Rapid Introduction to Machine Learning/ Deep Learning

Hyeong In Choi

Seoul National University

Lecture 3b
Aggregation and randomization

October 24, 2015

Table of contents

1. Objectives of Lecture 3b

Objective 1

Learn how predictor (regression function, classifier) depends on data

Objective 2

How to view data and its probability

Objective 3

Learn the bias-variance decomposition and how it sheds light on our understanding of errors

Objective 4

Learn the general framework of testing, validation and the model selection

Objective 5

How to use validation to get an estimate of generalization error

2. Bootstrap

2.1. Aggregation

Recall

- Regression

$$\varphi_a(x) = E_{\mathcal{D}}\, \varphi^{(\mathcal{D})}(x) = \int \varphi(x, \mathcal{D}) dP_{\mathcal{D}}$$

- Classification

$$\varphi_a(x) = \underset{y}{\operatorname{argmax}} P_{\mathcal{D}}(\varphi^{(\mathcal{D})}(x) = y)$$

Trouble with aggregation

- Aggregation is an operation on the set of data sets
- But in reality, there is only one data set available
- If there were multiple data sets, one may as well combine them all to create a single bigger data set

Bootstrap

Bootstrap is a way of artificially creating a multitude of data sets out of a single given one

2.2. Digression: statistic

Descriptive statistic

- A (descriptive) *statistic* is a function

$$S : \mathcal{D} \to \mathbb{R}^k$$

- If $\mathcal{D} = \{z_t\}_{t=1}^N$, we write

$$s(\mathcal{D}) = s(z_1, \cdots, z_N),$$

  where $z_t \in \mathfrak{Z}$
- s is used to estimate a parameter θ
  $\hat{\theta} = s(z)$ is an estimator of θ
- For rotational convenience, we use $z = (z_1, \cdots, z_N)$ in lieu of $\mathcal{D}$

Example

- mean

$$\hat{\theta} = \bar{z} = s(z) = \frac{1}{N}(z_1 + \cdots + z_N)$$

- variance

$$\hat{\theta} = \mathrm{Var}(z) = \frac{1}{N-1}\sum_{t=1}^{N}(z_i - \bar{z})^2$$

What about the confidence interval?

- If the probability distribution is known, one can estimate the confidence interval
- But what if no information an the probability distribution is known?

2.3. Bootstrap resampling

Data and empirical distribution

- Data $\mathcal{D} = \{z_t\}_{t=1}^{N}$
- Empirical (probability) distribution

$$\hat{F}(z) = \frac{1}{N} \sum_{t} \delta_{z_t}(z) \quad \text{for } z \in \mathfrak{Z}$$

$$\hat{F}(A) = \frac{1}{N} |\{t : z_t \in A\}| \quad \text{for } A \subset \mathfrak{Z}$$

Bootstrap resampling

- Out of $\mathcal{D} = \{z_1, \cdots, z_N\}$, or $z = (z_1, \cdots, z_N)$, draw the elements N times but with repetition
- Such set of N elements (with repetition) is called a bootstrap resample and is denoted by $z^* = (z_1^*, \cdots, z_N^*)$
- Some element of $\mathcal{D}$ may be picked many times and some not at all
- The ratio of elements left unpicked $\to 1/e \approx 36.8\%$ as $N \to \infty$
- The total number of distinct bootstrap resample is asymptotic to $\frac{1}{\sqrt{\pi N}} 2^{2N-1}$ as $N \to \infty$

