Decision Trees

10 March 2024 22:06

Why decision trees?

1. NonLinearity:

Unlike regression models, which assume a linear relationship between the predictors and the target, decision trees can handle complex interactions and non-linearities.

It can do without needing for any transformation of the features.

Decision trees can capture non-linear relationships between predictors and the target.

2. Interpretability:

Decision rules represented by the tree structure are easy to understand and visualize, making them accessible to executives who are non-experts.

[VP, Director] ← logit log(odds).

DTs are like a flow chart and business users love flow chart.

3. Scalability:

Decision trees are computionally efficient and scalable to large datasets, making them suitable for real-time applications and large-scale data processing

Training dataset = 50M rows → on big data

~~Linear / Logistic~~

↓

DT

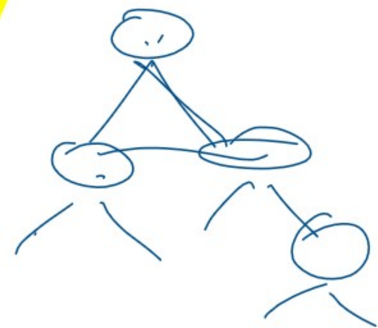
4. Handling Mixed Data Types

DTs can handle both numerical and categorical data without the need for one-hot encoding

5. Robustness to outliers and irrelevant features

DTs can handle outliers & noise in the data along with irrelevant features without significantly impacting the model performance.

[Gini impurity, Entropy] → to evalute the important features



6. Automatic selection of features

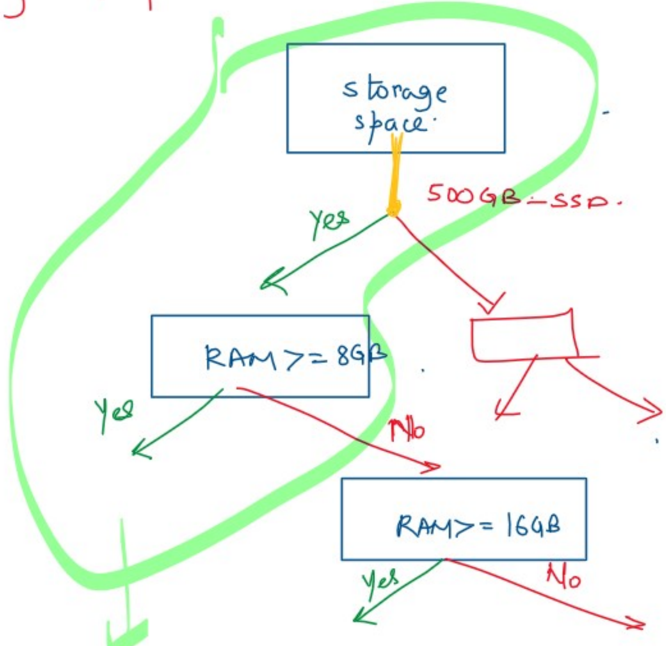
DTs perform automatic feature selection by identifying the most important features at each split.

Intuition behind Decision Trees

"Decision trees are everywhere."

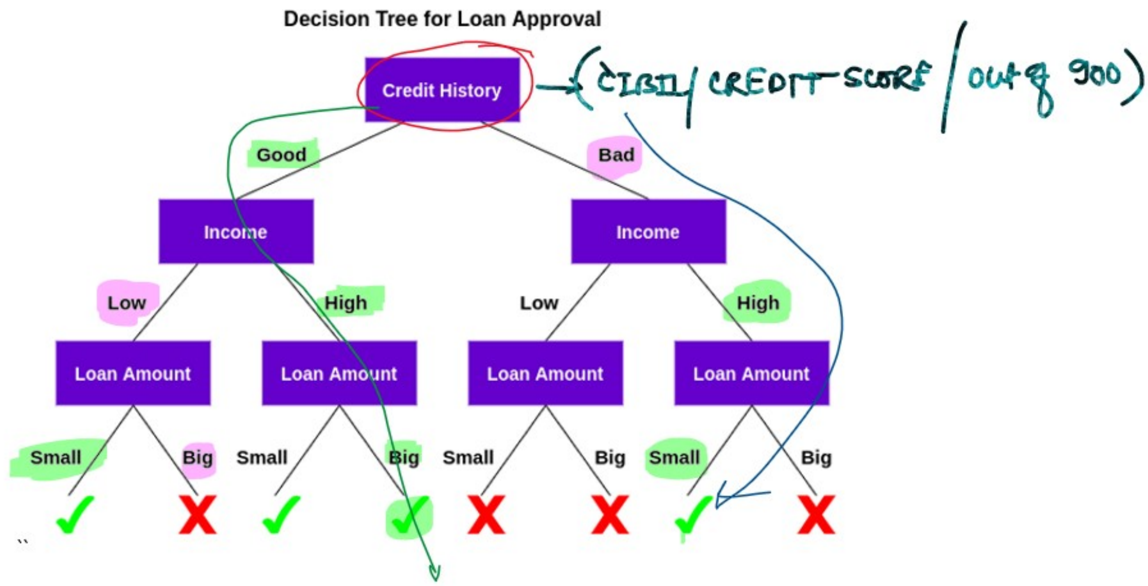
Senthil → is trying to buy a laptop.

