Trustworthy Aritificial Intelligence: Fundamentals of AI

Presented by

Dr. Narinder Singh Punn,

Assistant Professor,

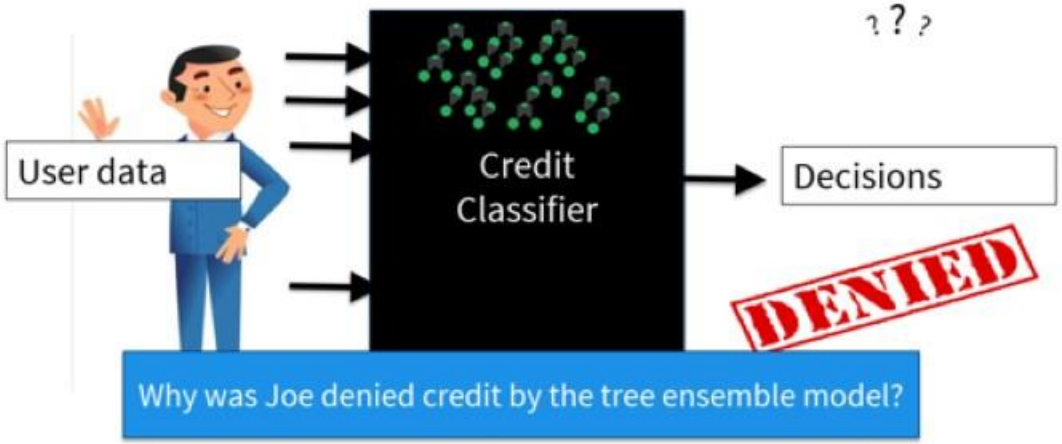
Dept. of CSE,

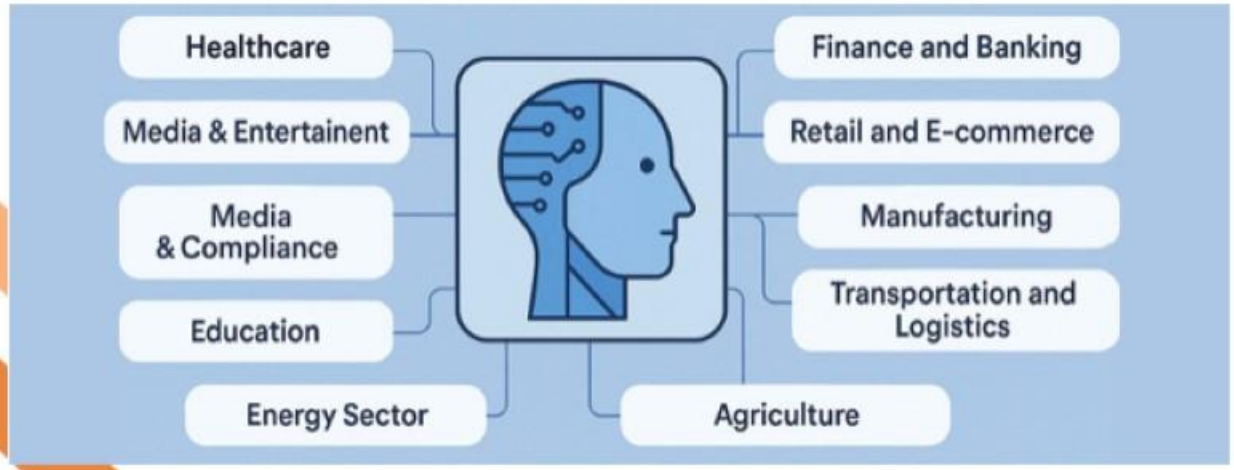
ABV-IIITM Gwalior

विश्वजीवनामृतं ज्ञानम्

Artificial Intelligence (AI)

- AI Systems are Ubiquitous





Source: "From Explainability to Model Quality and Back Again", Thirty-Fifth AAAI Conference on Artificial Intelligence

