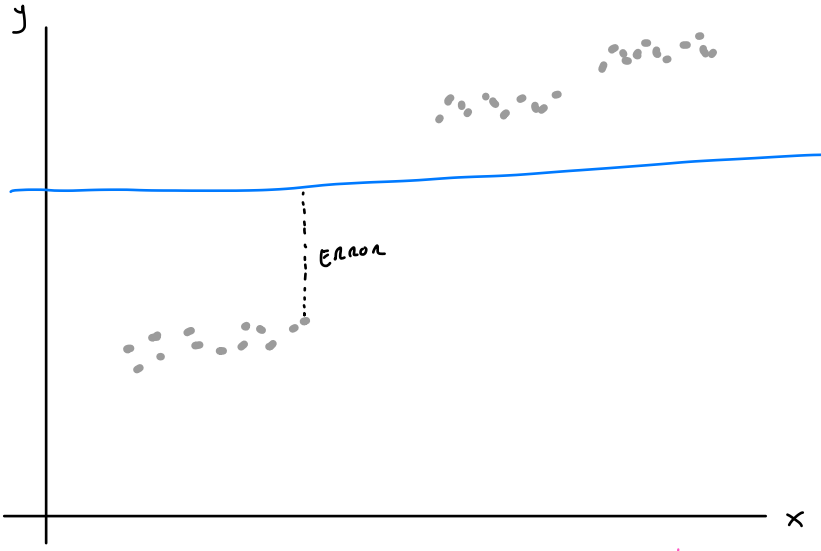
Decision Trees For Regression

CS 307

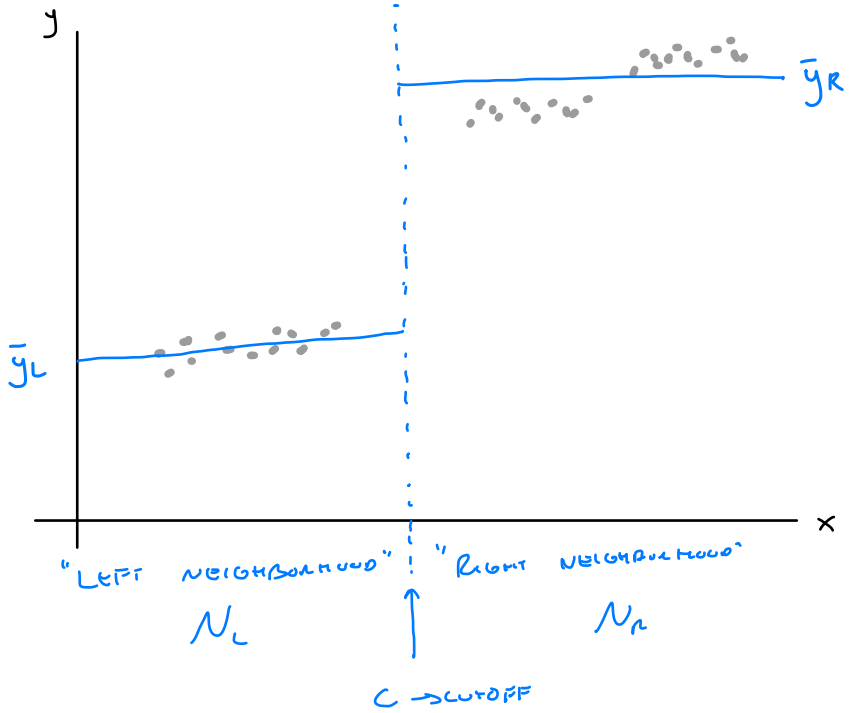


$$\bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$$

SUM SQUARES TOTAL

"OVERALL ERROR" → $SST = \sum_{i=1}^{n} (y_i - \bar{y})^2$

↳ "VARIANCE"



IDEA: CREATE NEIGHBORHOODS, PREDICT MEAN IN NEIGHBORHOOD

Decision Tree

$$SST = \sum_{i=1}^{n} (y_i - \bar{y})^2$$

$$SSE = \sum_{i \in N_L} (y_i - \bar{y}_L)^2 + \sum_{i \in N_R} (y_i - \bar{y}_R)^2$$

