Mathematics For Intelligent Systems
5

Dr.Don.S and Dr. Sunder Ram K

ML Experts

- Grady Jensen – linear regression and linear classification https://argmax.ai/ml-course/
- Nando de Freitas - Deep understanding of ML
- Kilian Weinberger – Deep understanding of ML
- Steve Brunton – Control theory

<table>
<tr><th colspan="3">Internal</th><th>External</th><th>Total</th></tr>
<tr><td>Components</td><td colspan="2">Weightage</td><td rowspan="4">Project Based
3 evaluations
Weightage: 30%
(7.5,7.5,15)

1^{st} evaluation nov 30
7.5
2^{nd} evaluation Dec 20^{th}
7.5
3^{rd} evaluation Jan 10^{th}
15 : report 5 +demo 10

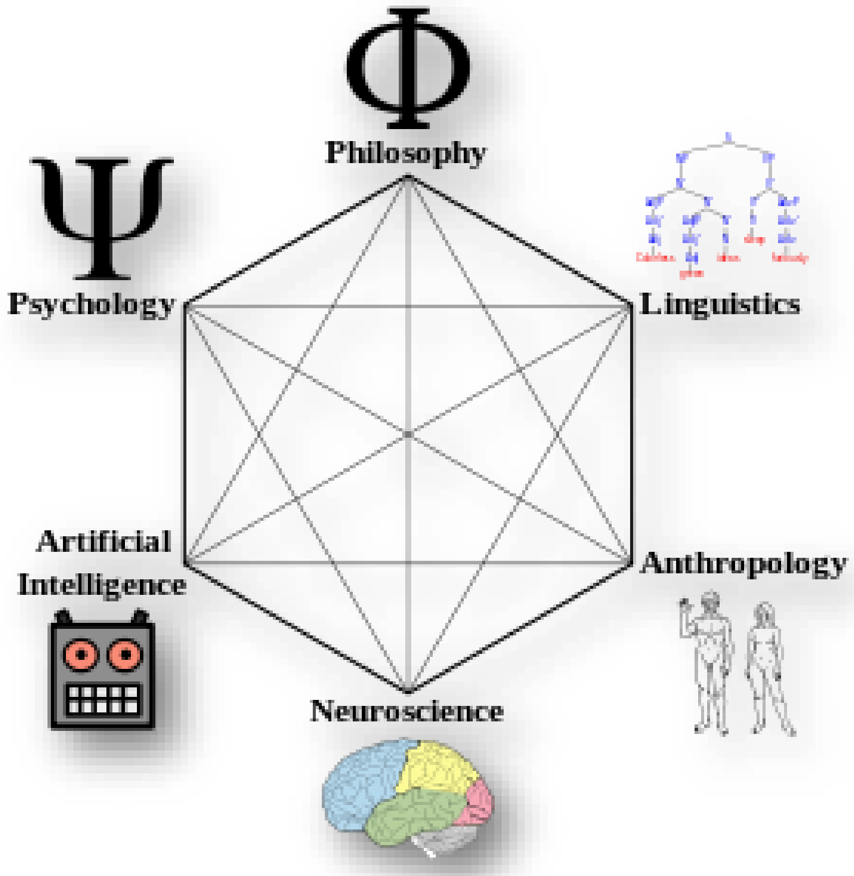
Negative mark for late submission
Max team size: 2</td><td rowspan="4">Internal + External=100</td></tr>
<tr><td>Assignments(2)
(3^{th} week of Nov ,
4^{rd} week of Dec)
Lab weekly evaluation - Summarizing research article, implementation, quiz based on research articles</td><td>30%</td><td rowspan="3">70%</td></tr>
<tr><td>Mid-Term</td><td>20%</td></tr>
<tr><td>Quiz (2)
(1^{st} week of Dec , -unit 2
1^{st} week of Jan) –unit 1 and 3</td><td>20%</td></tr>
</table>