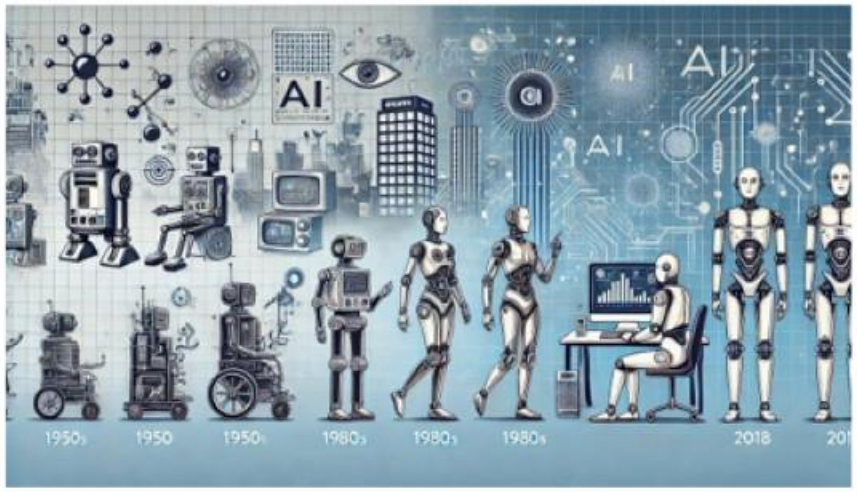
What is AI?

- AI (Artificial Intelligence) refers to the simulation of human intelligence in machines.
- It involves developing systems that can perform tasks that typically require human intelligence, such as:
  - Visual perception
  - Speech recognition
  - Decision-making
  - Language translation

How its is different from Traditional Programming?

Feature	Simple Programming (Traditional)	Artificial Intelligence (AI)
Problem Solving	Explicit, rule-based instructions	Learns patterns from data, often probabilistic
Adaptability	Requires manual updates for new conditions	Adapts and improves automatically with new data
Decision Making	Deterministic (fixed outcomes)	Probabilistic (predictions based on likelihood)
Complexity	Best for well-defined, rule-based problems	Excels at complex, ambiguous, and dynamic problems
Human Effort	High effort in defining all rules	High effort in data collection and model training, less in explicit rules
Transparency	Logic is usually clear and explainable	Can be a "black box," difficult to explain decisions

Evolution of AI



1642 First mechanical calculating machine built by French mathematician and inventor Blaise Pascal.

1837 First design for a programmable machine, by Charles Babbage and Ada Lovelace.

1943 Foundations of neural networks established by Warren McCulloch and Walter Pitts, drawing parallels between the brain and computing machines.

1950 Alan Turing introduces a test—the Turing test—as a way of testing a machine's intelligence.

1955 'Artificial intelligence' is coined during a conference devoted to the topic.

1965 ELIZA, a natural language program, is created. ELIZA handles dialogue on any topic; similar in concept to today's chatbots.

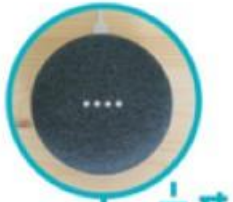
1980s

1997 Computer program Deep Blue beats world chess champion Garry Kasparov.

2002 iRobot launches Roomba, an autonomous vacuum cleaner that avoids obstacles.

2009 Google builds the first self-driving car to handle urban conditions.

2011 IBM's Watson defeats champions of US game show Jeopardy!



2011–2014 Personal assistants like Siri, Google Now, Cortana use speech recognition to answer questions and perform simple tasks.

