Z \in \{0,1\}^m$ is such that every sample $(x_i, y_i), i = 1, \ldots, m$ is included in subsample independently with probability $q_m \in (0,1]$, that is,

  $$Z^1, \ldots, Z^m \sim_{iid} \text{Bernoulli}(q_m)$$

- Averaged 1-NN: $\bar{h}_S^{NN}(x) = \mathbf{1}\left\{\mathbb{E}_Z[y_{\pi_1(x,Z)}] \geq \frac{1}{2}\right\}$

Theorem En.2 (Universal consistency of Averaged 1-NN)

Although 1-NN classifier is not universally consistent, averaged 1-NN classifier is universally consistent if $q_m \to 0$ *and* $mq_m \to \infty$ *as* $m \to \infty$.

Proof: Averaged 1-NN as a plugin estimator

- For an x, without loss of generality, let $x_1, \ldots, x_m$ indexed such that
$$\|x - x_1\| \leq \|x - x_2\| \leq \ldots \leq \|x - x_m\|, \quad \text{that is,} \quad \pi_i(x) = i \;\; (x_i \text{ is } i\text{-th NN})$$

- $\bar{h}_S^{NN}(x) = \mathbf{1}\left\{\widehat{\eta}(x) \geq \frac{1}{2}\right\}$, where
$$\widehat{\eta}(x) = \mathbb{E}_Z[y_{\pi_1(x,Z)}] = \sum_{i=1}^{m} y_i \cdot \underbrace{\mathbb{P}_Z(\pi_1(x,Z) = i)}_{= w_i(x)}$$
$w_i(x) =$ probability that i-th NN is nearest in subsample induced by Z

- $\pi_1(x, Z) = i$ if $Z_1 = Z_2 = \ldots = Z_{i-1} = 0$, $Z_i = 1$
$$\implies w_i(x) = \mathbb{P}_Z(\pi_1(x,Z) = i) = (1 - q_m)^{i-1} q_m$$

  - Observe: $\sum_{i=1}^{m} w_i(x) = 1 - (1 - q_m)^m < 1$.
  For completeness, we may define $(x_0 = x, y_0 = 0)$ with $w_0(x) = (1 - q_m)^m$

Proof: Recap conditions for Stone's theorem

For $\bar{h}_S^{NN}$ to be universally consistent, $w_0(x), \ldots, w_m(x)$ should satisfy:

(i) $\exists c$ such that, for every non-negative integrable function f with $\mathbb{E}_{x \sim \mathcal{D}_\mathcal{X}}[f(x)] < \infty$,

$$\mathbb{E}_{x, x_1, \ldots, x_m \sim \mathcal{D}^{m+1}} \left[\sum_{i=1}^{m} w_i(x) \cdot f(x_i) \right] \leq c \, \mathbb{E}_{x \sim \mathcal{D}_\mathcal{X}}[f(x)]$$

(ii) For all $a > 0$, $\lim_{m \to \infty} \mathbb{E}_{x, x_1, \ldots, x_m \sim \mathcal{D}^{m+1}} \left[\sum_{i=1}^{m} w_i(x) \cdot \mathbf{1}\{\|x_i - x\| > a\} \right] = 0$

(iii) $\lim_{m \to \infty} \mathbb{E}_{x, x_1, \ldots, x_m \sim \mathcal{D}^{m+1}} \left[\max_{0 \leq i \leq m} w_i(x) \right] = 0$

Proof: Satisfying conditions (ii)–(iii)

- Condition (iii): $\max_{0 \le i \le m} w_i(x) = \max\left\{q_m, \underbrace{(1-q_m)^m}_{\le e^{-q_m m}}\right\} \to 0$ if $q_m \to 0$ and $q_m m \to \infty$