Bootstrap schematics

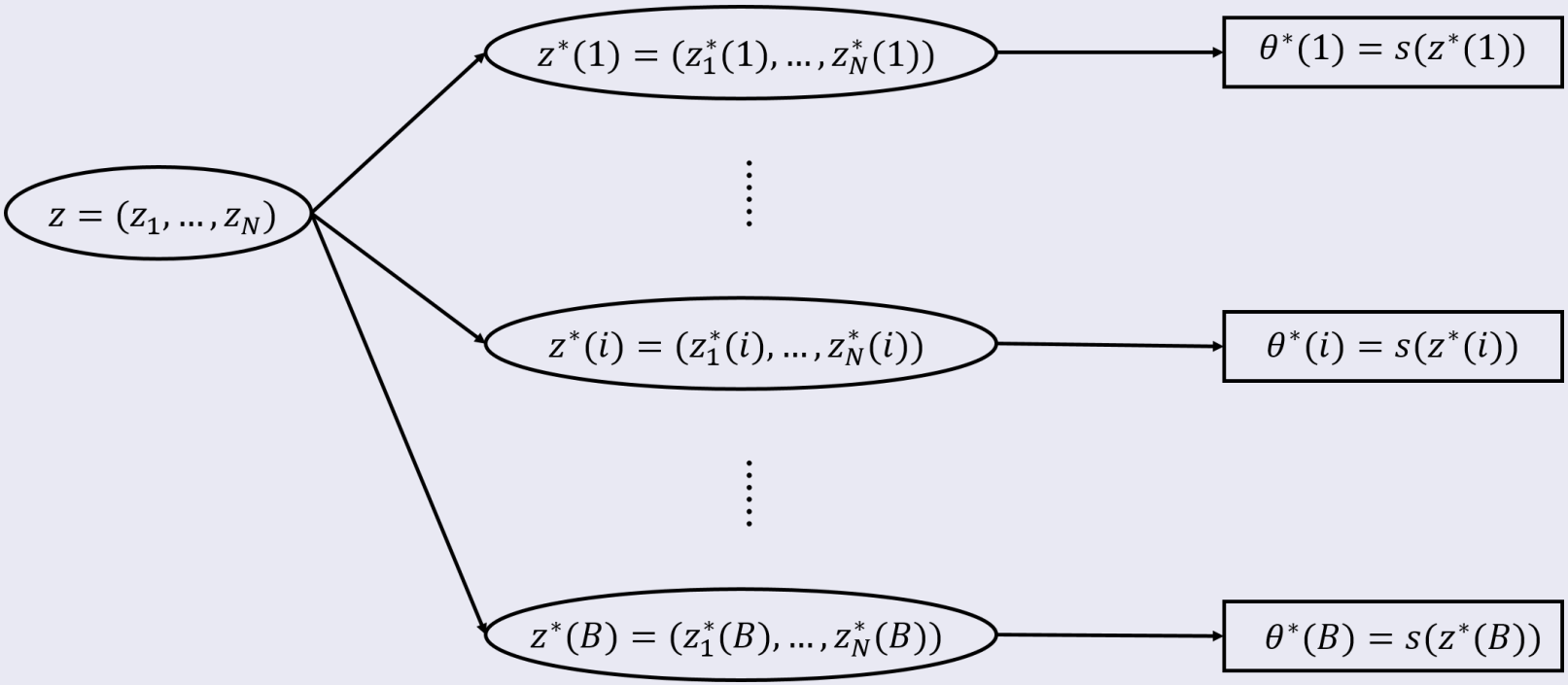
