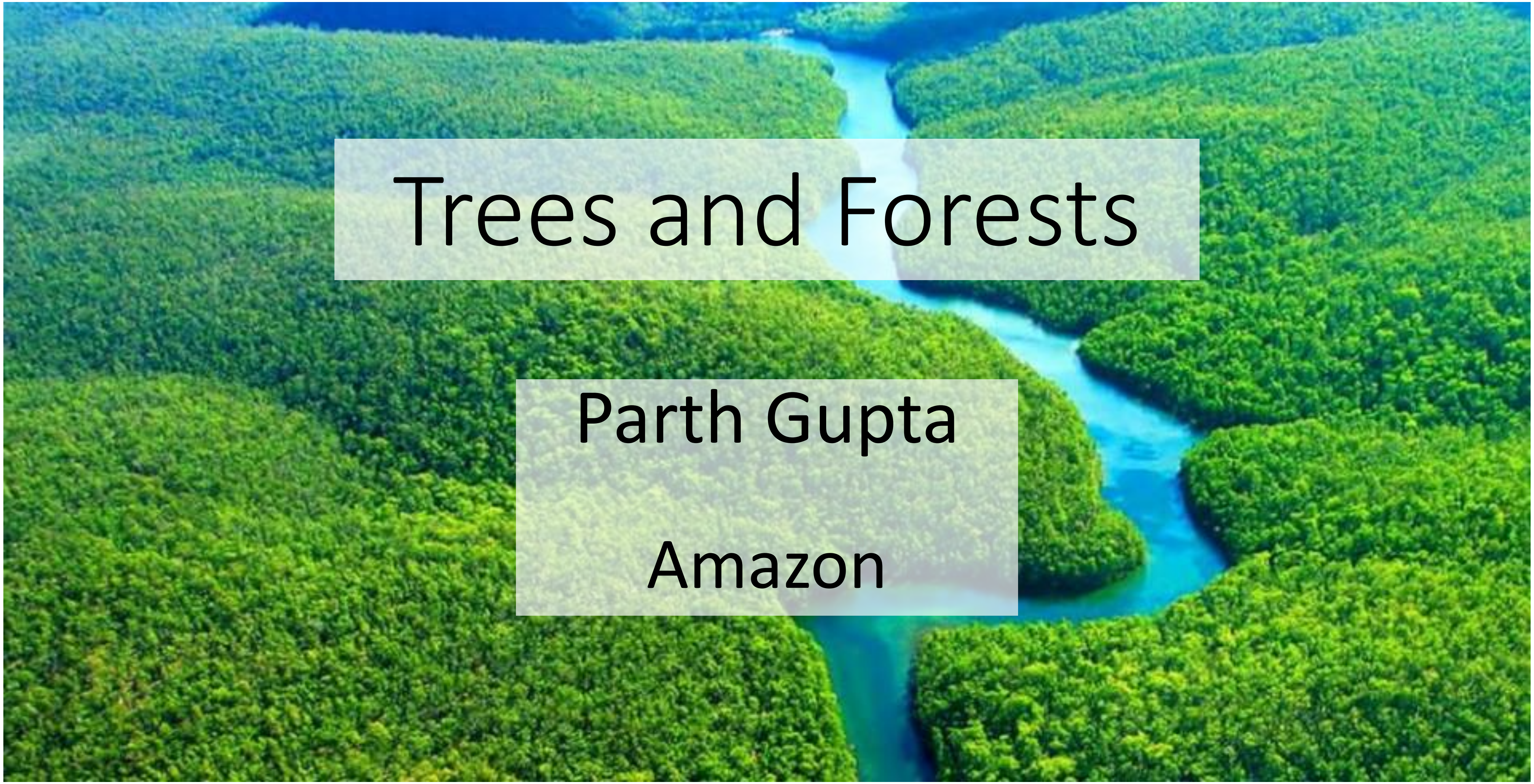

Trees and Forests
Parth Gupta
Amazon

(Decision) Trees and (Random) Forests

