

STAT 151A Lecture 41

Henry Liev

5 December 2025

Remark 0.1 (Fitting Regression Trees)

- (1) Repeated partition data into separate different y -values
- (2) Stop and use the mean of each group as predictions for members of this partition

Remark 0.2 (How to Choose Splits?)

→ Can't search over all possible trees (too many)

→ Instead use a greedy search

→ Minimize ErrSS

Pick a variable \vec{x}_j and a cutpoint c

Let $R_1(j, c) = \{i : x_{ij} < c\}$

$R_2(j, c) = \{i : x_{ij} \geq c\}$

$ErrSS(j, c) = \sum_{i \in R_1(j, c)} (y_i - \bar{y}_{R_1(j, c)})^2 + \sum_{i \in R_2(j, c)} (y_i - \bar{y}_{R_2(j, c)})^2$

For all $j = 1, \dots, p$

For all c , that produce different splits compute ErrSS(j,c)

Choose the split with the lowest ErrSS

Repeat subdividing individual split groups, ErrSS will always go down with more splits

Remark 0.3 (Better Approach)

- (1) Start by growing a very big tree with many, many splits (e.g. until no more than k observations per leaf)
- (2) Then prune the tree back down to make it smaller and (hopefully) overfit less
Basic approach: Consider a penalized ErrSS = $\sum_{l \in \text{leaves}} \sum_{i \in l} (y_{il} - \bar{y}_l)^2 + \alpha |T|$
Now evaluate full tree and all of its subleaves using penalized quantity as $\alpha \rightarrow \infty$
Produce a sequence of gradually smaller trees as $\alpha \rightarrow \infty$
Algorithm: “Weakest Link Cutting” Algorithm
- (3) We now have a list of trees of increasing complexity: pick the one with the lowest CV Error
What about categorical variables?
Categorical \vec{x}_j : can’t use splits of form: $\{x_{ij} < c\}, \{x_{ij} \geq c\}$
$$x_{ij} \in \begin{pmatrix} Red \\ Green \\ Blue \end{pmatrix}$$

Instead search over all possible ways to divide categories into 2 bins eg:
Search over $\{\{R, G\}, \{B\}, \{R\}, \{G, B\}, \{R, B\}, \{G\}\}$

Remark 0.4 (Categorical \vec{y} /Classification Trees)

What needs to change?

- (1) How do we get a partition from a leaf?
Must common: majority vote at leaf
- (2) Splitting criterion \rightarrow ErrSS is not well-suited for binary y ’s

New method: k categories for y

At leaf l , let $\bar{p}_{l1}, \dots, \bar{p}_{lK}$ be proportions observations in category $1, 2, \dots, K$ (so $\sum_{k=1}^K \bar{p}_{lk} = 1$)

How to decide on splits?

\rightarrow Gini index

For a given split: $G = n_{R_1} \sum_{k=1}^K \bar{p}_{kR_1} (1 - \bar{p}_{kR_1}) + n_{R_2} \sum_{k=1}^K \bar{p}_{kR_2} (1 - \bar{p}_{kR_2})$

For the whole tree: $\sum_{l \in \text{leaves}} n_l \sum_{k=1}^K \bar{p}_{kl} (1 - \bar{p}_{kl})$

\rightarrow Entropy

For the whole tree $D = \sum_{l \in \text{leaves}} n_l \sum_{k=1}^K \bar{p}_{kl} \log(\bar{p}_{kl})$

Why is this better than misclassification rate?

Because they tend to produce “purer” leaves

Remark 0.5 (Big Picture: Trees vs Linear Models)

Tree Advantages	LM Advantages
Very Interpretable	More flexible, esp continuous y in predicting different Inference
Easy to visualize/display	More stable - small perturbations of data can lead to different trees
Cases with many complex interactions between covariates	Random forest \rightarrow Categorizing predictions from many trees fit on random subsets of data/covariates