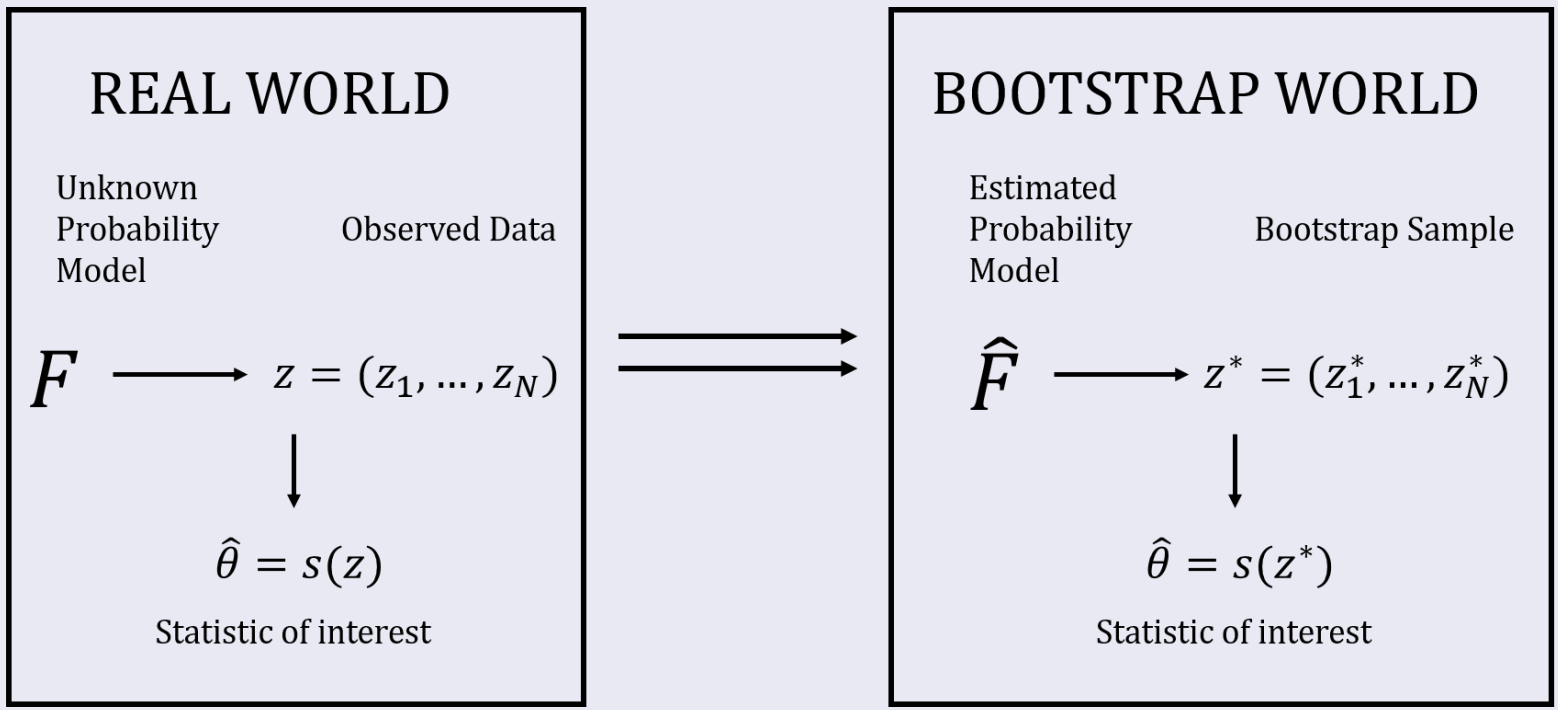


