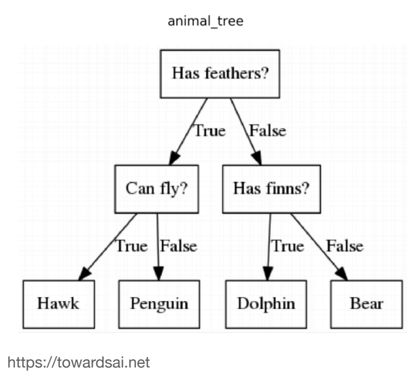
Decision tree models

Yulia Newton, Ph.D.
CS156, Introduction to Artificial Intelligence
San Jose State University
Spring 2021

What are decision trees in ML?

- Can be used for both classification and regression supervised problems
  - Most often used for classification
- Sometimes referred to as CART (Classification and Regression Trees)
- Provide a way to do supervised learning in non-parametric way
  - Non-parametric

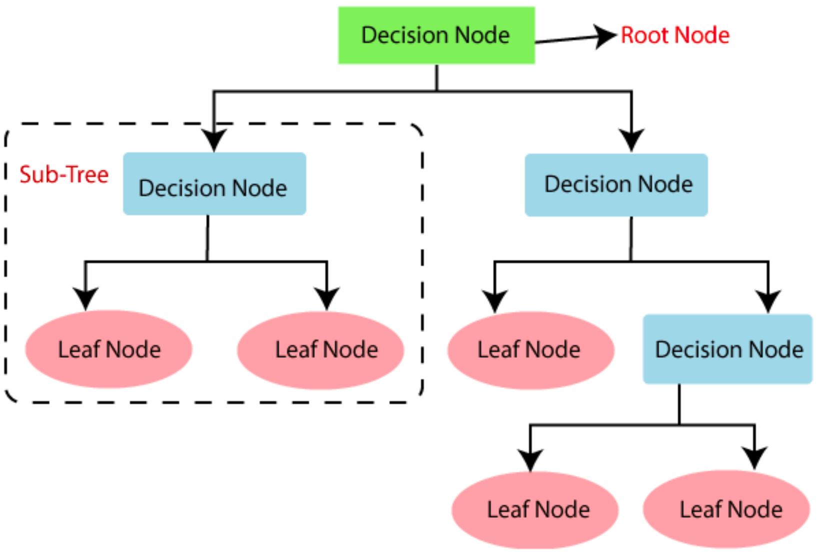


Based on a few simple attributes we can predict the type of animal

Terminology

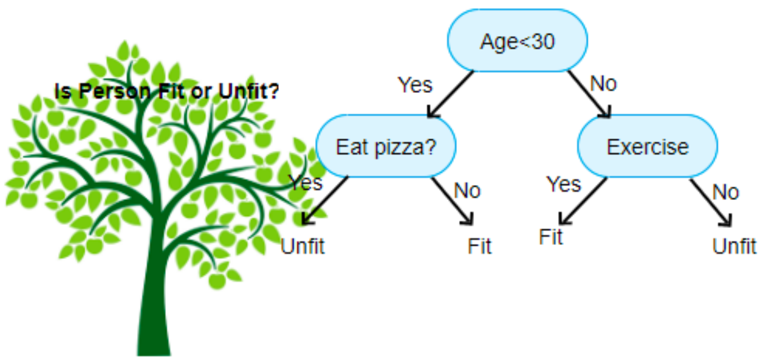
- **Decision tree** - hierarchical tree which can be traversed to make a decision
  - Decisions are made top to bottom, by splitting nodes into sub-nodes based on some criteria
  - Most decision trees are binary (bifurcating splits) but not a requirement
- **Root** - the initial node in the tree
  - Initial decision on which to split
- **Sub-tree** - a part of the decision tree with a non-root node at the root
  - Also called branch
- **Splitting** - process of splitting a tree based on a given criteria (e.g. value of an independent variable)
- **Decision node** - a node at which splitting occurs
- **Terminal node** - final node at which no splitting occurs
  - Also called a leaf
- **Tree pruning** - process of removing a sub-tree from a decision tree

