DECISION TREE

Introduction: decision tree in machine learning

AGENDA

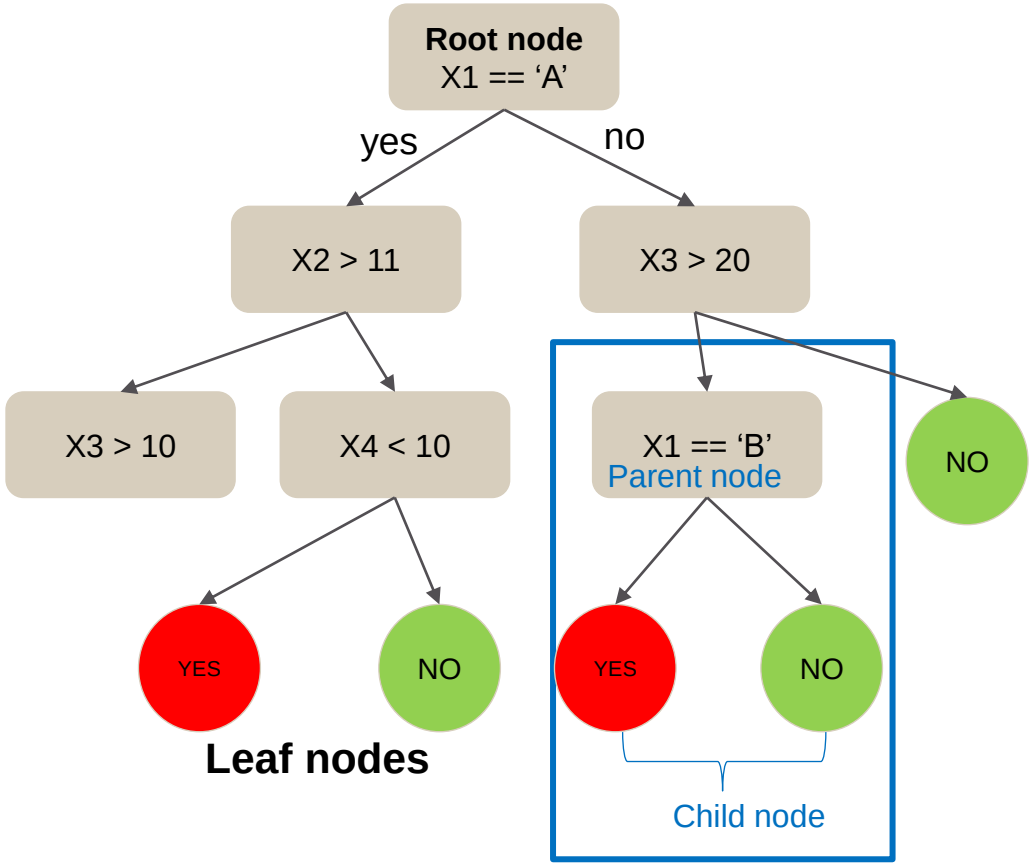
- Decision tree concept
- Construct a decision tree
- Decision tree regressor
- Ensemble tree base model

MACHINE LEARNING METHOD: DECISION TREE

- **Concept:**
  A tool to support coming up to the decision generally it is used a lot in binary classification question, but it can also be used in multiple classification question and even in regression