Course Outcome

- CO1: Understand and implement basic concepts and techniques of probabilistic graphical models needed for causal reasoning in AI
- CO2: Apply the concepts of linear algebra, optimization and probability theory for controlling real-world systems
- CO3: Identify the connection between the concepts of linear algebra, differential equation and probability theory
- CO4: Understand and implement latest data-driven modelling of linear and non-linear dynamical systems through modern matrix/tensor decomposition techniques

Motivation

Cognitive Science – scientific study of the Human brain,**Understanding Intelligence**



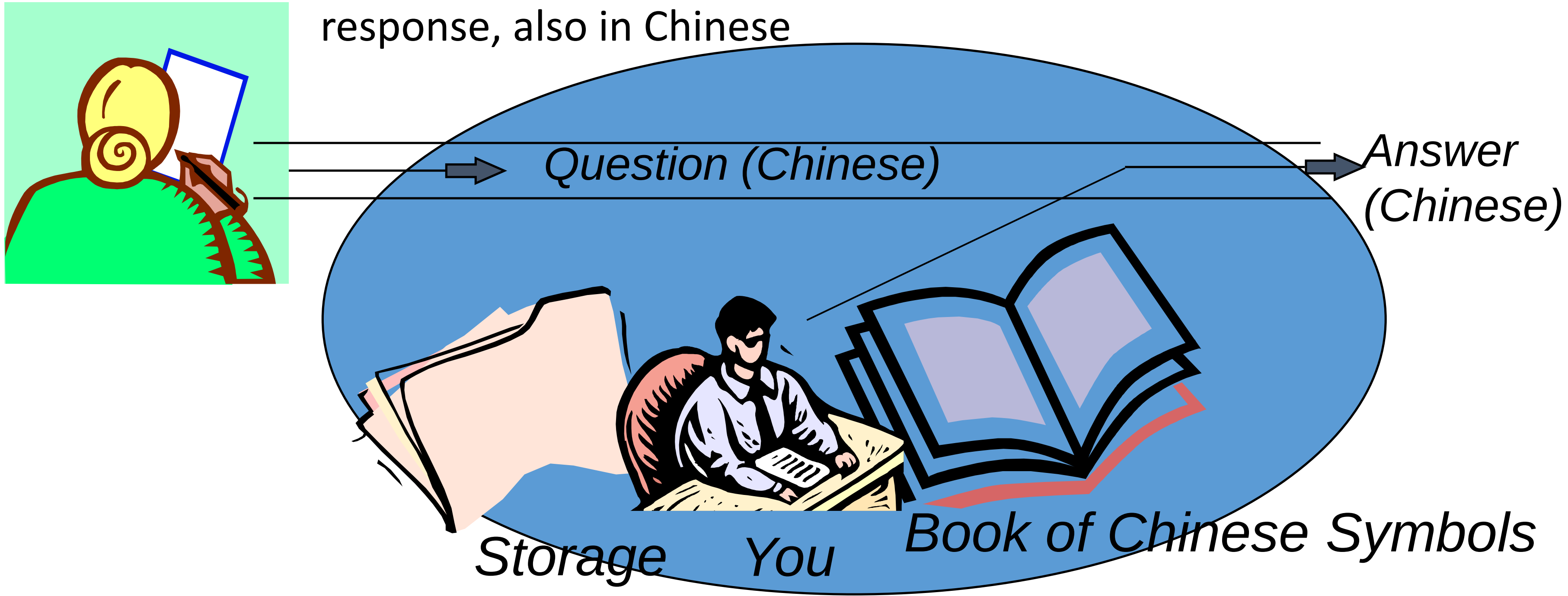
Testing "Intelligence" with the Turing Test

- 1950 – Alan Turing devised a test for intelligence called the Imitation Game
  - Ask questions of two entities, receive answers from both
  - If you can't tell which of the entities is human and which is a computer program, then you are fooled and we should therefore consider the computer to be intelligent

Which is the person?
Which is the computer?