Figure: Bootstrap procedure (repeated B times)

Real vs. bootstrap world



Bootstrap confidence interval

- Draw the histogram of $\hat{\theta}^*(1), \cdots, \hat{\theta}^*(B)$

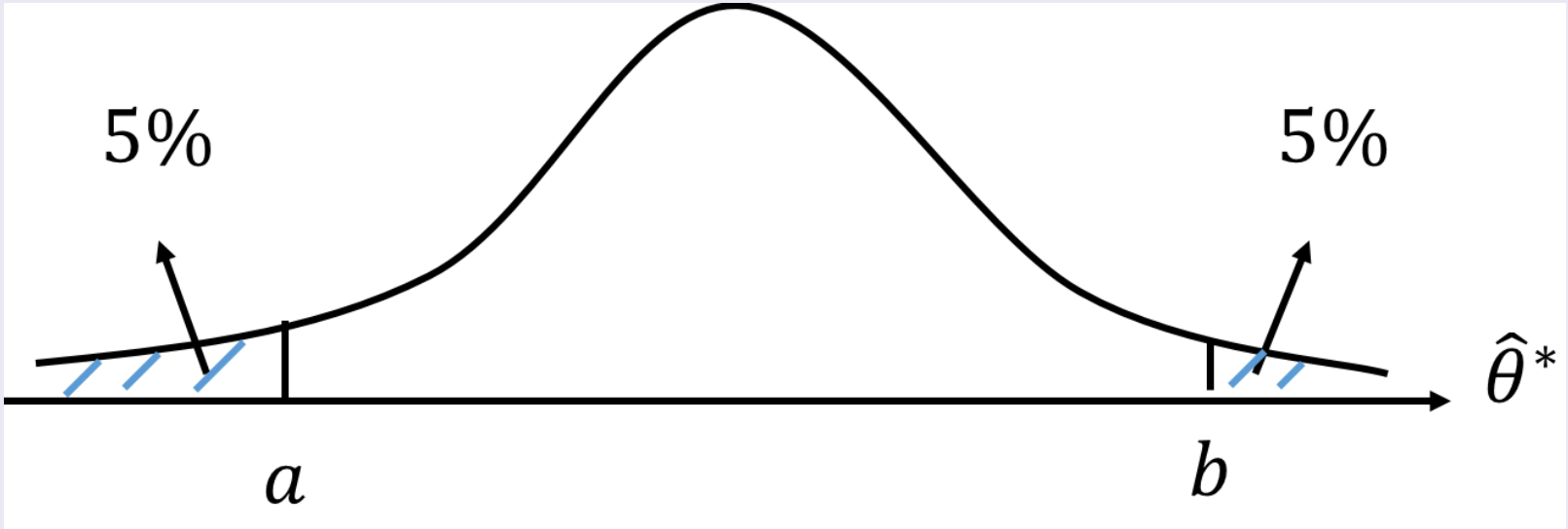


Figure: Histogram of $\hat{\theta}^*$

Theorem (Bickel and Freedman)

Assume that F has a finite variance σ^2. Let $x_1, \cdots, x_n$ be an IID sample $\sim F$. Define the usual sample mean and variance by

$$\begin{aligned} \mu_n &= \frac{1}{n}\sum_{i=1}^{n} x_i \\ s_n^2 &= \frac{1}{n}\sum_{i=1}^{n}(x_i - \mu_n)^2. \end{aligned}$$

Let $x_1^*, \cdots, x_m^*$ be a bootstrap resample $\sim \hat{F}$. (Note that here, m may differ from n.) Define the mean and the variance of the resample by

$$\mu_m^* = \frac{1}{m}\sum_{i=1}^{m} x_i^*$$
$$(s_m^*)^2 = \frac{1}{m}\sum_{i=1}^{m}(x_i^* - \mu_m^*)^2.$$

Then,

(a) $\sqrt{m}(\mu_m^* - \mu_n)$ weakly converges to $N(0, \sigma^2)$, as $n, m \to \infty$;

(b) $P(|s_m^* - \sigma| > \epsilon)$ almost surely converges to 0, as $m \to \infty$.

3. Bagging
3.1. Efficacy of aggregation

Regression

- $$\begin{aligned} E_{\mathcal{D}} |y - \varphi^{(\mathcal{D})}(x)|^2 &= |y|^2 - 2y \cdot E_{\mathcal{D}} \varphi^{(\mathcal{D})}(x) + E_{\mathcal{D}} |\varphi^{(\mathcal{D})}(x)|^2 \\ &= |y|^2 - 2y \cdot \varphi_a(x) + E_{\mathcal{D}} |\varphi^{(\mathcal{D})}(x)|^2 \end{aligned}$$

  $$|\varphi_a(x)|^2 = |E_{\mathcal{D}} \varphi^{(\mathcal{D})}(x)|^2 \leq E_{\mathcal{D}} |\varphi^{(\mathcal{D})}(x)|^2$$

  Therefore

  $$|y - \varphi^{(\mathcal{D})}(x)|^2 \leq E_{\mathcal{D}} |y - \varphi^{(\mathcal{D})}(x)|^2$$

- The loss of $\varphi_a(x)$ is always less than the expected (w.r.t $P_{\mathcal{D}}$) loss of any individual predictor

Classification

- Fix $\mathcal{D}$, hence $\varphi^{(\mathcal{D})}$. Let

$$Q(j|x) = P(\varphi^{(\mathcal{D})}(x) = j|x)$$

  This is the probability that $\varphi^{(\mathcal{D})}$ predicts that the label of x is j