PRICE
BRAND
RAM
ROM
CPU
GPU
DISPLAY

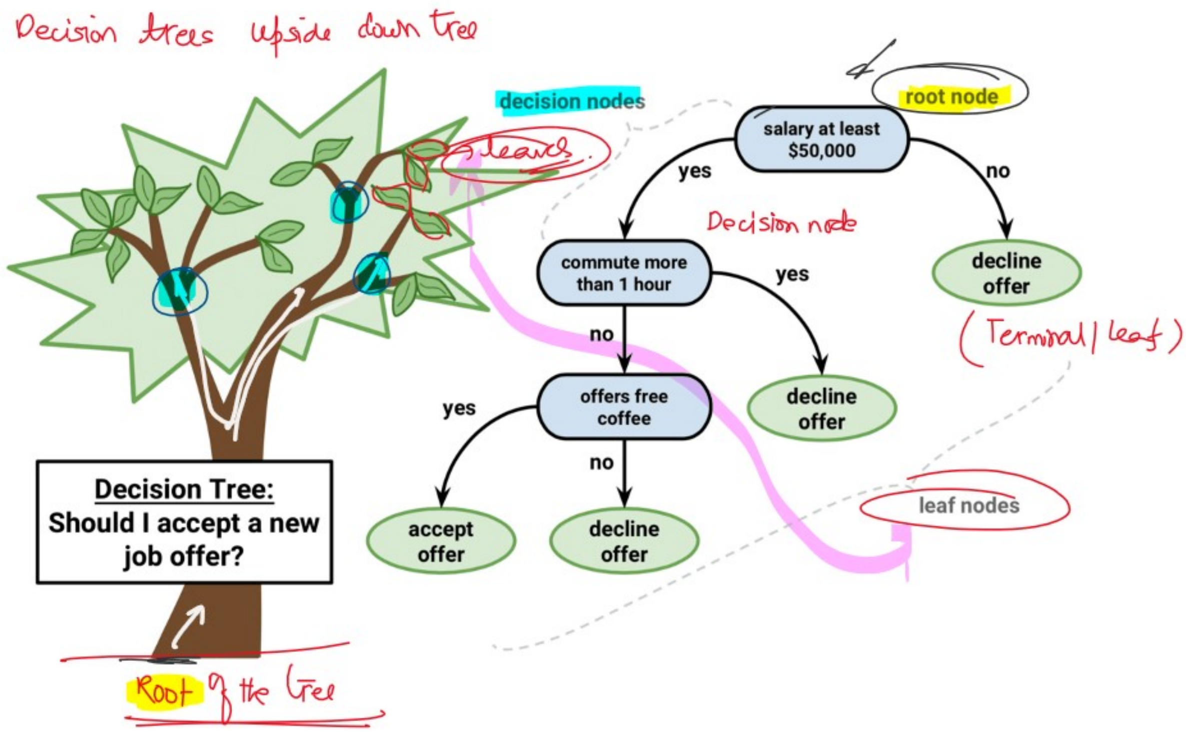


"Buy the laptop"

Loan approval Decision Trees



Decision trees upside down tree



Root Node:

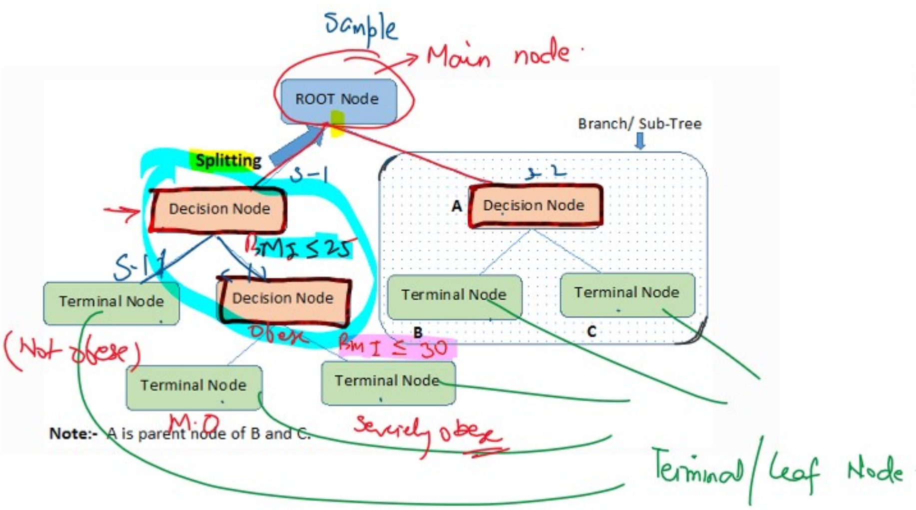
It represents entire population or sample and this further gets divided into two or homogeneous sets.

Splitting

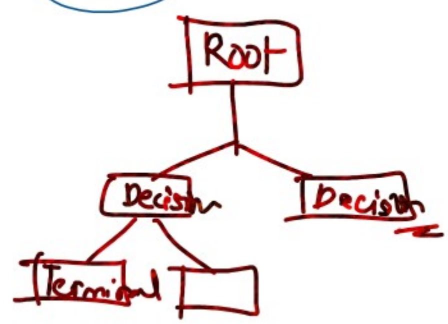
It is a process of dividing a node into two or more subnodes.

Decision Node:

When a sub-node splits into further sub-nodes, it is called a decision node.



Pruning



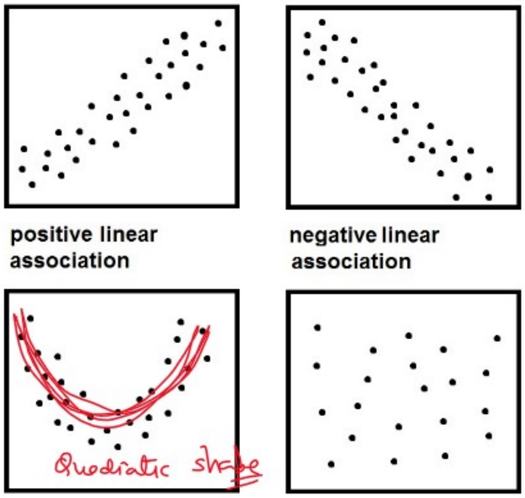
Leaf / Terminal Node:

Nodes which do not split is called a leaf/ terminal node.

Branch/ Sub-Tree

It is a sub-section of entire tree.

scatter plot



positive linear association

negative linear association

nonlinear association

no association

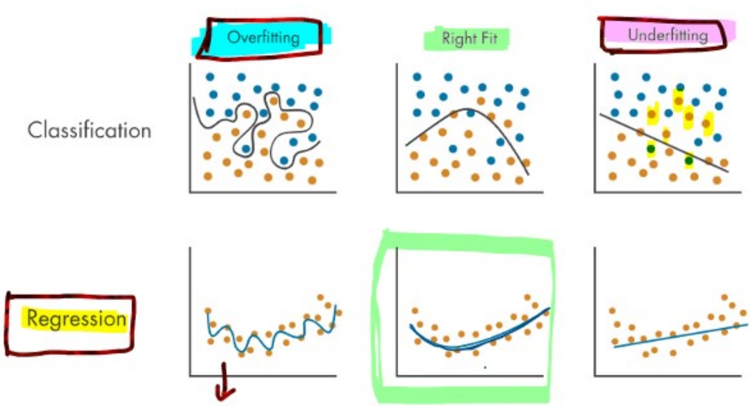
Disadvantage / Cons

overfitting: one of most practical difficulties for decision tree models.

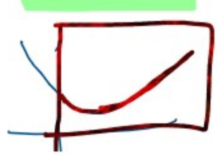
Overfitting and Underfitting

overfitting:

It occurs when a model learns the training data too well (too much), capturing noise or random fluctuations as well that may not be representative of true underlying patterns in the data.

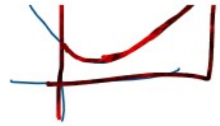


equation ?



