Class 1: What is AI?

Principles of ML & Deep Learning - SAILea

Anisha Raghu and Ananya Raghu

Icebreakers

- Name
- Grade
- A fun fact about yourself :)
- How much do you know about AI? (it’s ok if you do not have much past experience)

Machine Learning Models

- Decision Trees
- Random Forest
- Linear & Logistic Regression
- K-nearest neighbors
- Etc…

What is AI?

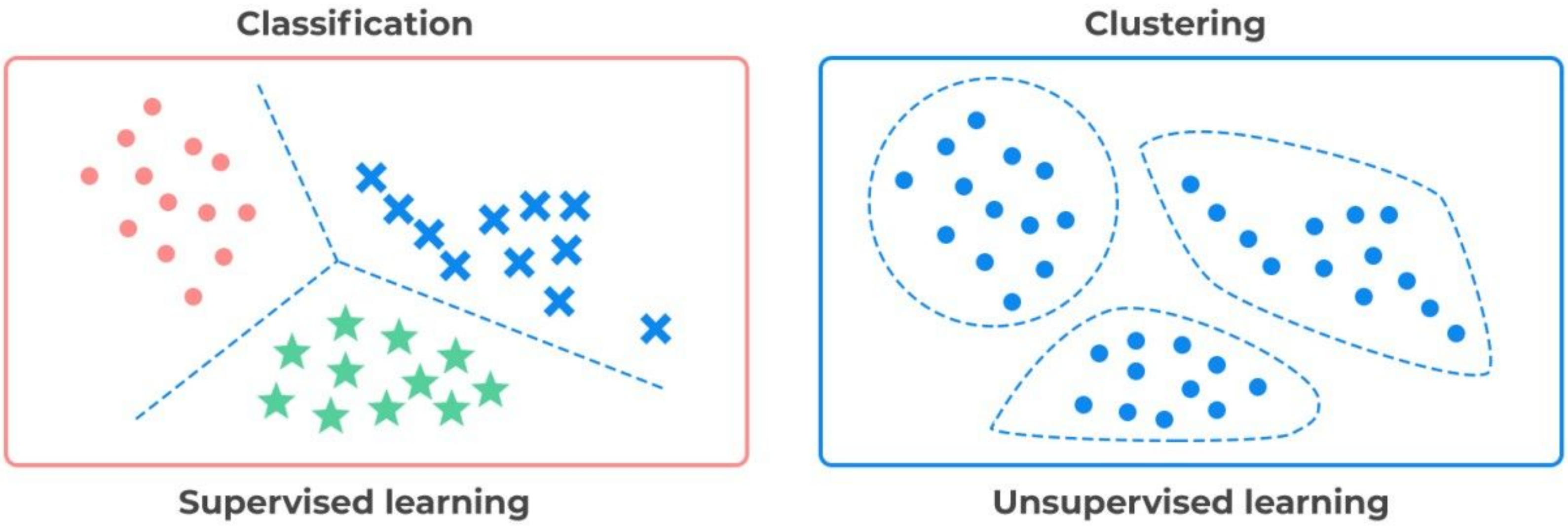
- Based on patterns
- Process large amounts of data and make predictions from it
- Traditional programming: does step by step commands - like a recipe using ingredients
- AI: Give computer various ingredients, and various outputs, and have it build or *learn* the recipe

Supervised, Unsupervised Learning, Reinforcement

- In simple terms, *supervised learning* is the process of using **labeled** examples to predict unlabeled examples

- *Unsupervised learning*: uses unlabeled data, discovers "structure" or underlying patterns in data

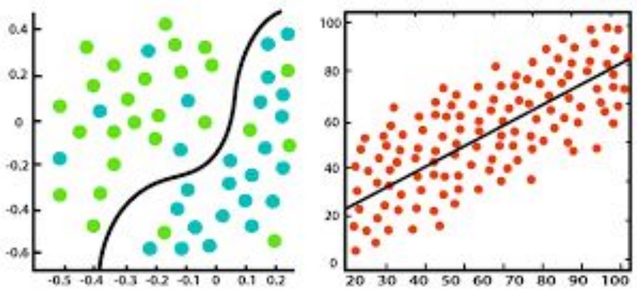
- *Machine Learning:* Process of learning to pick actions based on rewards and punishments from previous choices