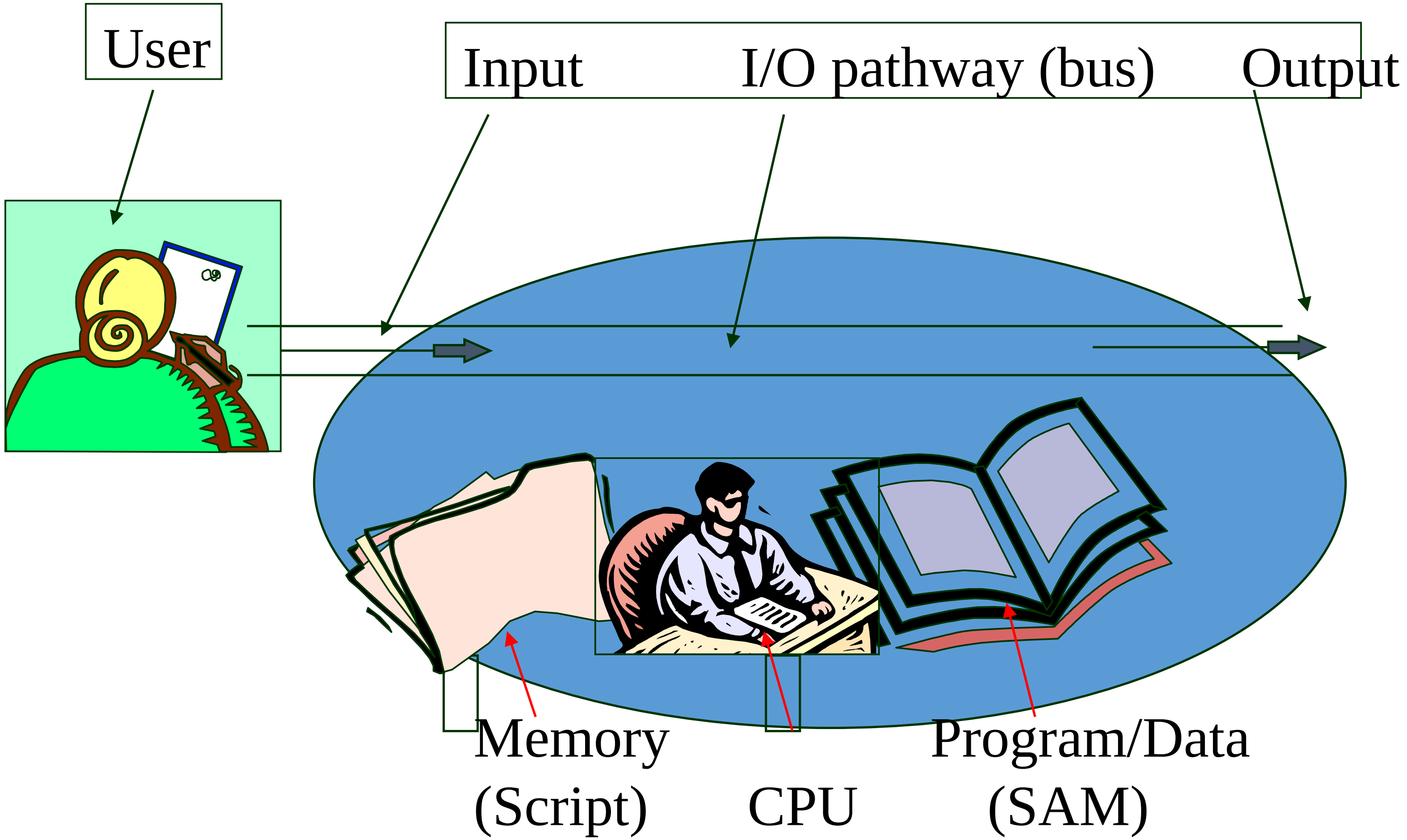
The Chinese Room Problem

- From John Searle, Philosopher, in an attempt to demonstrate that computers cannot be intelligent
  - The room consists of you, a book, a storage area (optional), and a mechanism for moving information to and from the room to the outside
    - a Chinese speaking individual provides a question for you in writing
    - you are able to find a matching set of symbols in the book (and storage) and write a response, also in Chinese