2014 Ian Goodfellow comes up with Generative Adversarial Networks (GAN).

2016 AlphaGo beats professional Go player Lee Sedol 4-1.

2018 Most universities have courses in Artificial Intelligence.

source: https://www.mohdmohana.com/post/the-history-of-artificial-intelligence/

Evolution of AI

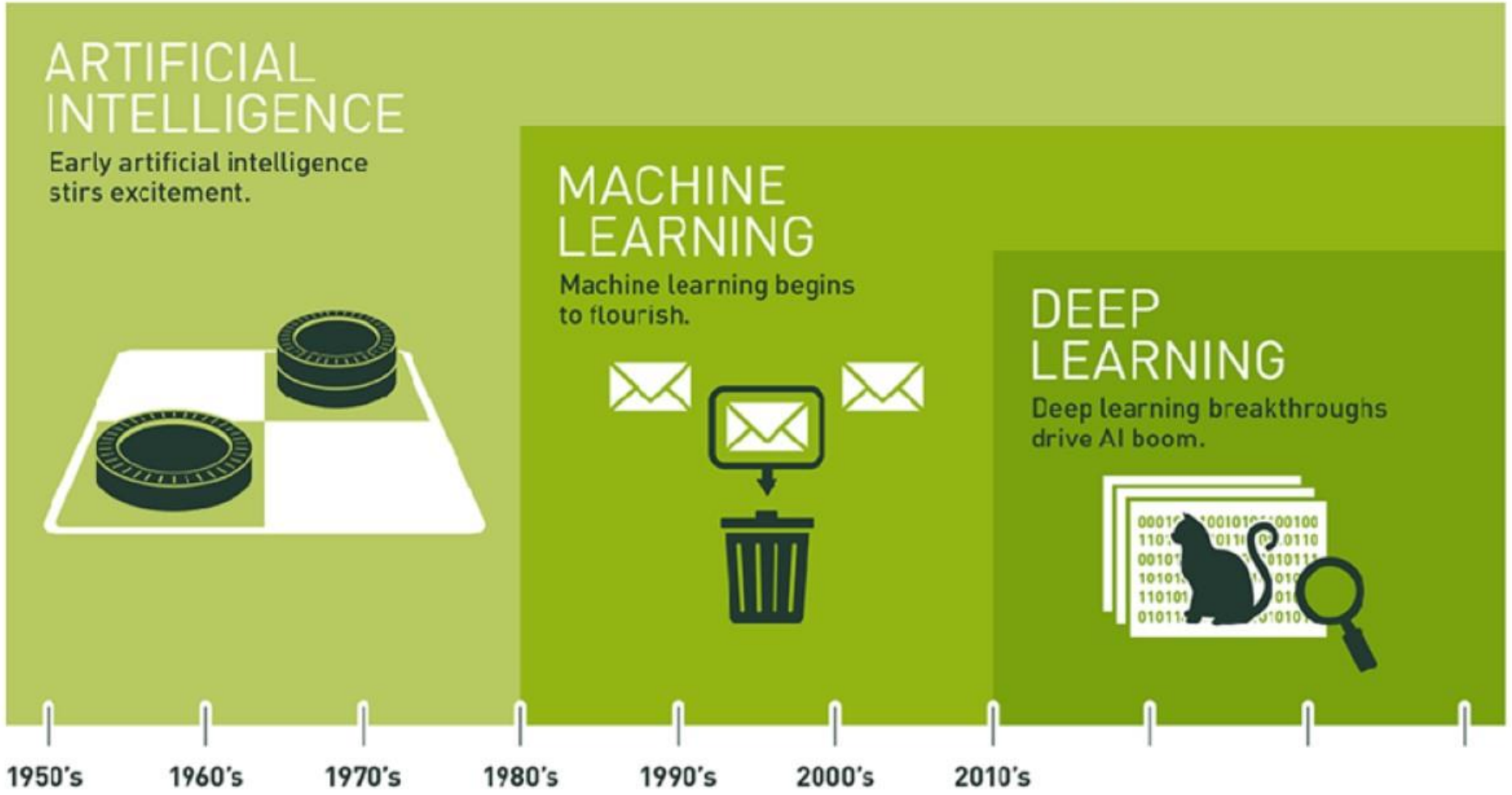


Image from: https://blogs.nvidia.com/blog/2016/07/29/whats-difference-artificial-intelligence-machine-learning-deep-learning-ai/

Learning and Adaptation

- Learning refers to the process by which a model or algorithm acquires knowledge from data to improve its ability to recognize patterns. This knowledge is typically represented in the form of:
  - *Model Parameters:* Numerical values that define the model's structure and behavior (e.g., weights in a neural network or coefficients in a logistic regression model).
  - *Decision Boundaries:* The dividing lines or surfaces in the feature space that separate different pattern classes.
- The learning process involves adjusting these parameters and decision boundaries based on the data, so that the model can make accurate predictions on new, unseen data.