SSE: error

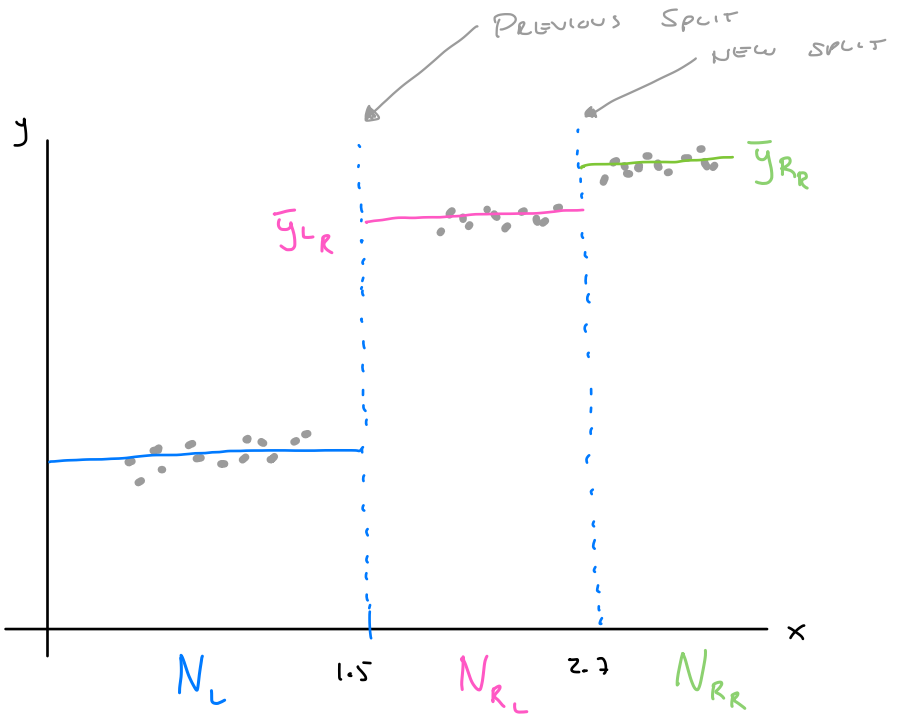
$\bar{y}_L$: mean of y in N_L

Find c that minimizes SSE

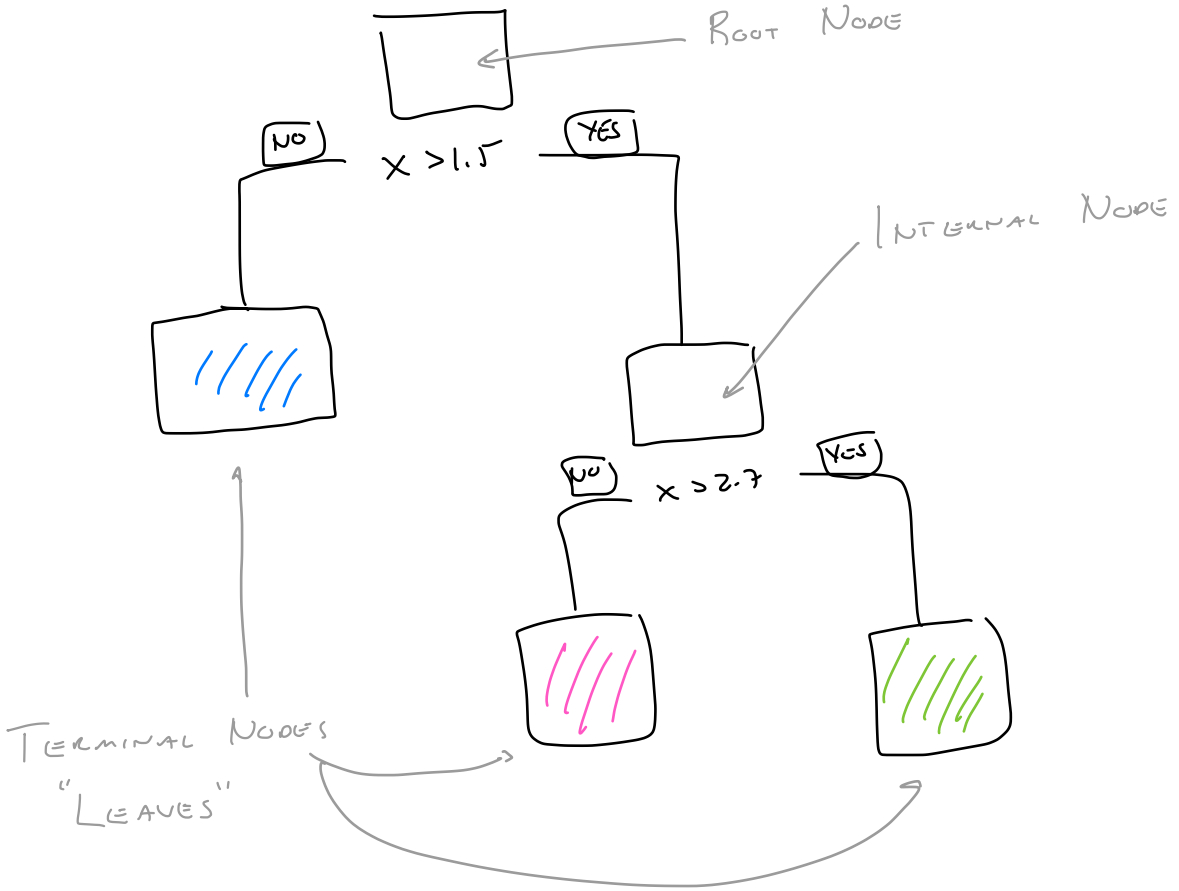
↳ considering all features

i	X	y
1	...	...
2	...	...
3	...	...
...	...	...
n	...	...

implied



$$SSE = \sum_{i \in N_L} (y_i - \bar{y}_L)^2 + \sum_{i \in N_{R_L}} (y_i - \bar{y}_{R_L})^2 + \sum_{i \in N_{R_R}} (y_i - \bar{y}_{R_R})^2$$



NOTE: THIS TREE IS "FLIPPED" COMPARED TO SKLEARN, WHICH GOES LEFT FOR "YES."

$$SST = \sum_{i=1}^{n} (y_i - \bar{y})^2$$

ONE SPLIT

$$SSE = \sum_{i \in N_L} (y_i - \bar{y}_L)^2 + \sum_{i \in N_R} (y_i - \bar{y}_R)^2$$

TWO SPLITS

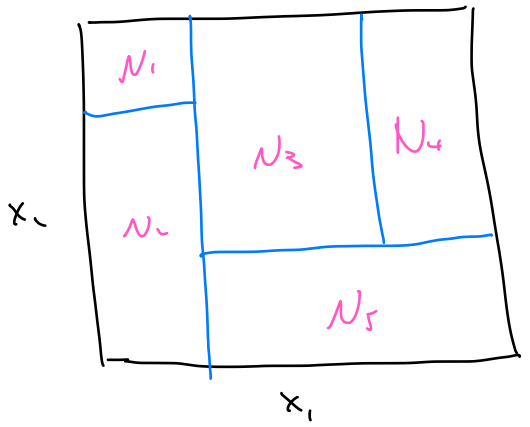
$$SSE = \sum_{i \in N_L} (y_i - \bar{y}_L)^2 + \sum_{i \in N_{RL}} (y_i - \bar{y}_{RL})^2 + \sum_{i \in N_{RR}} (y_i - \bar{y}_{RR})^2$$

NOTE:

$$R^2 = 1 - \frac{SSE}{SST}$$

Recursive Partitioning

Keep finding new splits in current terminal nodes



$$SSE = \sum_{j=1}^{\bar{J}} \sum_{i \in N_j} (y_i - \bar{y}_j)^2$$

mean of points in node j, N_j

WHEN TO STOP?

- DON'T ← BAD IDEA FOR A SINGLE TREE
- MIN SAMPLES SPLIT
- MAX DEPTH

} TUNING PARAMETERS

Sketch of Algorithm

- Init root node, set stop conditions
- For each current terminal node:
  - If stop conditions met:
    - Skip node
  - If no nodes can be split:
    - Stop algorithm, return tree
  - For each feature:
    - For each midpoint c:
      - Calculate and store error
  - Find j and c combination with lowest error
  - Split current node at using $x_j < c$