Basic structure of a decision tree



https://addepto.com/decision-tree-machine-learning-model

Toy example

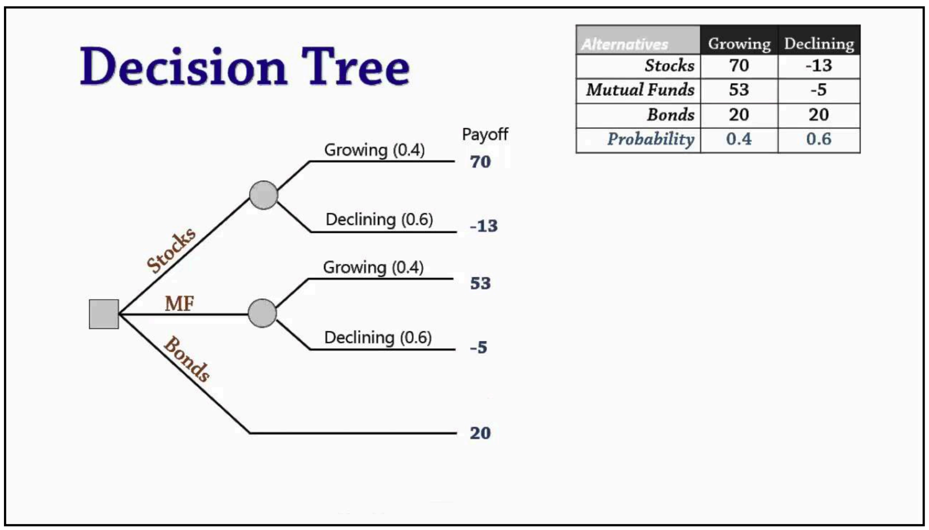


https://towardsai.net

Predict person’s fitness level based on their age, diet preferences, and whether they exercise

Toy example 2

- In practice decision trees are most often binary but they don’t have to be



Joshua Emmanuel

Major types of decision trees

- Categorical decision trees
  - The output/dependent variable is categorical
    - Classification tasks

- Continuous variable decision trees
  - The output/dependent variable is continuous
    - Regression tasks

A few notes about decision trees

- Non-parameteric models
- Have been shown to work well in making decisions in complex scenarios/ problems
  - Can produce simpler models than other methods (feature selection)
    - We prefer smaller trees - simpler explanations of dependent variable
      - Low depth, small number of nodes
  - Capture non-linear relationships well
- Both the model and its predictions are easily interpretable
- Work by computing relationships between each independent variable and the dependent variable

So how do we come up with a decision tree for a given problem?

- We pick an independent variable
  - If this variable is categorical, then we split the tree on each category
  - If this variable is continuous, then we can discretize it
    - E.g. age >= / < 18 yo
- We repeat until we only have leaf nodes
  - The faster we reach that point the better our decision tree is (shallow trees provide more robust models)

Example Data

Training Examples:

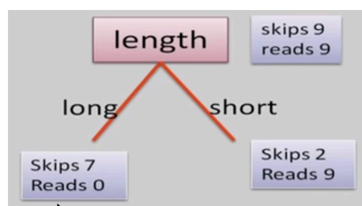
	Action	Author	Thread	Length	Where
e1	skips	known	new	long	Home
e2	reads	unknown	new	short	Work
e3	skips	unknown	old	long	Work
e4	skips	known	old	long	home
e5	reads	known	new	short	home
e6	skips	known	old	long	work

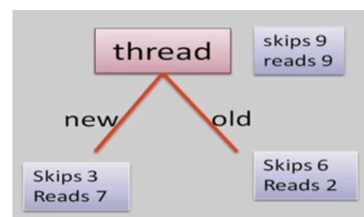
New Examples:

e7	???	known	new	short	work
e8	???	unknown	new	short	work

Machine Learning- Sudeshna Sarkar

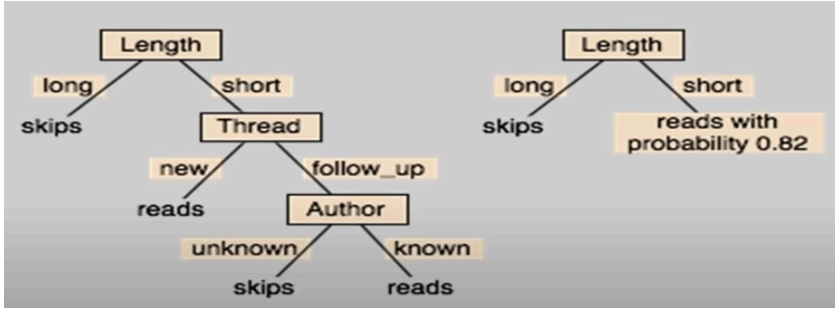
Many options for decision trees when there are multiple independent variables:





Leaf node because nothing to split it on

Continue until can make a prediction for the dependent variable on all paths



Machine Learning- Sudeshna Sarkar

Predict new/unseen samples by following decision nodes in the tree until reach a leaf

Example Data

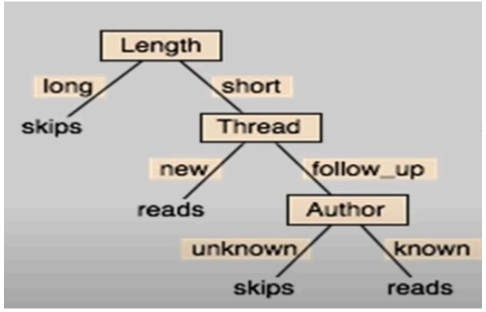
Training Examples:

	Action	Author	Thread	Length	Where
e1	skips	known	new	long	Home
e2	reads	unknown	new	short	Work
e3	skips	unknown	old	long	Work
e4	skips	known	old	long	home
e5	reads	known	new	short	home
e6	skips	known	old	long	work

New Examples:

e7	???	known	new	short	work
e8	???	unknown	new	short	work

Machine Learning- Sudeshna Sarkar



e7 and e8: short (Length) -> new (Thread)

Assumptions of the decision tree algorithms

- The goal is to find decision nodes and the splits which optimally separate the data into correct classes
- Initially all of the independent variables are considered for the root decision node
- Continuous independent variables need to be discretized or an optimal threshold must be computed (usually using the sum of squares of the residuals)
- The ordering of decision nodes is done using statistical methods
- If all classes are correctly separated by a decision node then the split is called "pure"

How to find a good tree that models your response variable

- Fitting a tree model is called “induction of a decision tree”
- The space of all decision trees is too large to use
- We have to utilize statistical techniques to find a root decision node and then go from there
- <u>Steps:</u>
  - Choose a root decision node based on a statistical step
  - For each of the subtrees use the same statistical step to pick the next independent variable for the next decision node
  - Continue until a prediction can be made with all paths (e.g. reach a leaf)
    - No root-to-leaf path should contain the same discrete attribute twice

Independent variable selection for a decision node

- There are several approaches for how a root decision node is selected
  - Information Gain
  - Gini index
    - Sometimes called Gini Impurity
  - Chi-square test

Information gain

- Represents a measurement of the amount of information that is gained by using the independent variable
  - Calculated using entropy
    - Amount of uncertainty/chaos/randomness
    - The less order/organization/relationship there is, the higher the entropy is
    - Sometimes entropy is used by itself for building a decision tree (without being a part of the information gain calculation)
  - Can be thought of as expected decrease in entropy if we partition data based on this variable
- A root decision node should be the one with the highest information gain
  - Then perform information gain calculation for the remaining independent variables at each split decision
- Used by ID3 (Iterative Dicotomizer3), C4.5, and C5.0 algorithms

Entropy: $$\boxed{E(s) = \sum_{i=1}^{c} -p_i \log_2 p_i}$$

p_i - probability of a category in a given class for a given input variable

Information gain (cont'd)

Information gain in data/current model if split on values of A

Entropy in the overall data/parent

Entropy in variable A

$$Gain(S, A) = Entropy(S) - \sum_{v \epsilon Values(A)} \frac{|S_v|}{|S|}.Entropy(S_v)$$

geeksforgeeks.org

S - set of all samples
A - independent variable
Values(A) - all unique value categories in A (all classes in A)
S_v - set of all samples where A = v

Information gain is biased to those variables with most observations

Information gain (cont'd)

$$Gain(S, A) = Entropy(S) - \sum_{v \epsilon Values(A)} \frac{|S_v|}{|S|}.Entropy(S_v)$$

```
For the set X = {a,a,a,b,b,b,b,b}
Total intances: 8
Instances of b: 5
Instances of a: 3
```

$$Entropy H(X) = -\left[\left(\frac{3}{8}\right) log_2 \frac{3}{8} + \left(\frac{5}{8}\right) log_2 \frac{5}{8}\right]$$

```
              = -[0.375 * (-1.415) + 0.625 * (-0.678)]
              =-(-0.53-0.424)
              = 0.954
```

geeksforgeeks.org

S - set of all samples
A - independent variable
Values(A) - all unique value categories in A (all classes in A)
S_v - set of all samples where A = v