Learning and Adaptation

- Two Main Types of Learning
  - *Supervised Learning:* The model learns from labeled examples, where the correct output (class label) is provided for each input data point. The goal is to find a function that maps input features to output labels.
  - *Unsupervised Learning:* The model learns from unlabeled data, where no output labels are provided. The goal is to discover hidden patterns or groupings within the data.
- Others: Self-supervised, Weakly supervised, Reinforcement learning, etc.

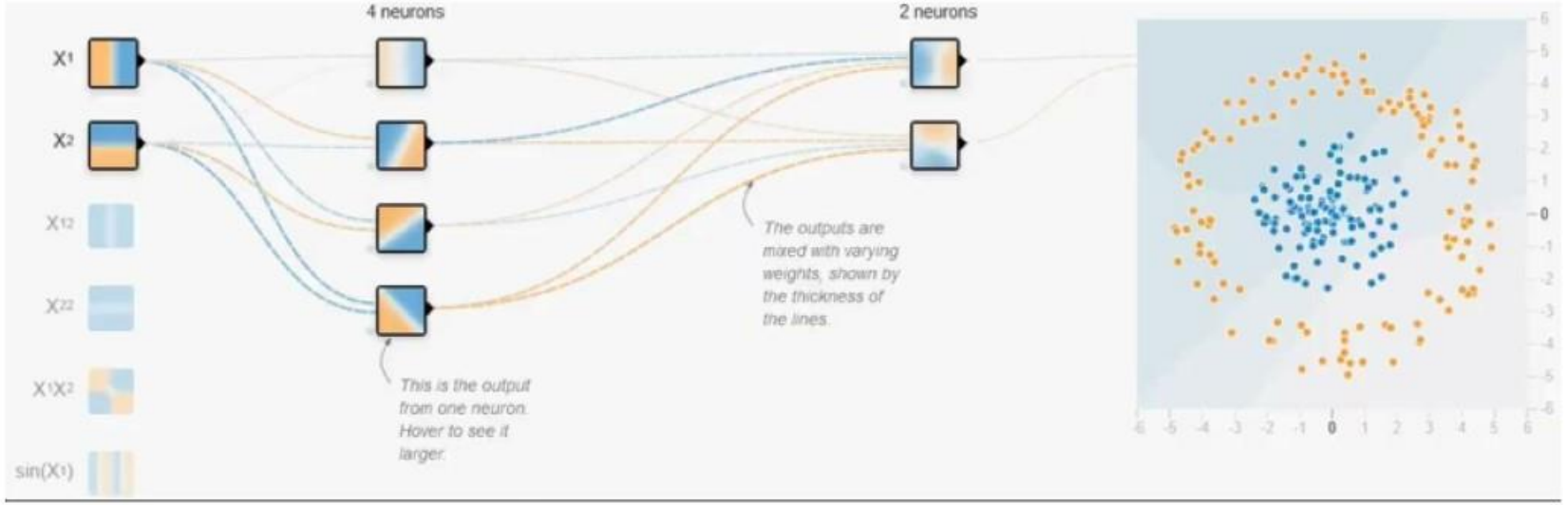
Learning and Adaptation

- Adaptation refers to the ability of a model or algorithm to adjust its behavior in response to changes in the environment or the data.
- Strategies:
  - *Online Learning:* The model is continuously updated with new data as it becomes available, allowing it to adapt to changing patterns.
  - *Transfer Learning:* Knowledge learned from one task or domain is transferred to a new task or domain, which can speed up learning and improve performance.
  - *Active Learning:* The model actively selects the most informative data points to learn from, which can reduce the amount of labeled data needed for training.

Decision Boundary

- A decision boundary is a dividing line or surface in the feature space that separates different classes in a classification problem.
- It represents the critical threshold where a model changes its prediction from one class to another.
- Points falling on one side of the boundary are assigned to one class, while points on the other side are assigned to a different class.

