Tree Algorithm 101

Agenda

- Linear Regression
- Tree
- Capstone Part 1

Source: Type source image link here

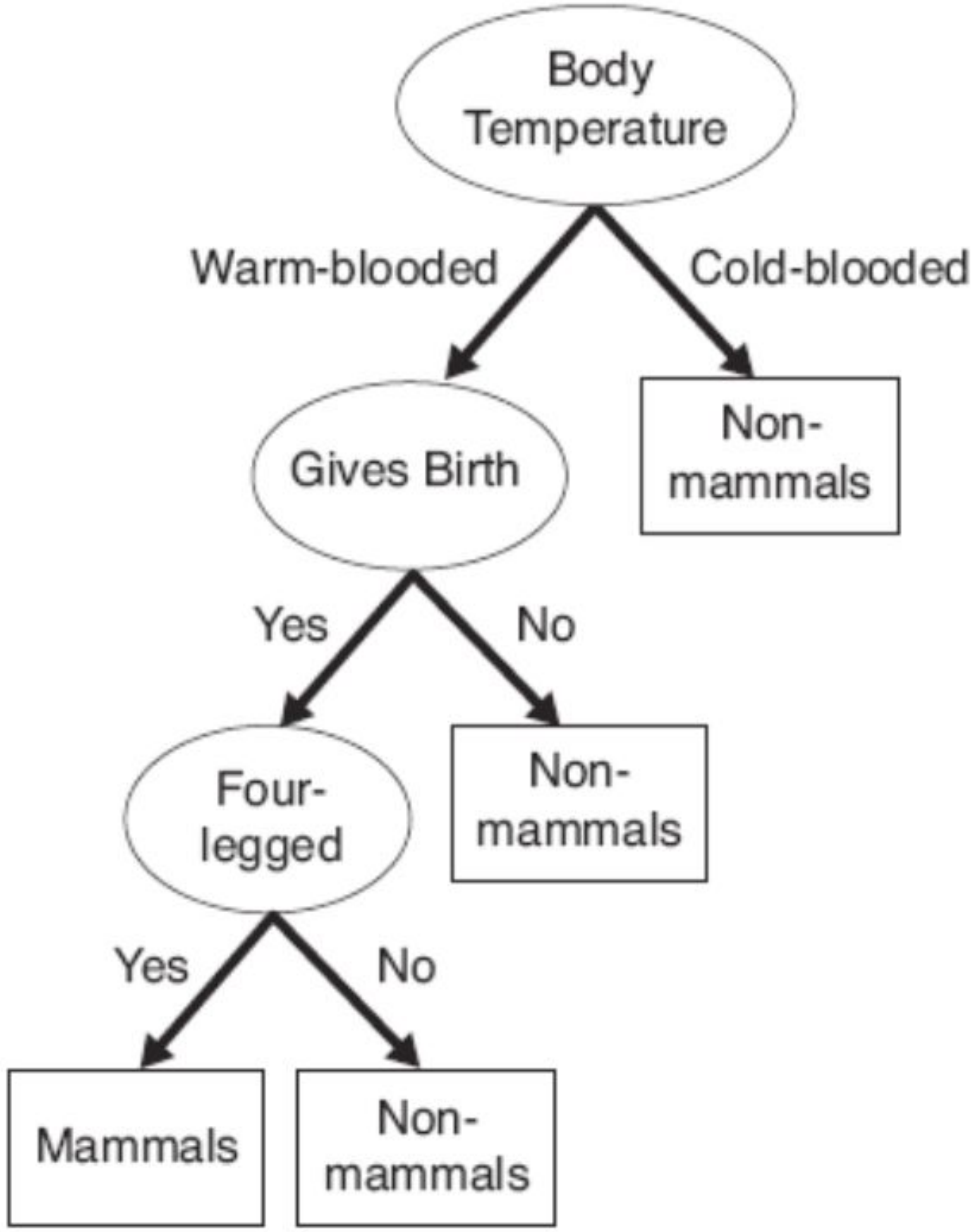
Tree

Agenda

- Decision trees?
- XGBoost

Source: Type source image link here

What They Look Like



Description of Decision Rules or Trees

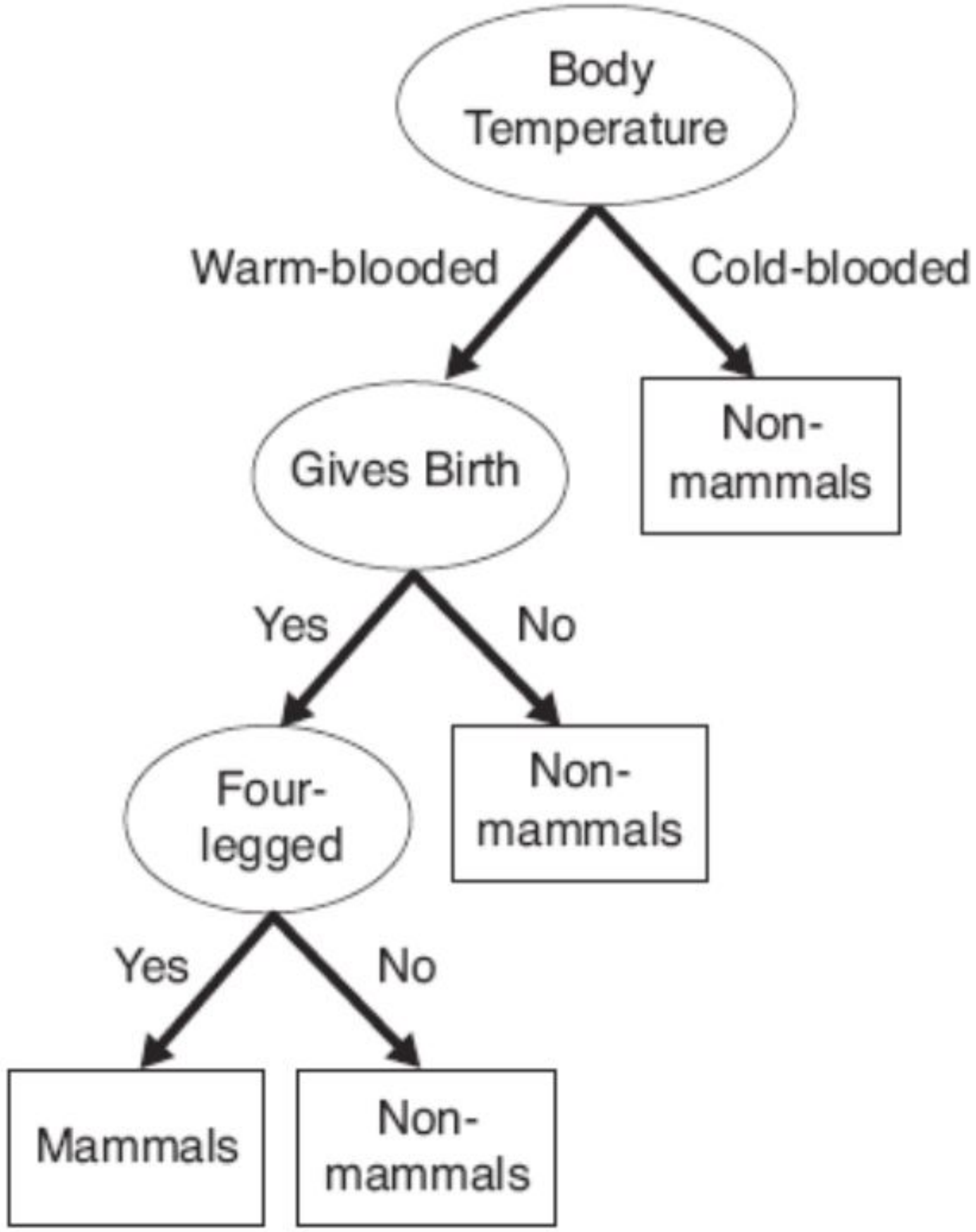
- Intuitive appeal for users
- Presentation Forms
  - "if, then" statements (decision rules)
  - graphically - decision trees

How to approve a loan application

- Bank - loan application
  - end result?
  - What criterion?

How they work?

- All paths
  - start at the root node
  - end at a leaf
  - Each path represents a decision rule
- All paths - mutually exclusive
  - for any one case - only one path will be followed
  - false decisions on the left branch
  - true decisions on the right branch