Gini index

- Remember that splitting data into correct classes on an independent variable means the split is pure
- Gini index/impurity - likelihood that a randomly chosen observation is incorrectly classified by a given decision node
  - How much the model deviates from a pure split
  - Values are in [0,1] interval
    - 0 - pure split
    - 1 - random split
    - A variable with lower Gini index is preferred

Gini index (cont'd)

$$Gini = 1 - \sum_{i=1}^{n}(p_i)^2$$

https://blog.paperspace.com/decision-trees/

Gini index (cont'd)

Independent variables (A–D) | Response variable (E)

INDEX	A	B	C	D	E
1	4.8	3.4	1.9	0.2	positive
2	5	3	1.6	1.2	positive
3	5	3.4	1.6	0.2	positive
4	5.2	3.5	1.5	0.2	positive
5	5.2	3.4	1.4	0.2	positive
6	4.7	3.2	1.6	0.2	positive
7	4.8	3.1	1.6	0.2	positive
8	5.4	3.4	1.5	0.4	positive
9	7	3.2	4.7	1.4	negative
10	6.4	3.2	4.7	1.5	negative
11	6.9	3.1	4.9	1.5	negative
12	5.5	2.3	4	1.3	negative
13	6.5	2.8	4.6	1.5	negative
14	5.7	2.8	4.5	1.3	negative
15	6.3	3.3	4.7	1.6	negative
16	4.9	2.4	3.3	1	negative

geeksforgeeks.org

We choose some split values for our independent variables:

A	B	C	D
>= 5	>= 3.0	>= 4.2	>= 1.4
< 5	< 3.0	< 4.2	< 1.4

Calculating Gini Index for Var A:

Value >= 5: 12

Attribute A >= 5 & class = positive: $\frac{5}{12}$

Attribute A >= 5 & class = negative: $\frac{7}{12}$

Gini(5, 7) = 1 – $\left[\left(\frac{5}{12}\right)^2 + \left(\frac{7}{12}\right)^2\right] = 0.4860$

Value < 5: 4

Attribute A < 5 & class = positive: $\frac{3}{4}$

Attribute A < 5 & class = negative: $\frac{1}{4}$

Gini(3, 1) = 1 – $\left[\left(\frac{3}{4}\right)^2 + \left(\frac{1}{4}\right)^2\right] = 0.375$

By adding weight and sum each of the gini indices:

$$gini(Target, A) = \left(\frac{12}{16}\right) * (0.486) + \left(\frac{4}{16}\right) * (0.375) = 0.45825$$

Chi-square test

- Remember from statistics that Chi-square distribution models variance
- Chi-square method finds statistical significance of the variations that exist between parent nodes and their children nodes
  - Observed vs. expected frequencies of the dependent variable value
- This method can perform multiple splits at a single decision node

$$chi - square = \sqrt{\frac{(Actual - Expected)^2}{Expected}}$$

https://blog.paperspace.com/decision-trees/

Regularization in decision trees

- Just like with other types of models we can use regularization to prevent overfitting the model
  - Regularization - adding small bias to the model to minimize model variability (performs well on the training data but not on new data)
- We add regularization in a heuristic way (hard to implement Ridge or Lasso methods with decision trees)
  - We add “limiting” hyper-parameters to the decision tree model to restrict finding the “best fit” tree

Regularization in decision trees (cont'd)

Some of the regularization parameters

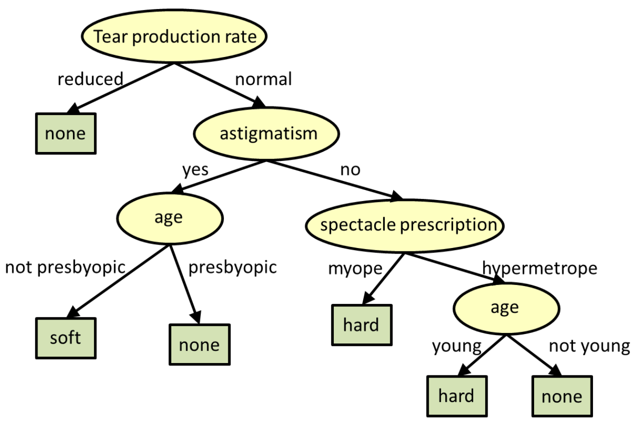
1. **Max_depth:** It is the maximal length of a path that is from root to leaf. Leaf nodes are not split further because they can create a tree with leaf nodes that takes many inspections on one side of the tree whereas nodes that contain very less inspection get again split.
2. **Min_sample_spilt:** It is the limit that is imposed to stop the further splitting of nodes.
3. **Min_sample_leaf:** A min number of samples that a leaf node has. If leaf nodes have only a few findings it can then result in overfitting.
4. **Max_leaf_node:** It is defined as the max no of leaf nodes in a tree. (Relatable article: What are the Model Parameters and Evaluation Metrics used in Machine Learning?)
5. **Max_feature_size:** It is computed as the max no of features that are examined for the splitting for each node.
6. **Min_weight_fraction_leaf:** It is similar to min_sample_leaf that is calculated in the fraction of total no weighted instances.