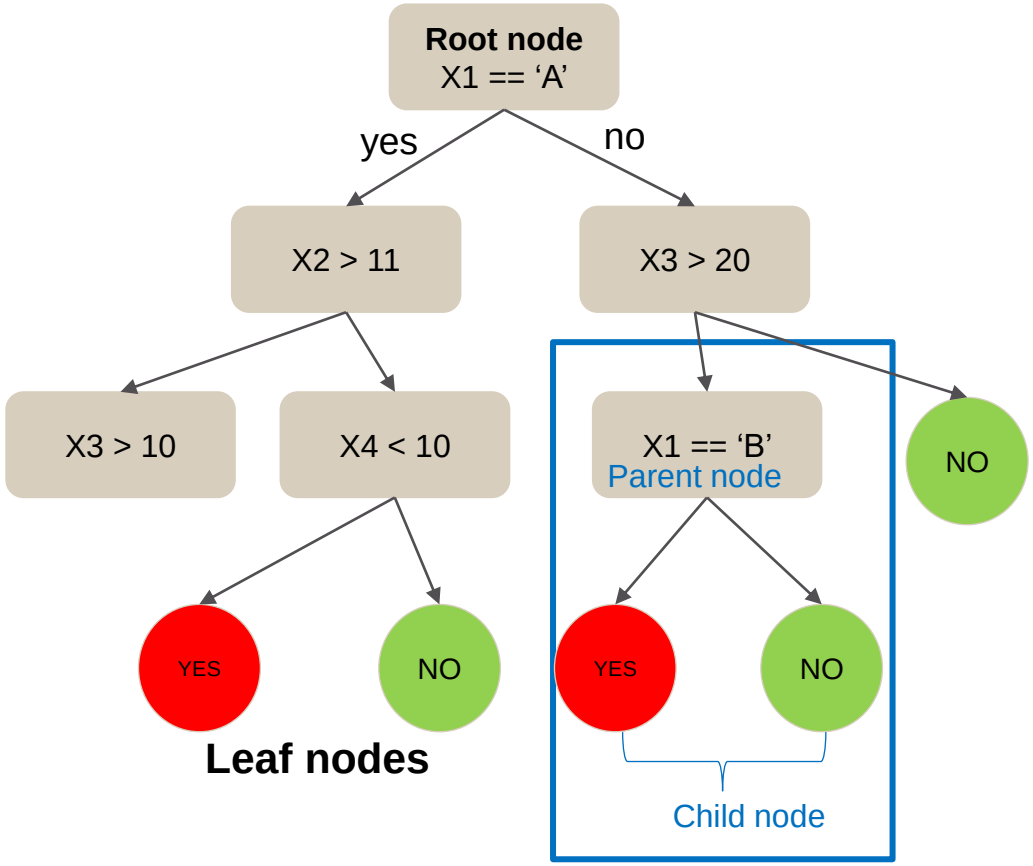
- **Many advanced model base:**
  Many ML methods are based on ensemble decision tree, such as **Random forest (bagging model) and XGBoost/ CatBoost (boosting model)**

- **Easier explained**
  It is commonly used in many project for its **explainability** for business. This can easily be graphically displayed (plot the trees) , can show the feature importance, how the feature is contributed to the model, to each individual prediction (SHAP values).