Chinese Room: An Analogy for a Computer

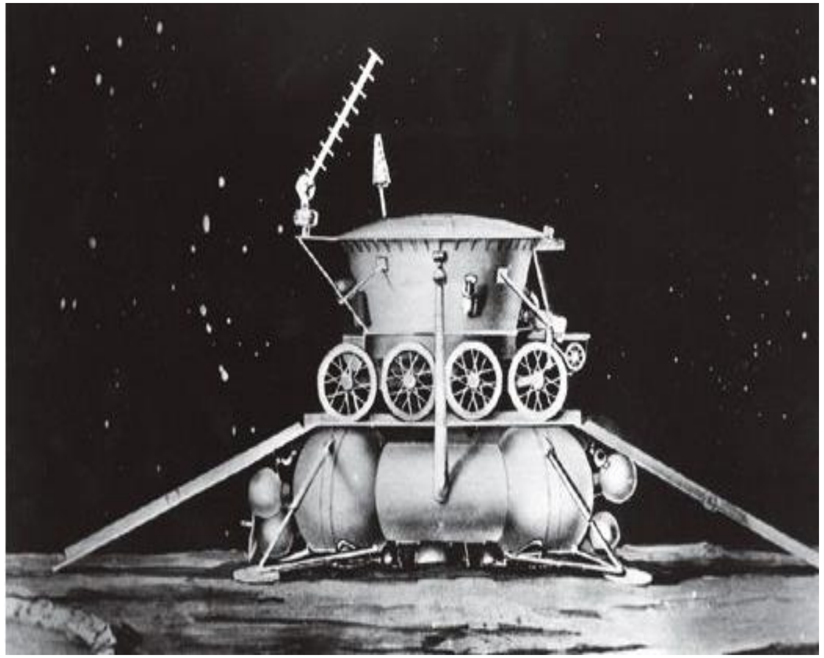
Note: Searle's original Chinese Room actually was based on a Script that was implemented in Chinese, our version is just a variation on the same theme



- You were able to solve the problem of communicating with the person/user and thus you/the room passes the Turing Test
- But did you understand the Chinese messages being communicated?
  - since you do not speak Chinese, you did not understand the symbols in the question, the answer, or the storage
  - can we say that you actually *used* any intelligence?
- By analogy, since you did not understand the symbols that you interacted with, neither does the computer understand the symbols that it interacts with (input, output, program code, data)
- Searle concludes that the computer is not intelligent, it has no "semantics," but instead is merely a symbol manipulating device
  - the computer operates solely on syntax, not semantics

What is Intelligent?

- "Intelligence denotes the **ability of an individual to adapt his thinking to new demands**; it is the common mental adaptability to new tasks and conditions of life" (William Stern, 1912)

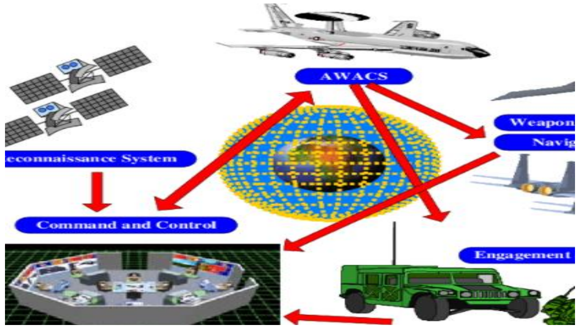


Intelligence must be able to perform

- perceive, reason and infer, solve problems, learn and adapt, apply common sense, apply analogy, recall, apply intuition, reach emotional states, achieve self-awareness