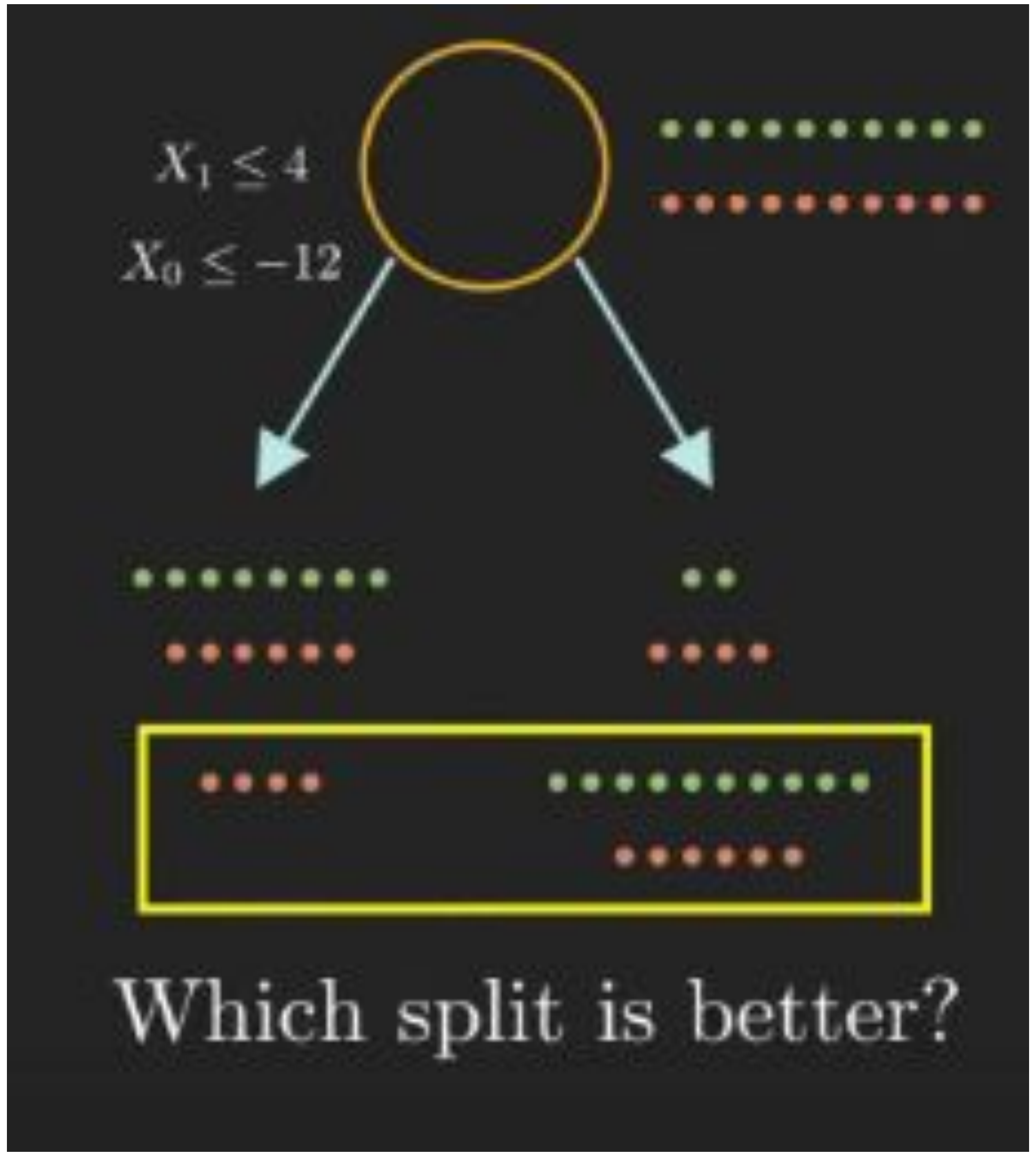
Goal

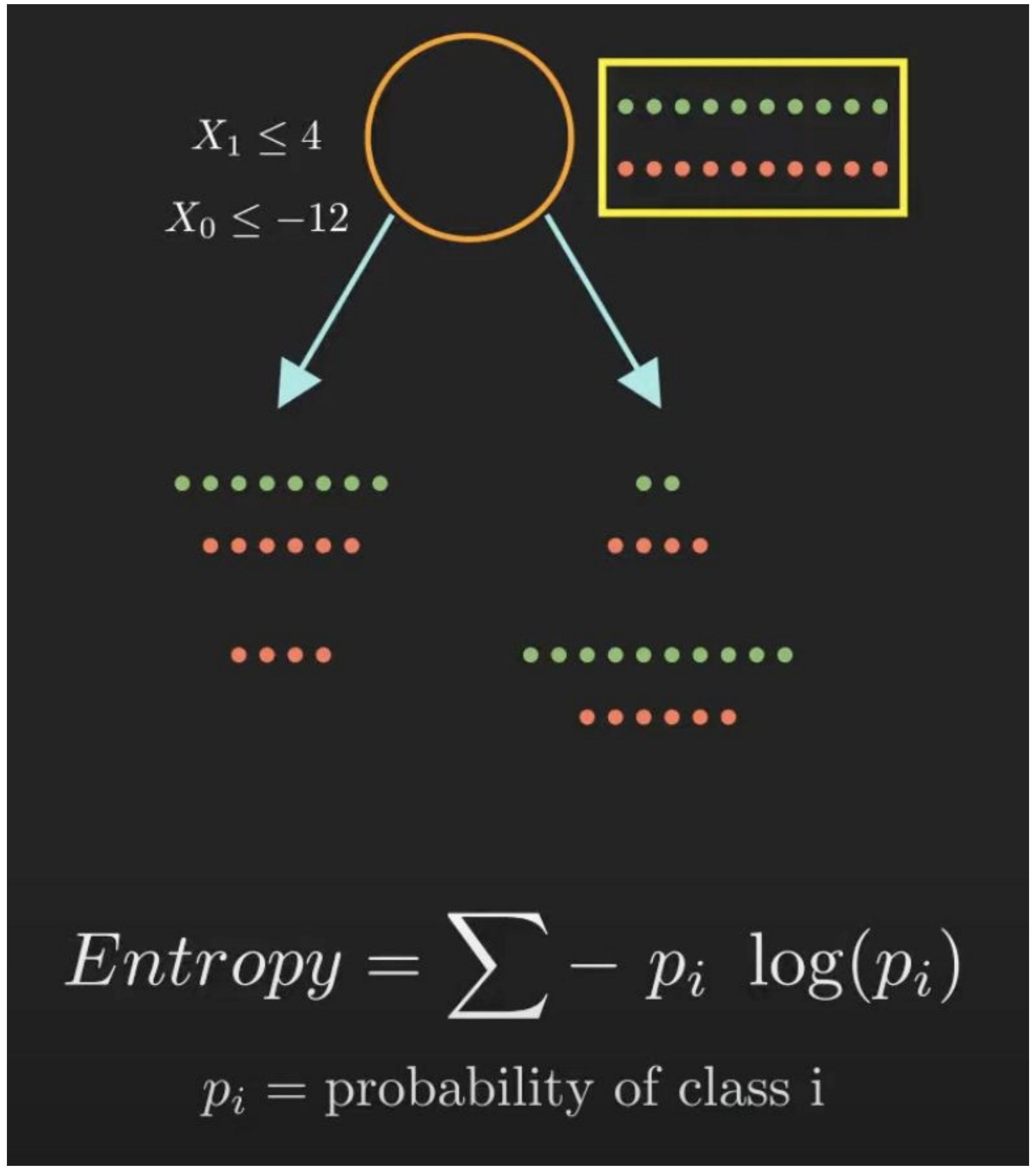
- Dual goal - Develop tree that
  - …

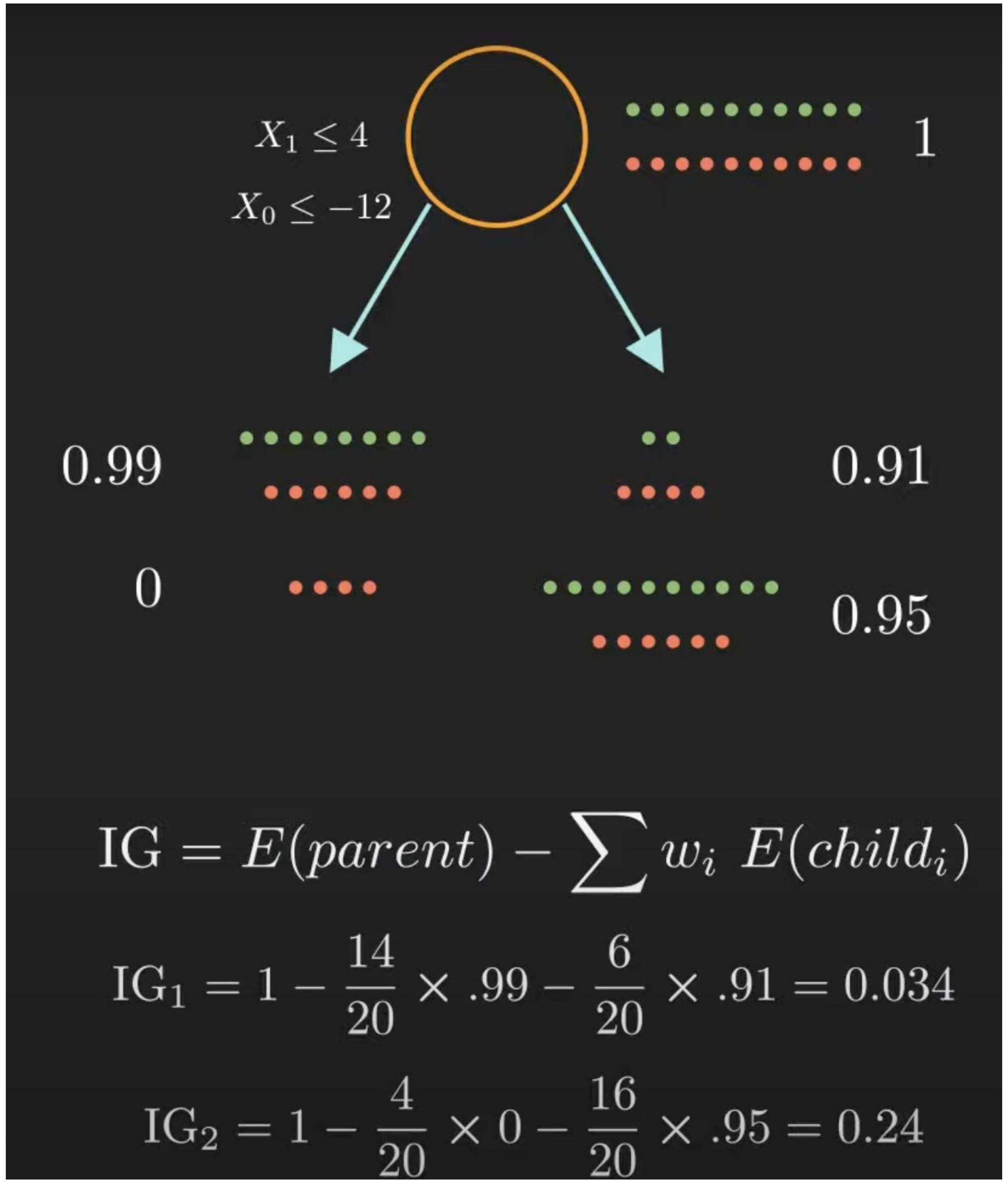
A small size tree?

- Why a small tree make senses?

Entropy and information gain







Discrete vs. Continuous Attributes

- Continuous variables attributes - problems for decision trees
  - increase computational complexity of the task
  - promote prediction inaccuracy
  - lead to overfitting of data

- Convert continuous variables into discrete intervals
  - “greater than or equal to” and “less than”
  - optimal solution for conversion
  - difficult to determine discrete intervals ideal
    - size
    - number

Making the Split

- Models induce a tree by recursively selecting and subdividing attributes
  - random selection - noisy variables
