Bias-Variance Tradeoff

$$\underbrace{\mathbb{E}[(h_D(x)-y)^2]}_{\text{test error}} = \underbrace{\mathbb{E}[(h_D(x)-\bar{h}(x))^2]}_{\text{variance}} + \underbrace{\mathbb{E}[(\bar{h}(x)-\bar{y}(x))^2]}_{\text{bias}} + \underbrace{\mathbb{E}[(\bar{y}(x)-y)^2]}_{\text{noise}}$$

Bagging (Breiman, 96)

Bagging Reduces Variance

$h_D \longrightarrow \bar{h}$

Weak Law of Large Numbers

i.i.d RVs x_i $\mathbb{E}[x_i] = \bar{x}$

$$\frac{1}{m}\sum_{i=1}^{m} x_i \longrightarrow \bar{x} \text{ as } m \to \infty$$

i.i.d. $D_1, \dots, D_m \sim P(X,Y)$

$h_1, \dots, h_m$

$$\frac{1}{m}\sum_{i=1}^{m} h_i \longrightarrow \bar{h} \text{ as } m \to \infty$$

Bagging Algo.

1. Sample m datasets: $D_1, \dots, D_m$ from P w.a.r. o/ replacement
2. For each D_i train $h_i(\,)$
3. $\hat{h}(x) = \frac{1}{m}\sum_{i=1}^{m} h_i(x)$

Adv

- Predicted var → classifier uncertainty
- unbiased of test error:

$(x_i, y_i) \in D \quad S_i = \{D_k \mid (x_i, y_i) \notin D_k\}$

$$\hat{h}_i(x) = \frac{1}{|S_i|}\sum_{k \in S_i} h_k(x)$$

out-of-bag error

$$\varepsilon_{oob} = \frac{1}{n}\sum_{(x_i,y_i)\in D} \left(\hat{h}_i(x_i) - y\right)^2$$

Solution: Bagging (Bootstrap Aggregating)

$Q(X,Y|D) \simeq P(X,Y)$

$Q((x_i,y_i)|D) = \frac{1}{n} \quad \forall (x_i,y_i) \in D$

$D_i \sim Q \qquad n = |D|$

Claim: Samples from Q aren't i.i.d. are from P

Proof

$P(X=x_i) = p_i \qquad \Omega = x_1, \dots, x_n$

$Q(X=x_i|D) =$

$D = \{x_1, x_2, \underbrace{x_i \dots x_i}_{k\text{-times}}, \dots x_n\}$

$$\sum_{k=1}^{n} \underbrace{\binom{n}{k} p_i^k (1-p_i)^{n-k}}_{\text{prob. of } k \text{ copies of } x_i \text{ in } D} \times \underbrace{\frac{k}{n}}_{\text{prob. of picking } x_i}$$

$$= \frac{1}{n}\sum_{k=1}^{n} \binom{n}{k} p_i^k (1-p_i)^{n-k} k$$

$\mathbb{E}[B(p_i, n)] = p_i n$

$= \frac{1}{n} p_i n$

$= p_i \quad \square$

Random Forests

Algo.

1. Sample $D_1, \dots, D_m$ u.a.r w/ replacement
2. Train h_j (decision tree) for D_j ← variance
   - randomly subsample $k \leq d$ features (w/o replacement)
3. $\hat{h}(x) = \frac{1}{m}\sum_{j=1}^{m} h_j(x)$

Adv.

1. 2 hyperparams: ⓐ m ⓑ k $= \sqrt{d}$
   - ⓐ → as large as comp. possible
2. Not much preprocessing necessary for DTs

Variants

1. $D_j = \{D_j^A, D_j^B\}$
   - D_j^A → build the tree
   - D_j^B → make predictions at leaves
   - * statistically consistent
2. Don't grow every tree fully
   - prune based on leave-out