Machine Learning
CSCE 5215

Decision Tree Algorithm

Instructor: Zeenat Tariq

Decision Trees

Branching methodology for decision making

Step 1: Create series of decision rules to partition dataset appropriately

Step 2: Use the rule that minimizes mixing of classes (clear rules) and split data at that level

Step 3: Continue splitting data using the next best rule

Step 4: Make prediction on classification problem

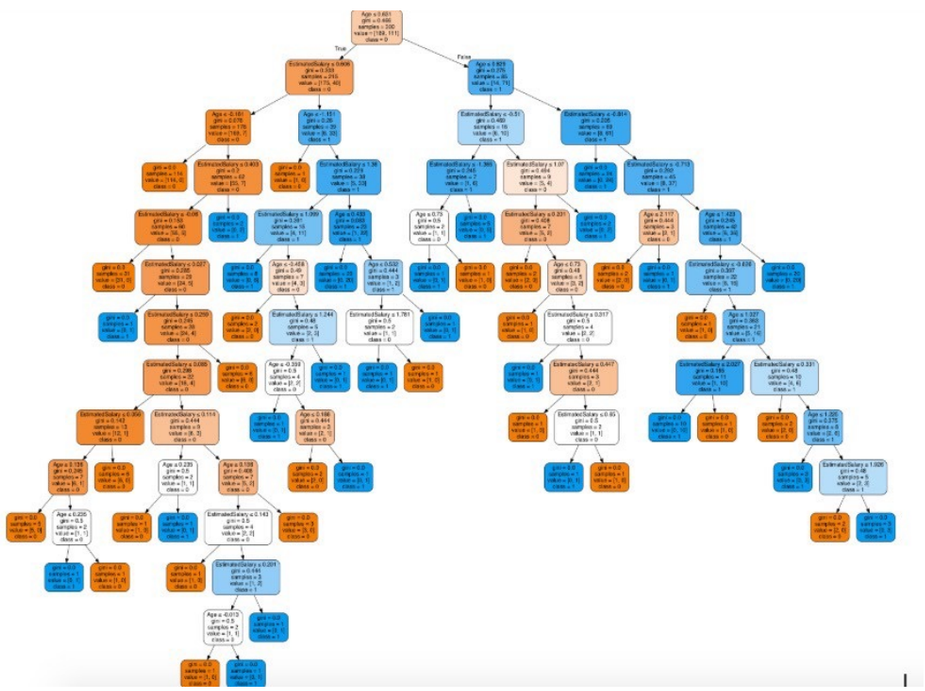
Decision tree : classification

Tree, subtree, roots and branches

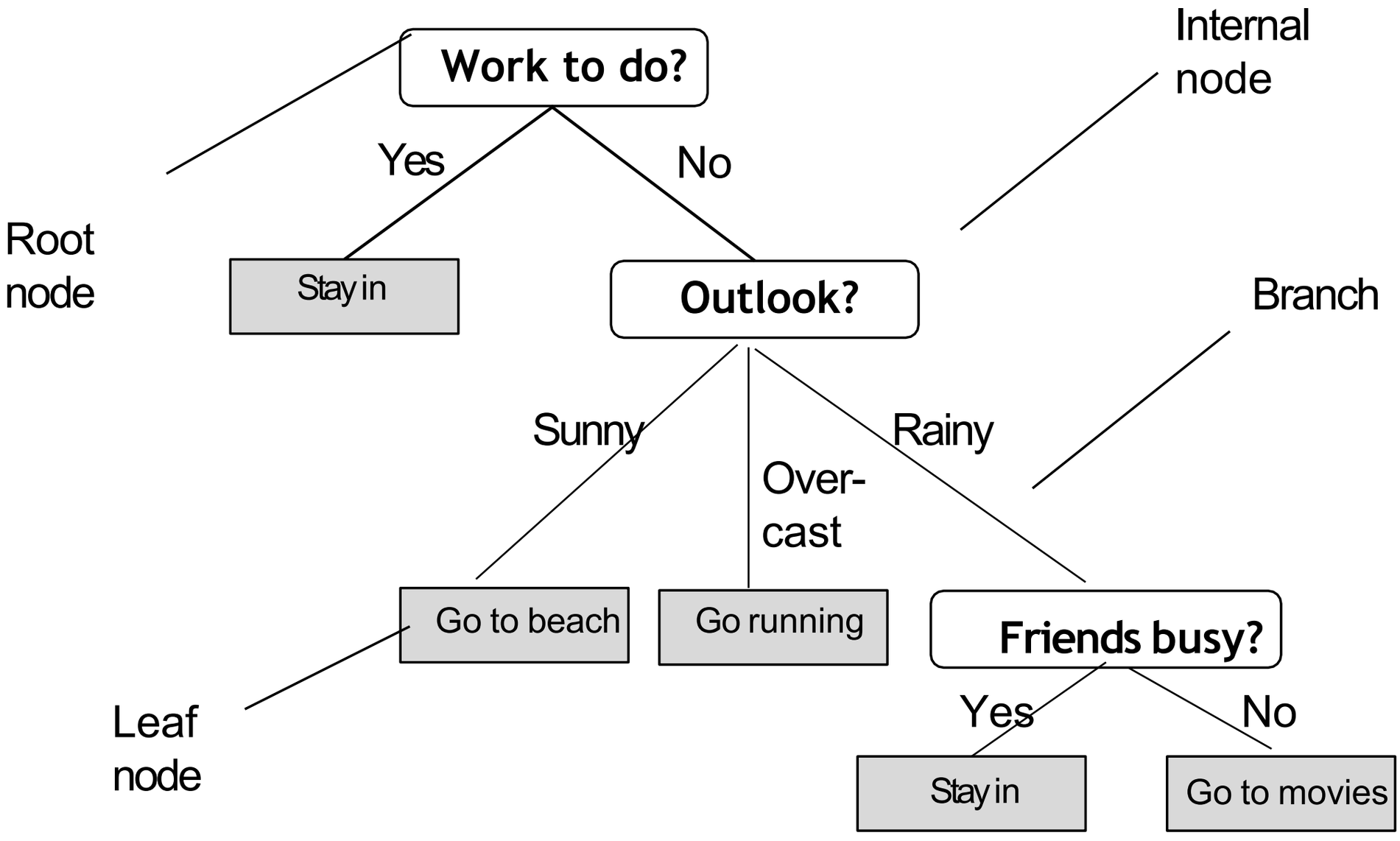
Root node, Decision node

Strategic splits of data→ Accuracy

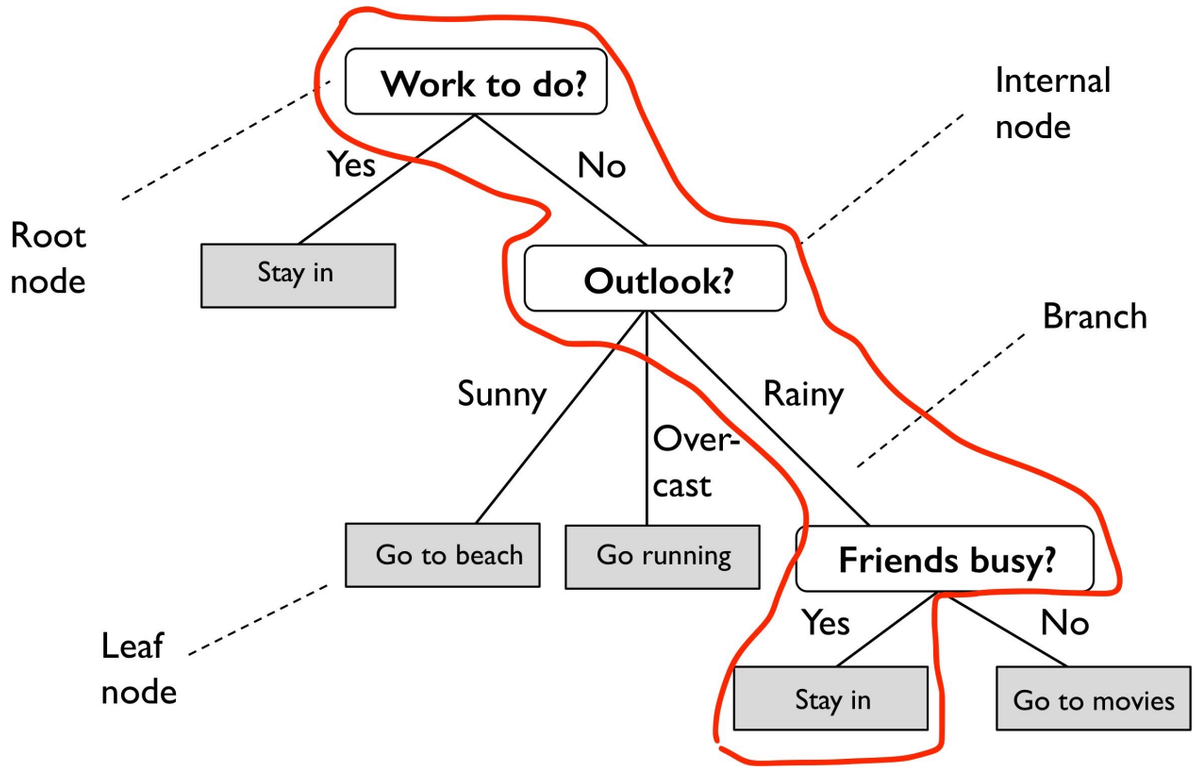
Categories (leaf node)