- Condition (ii):
  - Define $k_m = \left\lceil \sqrt{\frac{m}{q_m}} \right\rceil$. Note $\frac{k_m}{m} \to 0$ if $q_m m \to \infty$
  - Recall, if $k_m/m \to 0$, then $x_1, \ldots, x_{k_m} \to x$ in probability ... so $\|x - x_i\| < a$ for $i \le k_m$ We only need to consider sum over $i = (k_m + 1), \ldots, m$
  - **Exercise:** Assume $q_m m \to \infty$. Show that

$$\sum_{i=1}^{m} w_i(x) \cdot \mathbf{1}\{\|x_i - x\| > a\} \le \sum_{i=k_m+1}^{m} w_i(x) \le (1-q_m)^{k_m} \le e^{-\sqrt{q_m m}} \to 0$$

Proof: Satisfying condition (i)

- Since $w_i(x) = (1 - q_m)^{i-1} q_m$, we have $w_1(x) \geq w_2(x) \geq \ldots \geq w_m(x)$
  - Let $a_m = w_m(x)$ and $a_i = w_i(x) - w_{i+1}(x) \quad \implies w_i(x) = \sum_{j=i}^{m} a_j$

$$\mathbb{E}\left[\sum_{i=1}^{m} w_i(x) \cdot f(x_i)\right] = \mathbb{E}\left[\sum_{i=1}^{m} \sum_{j=i}^{m} a_j \cdot f(x_i)\right]$$

$$= \mathbb{E}\left[\sum_{j=1}^{m} a_j \sum_{i-1}^{j} f(x_i)\right]$$

$$= \sum_{j=1}^{m} a_j \underbrace{\mathbb{E}\left[\sum_{i-1}^{j} f(x_i)\right]}_{\leq C \cdot j \cdot \mathbb{E}[f(x)] \text{ by Stone's lemma}} \leq C \cdot \mathbb{E}[f(x)] \cdot \underbrace{\sum_{j=1}^{m} a_j j}_{= \sum_{i=1}^{m} w_i(x) \leq 1}$$

Optimal sample complexity with bagging

- Recall sample complexity $m_{\mathcal{H}}(\epsilon, \delta)$ = minimum number of samples needed to (ϵ, δ)-PAC learn $\mathcal{H}$ for any $\mathcal{D}$

$$\underbrace{C_1 \frac{d + \log_2(\frac{1}{\delta})}{\epsilon}}_{\text{impossible to PAC learn with fewer samples}} \leq m_{\mathcal{H}}(\epsilon, \delta) \leq \underbrace{C_2 \frac{d \ln(\frac{1}{\epsilon}) + \ln(\frac{1}{\delta})}{\epsilon}}_{\text{samples needed by ERM}}$$

- Can we match the lower bound? — Yes, using bagging (proof skipped)

Theorem En.3 (Larsen, *Bagging is an Optimal PAC Learner*, COLT 2023)

Let $VCdim(\mathcal{H}) = d$. *If* $m \geq C_3 \frac{d + \ln(\frac{1}{\delta})}{\epsilon}$, *then there is a bagging procedure that* (ϵ, δ)*-PAC learns* $\mathcal{H}$ *using* $T \geq 18 \ln\left(\frac{2m}{\delta}\right)$ *bootstrapped subsamples.*

Outline

- Voting and averaged predictors
  - Generalisation error bound for majority vote
  - If individual classifiers are consistent, then averaged classifier is also consistent
- Bagging
  - Ensemble of 1-NNs is universally consistent
  - Bagging leads to classifiers with optimal sample complexity (so better than ERM)
- Boosting
  - Weak learning: Performing better than random guessing (error $< \frac{1}{2}$ for binary case)
  - Adaboost: Combines weak learners to build a PAC learner; Achieves low training error

PAC (strong) learner

- $\mathcal{A}$ is a PAC (strong) learner for $\mathcal{H}$ if
  - there is function $m_{\mathcal{H}} : (0,1) \times (0,1) \to \mathbb{N}$

  such that for every
  - $\epsilon, \delta \in (0,1)$
  - $m \geq m_{\mathcal{H}}(\epsilon, \delta)$
  - distribution $\mathcal{D}$ that is realisable with respect to $\mathcal{H}$

$$\mathbb{P}_{S \sim \mathcal{D}^m} \left(L_{\mathcal{D}}(\mathcal{A}(S)) \leq \epsilon \right) \geq 1 - \delta$$

