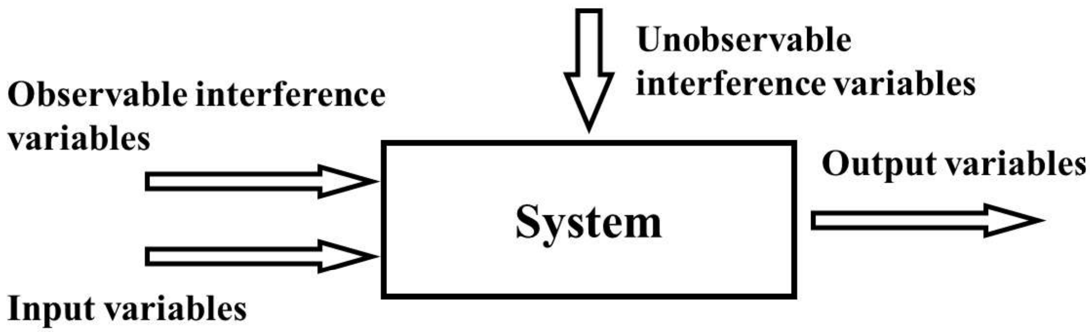
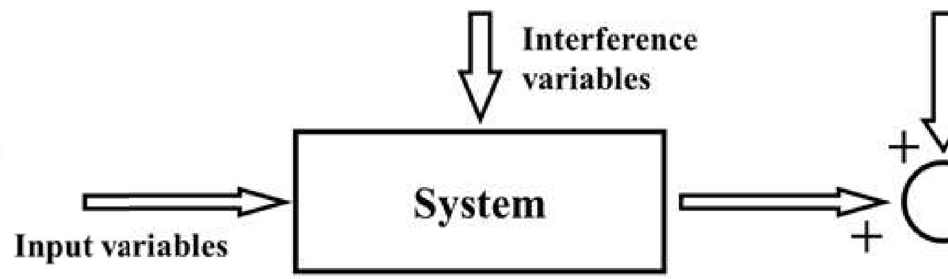
Content

Introduction

■ The concepts of system



System diagram



Univariate system diagram

What is system?

- A system represents an object, phenomenon, thing, pr
relationship.
- Its attributes must have corresponding input-output re
essential elements and unobservable interference variable

Types of system.

- Open-loop systems (with causal relationships)
- Closed-loop systems (with feedback effects)
- Sequential systems (without input effects)

Introduction

■ The concepts of system

➢ **A system is the research object of systems theory and other systems science. attributes and characteristics, including structure, function, behavior, and interc**