  - inefficient production of inaccurate trees

- Efficient models
  - examine each variable
  - determine which will improve accuracy of entire tree
  - problem - this approach decides best split without considering subsequent splits

Overfitting

- Error rate in predicting the correct class for new cases
  - overfitting of test data
  - very low apparent error rate
  - high actual error rate

Optimal Size

- Certain minimal size smaller tree
  - higher apparent error rate
  - lower actual error rate

- Goal
  - identify threshold
  - minimize actual error rate
  - achieve greatest predictive accuracy

Ending Tree Growth

- Grow the tree until
  - additional splitting produces no significant information gain
  - statistical test - a chi-squared test
  - problem - trees that are too small
  - only compares one split with the next descending split

Pruning

- Grow large tree
  - reduce its size by eliminating or pruning weak branches step by step
  - continue until minimum true error rate

- Pruning Methods
  - *reduced-error* pruning
  - divides samples into validation set and training set
  - training set is used to produce the fully expanded tree
  - tree is then tested using the validation set
  - weak branches are pruned
  - stop when no more improvement

Evaluating Decision Trees

- Method’s Appropriateness
- Data set or type
- Criteria
  - accuracy - predict class label for new data
  - scalability
    - performs model generation and prediction functions
    - large data sets
    - satisfactory speed
  - robustness
    - perform well despite noisy or missing data
  - intuitive appeal
    - results easily understood
    - promotes decision making