Parth Gupta

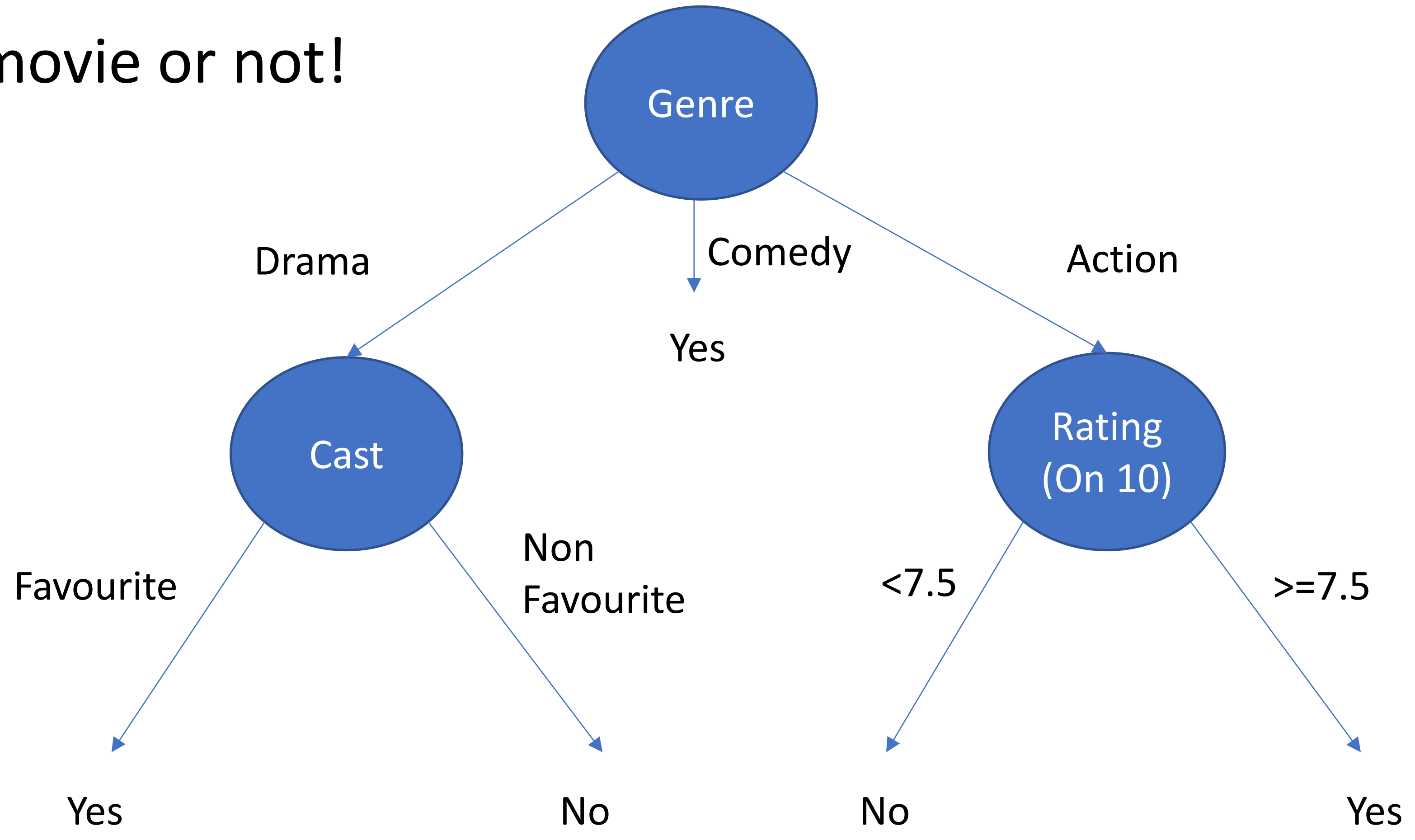
Amazon.com

What you know already!

