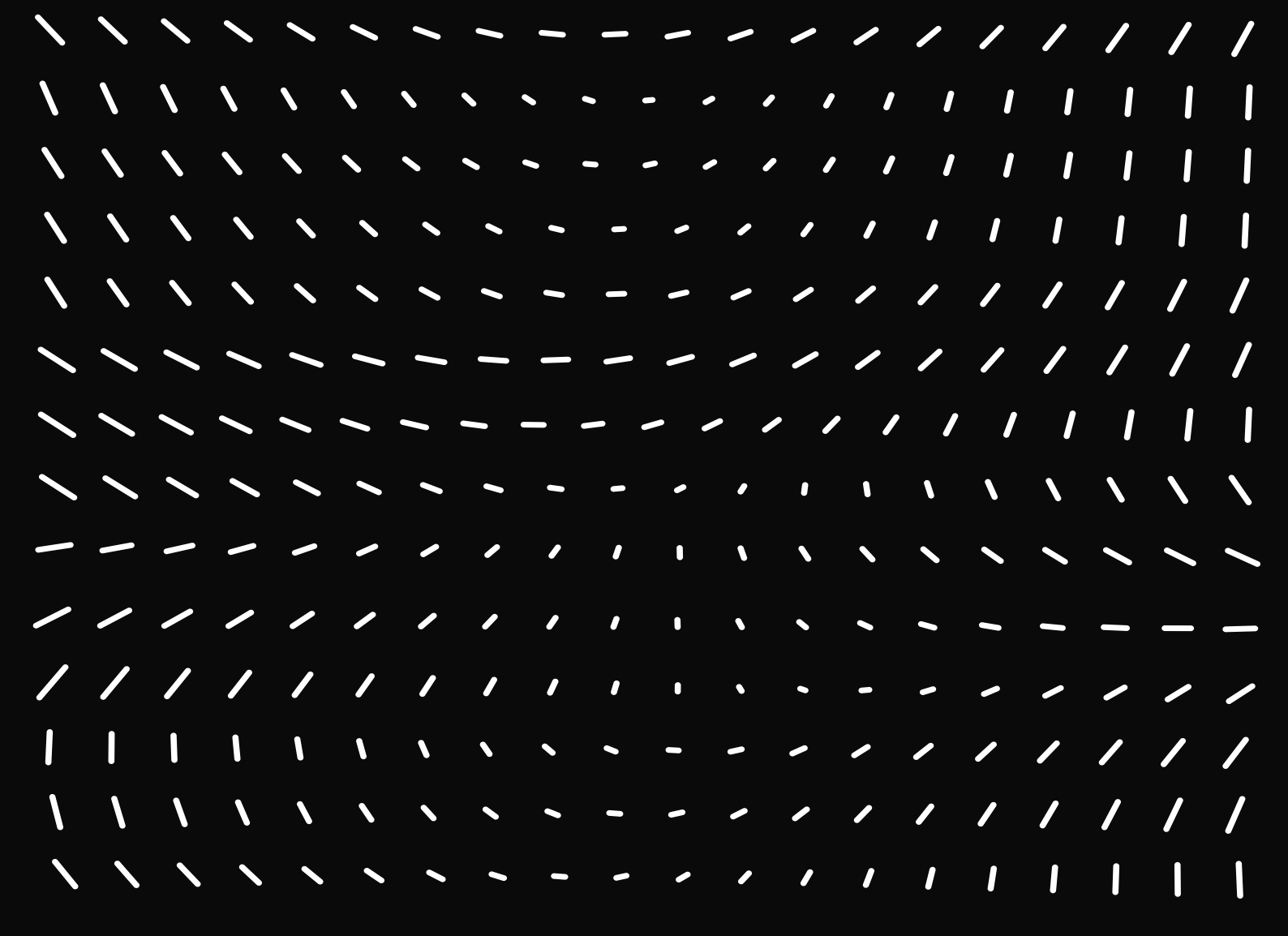
Additive Model & Decision Tree

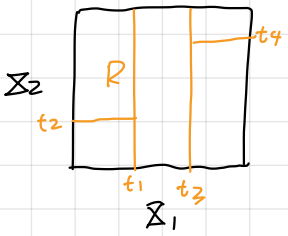


1. Generalized Additive Model

$$E(Y \mid X_1, X_2 \cdots X_p) = \alpha + f_1(X_1) + \cdots + f_p(X_p)$$

2. Decision Tree

$$f(x) = \sum_{m=1}^{M} c_m I\{(x_1, x_2) \in R_m\}$$



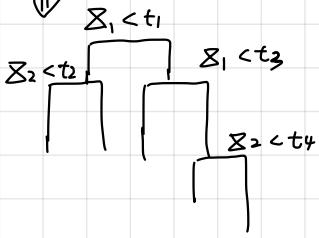
2.1 given data (x_i, y_i)

$$x_i = (x_{i1}, x_{i2}, \cdots, x_{ip})$$

$$f(x) = \sum_{m=1}^{M} c_m I(x \in R_m)$$

$$\Downarrow$$

$$\hat{c}_m = \text{avg}(y_i \mid x_i \in R_m)$$



binary partition

greedy algorithm

proof: given the data (x_i, y_i) and {splitting var. j / splitting point s}

$$R_1(j,s) = \{X \mid X_j \le s\} \qquad R_2(j,s) = \{X \mid X_j > s\}$$

object: $\min_{j,s}\left[\min_{c_1}\sum_{x_i \in R_1(j,s)}(y_i - c_1)^2 + \min_{c_2}\sum_{x_i \in R_2(j,s)}(y_i - c_2)^2\right]$

$$\Downarrow$$

$$c_1 = \text{avg}(y_i \mid x_i \in R_1(j,s)) \qquad c_2 = \text{avg}(y_i \mid x_i \in R_2(j,s))$$

repeat on R_1 and R_2.

2.2 Tree size.

strategy: grow large tree T_0. stop when minimum node size reached.

cost-complexity pruning: collapse any number of its internal (not terminal) nodes.

$|T|$: # of terminal nodes ($R_1 R_2 \cdots R_m$, index by m)

N_m: $\#\{x_i \in R_m\}$

$$\hat{c}_m = \frac{1}{N_m} \cdot \sum_{x_i \in R_m} y_i$$

$$Q_m(T) = \frac{1}{N_m} \cdot \sum_{x_i \in R_m} (y_i - \hat{c}_m)^2$$