Classification
Clustering
Supervised learning
Unsupervised learning

Regression vs Classification

- Regression helps predict CONTINUOUS values

- Classification simply classifies or predicts DISCRETE class labels



Classification

Regression

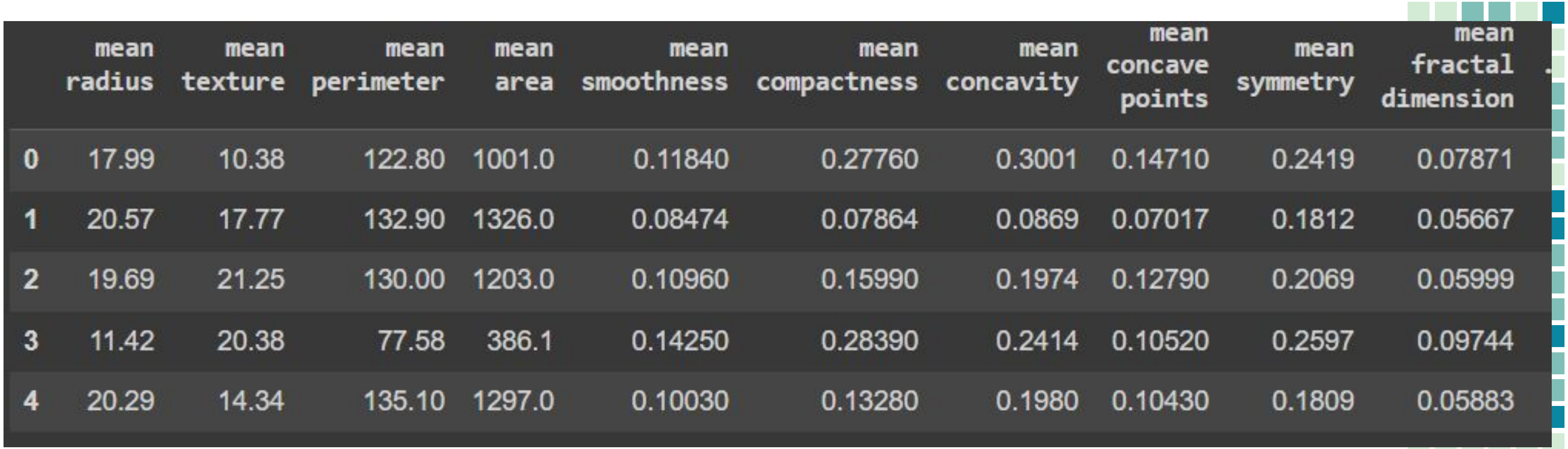
Source: https://www.javatpoint.com/regression-vs-classification-in-machine-learning

Practice question: I want to predict the temperature tomorrow in San Francisco. Would this be a classification or regression problem? What if I wanted to ask if it will rain or be sunny (assuming these are the only possible scenarios)?

Data

- While implementing machine learning, we generally split our data into three sets:
  - 1) **Training Data** - data used to fit the model
  - 2) **Validation Data** - Used to evaluate model performance at each step (epoch) of training. Note: model “sees” this, but does not “learn” directly from this
    - Example: I am designing a model - I would use validation dataset to decide hyperparameters of the model ⟶ for example, how many layers should be included?
  - 3) **Testing Data** - Used after training has completed to observe final performance of model
  -