- Supervised learning
- Target function $f: X \rightarrow y$
- Data $(\overrightarrow{x_1}, y_1), (\overrightarrow{x_2}, y_2) \ldots (\overrightarrow{x_N}, y_N)$
- Hypothesis space $\mathcal{H}$
- Hypothesis function $h: X \rightarrow y$
- Final hypothesis $g \approx f$
- In case of linear regression: $h(x) = w_0 + \sum w_i * x_i$

Decision Tree

- To watch a movie or not!



Decision Tree

- **Inference** is simply walking the tree
  - (Genre = action) $\rightarrow$ (Rating = 8.5) $\rightarrow$ Yes
- Also, in case of Boolean outcome, it can be represented as a logic predicate
  - (Genre = Drama $\wedge$ Cast = Favourite)

    $\vee$ (Genre = Comedy)

    $\vee$ (Genre = Action $\wedge$ Rating $\geq$7.5)

One of the most widely used and practical methods

Appropriate Problems for Decision Tree

- Instances are attribute-value pairs with fixed set of attributes and even better small number of values e.g. temperature ∈ {hot, mild, cold}
- Target has discrete output (Boolean or small number of outputs)
- Training data may contain errors/noise
- Training data may contain missing values

Basic Learning Algorithm (ID3)

Utility: Which attribute is able to classify the training examples alone with highest accuracy?

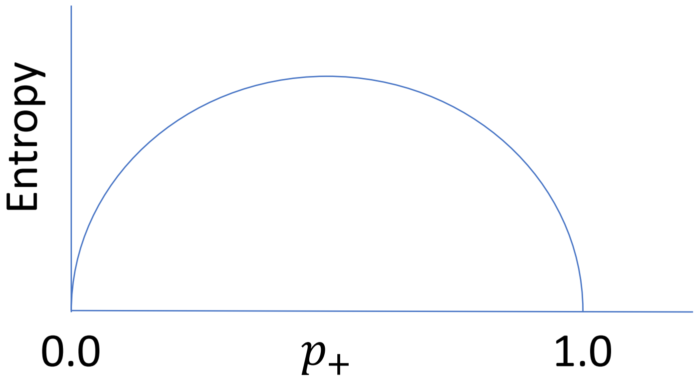
1. Select the root node intance using the utility
2. Branch out the tree with root node values and re-create the training data based on the root-node value
3. At each child-node select the instance using the same utility
4. Rearrange the dataset for each descendent node and repeat the process select the best node
5. Continue until the dataset is completely classified

Selecting the best attribute

- **Entropy** $E = -p_+ * \log p_+ \; - \; p_- * \log p_-$

$$p_+ = \frac{\#\, Positive\ samples}{\#\, Total\ samples} \;;\quad \mathrm{p}_- = \frac{\#\, Negative\ samples}{\#\, Total\ samples}$$

- Nature of p_+ $(or\ \ p_-)$



Entropy is highest when p_+ = 0.5 (i.e. equal number of +ve and -ve samples

Selecting the best attribute

- **Information Gain**

$$IG(S,A) = E(S) - \sum_{V \in values(A)} \frac{|S_V|}{|S|} * E(S_V)$$

- Measures effectiveness of an attribute *A* in classifying the training data S by measuring the expected reduction in Entropy.
- Select the attribute with maximum information gain (IG)

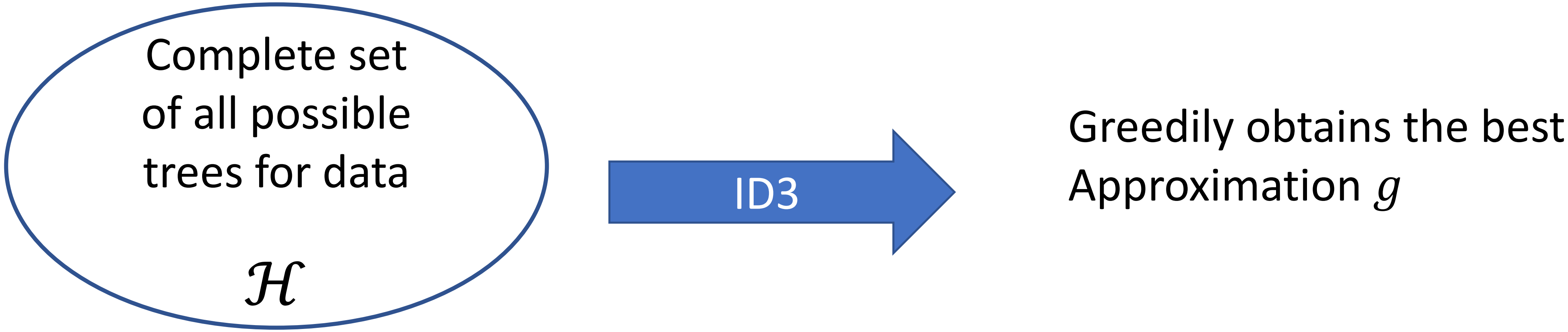
Note:
Entropy is a function of data (S) only
Information Gain is a function of data and the attribute (S, A)