$T \subseteq T_0$ means T can be obtained by pruning T_0

complexity criterion: $C_\alpha(T) = \sum_{m=1}^{|T|} N_m Q_m(T) + \alpha|T|$

objective: for each α, find $T_\alpha \subseteq T_0$ that minimize $C_\alpha(T)$

2.2.1

* for each α, there is a unique smallest T_α that minimize $C_\alpha(T)$

weakest link pruning: □ successively collapse internal nodes that produces the smallest per-node increase

in $\sum_{m} N_m Q_m(T)$)

until produce a single node tree

- collect a sequence of subtrees, must contain T_α
- use CV to find the best $\hat{\alpha}$ minimize MSE

$\Downarrow$

$T_{\hat{\alpha}}$

2.3. Tree for classification: (1, 2, ..., K categories)

{ in regression: $Q_m(T) = \frac{1}{N_m} \sum_{x_i \in R_m} (y_i - \hat{c}_m)^2$ ← called node impurity

in classification: $\hat{p}_{mk} = \frac{1}{N_m} \sum_{x_i \in R_m} I(y_i = k)$

$k(m) = \underset{k}{\operatorname{argmax}}\ \hat{p}_{mk}$ (the majority in node m)

$Q_m(T) =$ { misclassification error: (non differentiable)

$$\frac{1}{N_m} \sum_{i \in R_m} I(y_i \neq k(m)) = 1 - \hat{p}_{mk(m)}$$

Gini index:

$$\sum_{k \neq k'} \hat{p}_{mk} \hat{p}_{mk'} = \sum_{k=1}^{K} \hat{p}_{mk}(1 - \hat{p}_{mk})$$

Cross-entropy or deviance:

$$-\sum_{k=1}^{K} \hat{p}_{mk} \log \hat{p}_{mk}$$

- Gini index / cross-entropy used for T_0 tree growing
- misclassification rate use to guide cost-complexity pruning