Application of Intelligent Systems

Industrial Automation



Military Applications

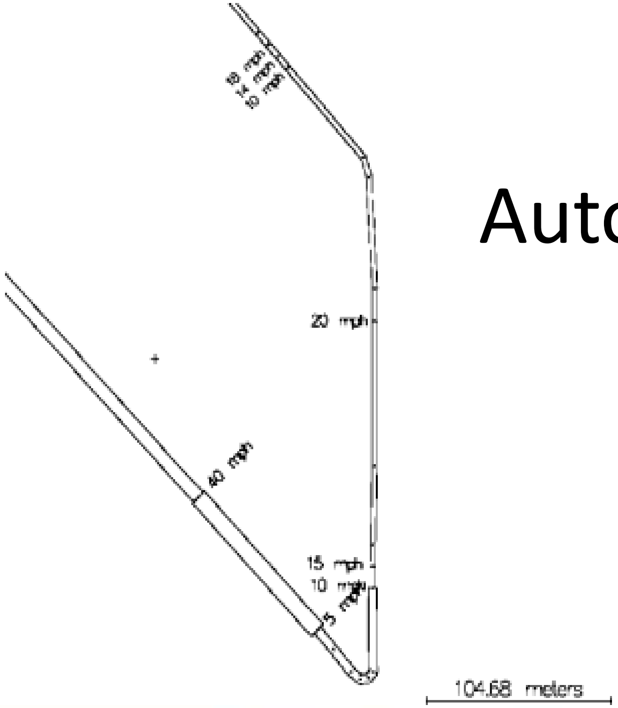
Clinical Applications

Challenges

- Uncertainty
- Dynamic World
- Time consuming computation
- Mapping
- ….

