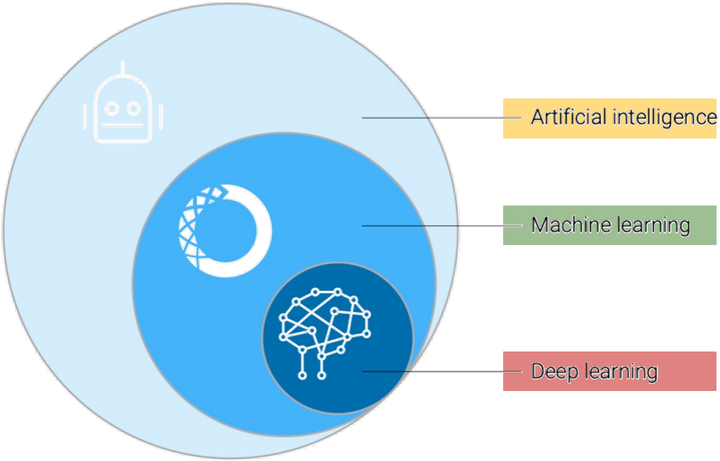
## Lecture 8

# Supervised Learning

**Haoyu Yue** / yohaoyu@washington.edu
Ph.D. Student, Interdisciplinary Urban Design and Planning
University of Washington

RE 519 Real Estate Data Analytics and Visualization
Course Website: www.yuehaoyu.com/data-analytics-visualization/
Autumn 2025

# Artificial Intelligence vs Machine Learning




Source: Visualizations for Machine Learning (Iris Series)

... make computers do the sorts of things that minds can do.

**Margaret Boden (AI: A Very Short Introduction) 2018**

... **activity** devoted to making machines intelligent, and intelligence is that quality that enables an entity to function appropriately and with foresight in its environment.

**Nils J. Nilsson 2010**

...the field of study that gives computers the ability to learn without being explicitly programmed.

**Samuel 1959**

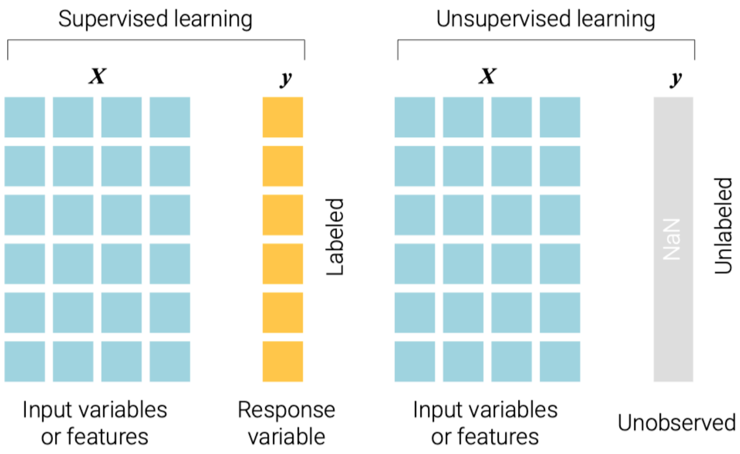
A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E.

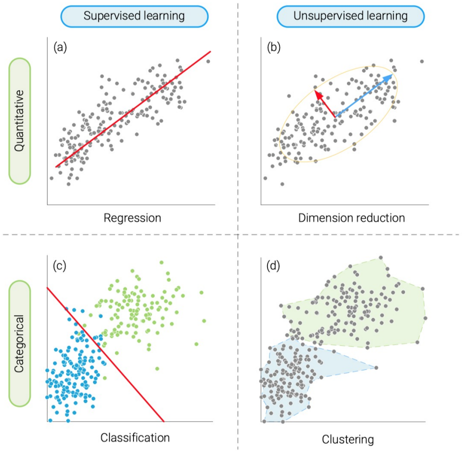
**McGraw-Hill 1997**

... is a subset of representation learning methods that use multiple layers of nonlinear processing units to learn hierarchical representations of data.