- Example: ERM over $\mathcal{H}$ if $\text{VCdim}(\mathcal{H}) < \infty$

γ-Weak learner for $\mathcal{H} \subseteq \{\pm 1\}^{\mathcal{X}}$

- Fix $\gamma \in (0, \frac{1}{2})$
- $\mathcal{A}$ is a γ-weak learner for $\mathcal{H}$ if
  - there is function $\tilde{m}_{\mathcal{H}} : (0,1) \to \mathbb{N}$

  such that for every
  - $\delta \in (0,1)$
  - $m \geq \tilde{m}_{\mathcal{H}}(\delta)$
  - distribution $\mathcal{D}$ that is realisable with respect to $\mathcal{H}$

$$\mathbb{P}_{S \sim \mathcal{D}^m} \left(L_{\mathcal{D}}(\mathcal{A}(S)) \leq \frac{1}{2} - \gamma \right) \geq 1 - \delta$$

Weak learnability vs PAC learnability

Strong and weak learnability

$\mathcal{H}$ is PAC (strong) learnable if there exists a PAC (strong) learner $\mathcal{A}$ for $\mathcal{H}$

$\mathcal{H}$ is γ-weak learnable if there exists a γ-weak learner $\mathcal{A}$ for $\mathcal{H}$

Why are we interested in weak learning?

- Weak vs strong learning
  - Strong learner: Almost accurate, test error $\leq \epsilon$
  - Weak learner: Better than random guessing, test error $\leq \frac{1}{2} - \gamma$
- Exercise: Argue that
  - $\mathcal{H}$ is PAC learnable $\implies$ $\mathcal{H}$ is γ-weak learnable for all $\gamma < \frac{1}{2}$
  - $\text{VCdim}(\mathcal{H}) < \infty$ $\implies$ ERM over $\mathcal{H}$ is also weak learner for $\mathcal{H}$
- We will use ERM over a simple class $\mathcal{H}'$ to weakly learn a complex class $\mathcal{H}$
  - Choose $\mathcal{H}'$ such that ERM is efficiently implementable

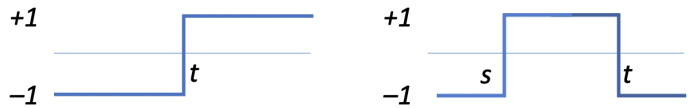
Weak learning with decision stumps

- Recall class of decision stumps over $\mathbb{R}$

$$\mathcal{H}_{ds-1} = \left\{ h_{t,b}(x) = b \cdot \text{sign}(x - t) \ : \ b \in \{\pm 1\}, t \in \mathbb{R} \right\}$$

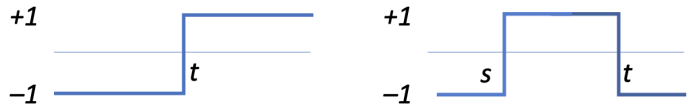
- Class of intervals over $\mathbb{R}$

$$\mathcal{H}_{int} = \left\{ h_{s,t,b}(x) = b \text{ for } x \in (s,t), \text{ and } -b \text{ otherwise } : \ b \in \{\pm 1\}, s, t \in \mathbb{R}, s < t \right\}$$



Weakly learning intervals using decision stumps

- Let $\mathcal{D}$ realisable with respect to $\mathcal{H}_{int}$
  - True labels $y = h_{s^*,t^*,b^*}(x)$ for some $h_{s^*,t^*,b^*} \in \mathcal{H}_{int}$
- Fitted stump will err at least one of three intervals $(-\infty, s^*]$, (s^*, t^*) or $[t^*, \infty)$
- Under marginal $\mathcal{D}_{\mathcal{X}}$, one of the intervals has probability $\leq \frac{1}{3}$
  - Let decision stump err on interval with smallest probability