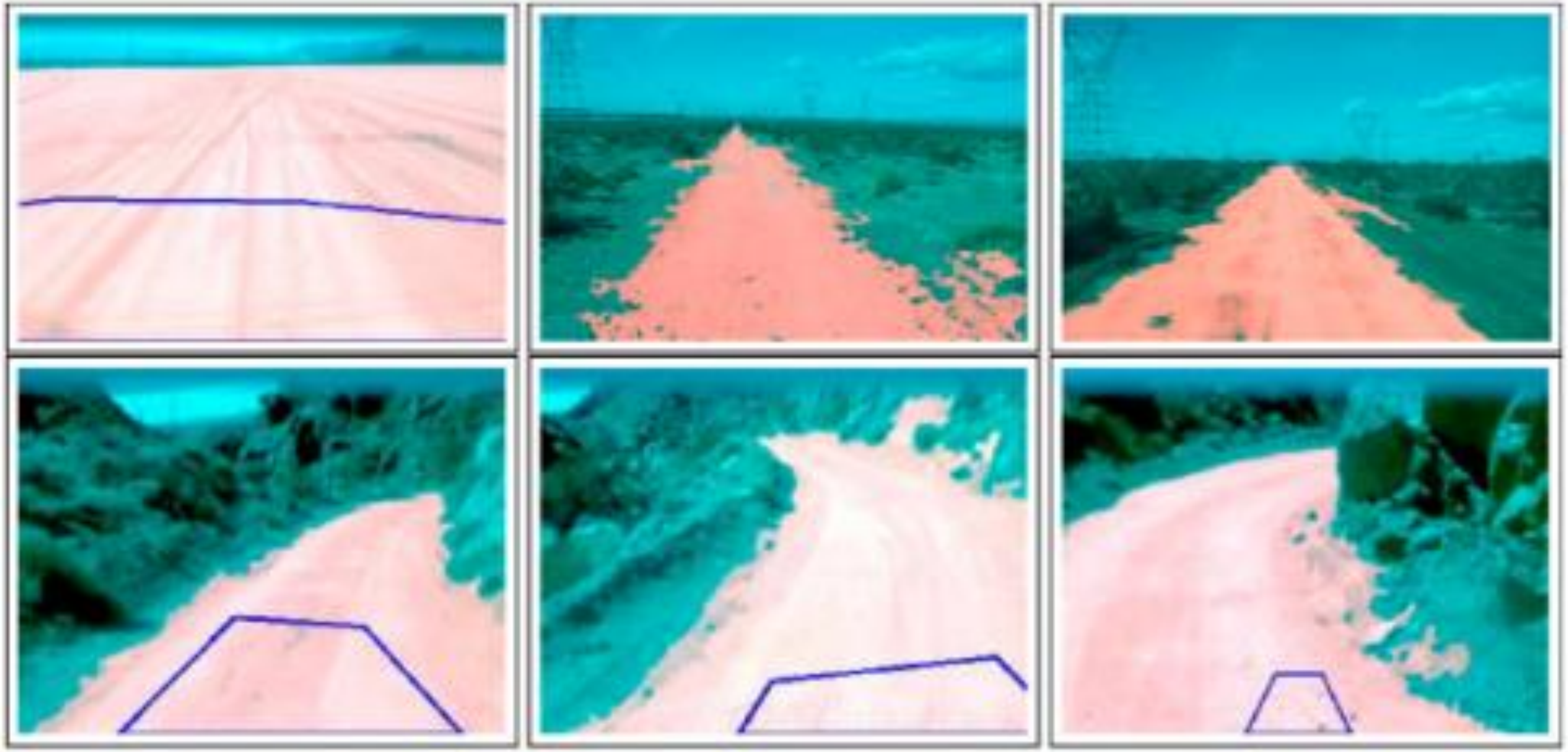
Why we study MIS5

(a)