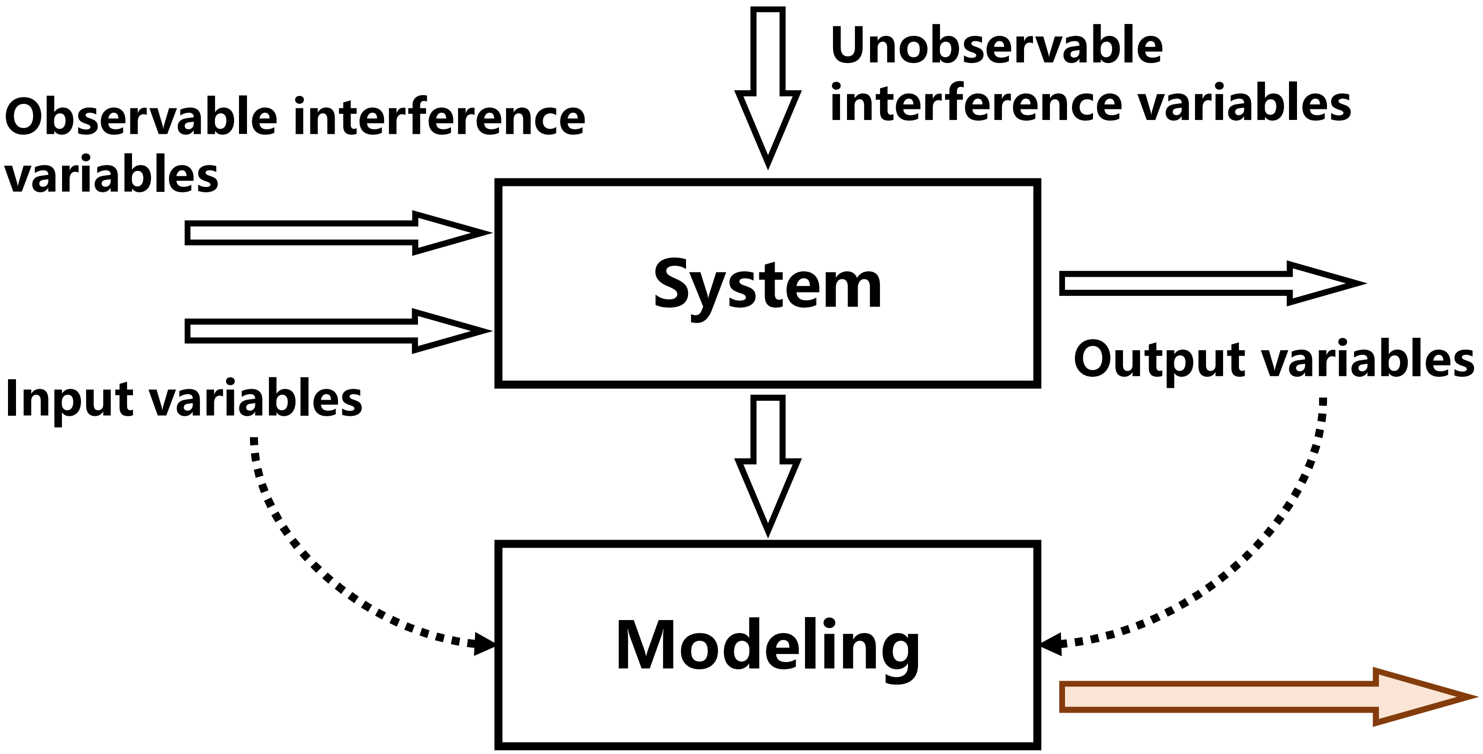
Example: computer

- A computer's hardware system is composed of arithmetic controllers, memory, and input/output devices, and the h system is a subsystem of the computer system.

Example: clock

- A clock is assembled from components such as gears, springs, and pointers in a certain way, but a pile of gears, springs, and pointers placed randomly together cannot form a clock

■ The concepts of model



Different types:

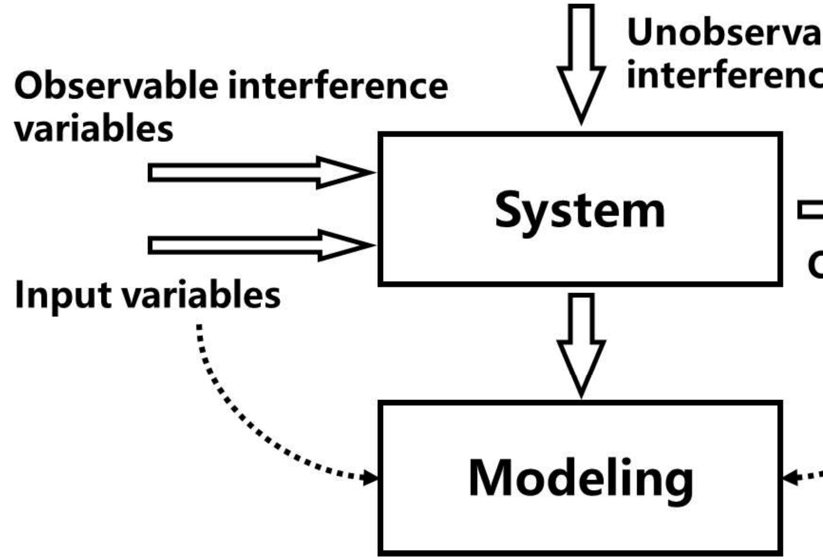
- ☐ Equation express
- ☐ Mathematical rel
- ☐ Natural laws
- ☐ Physical laws
- ☐ Assumptions
- ☐ Examples
- ☐ Schemes
- ☐ Hierarchical struc

What is models ?