Decision Tree Terminology



Decision Trees as Rulesets



IF Work to do = No **and** Outlook = Rainy

and Friends busy = Yes

THEN Stay Inside

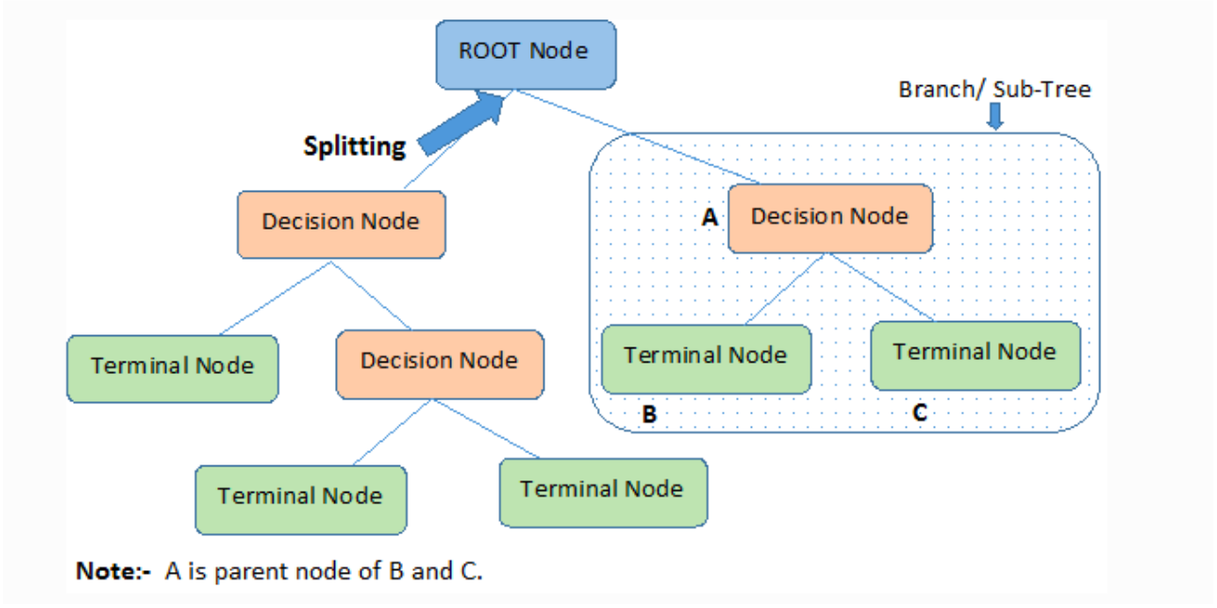
Rationale

Why use the Decision Tree?

1. Makes decisions under uncertainty

2. Aids in analysis of different decision outcomes

3. Allows making optimal decisions



When to use Decision Tree ?

1. If the training dataset contains errors or missing values

2. For problems where instances are represented by attribute value pairs

3. When the target function has discrete output values

Decision Tree in Pseudocode

GenerateTree($\mathcal{D}$):

- if $y = 1 \; \forall \; \langle \mathbf{x}, \mathbf{y} \rangle \in \mathcal{D}$ or $y = 0 \; \forall \; \langle \mathbf{x}, y \rangle \in \mathcal{D}$:
    - return Tree
- else:
    - Pick best feature x_j:
        - $\mathcal{D}_0$ at $\text{Child}_0 : x_j = 0 \; \forall \; \langle \mathbf{x}, y \rangle \in \mathcal{D}$
        - $\mathcal{D}_1$ at $\text{Child}_1 : x_j = 1 \; \forall \; \langle \mathbf{x}, y \rangle \in \mathcal{D}$

    return Node(x_j, GenerateTree($\mathcal{D}_0$), GenerateTree($\mathcal{D}_1$))

Generic Tree Growing Algorithm

1) Pick the feature that, when parent node is split, results in the largest information gain

2) Stop if child nodes are pure or information gain <= 0

3) Go back to step 1 for each of the two child nodes

Generic Tree Growing Algorithm

1) Pick the feature that, when parent node is split, results in the largest information gain

2) Stop if child nodes are pure or information gain <= 0

3) Go back to step 1 for each of the two child nodes

- How make predictions of features in dataset not sufficient to make child nodes pure?

