__/__/____

Decision Trees. $(n \log n) \leftarrow O(N)$

Common metrics:

1. Gini Impurity: $1 - \Sigma p^2$
2. Information Gain: ~~Entropy of Parent~~

   Entropy parent $- \Sigma w$ (Entropy of split)

   where entropy $= \Sigma - p \log_2 p$

Stopping Conditions:

1. Max Depth
2. Min sample per leaf
3. Pruning
4. CV error
5. ~~Ex~~

Psudocode:

```
func BuildTree (Data, Target)
  if stopping_cond: Return leaves
  else: if all in same node: return leaves
    else: choose best attribute and split
      for each split: Build Tree (split)
```

__/__/__

Pruning

Pre Pruning: Early stopping conditions like max depth, min split, samples per leaf etc.

Post Pruning: Pruning after fit.

- Cost Complexity Pruning:

Calculate Tree score for each tree and chose the tree with the lowest tree score.

Tree Score = SSR + α * Tree Splits

Where α is the tree complexity penalty.

For choosing α: Try various α and chose the one with lowest validation error.