**Ian Goodfellow, Yoshua Bengio, Aaron Courville 2016**

# Supervised vs Unsupervised Learning



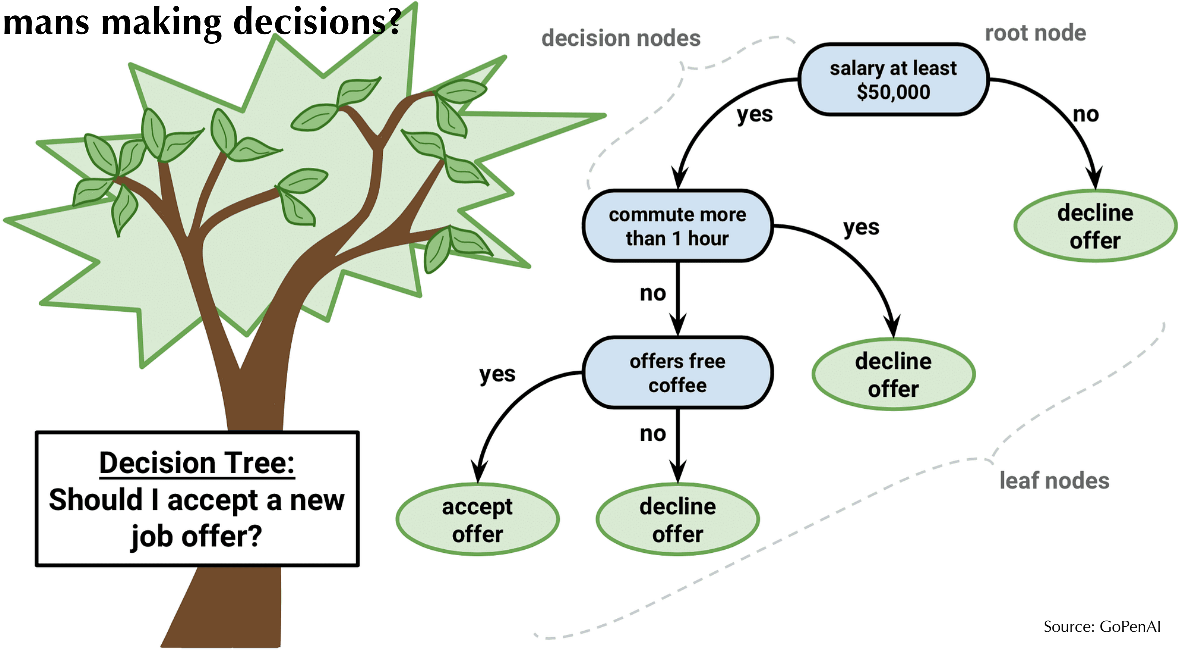

Source: Visualizations for Machine Learning (Iris Series)

**Supervised learning**: Learning a function that maps inputs to outputs using labeled examples (Bishop, 2006).

**Unsupervised learning**: Learning hidden structure from unlabeled data (Hastie, Tibshirani & Friedman, 2009).

# Decision Tree

## How are humans making decisions?

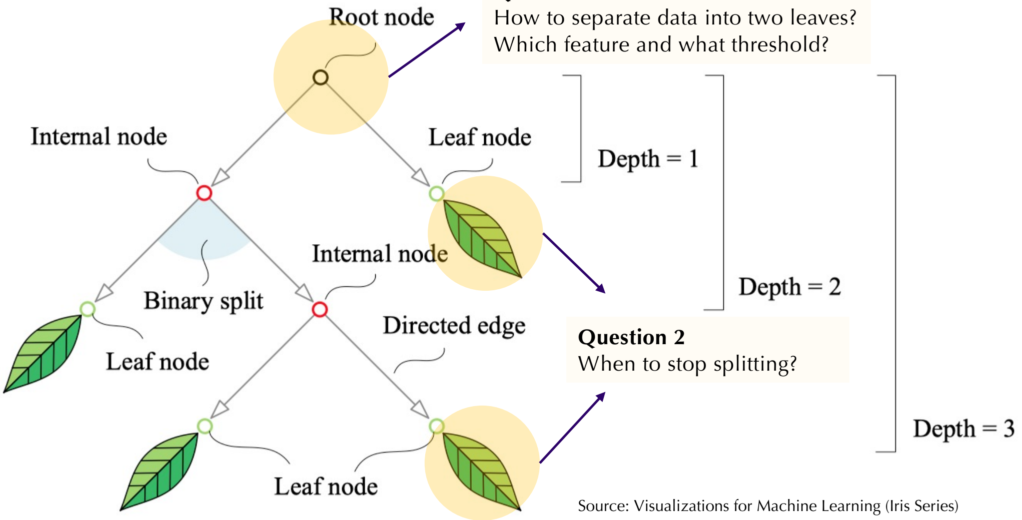



Source: GoPenAI

# Decision Tree

## Structure - CART (Classification And Regression Tree)

A decision tree is **a set of rules** that can be learned from data and used to predict an unknown value. It could be used for both regression and classification.