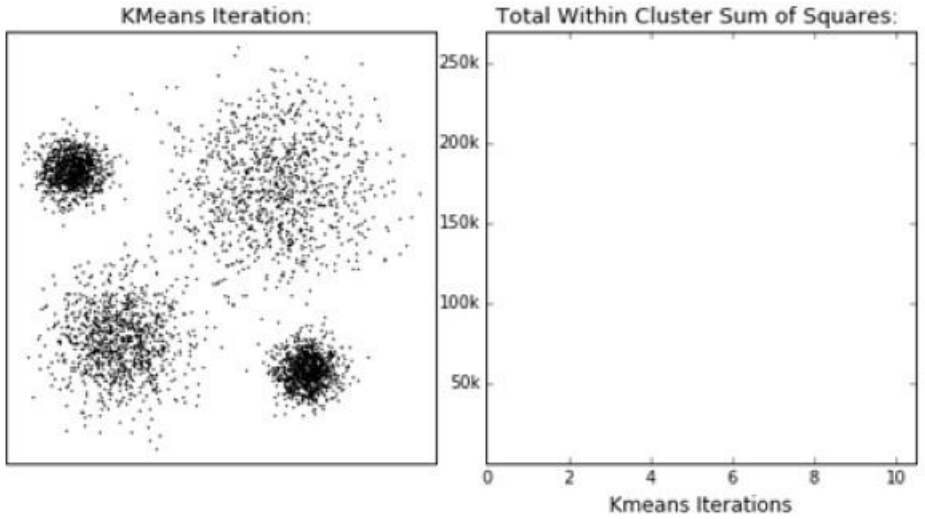
Supervised Learning



Decision Boundary

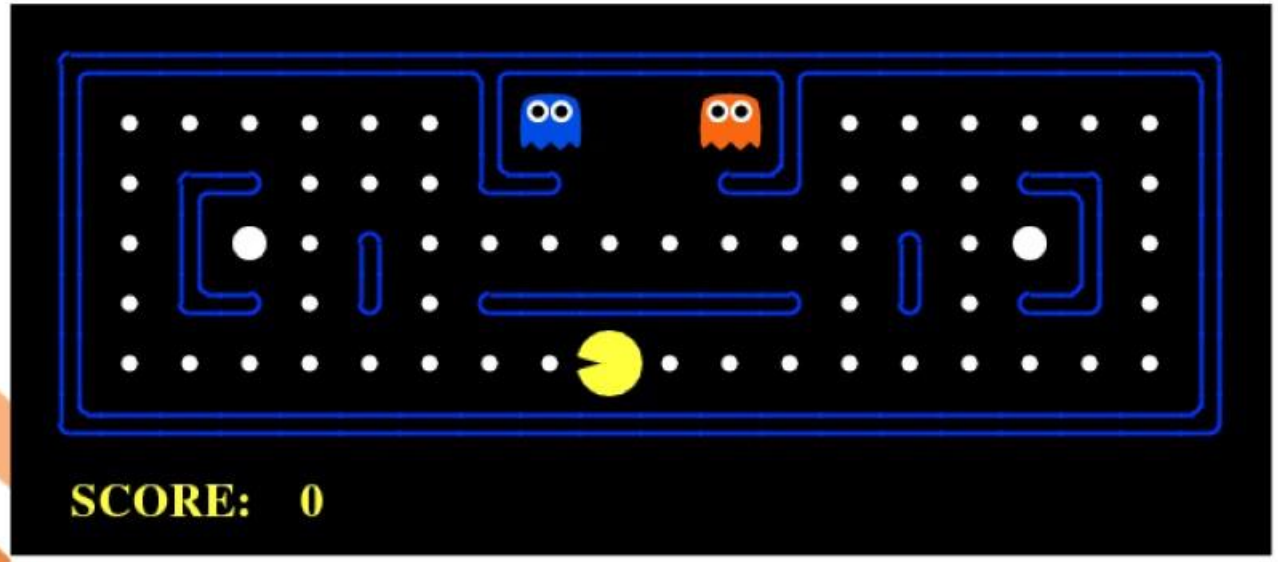
- A decision boundary is a dividing line or surface in the feature space that separates different classes in a classification problem.
- It represents the critical threshold where a model changes its prediction from one class to another.
- Points falling on one side of the boundary are assigned to one class, while points on the other side are assigned to a different class.