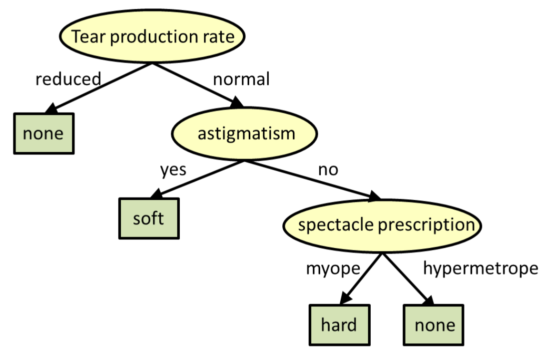
https://www.analyticssteps.com/blogs/introduction-decision-tree-algorithm-machine-learning

Tree pruning

- Another way to add regularization to a decision tree is by pruning it
- Pruning has been shown to significantly improve prediction accuracy
- Types of pruning
  - Pre-pruning
    - Replace stop criterion in the decision tree induction
      - Minimum information gain or max Gini index
    - More efficient than post-pruning (do not need to run induction on the whole training set)
      - Pre-mature termination
  - Post-pruning
    - A way to simplify a tree after induction is completed
    - Decision nodes and sub-trees are replaced with leaf nodes
      - At random or in some systematic way

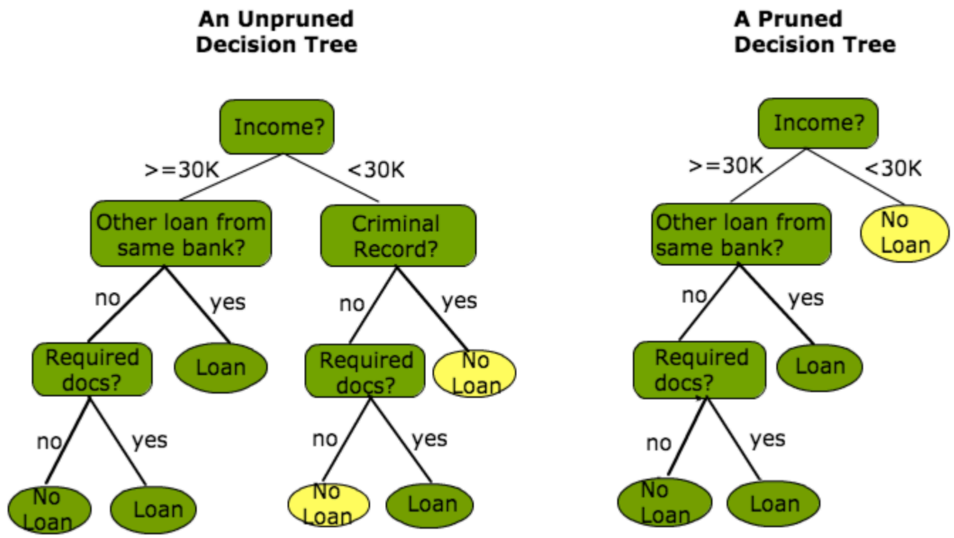
Tree post-pruning example 1





https://www.cs.cmu.edu/~bhiksha/courses/10-601/decisiontrees/

Tree post-pruning example 2

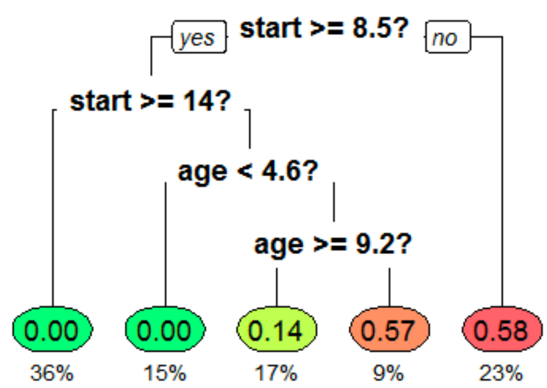


medium.com

Regression trees

- Instead of predicting a categorical outcome variable regression trees predict a continuous variable
- The leaf values are usually computed as an average dependent variable value across all observations in that node