- $$r = \int \sum_j Q(j|x)P(j|x)dp(x)$$

  is the probability of correct classification by $\varphi^{(\mathcal{D})}$
- Recall the optimal (Bayes) classifier

$$f(x) = \max_j P(j|x)$$

Let

$$r^* = \int f(x) dP(x) = \int \max_j P(j|x) dx$$

the correct classification rate of the optimal classifier

- r^* is the highest possible correct classification rate

Definition

We say the predictor φ defined by (**??**) is order correct for the input x if

$$\operatorname*{argmax}_j Q(j|x) = \operatorname*{argmax}_j P(j|x).$$

Remark

Order-correct predictor can be quite worse than the Bayes predictor. For example, let

$$P(1|x) = 0.9 \qquad P(2|x) = 0.1$$
$$Q(1|x) = 0.6 \qquad Q(2|x) = 0.4.$$

Then φ is order-correct, but the correct classification rate for x is seen to be

$$Q(1|x)P(1|x) + Q(2|x)P(2|x) = 0.58.$$

But for the Bayes classifier, the correct classification rate is 0.9.

Theorem (Breiman)

If individual classifies are reasonable in the sense that they are order-correct for most of inputs, then the correct classification rate r_A of the aggregate classifier approaches the optimal rate r^*

- Order-correctness is far from optimal, but though aggregation, one can achieve very high correct classification rate

3.2. Bagging

Bagging procedure

- Given a data set $\mathcal{D} = z^{(t)}{}_{t=1}^{N}$, create B bootstrap resamples $\mathcal{D}_1, \cdots, \mathcal{D}_B$
- Use these resamples, construct
  - Regression

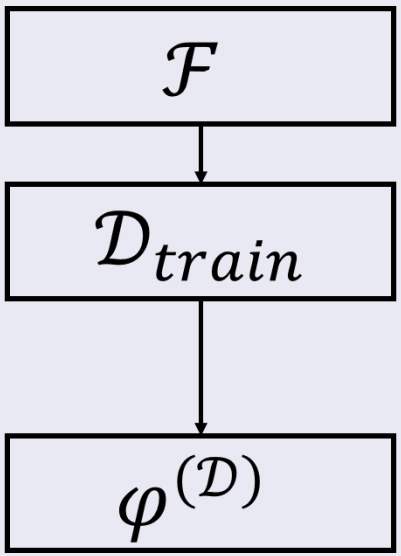
$$\varphi_b(x) = \frac{1}{B} \sum_{k=1}^{B} \varphi^{(\mathcal{D}_k)}(x)$$

  - classification

$$\varphi_b(x) = \underset{j}{\operatorname{argmax}} |\{k : \varphi^{(\mathcal{D}_k)}(x) = j\}|$$

3.2.2

- These bagging predictors converge to the corresponding aggregate predictors at $B \to \infty$
- In practice, to reap the benefit of bagging the bootstrap resamples $\mathcal{D}_1, \cdots, \mathcal{D}_B$ should have smaller correlation