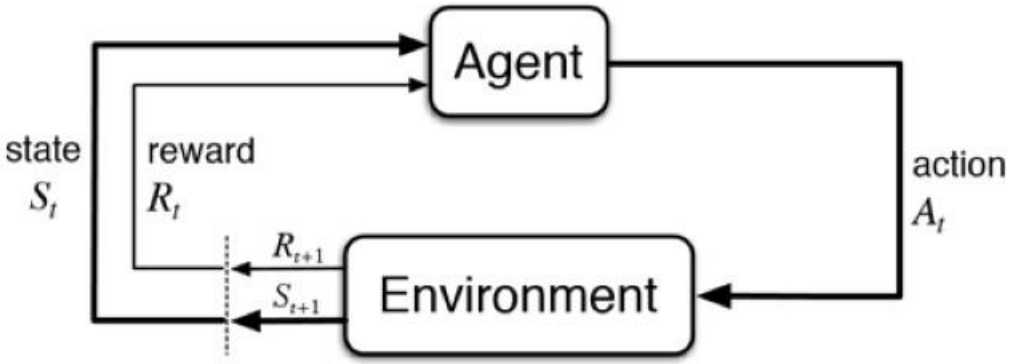
Unsupervised learning



Decision Boundary

- Reinforcement learning: An agent learns to make decisions by interacting with an environment. The agent receives feedback in the form of rewards or penalties, and it aims to maximize its cumulative reward over time by choosing the best actions in each state of the environment.





Decision Boundary

- Binary-Class Classification:
  - There are only two classes (e.g., "spam" vs. "not spam," "positive" vs. "negative").
  - The decision boundary can be a simple line (for linearly separable data) or a more complex curve or surface (for non-linearly separable data).
  - Examples of models with linear decision boundaries:
    - Logistic regression
    - Linear support vector machines (SVMs)
  - Examples of models with non-linear decision boundaries:
    - Non-linear SVMs
    - Decision trees
    - Neural networks

Decision Boundary

- Multi-Class Classification:
  - There are more than two classes (e.g., classifying images into "dog," "cat," "bird," etc.).
  - The decision boundaries become more complex, often involving multiple lines or surfaces dividing the feature space into different regions.
  - There are two main approaches to multi-class classification:
    - One-vs-All (OvA): Train a separate binary classifier for each class, where one class is considered "positive" and the rest are considered "negative."
    - One-vs-One (OvO): Train a binary classifier for every pair of classes.
    - Use probabilistic approach

Likelihood Function

- Likelihood is a measure of how well a particular statistical model explains observed data.
  - Model: A simplified representation of a real-world phenomenon. It often involves parameters that control its behavior.
  - Data: The observations or measurements we collect from the real world.
  - Likelihood Function: A function that takes the model's parameters as input and outputs a number representing how well those parameters explain the observed data. Higher likelihood means the model is a better fit to the data.

MLE

- To determine the values for the parameters in θ, we maximize the probability of generating the observed samples.

Definition 1: (Likelihood function and MLE)
Given n data points $x_1, x_2, \ldots, x_n$ we can define the likelihood of the data given the model θ (usually a collection of parameters) as follows.

$$\hat{P}(\text{dataset} \mid \theta) = \prod_{k=1}^{n} \hat{P}(x_k \mid \theta). \tag{1.1}$$

The maximum likelihood estimate (MLE) of θ is

$$\hat{\theta}_{MLE} = \arg\max_{\theta} \hat{P}(\text{dataset} \mid \theta). \tag{1.2}$$

Coin Flip Example

- Let $\theta = \{P(\text{Head}) = q\}$ be the parameter for a coin flip model. The Maximum Likelihood Estimate (MLE) for θ is $\hat{\theta}$, given by:

$$\hat{q} = \frac{\text{Number of heads}}{\text{Total number of flips}}$$

Coin Flip Example

Assume we observe n samples $x_1, x_2, \ldots, x_n$ with n_1 heads and n_2 tails.
Then, the likelihood function is

$$P(D \mid \theta) = \prod_{i=1}^{n} P(x_i \mid \theta) = q^{n_1}(1-q)^{n_2}.$$

We now find $\hat{q}$ that maximizes this likelihood function, i.e., $\hat{q} = \arg\max_q q^{n_1}(1-q)^{n_2}$.