Example Dataset for ML



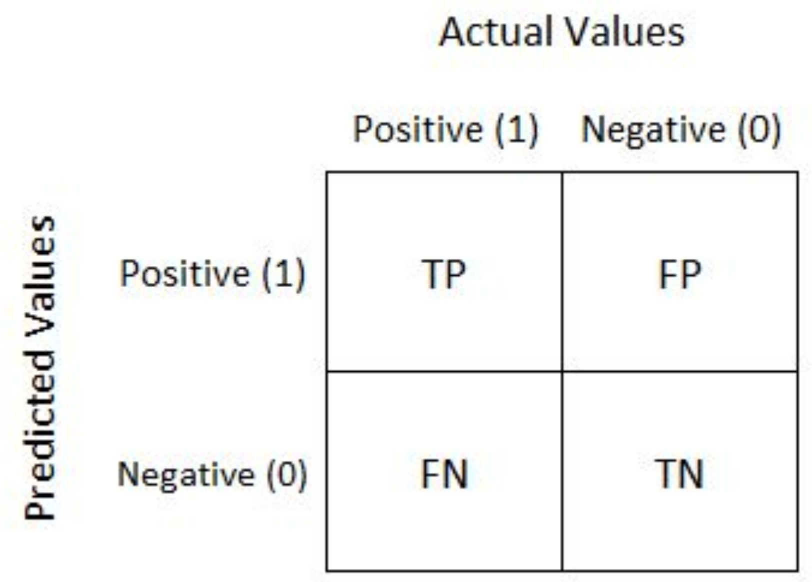
	mean radius	mean texture	mean perimeter	mean area	mean smoothness	mean compactness	mean concavity	mean concave points	mean symmetry	mean fractal dimension
0	17.99	10.38	122.80	1001.0	0.11840	0.27760	0.3001	0.14710	0.2419	0.07871
1	20.57	17.77	132.90	1326.0	0.08474	0.07864	0.0869	0.07017	0.1812	0.05667
2	19.69	21.25	130.00	1203.0	0.10960	0.15990	0.1974	0.12790	0.2069	0.05999
3	11.42	20.38	77.58	386.1	0.14250	0.28390	0.2414	0.10520	0.2597	0.09744
4	20.29	14.34	135.10	1297.0	0.10030	0.13280	0.1980	0.10430	0.1809	0.05883

Each row can be thought of as a point in n-dimensional space, where n is the number of features!

Metrics

- After training our model, how can we evaluate its performance?
- Basic metric: **accuracy**
  - (number of correct decisions)/total decisions

Continued – Confusion Matrices



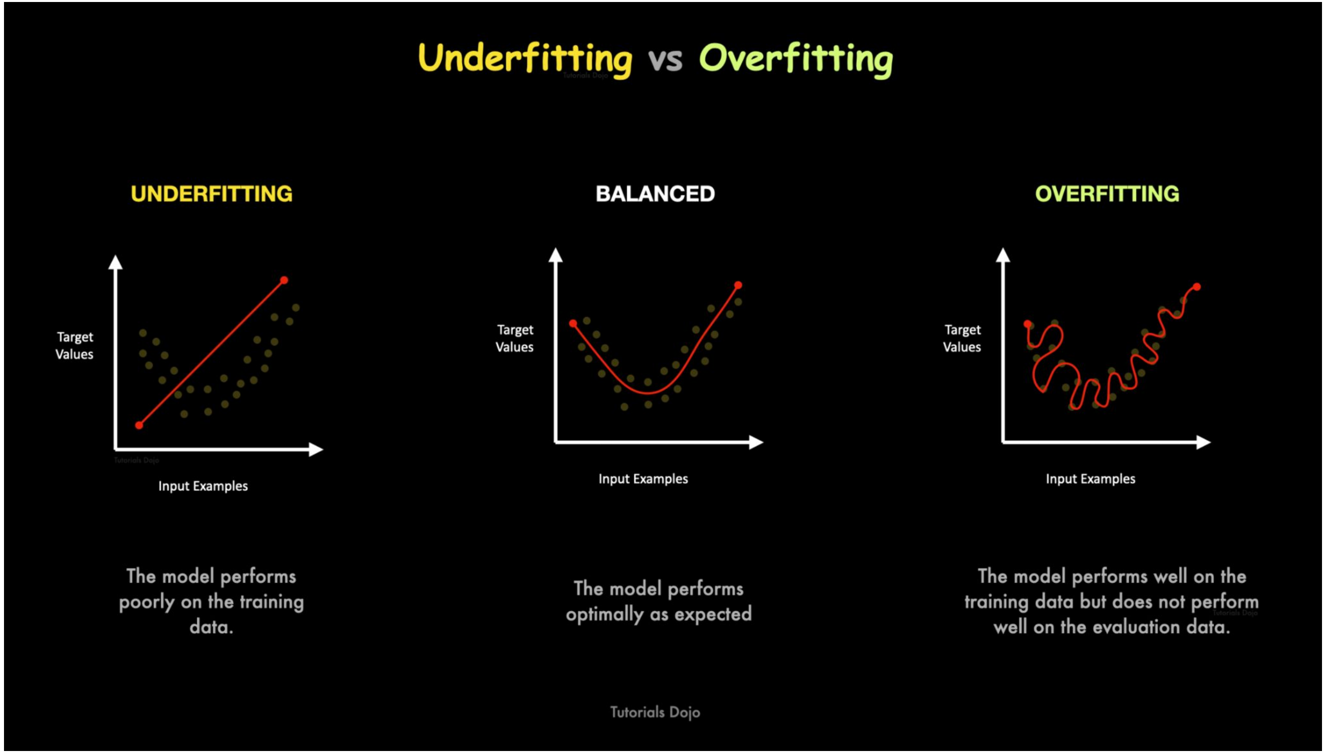
		Actual Values	
		Positive (1)	Negative (0)
Predicted Values	Positive (1)	TP	FP
	Negative (0)	FN	TN