Stumps better than random guessing for learning intervals

- Example: Assume $b^* = +1$ and $\mathbb{P}_{x \sim \mathcal{D}_{\mathcal{X}}}(x \geq t^*) \leq \frac{1}{3}$
  - Choose $h_{t,b}$ with $t = s^*, b = +1$
  - $L_{\mathcal{D}}(h_{t,b}) \leq \frac{1}{3}$ ... Verify formally
- Verify: For every true $h_{s^*,t^*,b^*} \in \mathcal{H}_{int}$,
  - there is some $h_{t,b} \in \mathcal{H}_{ds-1}$ with $L_{\mathcal{D}}(h_{t,b}) \leq \frac{1}{3}$
- We use ERM over $\mathcal{H}_{ds-1}$ to find above $h_{t,b}$
- Claim: ERM over $\mathcal{H}_{ds-1}$ weakly learns any $h_{s^*,t^*,b^*} \in \mathcal{H}_{int}$

$\frac{1}{12}$-Weak learning using ERM over $\mathcal{H}_{ds-1}$

- Take $\epsilon = \frac{1}{12}$ ... can take any $\epsilon < \frac{1}{6}$ (affects γ)
- $\mathcal{A}_{ERM_{ds-1}}$ = ERM over $\mathcal{H}_{ds-1}$
- For $m \geq m_{\mathcal{H}_{ds-1}}(\epsilon, \delta)$

$$\begin{aligned} L_{\mathcal{D}}\left(\mathcal{A}_{ERM_{ds-1}}(S)\right) &< L_{\mathcal{D}}(\mathcal{H}_{ds-1}) + \epsilon \qquad \text{with probability } 1-\delta \\ &\leq \frac{1}{3} + \frac{1}{12} \\ &= \frac{1}{2} - \frac{1}{12} \end{aligned}$$

- Define $\tilde{m}_{\mathcal{H}_{int}}(\delta) = m_{\mathcal{H}_{ds-1}}(\epsilon, \delta) \implies \mathcal{A}_{ERM_{ds-1}}$ $\frac{1}{12}$-weakly learns $\mathcal{H}_{int}$

Boosting

- Create a strong learner using several weak learners
  - Learn several weak learners, each perfectly fitting part of data
  - Use majority voting of weak learners
- Example: Majority vote of three decision stumps perfectly learns h_{s^*,t^*,b^*}
  - Some estimation error if we use ERM, but majority vote still PAC learns

Adaptive Boosting (AdaBoost)

- Use an iterative approach for boosting
- Learn a predictor h using weak learning
- Find parts of data that are not fitted well using h
- Learn new rule h' giving more importance to poorly learned data ...
- Iterate till all parts are learned well by some predictor

Weighted empirical risk

- Need to solve ERM giving more importance to poorly learned parts of data
- Weighted empirical risk of m samples
  - Probability weight vector $\mathbf{D} = (\mathbf{D}_1, \ldots, \mathbf{D}_m) \in [0,1]^m$ with $\sum_i \mathbf{D}_i = 1$
  - Weighted risk: $L_{\mathbf{D}}(h) = \sum_{i=1}^{m} \mathbf{D}_i \mathbf{1}\{h(x_i) \neq y_i\}$
  - $L_S(h) = L_{\mathbf{D}}(h)$ for $\mathbf{D} = (\frac{1}{m}, \ldots, \frac{1}{m})$
- Weighted ERM for specified training set S and weight $\mathbf{D}$:

$$\underset{h \in \mathcal{H}}{\text{minimise}}\ L_{\mathbf{D}}(h)$$

Key components of AdaBoost

- WL = γ-weak learner used to solve weighted ERM
- $h_1, h_2, \ldots$ = predictors learned in different iterations using WL
- $\mathbf{D}^{(t)}$ = weights on training samples used for learning h_t
  - $\mathbf{D}^{(1)} = (\frac{1}{m}, \ldots, \frac{1}{m})$
- Final predictor $h_{ada}(\cdot) = \text{sign}\left(\sum_{t=1}^{T} w_t h_t(\cdot)\right)$ $T = \#\text{iterations}$
  - Weighted majority vote
  - w_t depends on our trust of predictor h_t (high if h_t has low weighted risk)