- ☐ **A model is the main means of describing a system, which abstraction of specific behavioral patterns of the syste external characteristics of the system or the input-outpu the system are concerned**

■ Classification of system modeling approaches

Concerning randomness or uncertainty?
(When the state of the system described is determined, the system's output response is certain or uncertain?)



No

Yes

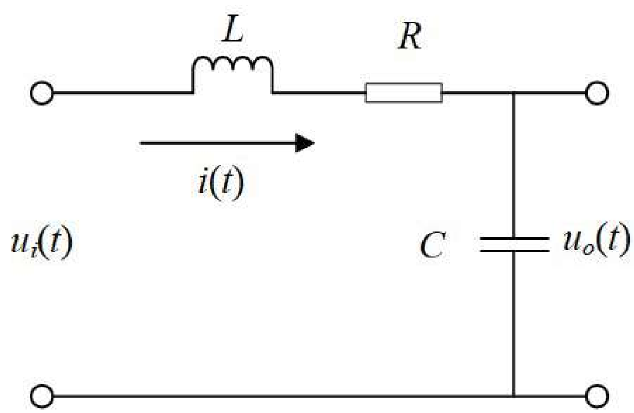
Deterministic modeling
- ☐ Differential equation modeling
- ☐ Transfer function modeling
- ☐ State space modeling
- ☐ AI-based modeling

Uncertainty modeling
- ☐ Evidence theory modeling
- ☐ Bayesian theory modeling
- ☐ Markov process modeling

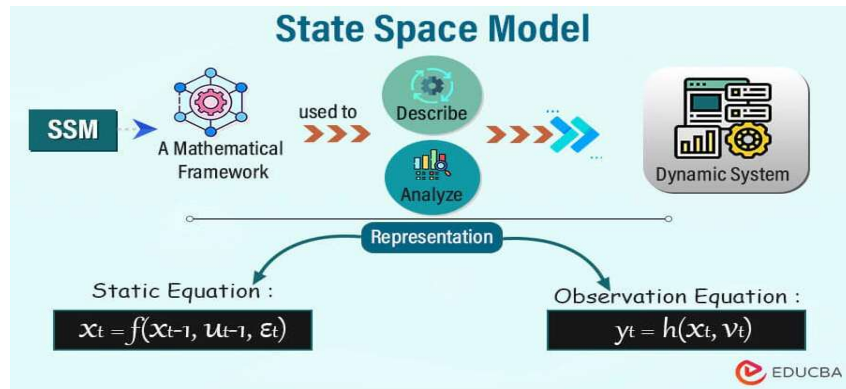
Introduction

Classification of system modeling approaches: Deterministic modeling

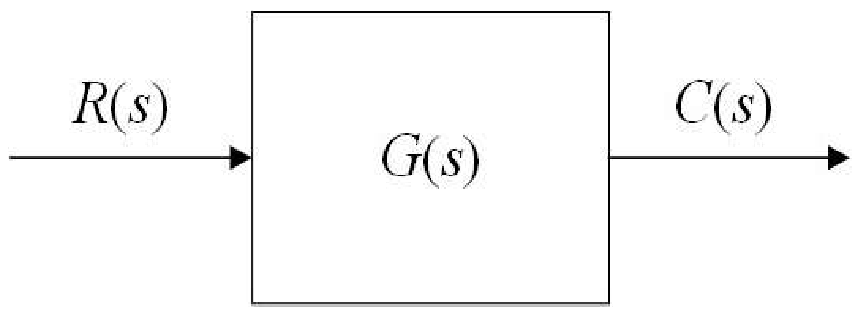
- A **mathematical model** of a system describes the **relationship** between internal **physic**
  variables) using mathematical expressions.
- In system modeling, deterministic models refer to models that do not contain any **random**