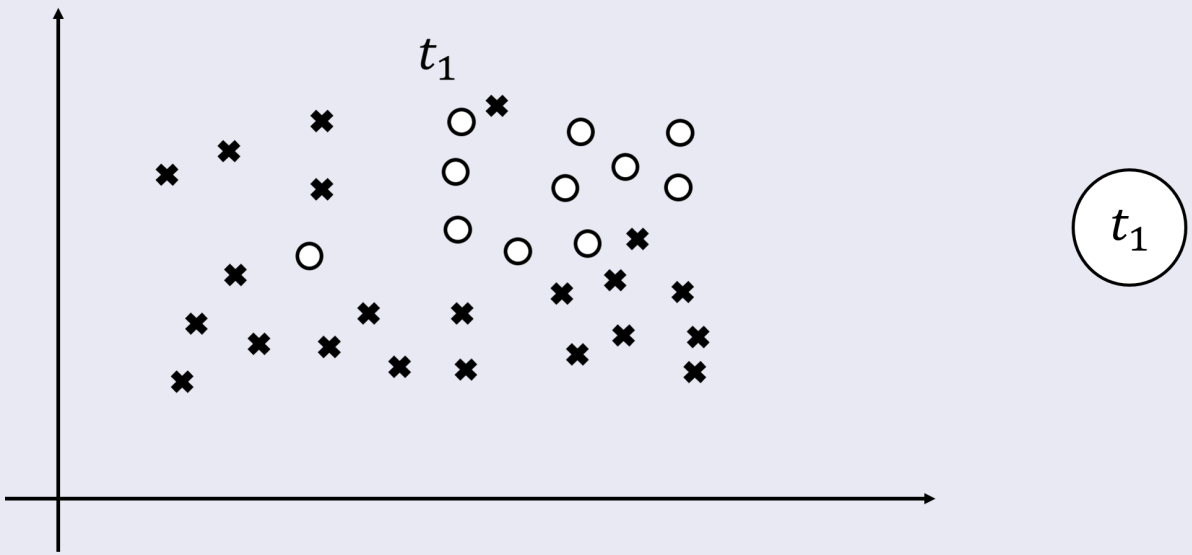


- $\varphi^{(\mathcal{D})}$ is applied to $\mathcal{D}_{test}$ to get the error estimate: a proxy for generalization error

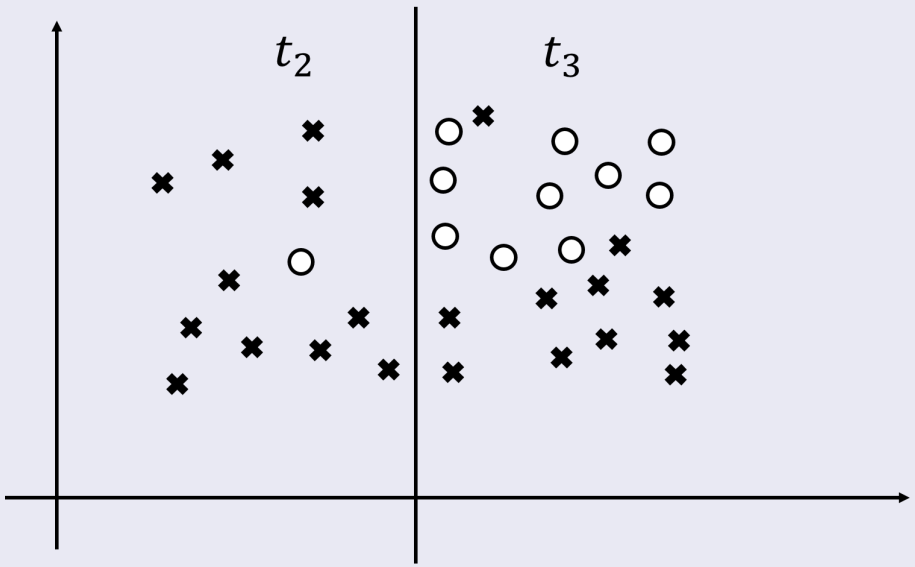
4. CART

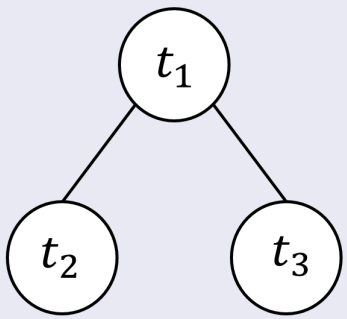
4.1. Idea of CART

4.1.1.