Popular choices for w_t and $\mathbf{D}^{(t)}$

- Define $\epsilon_t = L_{\mathbf{D}^{(t)}}(h_t) \leq \frac{1}{2} - \gamma$
- Choice of w_t ... something that theoretically fits well

$$w_t = \frac{1}{2} \ln \left(\frac{1}{\epsilon_t} - 1 \right)$$

- Update of $\mathbf{D}_1^{(t)}, \ldots, \mathbf{D}_m^{(t)}$ ... exponential update reduces error faster

$$\mathbf{D}_i^{(t+1)} \propto \mathbf{D}_i^{(t)} \exp\big(- w_t \cdot \underbrace{y_i h_t(x_i)}_{\in\{\pm 1\}} \big)$$

  - Reduce relative weight of x_i if $h_t(x_i)$ is correct, else increase

AdaBoost

Input:

- Training samples $S = \{(x_1, y_1), \ldots, (x_m, y_m)\}$
- Weak learner WL that solves weighted ERM
- Number of rounds T

Initialisation:

- $\mathbf{D}^{(1)} = (\frac{1}{m}, \ldots, \frac{1}{m})$

AdaBoost

Iterations:

- For $t = 1, 2, \ldots, T$
  - $h_t =$ minimum weighted ERM solution obtained from WL
  - Compute $\epsilon_t = L_{\mathbf{D}_i^{(t)}}(h_t)$ and $w_t = \frac{1}{2} \ln\left(\frac{1}{\epsilon_t} - 1\right)$
  - Update $\mathbf{D}_i^{(t+1)} = \dfrac{\mathbf{D}_i^{(t)} \exp\left(-w_t y_i h_t(x_i)\right)}{\sum_{j=1}^{m} \mathbf{D}_j^{(t)} \exp\left(-w_t y_j h_t(x_j)\right)}$

Output:

- Weighted majority classifier $h_{ada}(\cdot) = \text{sign}\left(\sum_{t=1}^{T} w_t h_t(\cdot)\right)$

Performance of AdaBoost

- AdaBoost can overfit training data

$$L_S(h_{ada}) < \exp(-2\gamma^2 T) \qquad \dots \; L_S(h_{ada}) = 0 \text{ for } T = \frac{\ln m}{2\gamma^2}$$

- AdaBoost can still generalise
  - Fix $T = \frac{\ln m}{2\gamma^2}$, and assume WL is strong learner for class $\mathcal{B}$ with $\text{VCdim}(\mathcal{B}) = d_{\mathcal{B}}$

$$L_{\mathcal{D}}(h_{ada}) = O\left(\sqrt{\frac{d_{\mathcal{B}}(\ln m)^2 \log_2(d_{\mathcal{B}} \ln(m)) + \ln(\frac{1}{\delta})}{m}}\right) \qquad \text{with probability } 1-\delta$$

- Application: Viola-Jones face detection (Understanding ML book, Section 10.4)

Empirical risk of AdaBoost

Theorem En.4 (Upper bound on empirical risk of AdaBoost)

Assume, at every iteration, WL returns a hypothesis with $\epsilon_t \leq \frac{1}{2} - \gamma$

Choose $w_t = \frac{1}{2} \ln \left(\frac{1}{\epsilon_t} - 1 \right)$ *and* $\mathbf{D}_i^{(t+1)} \propto \mathbf{D}_i^{(t)} \exp\left(- w_t \cdot y_i h_t(x_i) \right)$

Then, after T *iterations,*

$$L_S(h_{ada}) \quad = \quad \frac{1}{m} \sum_{i=1}^{m} \mathbf{1}\left\{ h_{ada}(x_i) \neq y_i \right\} \quad < \quad \exp(-2\gamma^2 T)$$

$$\text{Choosing } \; T = \left\lceil \frac{\ln m}{2\gamma^2} \right\rceil \qquad \Longrightarrow \qquad L_S(h_{ada}) < \frac{1}{m} \qquad \Longrightarrow \qquad L_S(h_{ada}) = 0$$