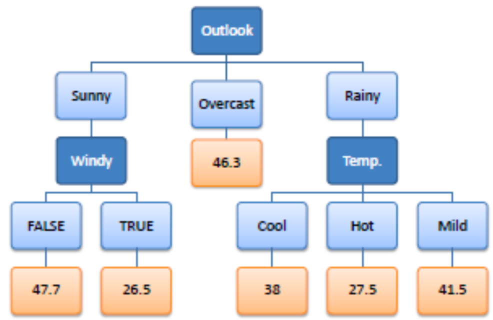
Probability of kyphosis after surgery:



en.wikipedia.org

Regression trees predict a continuous outcome

Predictors				Target
Outlook	Temp	Humidity	Windy	Hours Played
Rainy	Hot	High	False	25
Rainy	Hot	High	True	30
Overcast	Hot	High	False	46
Sunny	Mild	High	False	45
Sunny	Cool	Normal	False	52
Sunny	Cool	Normal	True	23
Overcast	Cool	Normal	True	43
Rainy	Mild	High	False	35
Rainy	Cool	Normal	False	38
Sunny	Mild	Normal	False	46
Rainy	Mild	Normal	True	48
Overcast	Mild	High	True	52
Overcast	Hot	Normal	False	44
Sunny	Mild	High	True	30



https://www.saedsayad.com/decision_tree_reg.htm

Advantages of using decision trees

- Have been shown to work well in making decisions in complex scenarios/problems
- Can produce simpler models than other methods
- Capture non-linear relationships well
- Can deal with continuous as well as categorical variables
- Both the model and its predictions are easily interpretable and do not require special expertise
- Requires little to no data preprocessing

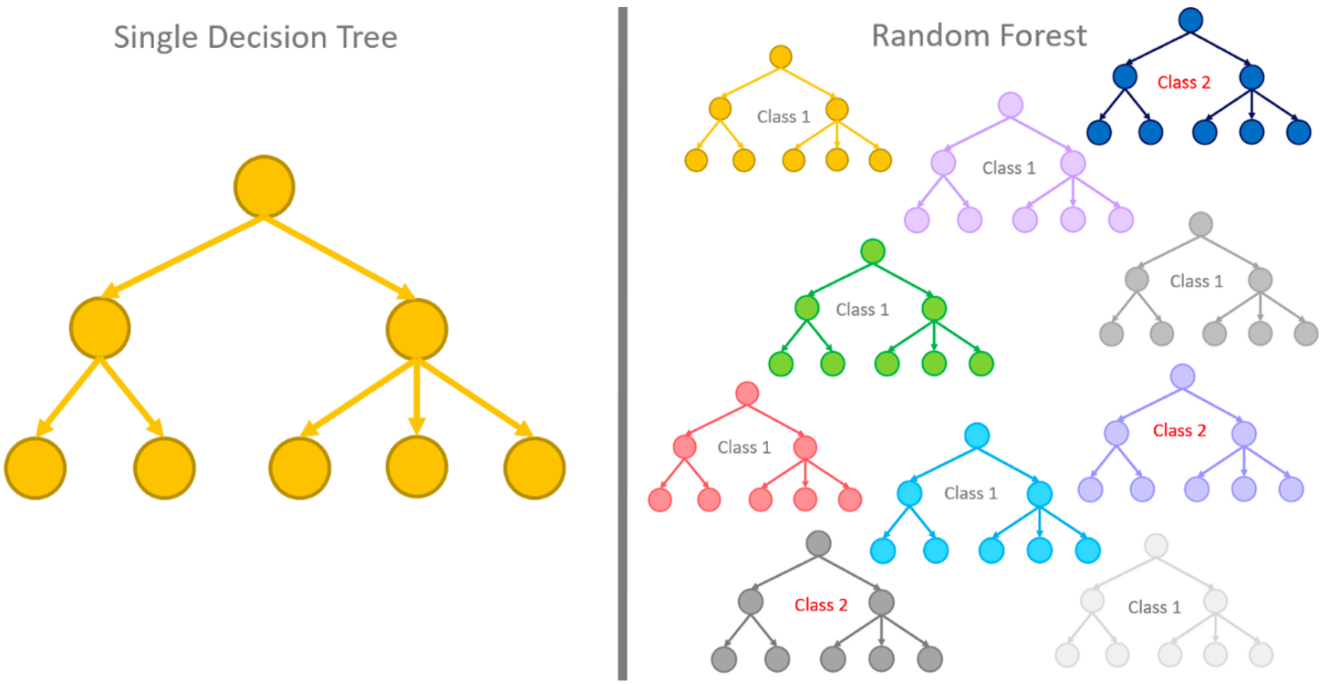
Disadvantages of using decision trees

- Might change a lot if there are even small changes to the training data
- Large trees are harder to interpret
- Without regularization can grow a lot in depth