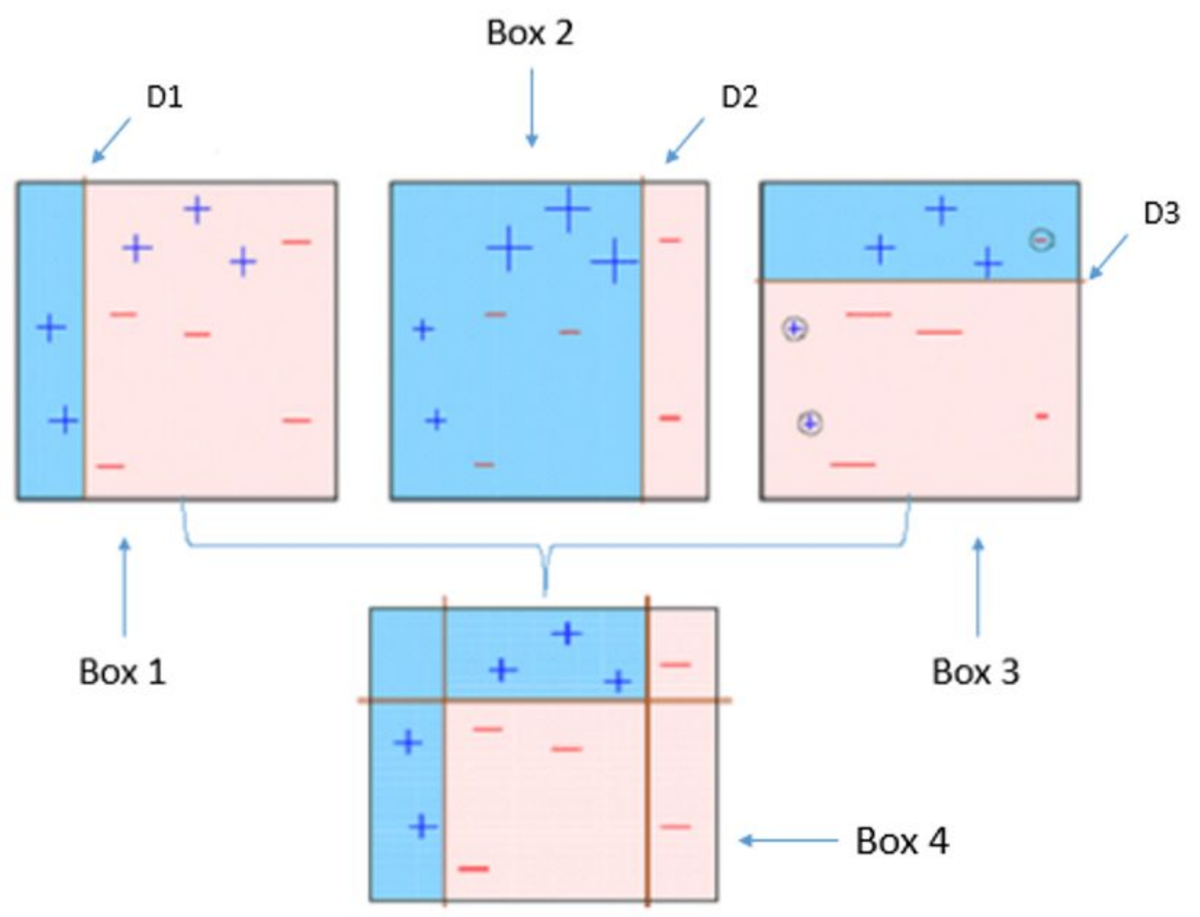
Decision Tree Limitations

- No backtracking
  - local optimal solution not global optimal solution
  - *lookahead* features may give us better trees

- Rectangular-shaped geometric regions
  - in two-dimensional space
    - regions bounded by lines parallel to the x- and y- axes
  - some linear relationships not parallel to the axes

What is XGBoost

- Stands for:
  - –eXtreme Gradient Boosting.
- XGBoost is a powerful iterative learning algorithm based on gradient boosting.
- Robust and highly versatile, with custom objective loss function compatibility.



How does XGBoost work?

- Tree-Based Boosting algorithm.

- 4 Critical Parameters for Tuning:

  –η: ETA or “Learning Rate”

  –max_depth: Controls the “height” of the tree via splits.

  –γ: Minimum required loss for the model to justify a split.

  –λ: L2 (Ridge) regularization on variable weights.

Why use Xgboost?

- All of the advantages of gradient boosting, plus more.
- Frequent Kaggle data competition champion.
- Utilizes CPU Parallel Processing by default.
- Two main reasons for use:
  1. Low Runtime
  2. High Model Performance

Tuning XGBoost

- In order to produce the optimal XGBoost model, a grid-search method was employed against a hyper-grid of possible parameters.