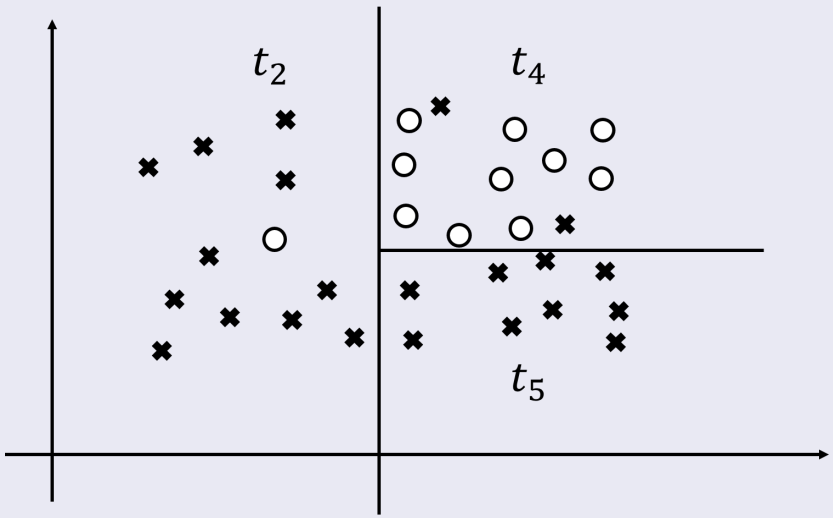


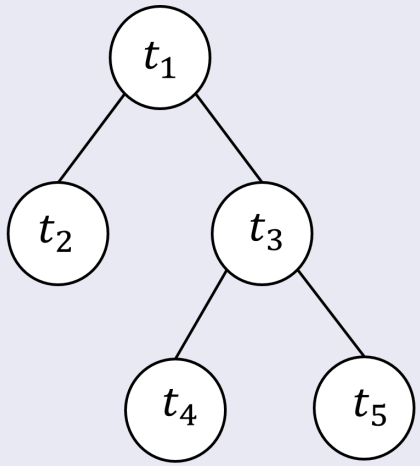
4.1.2.





4.1.3.





4.1.4

Each corresponds to a region in $\mathfrak{X}$ and the tree to a partition of $\mathfrak{X}$

4.2. Splitting and pruning

Splitting

- Splitting is done in such a way that the resulting child nodes have higher purity (i.e., split to result in more discriminate power)

Pruning

- Two many splitting result in overfitting problem
- One has to prune back the tree branches

4.3. Predictor

Definition

Let $i, j \in \mathcal{Y}$ be labels. A cost $C(i|j)$ of misclassifying class j as class i is defined to be a nonnegative number satisfying $C(j|j) = 0$ for all $j \in \mathcal{Y}$.

Definition

Let t be a node. The class label $j^*(t)$ of t is defined to be

$$j^*(t) = \operatorname*{argmin}_i \sum_j C(i|j)p(j|t).$$

In case there is a tie, tie breaking is done by some arbitrary rule.

- $\sum_j C(i|j)P(j|t)$ is the expected cost of misclassification

5. Random forests

5.1. Random forest recipe

Random forest recipe

- Fix $m << d$ (typically $m = \sqrt{d}$)
  Fix N_{min} and B
- Recipe
  Do for $k = 1$ to B
  - Draw a bootstrap resample $\mathcal{D}^{(k)}$ from $\mathcal{D}$. Let $OOB(k)$ be the set of data points not in $\mathcal{D}^{(k)}$. ("OOB" stands for "out of bag".) Thus we have the disjoint union:

$$D = \mathcal{D}^{(k)} \cup OOB(k)$$

  - Grow a tree T_k using the x-component of $\mathcal{D}^{(k)}$ by applying the following splitting rule:

  - At each node, randomly select m features (out of total d features)
  - Do the split on these m features using some impurity measure (Gini, entropy, misclassification rate,etc.)
  - Stop splitting the node if it contains fewer tree $N_{\min}$ element
  - Do not prune
- From each tree T_k, get the predictor φ_k for $k = 1, \cdots, B$.

Get the final predictor $\zeta(x)$:

- For regression:

$$\zeta(x) = Av_k \varphi_k(x) = \frac{1}{B} \sum_{k=1}^{B} \varphi_k(x)$$

- For classification:

$$\zeta(x) = Plur_k \varphi_k(x) = \underset{j \in \mathcal{Y}}{\operatorname{argmax}} |\{k : \varphi_k(x) = j\}|.$$

5.2. Why do random forests work?

Breiman

The smaller the correlation between the splits among the nodes, the more accurate the random forests predictor becomes

5.2.2

The random forests recipe is designed to reduce such correlation

5.3. Some thoughts on bias-variance tradeoff

Random forests

- RF has very small bias
- RF reduces the variance

Boosting

- Boosting reduces bias