Source of Image: https://towardsdatascience.com/understanding-confusion-matrix-a9ad42dcfd62

- Way to visualize the performance of an algorithm

- Summarizes performance of a classification algorithm

Overfitting versus Underfitting



Underfitting vs Overfitting

UNDERFITTING

The model performs poorly on the training data.

BALANCED

The model performs optimally as expected

OVERFITTING

The model performs well on the training data but does not perform well on the evaluation data.

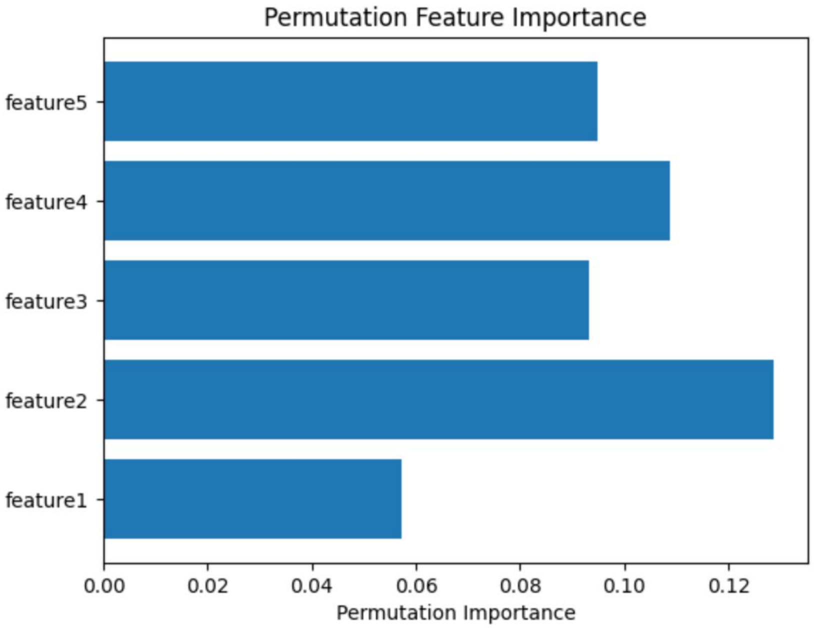
Tutorials Dojo

Source: https://www.linkedin.com/pulse/underfitting-vs-overfitting-simplified-jon-bonso

Feature Importance

- Indicates how much each feature contributes to the overall prediction

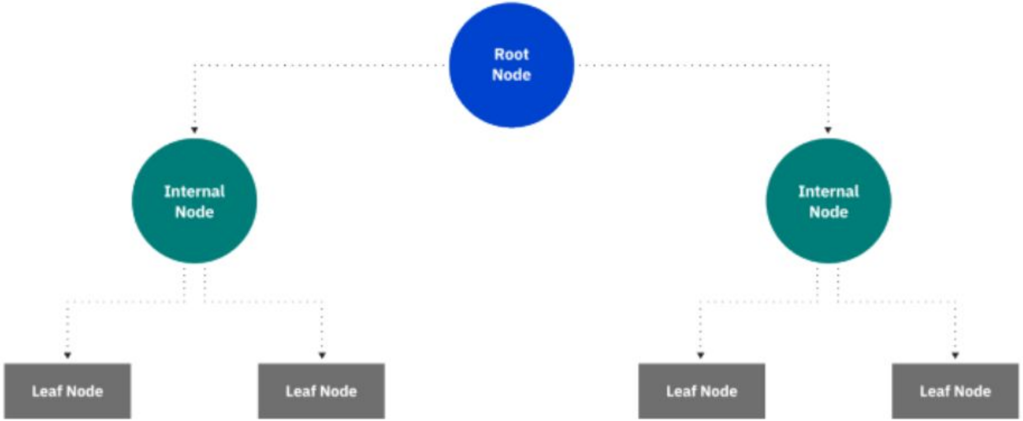
- **Permutation Feature Importance:** Permute one feature in the dataset and see how it affects the prediction.