To do so, we set the derivative to 0:

$$0 = \frac{\partial}{\partial q} q^{n_1}(1-q)^{n_2} = n_1 q^{n_1-1}(1-q)^{n_2} - q^{n_1} n_2 (1-q)^{n_2-1},$$

which is equivalent to

$$q^{n_1-1}(1-q)^{n_2-1}(n_1(1-q) - q n_2) = 0,$$

which yields

$$n_1(1-q) - q n_2 = 0 \Leftrightarrow \boxed{q = \frac{n_1}{n_1 + n_2}}$$

Log Likelihood

- When working with products, probabilities of entire datasets often get too small. A possible solution is to use the log of probabilities, often termed log likelihood.

$$\log \hat{P}(\text{dataset} \mid M) = \log \prod_{k=1}^{n} \hat{P}(x_k \mid M) = \sum_{k=1}^{n} \log \hat{P}(x_k \mid M).$$

MLE Algorithm

Given n data points $x_1, x_2, \ldots, x_n$ and a model θ represented by an expression for $P(X \mid \theta)$, perform the following steps:

1. Compute the log-likelihood

$$L = \log \prod_{i=1}^{n} P(x_i \mid \theta) = \sum_{i=1}^{n} \log P(x_i \mid \theta).$$

2. For each parameter γ in θ, find the solution(s) to the equation $\frac{\partial L}{\partial \gamma} = 0$.

3. The solution $\hat{\gamma}$ that satisfies $\frac{\partial^2 L}{\partial \hat{\gamma}^2} \leq 0$ is the MLE of γ.

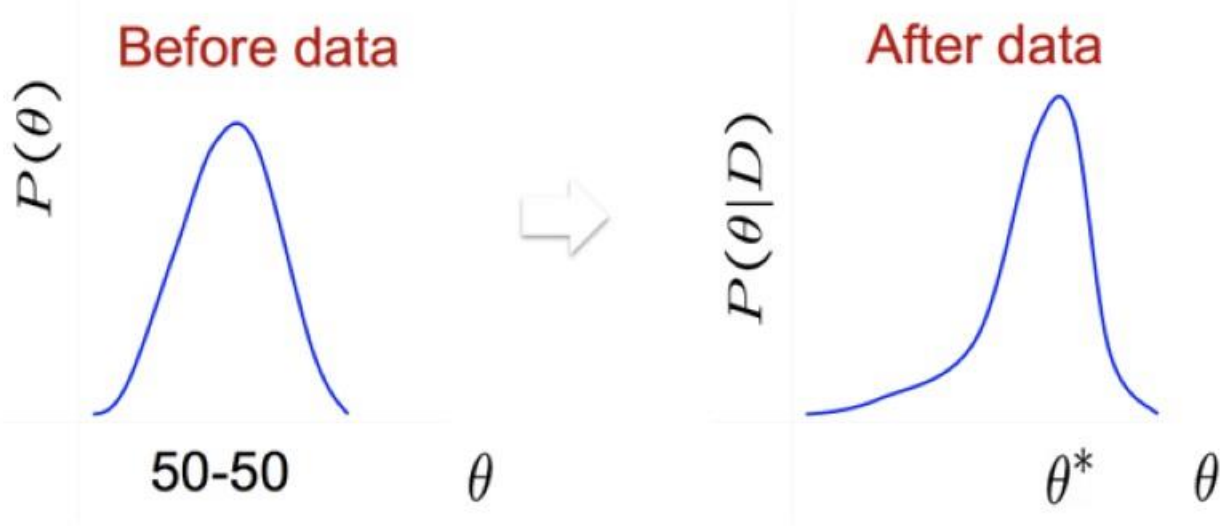
Bayesian Classifiers

- Bayesian classifiers are the statistical classifiers based on Bayes' Theorem.
- Bayesian classifiers can predict class membership probabilities i.e. the probability that a given tuple belongs to a particular class.
- It uses the given values to train a model and then it uses this model to classify new data.

Bayesian Classifiers

Bayesian Classifiers

- There are only two events possible for the given question:
  - A: It is going to rain tomorrow
  - B: It will not rain tomorrow.
- If you think intuitively
  - It's either going to be raining today or it is NOt going to be raining today
  - So technically there is 50% CHANCE OF RAIN tomorrow. Correct?