Design choices

- How to split
  - what measurement/criterion as measure of goodness
  - binary vs multi-category split
- When to stop
  - if leaf nodes contain only examples of the same class
  - feature values are all the same for all examples
  - statistical significance test

Types of decision trees

ID3 – Iterative Dichotomizer 3

One of the earlier/earliest decision tree algorithms

Quinlan, J. R. 1986. Induction of Decision Trees. Mach. Learn. 1, 1 (Mar. 1986), 81-106.

cannot handle numeric features no pruning, prone to overfitting

short and wide trees (compared to CART)

maximizing information gain/minimizing entropy

discrete features, binary and multi-category features

C4.5

continuous and discrete features

Ross Quinlan 1993, Quinlan, J. R. (1993). C4. 5: Programming for machine learning. *Morgan Kauffmann*, *38*, 48.

continuous is very expensive, because must consider all possible ranges

handles missing attributes (ignores them in gain compute)

post-pruning (bottom-up pruning) Gain

Ratio

CART

Breiman, L. (1984). *Classification and regression trees*. Belmont, Calif: Wadsworth International Group.

continuous and discrete features

strictly binary splits (taller trees than ID3, C4.5)

binary splits can generate better trees than C4.5, but tend to be larger and harder to interpret; k-attributes has a ways to create a binary partitioning

variance reduction in regression trees

Gini impurity, twoing criteria in classification trees

cost complexity pruning

Other

- CHAID (CHi-squared Automatic Interaction Detector); Kass, G. V. (1980). "An exploratory technique for investigating large quantities of categorical data". *Applied Statistics*. 29 (2): 119–127.
- MARS (Multivariate adaptive regression splines); Friedman, J. H. (1991). "Multivariate Adaptive Regression Splines". *The Annals of Statistics*. 19: 1
- C5.0 (patented)
- ...

The Splitting Criterion

Goal of learning Models

Generalize well

Model selection criteria :

Inductive bias

Ability to learn a general rule from a finite set of examples

Restriction bias

Constraining the set of models a ML algorithm will consider

Preference bias

Prefer certain models over others

Selection Criteria

Metric for information content of a set of training data D	Measure of impurity
Entropy (or Log Loss)	$H(p) = -\sum_{i=1}^{n} p_i \log_2 p_i$
Gini index	$H(p) = \sum_{i=1}^{n} p_i(1-p_i)$
Information Gain	$G(D,A) = I(D) - \sum_{i=1}^{n} \frac{\lvert D_i \rvert}{\lvert D \rvert} I(D_i)$
Misclassification Error	1-error

Decision Trees: Avoiding Overfitting

- Models that overfit do not generalize well

- Overfitting risk
  - simply ask enough questions to filter to every individual sample as a leaf node

- Appropriate hyperparameters to control overfitting:
  - Minimum samples at which to split
    - lower means more overfitting, higher means not learning the data
    - similar for minimum samples per leaf node
  - More crudely, just fix the maximum depth of the tree
  - Consider selecting subsets of features
    - forms the basis of tree selection in random forest!

Pros and Cons of Decision trees

Easily understood, interpreted and readable

Implicitly perform feature selection

Simple to visualize

High computation time with high number of attributes (poor performance)

Considers only one attribute at a time

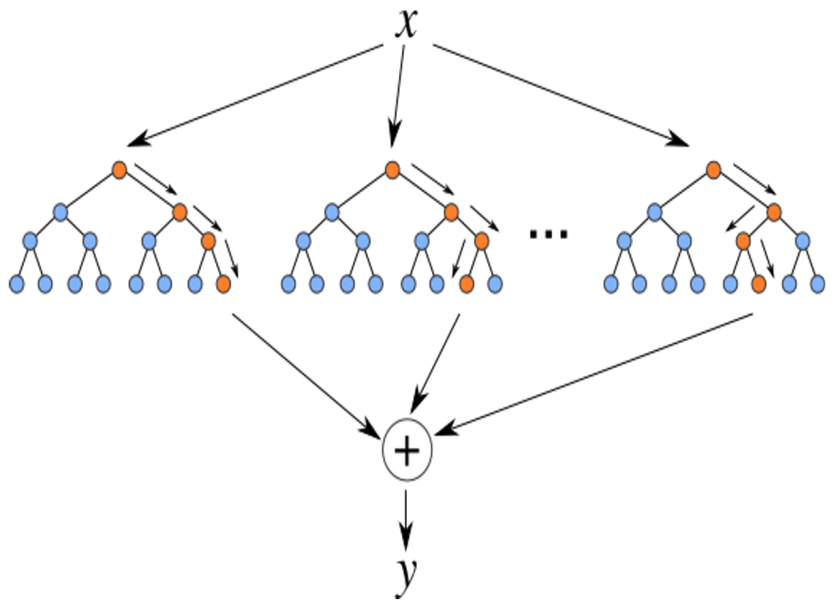
Random Forest

Decision tree learning is fast, but not robust...

Decision trees tend to overfit

Decision trees are created with random subsets of features

Combining these concepts produces a Random Forest classifier



Random Forest, pros and cons

- Robust behavior in many real-world situations
- Relatively fast for an ensemble method (since decision tree learning is fast)
- Works well when there is no strong relation among features
- More hyperparameters to potentially fit than decision trees alone
- Not easily interpretable