Autonomous Driving

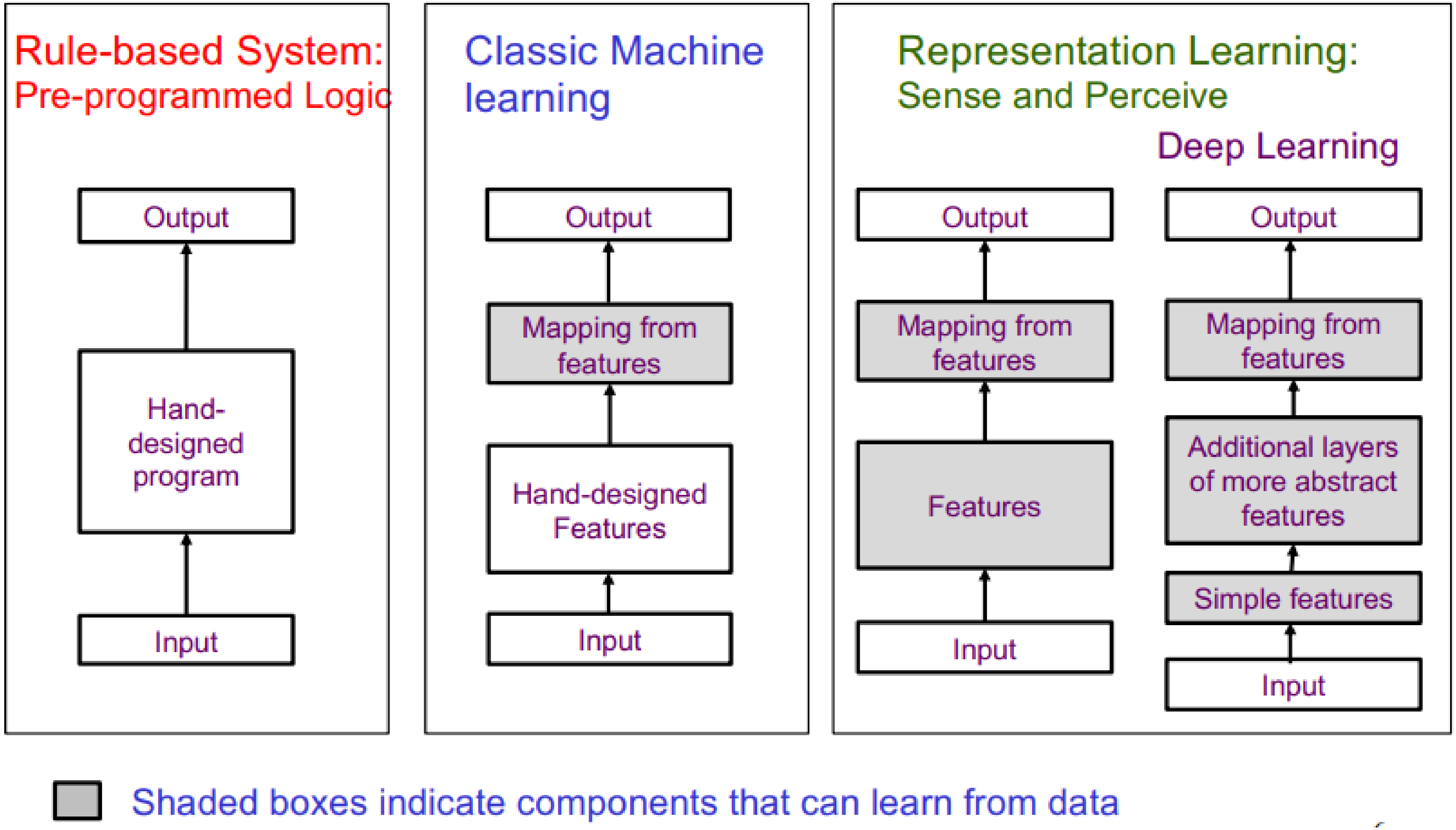
(a) (b) (c)