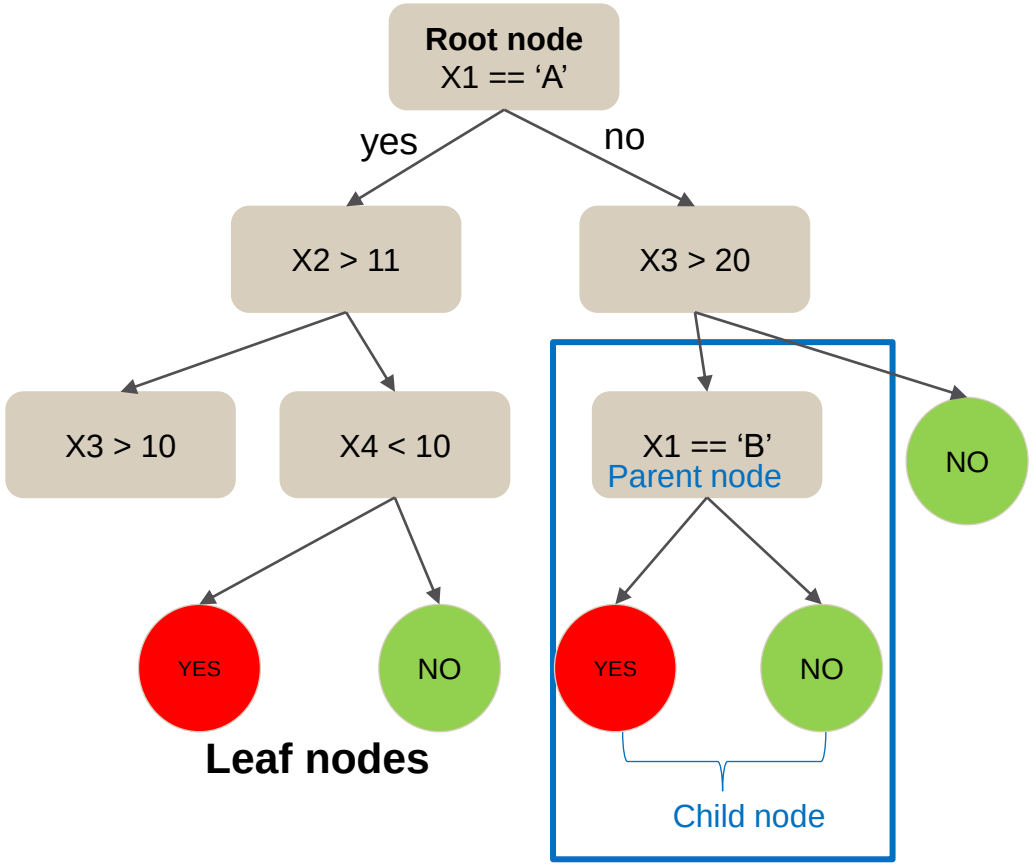
CONSTRUCT A TREE

- **Select the features to use in a node**
  It would try out all the features and values
- **Which split: maximize information gain**
  Impurity of a node: Entropy, Gini impurity
  - Entropy = Σ - P_i log(P_i) [0-1]
  - Gini impurity = 1 - $\Sigma(P_i)^2$ [0-0.5] (computational efficient)
  - Information Gain = I(parent) – Σ w_i I($child_i$)
- **Stopping points:**
  - minimize the wrongly predicted class in each class (pure subset, or defined stopping points)
  - Stopped at a certain level of depth



DECISION TREE REGRESSOR

- **Regression problem:**
  Decision tree generally is made for solving classification problem, but it can also be used in regression problem.
- **Do the split:**
  1. Scan: the same as before scan all the features and the values
  2. Compute the impurity of the node!
     Maximize Variance reduction
  - Var = 1/n Σ $(Y_i - Y_h)$ [0-1]
  - Variance reduction = Var(parent) – Σ w_i Var($child_i$)
- **Prediction:**
  the mean of the target value in a leaf node



ADVANCED ENSEMBLE MODEL FROM DECISION TREE

- Decision tree is a greedy search algo, do not go back to change the previous split.
  Therefore to get more precision and stable model, we often use ensemble model.

Bagging:

- Bagging method is to train multiple models, using different subset of data with replacement (bootstrap) independently and parallelly.
- Well known models: Radom forest Bagging (bootstrap aggregation)

Boosting

- Boosting are not independent, models has to be trained sequentially, only the wrongly predicted records would be pass to the next models.
- Well known models: AdaBoost, gradient boosting, Xgboost, lightgbm, CatBoost

Q&A

APPENDIX: ENSEMBLE TREE BASE MODELS

Bagging:

- Random forest

It is transform of bagging method of decision tree, it choose random subset and even random features to train an individual decision tree. Finally combine the result to choose the majority predicted class for classifier or mean/median of the predicted value for regressor.

Boosting

- Adaboost:

decision tree to create only one stomps that is depth 1 decision tree. It is a transform of boosting, updating the weight of the records, increase the wrongly predicted weight and decrease the correctly predicted weight and normalize it to create another model. In the end also choose the majority of the prediction.

- Gradient boosting:

After having the base model, a sequence of decision trees are computed to predict the residuals. It would use the base model predict value and add up all tree's residual (errors) with a learning rate.