Proof (see Understanding ML, Theorem 10.2)

- Verify $\mathbf{1}\{h(x) \neq y\} \leq e^{-yh(x)}$. So

$$L_S(h_{ada}) \leq \frac{1}{m} \sum_{i=1}^{m} e^{-y_i h_{ada}(x_i)}$$

- Define $f_t(\cdot) = \sum_{p \leq t} w_p h_p(\cdot)$ ... $f_T = h_{ada}$, $f_0 =$ zero function

- Define $Z_t = \frac{1}{m} \sum_{i=1}^{m} e^{-y_i f_t(x_i)}$ ... $Z_0 = 1$, $L_S(h_{ada}) \leq Z_T$

- Enough to show: $Z_T \leq e^{-2\gamma^2 T}$

Proof sketch

$$Z_T = \frac{Z_T}{Z_0} = \frac{Z_1}{Z_0} \cdot \frac{Z_2}{Z_1} \cdot \ldots \cdot \frac{Z_T}{Z_{T-1}}$$

- Enough to show: $\frac{Z_{t+1}}{Z_t} \leq e^{-2\gamma^2}$ for every $t = 0, 1, \ldots$

$$\frac{Z_{t+1}}{Z_t} = \frac{\sum_i e^{-y_i f_{t+1}(x_i)}}{\sum_j e^{-y_j f_t(x_j)}} = \frac{\sum_i e^{-y_i f_t(x_i)} e^{-y_i w_{t+1} h_{t+1}(x_i)}}{\sum_j e^{-y_j f_t(x_j)}}$$

$$= \sum_i \frac{e^{-y_i f_t(x_i)}}{\sum_j e^{-y_j f_t(x_j)}} e^{-y_i w_{t+1} h_{t+1}(x_i)}$$

Proof sketch

- Probability weight updates

$$\mathbf{D}_i^{(t+1)} \propto \mathbf{D}_i^{(t)} e^{-w_t y_i h_t(x_i)} \propto \mathbf{D}_i^{(t-1)} e^{-w_{t-1} y_i h_{t-1}(x_i)} e^{-w_t y_i h_t(x_i)}$$

$$\ldots \propto \underbrace{\mathbf{D}_i^{(1)}}_{=1/m} \underbrace{e^{-\sum_{p \le t} w_p y_i h_p(x_i)}}_{=e^{-y_i f_t(x_i)}}$$

- $\mathbf{D}_i^{(t+1)} \propto e^{-y_i f_t(x_i)}$ and $\sum_{i=1}^{m} \mathbf{D}_i^{(t+1)} = 1$

$$\implies \mathbf{D}_i^{(t+1)} = \frac{e^{-y_i f_t(x_i)}}{\sum_j e^{-y_j f_t(x_j)}} \implies \frac{Z_{t+1}}{Z_t} = \sum_{i=1}^{m} \mathbf{D}_i^{(t+1)} e^{-y_i w_{t+1} h_{t+1}(x_i)}$$

Proof sketch

- From definition of w_t, we get $e^{w_{t+1}} = \sqrt{\frac{1}{\epsilon_{t+1}} - 1}$

- Also $\epsilon_{t+1} = L_{\mathbf{D}^{(t+1)}}(h_{t+1}) = \sum_{i=1}^{m} \mathbf{D}_i^{(t+1)} \mathbf{1}\{h_{t+1}(x_i) \neq y_i\} = \sum_{i\,:\,y_i h_{t+1}(x_i) = -1} \mathbf{D}_i^{(t+1)}$

$$
\begin{aligned}
\frac{Z_{t+1}}{Z_t} &= \sum_{i=1}^{m} \mathbf{D}_i^{(t+1)} e^{-y_i w_{t+1} h_{t+1}(x_i)} \\
&= e^{w_{t+1}} \sum_{i: y_i h_{t+1}(x_i) = -1} \mathbf{D}_i^{(t+1)} \quad + \quad e^{-w_{t+1}} \sum_{i: y_i h_{t+1}(x_i) = 1} \mathbf{D}_i^{(t+1)} \\
&= e^{w_{t+1}} \epsilon_{t+1} + e^{-w_{t+1}} (1 - \epsilon_{t+1}) \\
&= 2\sqrt{\epsilon_{t+1}(1 - \epsilon_{t+1})}
\end{aligned}
$$