Random forest models

- An ensemble learning method based on decision trees
  - A forest of decision trees
- Create a number of different decision trees from subsets of the independent variables
- Useful for datasets with large number of features (independent variables)
- Solves the tendency of single decision trees to overfit the model to the training data
- Average the prediction across multiple decision trees in the forest

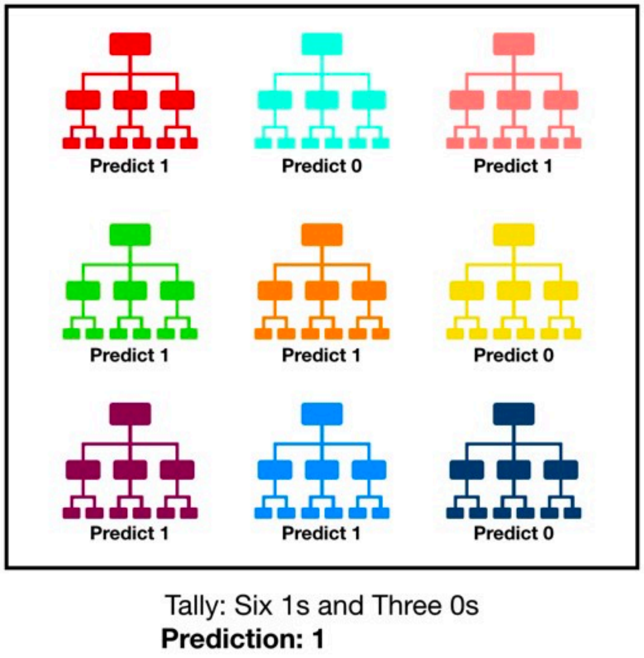
Random forest models (cont'd)



https://towardsdatascience.com

Majority classification is *Class 1*

Random forest models (cont'd)

