CS 307 Lecture 9

Decision Trees (for classification)

Regression

- Full binary tree

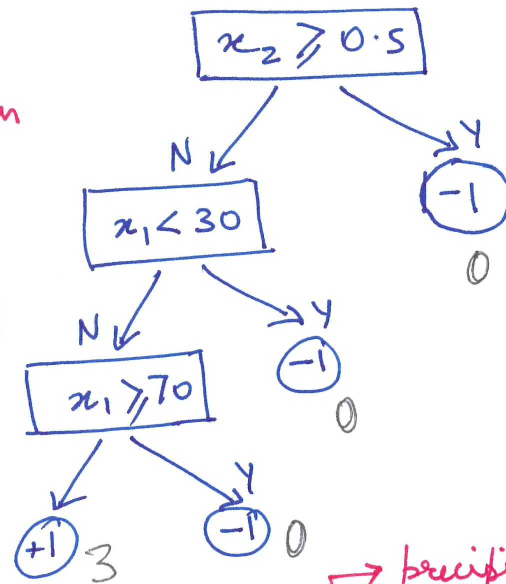
  Every vertex has either 0 children (leaves) or 2 children]– internal nodes/vertices

- Label of internal nodes:

  Feature compared with threshold

- Label of leaves: +1, −1.

  $\in \mathbb{R}$



Input: (x_1, x_2)

x_1 → Temperature, x_2 → precipitation

Decision Tree (S, k)

if $|S| \leq k$:

  return Leaf(label $= \underset{a \in \{+1,-1\}}{\operatorname{argmax}} \hat{p}_S(a)$)

$\hat{p}_S(a) = \dfrac{|\{(x,y) \in S \mid y = a\}|}{|S|}$

[Leaf(label $= \frac{1}{n} \sum_{i=1}^{n} y^{(i)}$]

else:

  for each j, θ:

$x_j^{(1)} < x_j^{(2)} < \dots < x_j^{(n)}$

$S_N^{j,\theta} = \{(x,y) \mid x_j < \theta\}$ $y_N = \frac{1}{|S_N|} \sum_{i \in S_N} y^{(i)}$

$S_Y^{j,\theta} = \{(x,y) \mid x_j \geq \theta\}$ $y_Y = \frac{1}{|S_Y|} \sum_{i \in S_Y} y^{(i)}$

$C(j,\theta) = \left(1 - \max_{a \in \{+1,-1\}} \hat{p}_{S_N}(a)\right) + \left(1 - \max_{a \in \{+1,-1\}} \hat{p}_{S_Y}(a)\right)$

$\sum_{i \in S_N} (y_N - y^{(i)})^2 + \sum_{i \in S_Y} (y_Y - y^{(i)})^2$

$j^*, \theta^* = \operatorname{argmin} C(j,\theta)$

return Node(label="$j^* \geq \theta^*$", DecisionTree$(S_N^{j^*,\theta^*}, k)$,

  DecisionTree$(S_Y^{j^*,\theta^*}, k)$)

Decision Trees are prone to overfitting

Bagging : ~~Boosting~~ Bootstrapping Aggregation

Bootstrapping ~~Boosting~~

- Given training set S, construct training sets $S^{(1)}, S^{(2)} \dots S^{(k)}$ by picking (with replacement) m examples from S
- Build decision tree $T^{(1)}, T^{(2)} \dots T^{(k)}$ on $S^{(1)} \dots S^{(k)}$.

Aggregating

- On a new example (x)

$$\hat{y} = \text{maj}_k\, T^{(i)}(x) \quad \text{(classification)}$$

$$\hat{y} = \frac{1}{k}\sum_{i=1}^{k} T^{(i)}(x) \quad \text{(regression)}$$

Random Forests

- Given Training set S, construct training sets $S^{(1)} \dots S^{(k)}$ by picking (with replacement) m examples from S.
- Build a decision tree $T^{(i)}$ on $S^{(i)}$ as follows.
  - ~~For~~ At each stage of the decision tree construction pick a random subset of features I, and you use one of the features in I to split.
- On a new example x.
  "Aggregate" the outputs of $T^{(i)}(x)$.

k-Nearest Neighbours :

No hypothesis constructed from the training set.

On a new example x. $S = \{(x^{(1)}, y^{(1)}) \dots (x^{(n)}, y^{(n)})\}$.

- Compute permutation of S say $(x^{\pi(1)}, y^{\pi(1)}) \dots (x^{\pi(n)}, y^{\pi(n)})$ such that

$$d(x, x^{\pi(i)}) \leq d(x, x^{\pi(i+1)}) \quad \left[d(x,y) = \sqrt{\sum_{j=1}^{d} (x_j - y_j)^2} \right.$$

- Output $y = \text{Maj}(y^{\pi(1)}, y^{\pi(2)} \dots y^{\pi(k)})$ (classification)

$$y = \frac{1}{k} \sum_{i=1}^{k} y^{\pi(i)} \quad \text{(regression)}$$

Examples that are close by have outputs that are close.

c-Lipschitz : $c \, d(x^{(i)}, x^{(j)}) \geq d(y^{(i)}, y^{(j)})$