Proof sketch

$$
\begin{aligned}
\frac{Z_{t+1}}{Z_t} &= 2\sqrt{\epsilon_{t+1}(1-\epsilon_{t+1})} \\
&\leq 2\sqrt{(\tfrac{1}{2}-\gamma)(\tfrac{1}{2}+\gamma)} && \dots\ \epsilon_{t+1} \leq \tfrac{1}{2}-\gamma, \text{ and } a(1-a) \text{ increases for } a \leq \tfrac{1}{2} \\
&= \sqrt{1-4\gamma^2} \\
&< \sqrt{e^{-4\gamma^2}} && \dots\ 1-a \leq e^{-a}\ (< e^{-a} \text{ if } a > 0) \\
&= e^{-2\gamma^2}
\end{aligned}
$$

Generalisation error for AdaBoost

Theorem En.5 (Generalisation error bound for AdaBoost)

Assume WL is strong learner for base hypothesis class $\mathcal{B}$ with $\mathrm{VCdim}(\mathcal{B}) = d_{\mathcal{B}}$

Assume T is odd and AdaBoost satisfies $L_S(h_{ada}) < \exp(-2\gamma^2 T)$

With probability $1 - \delta$,

$$L_{\mathcal{D}}(h_{ada}) < \exp(-2\gamma^2 T) + C\sqrt{\frac{d_{\mathcal{B}} T \log_2(d_{\mathcal{B}} T) \ln m + \ln(\frac{1}{\delta})}{m}} \qquad C = \text{ constant}$$

$$T = \left\lceil \frac{\ln m}{2\gamma^2} \right\rceil \qquad \Longrightarrow \qquad L_{\mathcal{D}}(h_{ada}) < C\sqrt{\frac{d_{\mathcal{B}} (\ln m)^2 \log_2\left(\frac{d_{\mathcal{B}} \ln(m)}{2\gamma^2}\right) + \ln\left(\frac{1}{\delta}\right)}{2\gamma^2 m}}$$

Boosting in agnostic setting

- So far: PAC learn $\mathcal{H}$ using γ-weak learners for $\mathcal{H}$
  - Key assumption: $\mathcal{D}$ is realisable w.r.t $\mathcal{H}$, that is, $L_{\mathcal{D}}(\mathcal{H}) = \inf_{h \in \mathcal{H}} L_{\mathcal{D}}(h) = 0$
- Ben-David et al., *Agnostic Boosting*, COLT 2001:
  - Given access to a γ-weak agnostic learner, a variant of AdaBoost returns $\widehat{h}$ such that

$$L_{\mathcal{D}}(\widehat{h}) \leq c_\gamma \cdot \left(L_{\mathcal{D}}(\mathcal{H})\right)^{c'_\gamma} + \epsilon \qquad \epsilon \text{ can be made small, } c_\gamma, c'_\gamma \text{ depend only on } \gamma$$

- Diakonikolas et al., *Boosting in the Presence of Massart Noise*, COLT 2021:
  - Noise PAC setting: $h^* \in \mathcal{H}$, but noisy labels $err(x) = \mathbb{P}(y \neq h^*(x)|x) \in [0, \eta]$, $\eta < 1/2$
  - Given access to a (α, γ)-Massart weak learner, a boosting algorithm returns $\widehat{h}$ such that

$$L_{\mathcal{D}}(\widehat{h}) \leq \eta \cdot (1 + O(\alpha))$$