**Question 1**
How to separate data into two leaves?
Which feature and what threshold?

**Question 2**
When to stop splitting?

Source: Visualizations for Machine Learning (Iris Series)

# Decision Tree

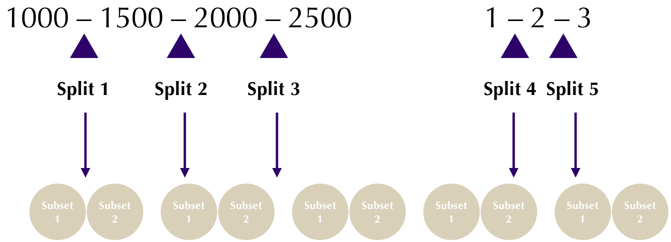
## Question 1: How to separate data into two leaves?

| X - SQFT | X - Bed | Y |
|---|---|---|
| 1000 | 1 | 1 |
| 1000 | 2 | 0 |
| 1500 | 2 | 1 |
| 2000 | 3 | 1 |
| 2500 | 2 | 0 |

The tree prefers splits that make the left and right groups **as pure as possible**.

But, how to define purity or impurity?

There will be 5 ways to split the data:




Calculate the **purity** for each split and pick the purer one.

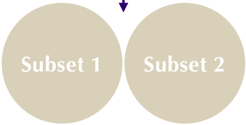
*Notes: if feature X is categorical, CART will try all pairs of subsets 1 & 2.*

# Decision Tree

## Question 1: How to separate data into two leaves?

1000 – 1500

Split 1




GINI = 0.5 GINI = 0.44

**For classification**: GINI Index $GINI(t) = 1 - \sum_{j} p(j|t)^2$

Total pure (only single value): GINI = 0

**For regression**: Variance $\text{Weighted Impurity} = \frac{1}{N_{total}} \sum_{i \in left} (y_i - \bar{y}_t)^2 + \frac{1}{N_{total}} \sum_{i \in right} (y_i - \bar{y}_t)^2$

| X - SQFT | X - Bed | Y | Subset |
|---|---|---|---|
| 1000 | 1 | 1 | 1 |
| 1000 | 2 | 0 | 1 |
| 1500 | 2 | 1 | 2 |
| 2000 | 3 | 1 | 2 |
| 2500 | 2 | 0 | 2 |

There are many other indices to show impurity, but GINI and weighted variance are most commonly used in CART.

# Decision Tree

## Question 2: When to stop splitting?

**If we have too few samples in a node**, for example, only 2 samples in a node, it will be meaningless to split

**If we have a tree with too large depth**, for example, we have a tree with depth = 100, it becomes too complex.

**If we cannot get purer splits,** for example:

- All samples in a node have the same X (features).
- All samples in a node have the same Y (labels).
- Cannot get a smaller impurity.

| X - SQFT | X - Bed | Y |
|---|---|---|
| 1000 | 1 | 1 |
| 1000 | 1 | 0 |
| 1500 | 2 | 1 |
| 2000 | 3 | 1 |
| 2500 | 2 | 1 |
| 3500 | 4 | 1 |
| 3500 | 4 | 0 |
| 4000 | 5 | 0 |
| 4000 | 5 | 1 |

# Decision Tree

## Overfitting

A decision tree has a serious problem of overfitting: decision trees tend to grow very deep and complex if we do not restrict their growth.

We can limit the complexity of trees by setting some important **hyperparameters** before training:

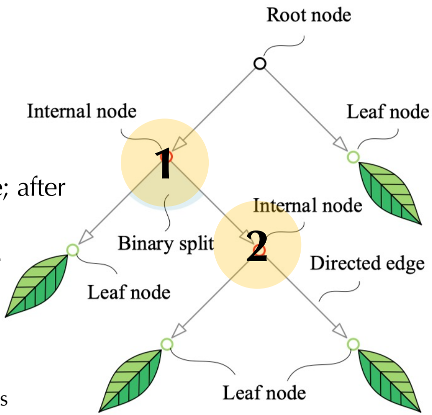
- Max depth
- Minimal samples split
- Minimal samples leaf
- Max leaf nodes

**How to decide the hyperparameters?** → **Cross Validation!**

**Cost Complexity Pruning (CCP)**

*A after-training method*

1. Prune 1 and 2, calculate **the ratio of impurity increase/leaf decrease**
2. Compare and decide which to prune; after prune, set it as one subtree
3. Continue 1~2 until we cannot prune
4. Calculate the cost for all subtrees
5. Pick one with a smaller cost