$ax^2 + bx + c$

quation:



ax^2+bx+c

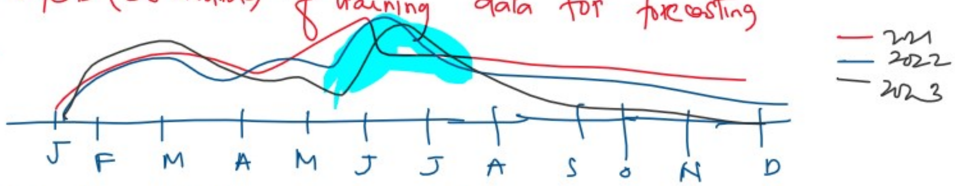
Signs of overfitting:

- Model performs exceptionally well on the training data however it fails to generalize to new/unseen data.
  (performs poorly on the test set)

Reasons for overfitting

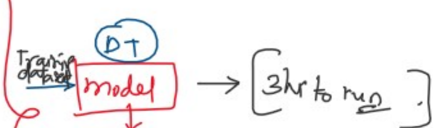
- using a highly complex model with too many parameters.
- having insufficient data to support the complexity of the model.

  For ex: Need 3 years (36 months) of training data for forecasting use-cases.



- training the model for a lot number of epochs.

  One epoch means in ML, one complete pass of the training dataset through the algorithm.



Mathematical model with parameters

Mitigation:

- Use simpler models ⇒ a) less no. of features

  ~~100~~ 5 — 20

  (5–20)

  b) Do not overkill with the no. of training parameters.

- Use/increase the amount of training data.

- Regularization techniques to penalize overly complex models.

Error	Overfitting	Right Fit	Underfitting
Training	Low	Low	High
Test	High	Low	High

all the modelers pursue this

Example: Correlation vs Causality

X: Coding hours
Y: Exam score (out of 100)
Z: Lines of code.

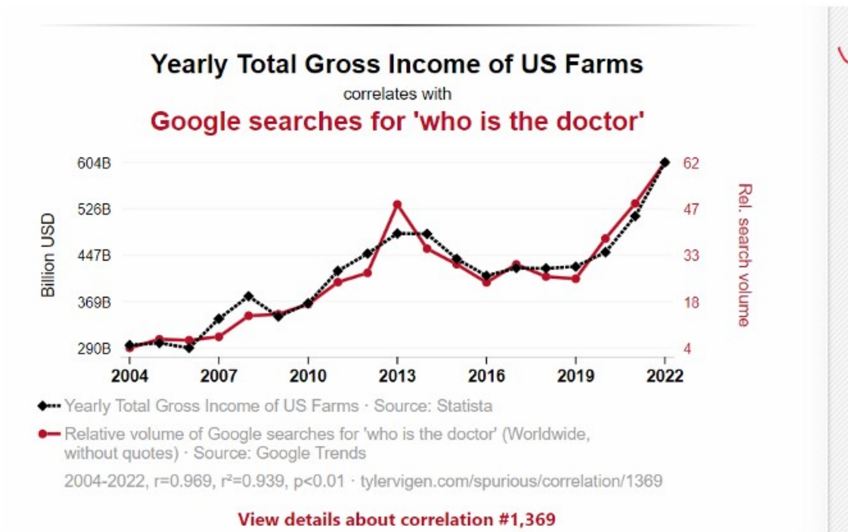
Python Exam

[X and Y are highly correlated]

~~High Z → Better Y (exam score)~~

(Not correct)

conclusion: X and Z are highly correlated but they are not related.



Y → $\boxed{X_{10}}$ → 80+

without quotes) · Source: Google Trends
2004-2022, r=0.969, r²=0.939, p<0.01 · tylervigen.com/spurious/correlation/1369

View details about correlation #1,369

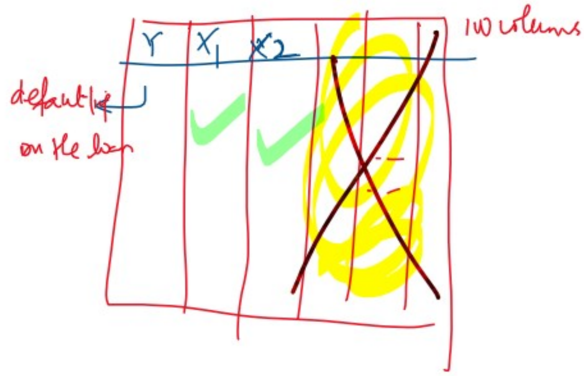
https://www.tylervigen.com/spurious-correlations

Underfitting

- Underfitting occurs when a model is too simple to capture the underlying patterns in the training data, resulting in poor performance on both training & testing.

Signs of underfitting

- the model performs poorly on training data
- model is very simple, with almost no features.
- it also performs poorly on test set.



Reasons for underfitting

- too simple model with too few parameters
- insufficient training or not allowing the model to learn enough during training.