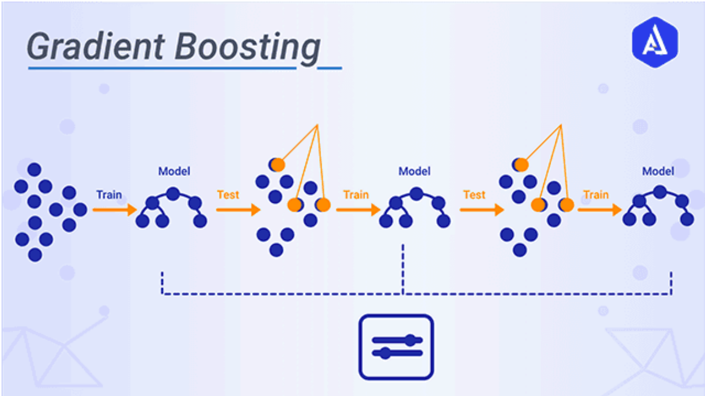


https://towardsdatascience.com

Gradient boosting models

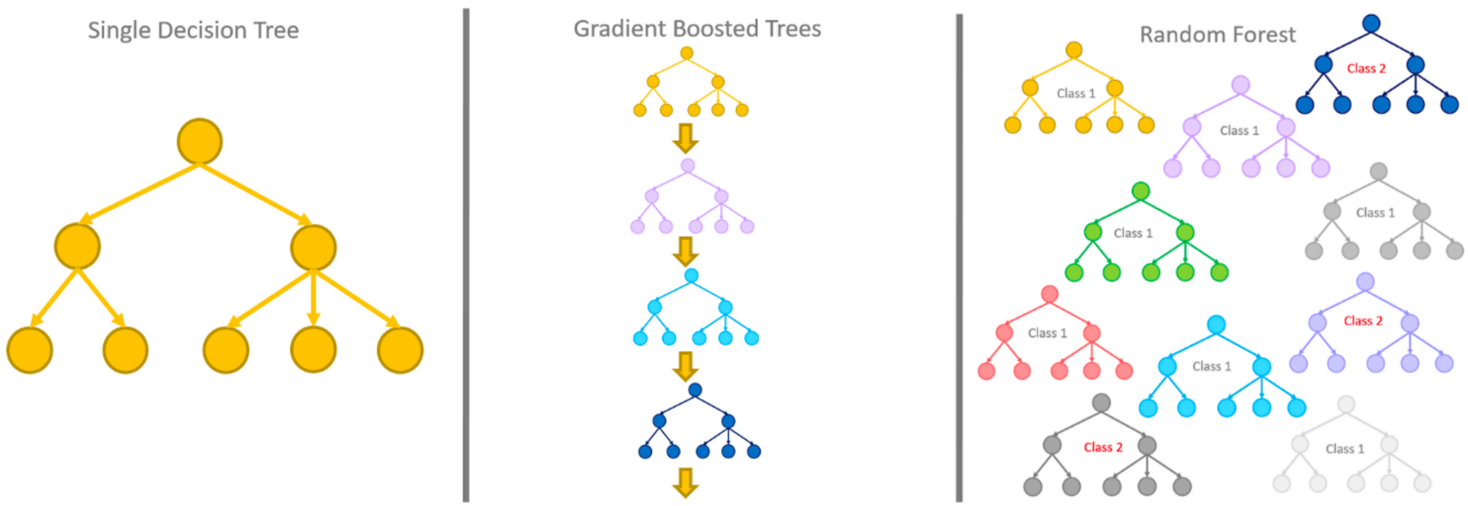
- Decision tree based model that aggregates less accurate models into a more accurate model
  - We usually mean shallow decision trees
  - Each weak predictor compensates the weaknesses of its predecessor predictors
- Boosting reduces model bias
- Big idea: one is weak, together we are strong, learning from past is the best

Train-test cycle:



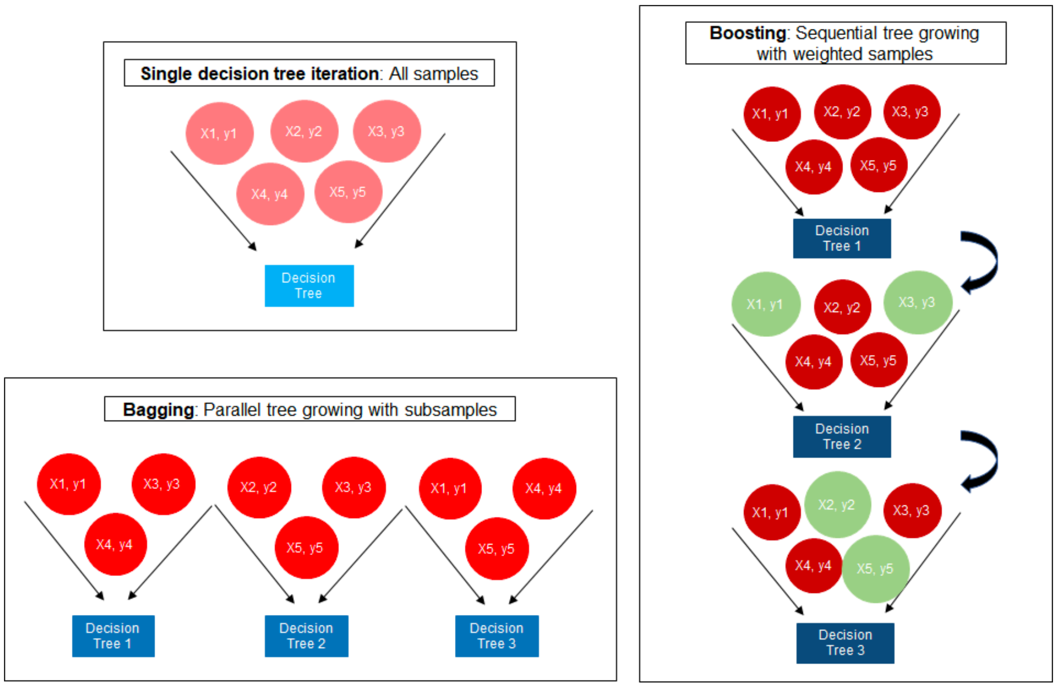
https://www.akira.ai/glossary/gradient-boosting/

Gradient boosting vs. random forest



https://medium.com

Gradient boosting models (cont'd)



https://towardsdatascience.com

Some concluding remarks

- Let’s look at some python code examples in Jupyter notebooks
  - *DecisionTrees.Breast.ipynb*