Source:https://towardsdatascience.com/feature-importance-in-machine-learning-explained-443e35b1b284

Decision Tree

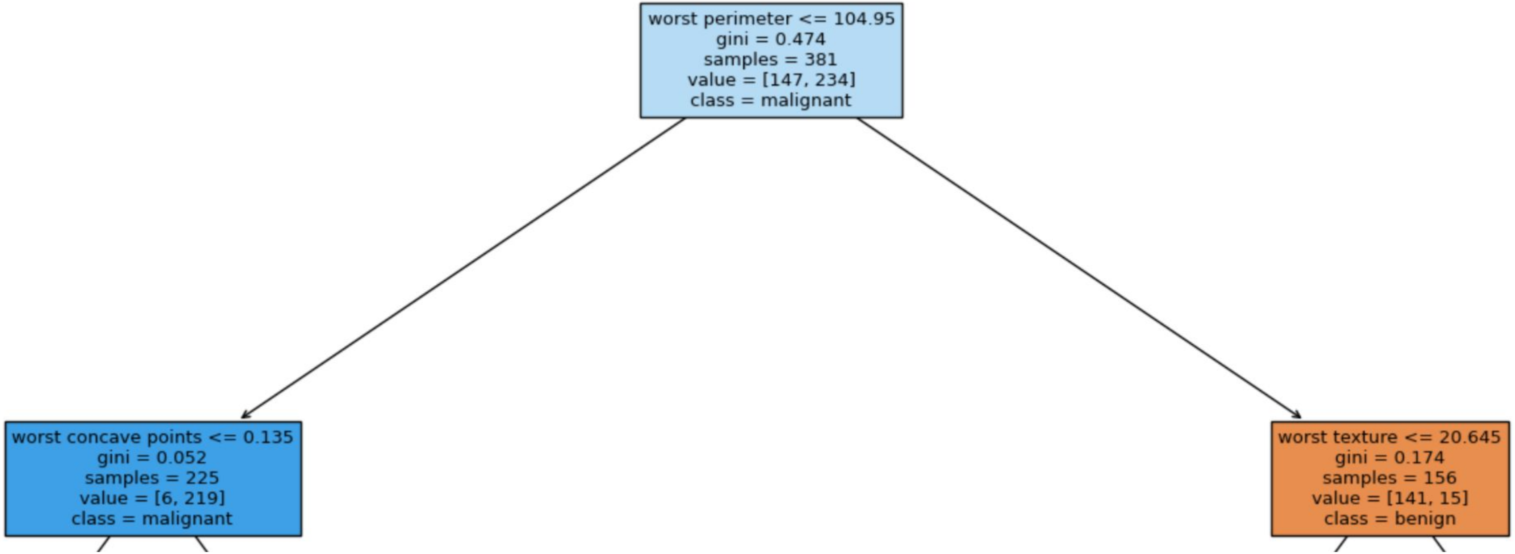
- A decision tree is a supervised learning algorithm
- Used for both classification and regression tasks.
- Tree structure, which consists of a root node, branches, internal nodes and leaf nodes.



Decision Tree Explained

- How does a decision tree decide what is the condition at each node?
- A commonly used method is via the **gini** calculation, which measures “impurity” in a distribution:
  - $Gini = 1 - \sum_{i=1}^{n}(p_i)^2$ Where p_i is the probability of an object being classified to a particular class

Example: Gini Calculation



worst perimeter <= 104.95
gini = 0.474
samples = 381
value = [147, 234]
class = malignant

worst concave points <= 0.135
gini = 0.052
samples = 225
value = [6, 219]
class = malignant

worst texture <= 20.645
gini = 0.174
samples = 156
value = [141, 15]
class = benign

In the second layer, the gini value for the blue box is 0.052 because it is 1 - (6/225)^2 - (219/225)^2. This can be thought of as a collection of 6 blue marbles, 219 red marbles, which is close to being very pure.

How do Decision Trees use the Gini Index?

- First, the decision tree will choose a split function.
- Based on split function, finds probability of going to left node or going to right node (P_R , P_L)
- Then, can compute gini for left node and gini for right node
- The decision will try to pick the split function that minimizes P_L*$Gini_L$ + P_R*$Gini_R$

Any questions?