Mitigation

- increasing the model complexity by adding more parameters.
- ensure sufficient training time and training data.
- use a sophisticated model.

- cross-validation (CV)
- regularization

to strike the right balance between underfitting & overfitting.

- regularization
- hyperparameter tuning

to strike the right balance between underfitting & overfitting.

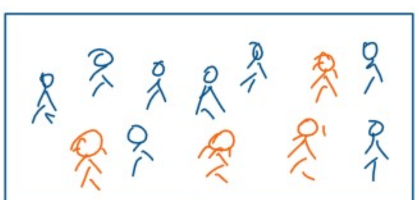
Where to split?

DTs use algorithm(s) to decide to split a node in two or more subnodes basis a criterion – Gini algorithm

Gini algorithm

Want to group students based on target variable – playing cricket or not.

Variable #1 Gender: Split on Gender



N=30 students – [across the classes 9 and 10]

↓

Play cricket = $\boxed{15}$ (50%)

↓

(# Total # students who play cricket)

sub-node-1

Male

male 20 students

↓

13

$\frac{13}{20} \times 100 = 65\%$

sub-node-2

Female

10 female students

↓

2

$\frac{2}{10} \times 100 = 20\%$

13+2

Calculate Gini for sub-nodes, using formula.

$$(p^2 + q^2)$$

where p = probability of success

and q = probability of failure

and q = probability of failure

① Gini index for sub-node male $= \underbrace{(0.65 * 0.65)}_{p^2} + \underbrace{(0.35 * 0.35)}_{q^2}$

$p = 0.65$

$1 - p = q = 1 - 0.65 = 0.35$

$= 0.4225 + 0.1225$

$= .5450$

$\approx .55$

② Gini index for sub-node female $= (0.2 * 0.2) + (0.8 * 0.8)$

$p = 0.2$

$q = 1 - p = 0.8$

$= 0.04 + 0.64$

$= 0.68$

calculated weighted gini index for split at gender

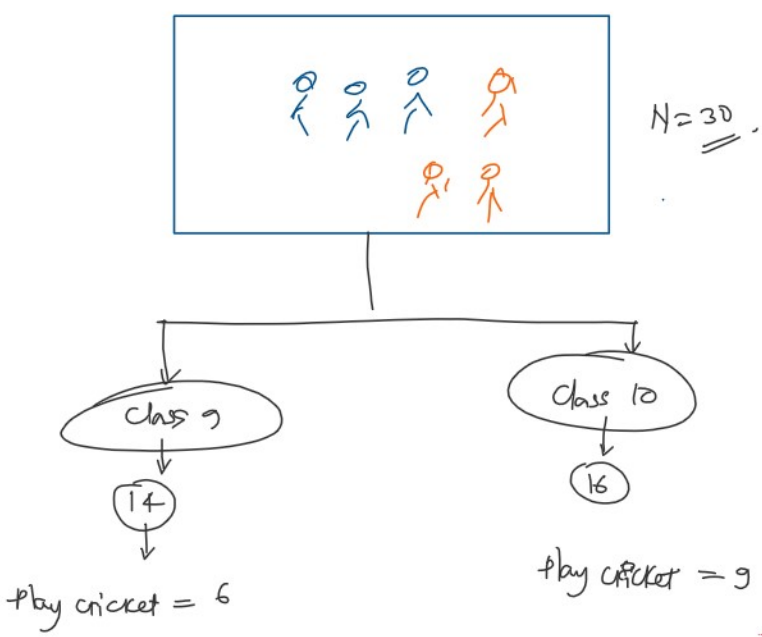
(# students in each M & F)

$= \left(\frac{20}{30} \times 0.55 + \frac{10}{30} \times 0.68\right)$

= 20*.55/30 + 10*.68/30=0.5933

split @ gender → Gini index → 0.5933.

Variable #2 class



Play cricket = 6

$\frac{6}{14} \times 100 = 43\%$

Play cricket = 9

$\frac{9}{\cancel{16}_{4}} \times \cancel{100}^{25}\% = 56\%$

9/16=0.5625

Gini-index for subnode class IX

$= (0.43 * 0.43) + (0.57 * 0.57)$

$= 0.51$

Gini-index for sub-node class X

$= (0.56 * 0.56) + (0.44 * 0.44)$

$= 0.51$

weighted Gini-index is = 0.51

Conclusion: Gini index for Gender is higher than split on class, the node (root) split will take place on gender.

Gini impurity = (1 − Gini)

Gini impurity) split on Gender $= (1 - 0.59)$

$= 0.41$.

Gini

~~or~~

Entropy