Bayesian Classifiers

- *Bayesian theorem argues that the probability of an event taking place changes if there is information available about a related event.*

- This means that if you recall the previous weather conditions for the last week, and you remember that it has actually rained every single day, your answer will no longer be 50%.
- The Bayesian approach provides a way of explaining how you should change your existing beliefs in the light of new evidence.
- Bayesian rule's emphasis on prior probability makes it better suited to be applied in a wide range of scenarios.

Bayes Formula

$$\underbrace{P(\theta|x)}_{\text{Posterior}} = \frac{\overbrace{P(x|\theta)}^{\text{Likelihood}} \times \overbrace{P(\theta)}^{\text{Prior}}}{\underbrace{P(x)}_{\text{Marginal}}}$$

Posterior $P(\theta|x)$: The updated probability of the parameter θ after observing the data x.

Likelihood $P(x|\theta)$: The probability of observing the data x given a particular value of the parameter θ.

Prior $P(\theta)$: The initial probability of the parameter θ before observing any data.

Marginal $P(x)$: The total probability of observing the data x, averaged over all possible values of θ.

Bayes Formula

Given a dataset and a model M with prior $P(M)$, the maximum a posteriori (MAP) estimate of M is

$$\hat{\theta}_{MAP} = \arg\max_{\theta} P(\theta \mid \text{dataset}) = \arg\max_{\theta} P(\text{dataset} \mid \theta) P(\theta). \tag{1.5}$$

- If we only have very few samples, MLE may not yield accurate results, so it is useful to take into account prior knowledge. When the number of samples gets large, the effect of prior knowledge will diminish.

MLE vs MAP

- Maximum Likelihood Estimation (MLE): This approach seeks to find the parameters (θ) that maximize the likelihood (P(x|θ)). It asks: *"What parameters make the observed data most probable?"* Most common loss functions, like Cross-Entropy in classification and Mean Squared Error in regression, are derived from maximizing the likelihood. MLE makes no assumption about the parameters beforehand.

- Maximum a Posteriori (MAP): This approach seeks to find the parameters (θ) that maximize the entire posterior probability. It asks: *"Given the data, what is the most probable set of parameters?"* MAP incorporates a prior (P(θ)), which is a belief about what the parameters should look like before seeing any data.

Bayes Decision Rule

- Classification is the task of predicting a (discrete) output label given the input data. The most optimal algorithm is called the Bayes decision rule.

If we know the conditional probability $P(x \mid y)$ and class prior $P(y)$, use (1.4) to compute

$$P(y = i \mid x) = \frac{P(x \mid y = i)P(y = i)}{P(x)} \propto P(x \mid y = i)P(y = i) = q_i(x) \tag{2.1}$$

and $q_i(x)$ to select the appropriate class. Choose class 0 if $q_0(x) > q_1(x)$ and 1 otherwise. In general choose the class $\hat{c} = \arg\max_c \{q_c(x)\}$.

Bayes Decision Rule

- Because our decision is probabilistic, there is still chance for error.
- The Bayes error rate (risk) of the data distribution is the probability an instance is misclassified by the Bayes decision rule.
- For binary classification, the risk for sample is

$$R(x) = \min\{P(y = 0 \mid x), P(y = 1 \mid x)\}. \tag{2.2}$$

Bayes Decision Rule

- Expected value of risk

$$
\begin{aligned}
E[r(x)] &= \int_x r(x)P(x)dx \\
&= \int_x \min\{P(y=0 \mid x), P(y=1 \mid x)\}dx \\
&= P(y=0)\int_{L_1} P(x \mid y=0)dx + P(y=1)\int_{L_0} P(x \mid y=1)dx,
\end{aligned}
$$

where L_i is the region over which the decision rule outputs label *i*.

Bayes Decision Rule

- The risk value we computed assumes that both errors (assigning instances of class 1 to 0 and vice versa) are equally harmful.
- In general, we can set the weight penalty $L_{i,j}(x)$ for assigning instances of class *i* to class *j*. This gives us the concept of a loss function

$$E[L] = L_{0,1}(x)P(y=0)\int_{L_1} P(x \mid y=0)dx + L_{1,0}(x)P(y=1)\int_{L_0} P(x \mid y=1)dx. \qquad (2.3)$$

THANK YOU