Rerun on training data and get the errors

$$\text{Cost} = \text{Training Error} + \alpha \times \text{Number of Leaf}$$

$\alpha$: hyperparameters, decided by us

Number of Leaf: Represent the complexity of trees

# Decision Tree

## Pros and Cons of Decision Trees

PROS

- Easy to learn the tree from the data, and easy to predict
- Can be used for both classification and regression
- Input features can be both continuous and discrete
- Nice performance in general
- Easy to visualize and explain for small trees
- Give an idea of which variables are important (tend to show up at the top of the tree)

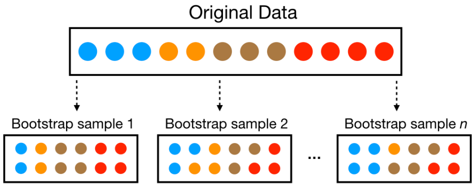
CONS

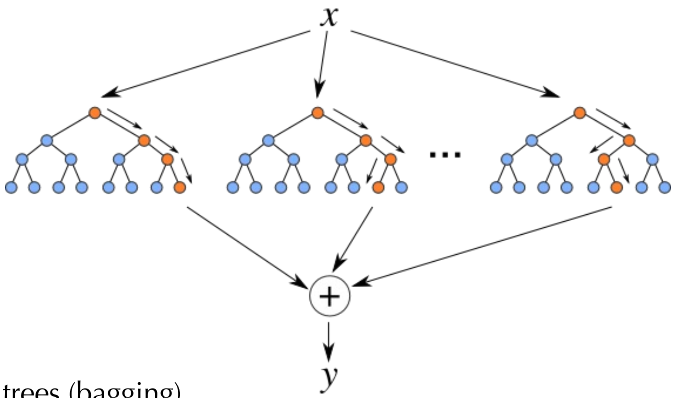
- So many possible tree structures, and our method may not be able to find the best one
- Will not consider interaction between features. Only use one feature in each split.

# Tree Ensembles

## Random Forests

- Sum predictions across multiple decision trees



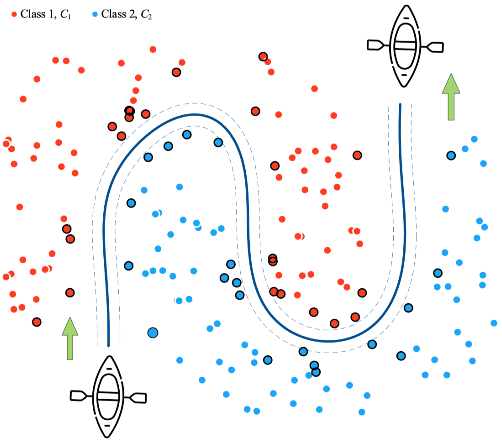




- **Random forests:** bootstrap training data and aggregate trees (bagging), ensemble of independent strong learners
- **Gradient boosting machines:** combine weak learners, let new trees improve on previous ones (gradient boosting)

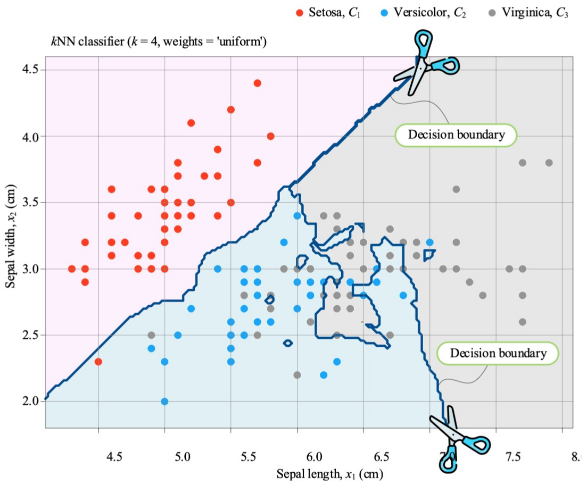
Source: https://www.datasciencecentral.com/decision-tree-vs-random-forest-vs-boosted-trees-explained/; https://datasciencedojo.com/blog/bootstrap-sampling/

# Other Supervised Learning Methods

**Support Vector Machines (SVM)**
Finds the decision boundary that maximizes the margin between classes.




**K-Nearest Neighbor Classification (KNN)**
Classifies a new point based on the majority label of its k nearest neighbors.



# Reminders

# Thank you!

**Haoyu Yue** / yohaoyu@washington.edu
Ph.D. Student, Interdisciplinary Urban Design and Planning
University of Washington

RE 519 Real Estate Data Analysis and Visualization
Course Website: www.yuehaoyu.com/data-analytics-visualization/
Autumn 2025

The course was developed based on previous instructors: Christian Phillips, Siman Ning, Feiyang Sun
Cover page credits: Visax