- Module 1 Dr. Don.S
  - Data Driven Dynamical Systems: Motivation and Challenges, Dynamic Mode decomposition, Sparse identification of Non-linear Dynamics.
- Module 2 Dr. Sunder Ram K
  - Probability theory, Bayesian Networks (BNs), Representation Learning in Bayesian Networks, Markov Random Fields- MRF, Inference, Message Passing, Learning in Markov Networks, Numerical Optimization, MRFs and BNs Monte Carlo Method.
- Module 3 Dr.Don.S
  - Linear Control Theory: Closed loop Feedback Control, LTI, Controllability and Observability, Optimal Full State Control, Optimal Full-State Estimation, The Kalman Filter.

Traditional Computer System Vs Machine Learning Vs Artificial Intelligence

Current AI Models



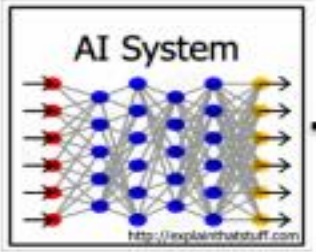
AI vs Human Intelligence

- If you are driving a car and see a soccer ball roll into the street,
- Your immediate and natural reaction is to stop the car since we can assume a child is running after the ball and isn't far behind.

Role of Probabilistic Systems

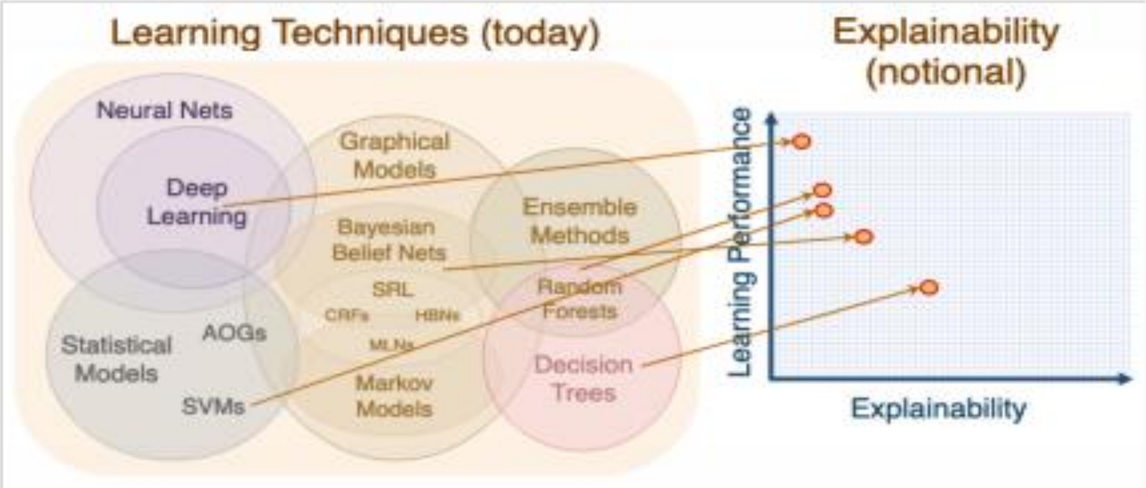
- Driver reaches the decision to stop the car based on **experience of natural data and assumptions about human behavior**.
  - But, a traditional computer likely wouldn’t reach the same conclusion in real-time, because today’s **systems are not programmed to mine noisy data efficiently** and to make decisions based on environmental awareness.
  - You would want a **probabilistic system** calling the shots-one that could quickly assess the situation and act (stop the car) immediately.

PGMs in Explainable AI

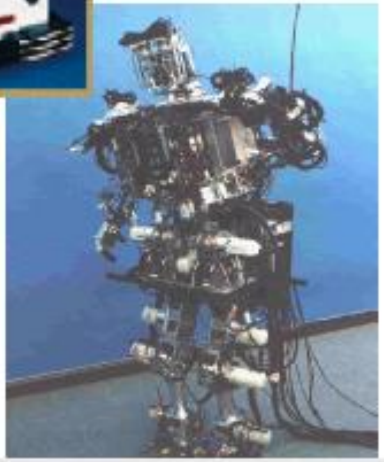


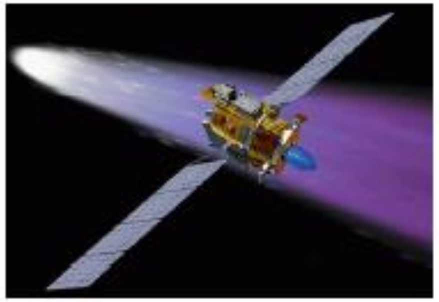
Why did you do that?
Why not something else?
When do you succeed?
When do you fail?
When can I trust you?
How do I correct an error?

Anecdote: Medical AI
Decisions can be worse with AI
e.g., Patient discharge to a nursing home



Robotics Today





```
import numpy as np

A_t_minus_1 = np.array([[1.0,  0,   0],
                        [  0,1.0,   0],
                        [  0,  0, 1.0]])

state_estimate_t_minus_1 = np.array([0.0,0.0,0.0])
control_vector_t_minus_1 = np.array([4.5, 0.05])
process_noise_v_t_minus_1 = np.array([0.01,0.01,0.003])
yaw_angle = 0.0 # radians
delta_t = 1.0 # seconds
def getB(yaw,dt):
   B = np.array([[np.cos(yaw)*dt, 0],
                 [np.sin(yaw)*dt, 0],
                 [0, dt]])
  return B
def main():
  state_estimate_t = A_t_minus_1 @ (
    state_estimate_t_minus_1) + (
    getB(yaw_angle, delta_t)) @ (
    control_vector_t_minus_1) + (
    process_noise_v_t_minus_1)

  print(f'State at time t-1: {state_estimate_t_minus_1}')
  print(f'Control input at time t-1: {control_vector_t_minus_1}')
  print(f'State at time t: {state_estimate_t}')
main()
```