Selecting the leaf node

- At some point $|S_V|$ will become all +ve or –ve, declare that node to be a lead node. → Prone to overfit.
- In practice, stop when there are not enough samples further and choose the maximum label from the subtree.

Hypothesis Search Space: Two Components

- Current State
  - Assumption that current state contains all information needed for a solution
- Evaluation Function
  - Based on which, the next state is chosen



Huge: Brute force is impossible!

ID3 Advantages

- **Complete Search Space:** Which contains the target function, dissimilar to some methods that search incomplete hypothesis spaces.
- **Uses ALL training samples to go to next state:** Less sensitive to noise in the data. Btw, leaf nodes should be handled *smartly.*

ID3 Disadvantages

- **Single Current Hypothesis:** It does not know about other consistent hypotheses.
- **No Backtracking:** In its pure form, the search decisions can not be reverted or backtracked. Though, there are some extensions which allow it e.g. C4.5

Decision Tree is prone to Overfitting!

- Due to
  1. **Stochastic Noise**: Some noisy training examples which ID3 tries to fit to.
  2. **Deterministic Noise:** Not enough samples to make *stable* inferences on.
- Two approaches
  1. Early stopping: Easy but in practice very difficult to decide when to stop.
  2. Post-pruning: Usually more successful (discussed next)

- In both approaches, the key question is about a correct tree size.

Reduced Error Pruning

- Set aside some validation data on which the tree can be evaluated
- For each node
  - remove the subtree below it and make it a lead node with highest classification e.g. $p_+ > p_-$ then +ve else –ve
  - Measure the accuracy on validation set
- Remove the node for which the validation accuracy was maximum after removal and greater than before removal.
- Repeat until validation accuracy starts to decrease.

Handling Continuous Values

- Convert them into discrete values
- Select values where the target changes and select the one which has highest information gain
- Can have multiple levels

Handling Missing Values

- Simply ignore them
- And, do not use them into IG calculations

Ensemble methods

- **Voting from multiple classifiers (Bagging)**
  1. Train multiple classifiers on the same dataset
     - Decision tree, logistic regression, SVM, Neural Networks etc.
     - Take decision from each of them and take the maximum vote.
  2. Train one classifier on multiple datasets
     - Create M sub-datasets from the given data (sampling with replacement)
     - Train the same learning algorithm e.g. decision tree on each of them
     - Take the maximum vote from such multiple classifiers each trained on a different sample

Rationale behind Ensembles

- Bagging tends to reduce the bias hence a good alternative to regularization
- Even though each Tree overfits, they overfit to different things. Through voting, you can get away from the overfitting phenomenon.

Random Forests

- Motivation
  - Computationally, the most expansive task in ensembles is to train the decision tree.
  - Especially, deep structures with large datasets sometimes make it prohibitively expensive
- Efficient and surprisingly effective solution:
  - Fix the size of the tree; and
  - Use random nodes
- Such collection of trees are called random forests.

Random Forest: Parameters

- Data (D), depth (d) and number of trees (k)
- The features/nodes selected at the branches of the tree are selected randomly (usually with replacement)
- The prediction is made at the leaf nodes of the tree with maximum likelihood i.e. $p_+ > p_-$ then +ve otherwise –ve.

Thanks!