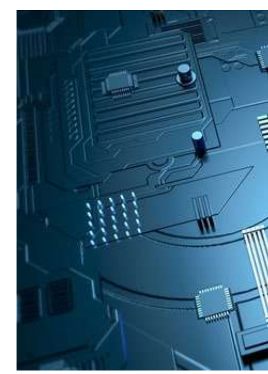
Differential equation modeling



State space modeling



Transfer function modeling

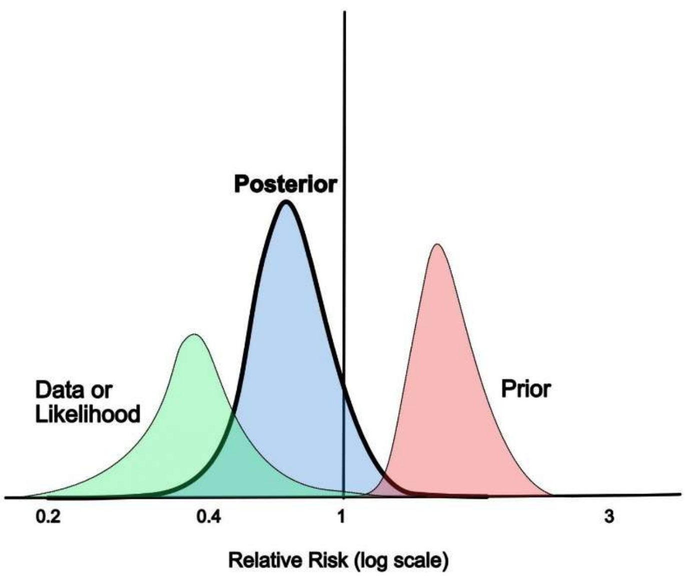


AI-base

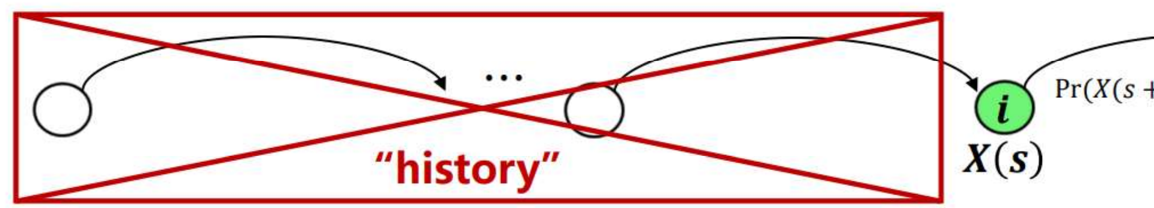
Introduction

Classification of system modeling approaches: Uncertainty modeling

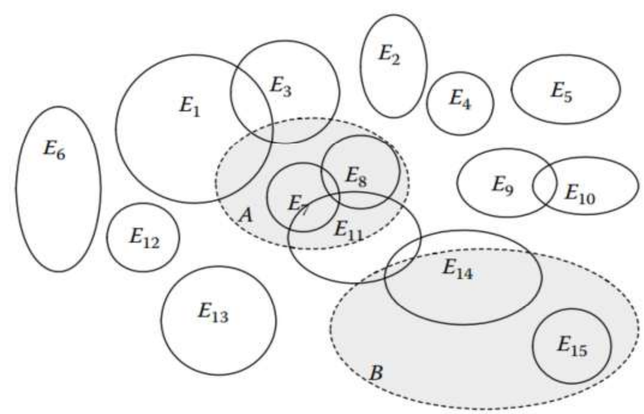
- Uncertainty modeling refers to the process of representing and quantifying the va **uncertainty** that exist within a system or a model.
- It applies to predictions of future events, to physical measurements that are already unknown.



Bayesian theory



Markov Proce



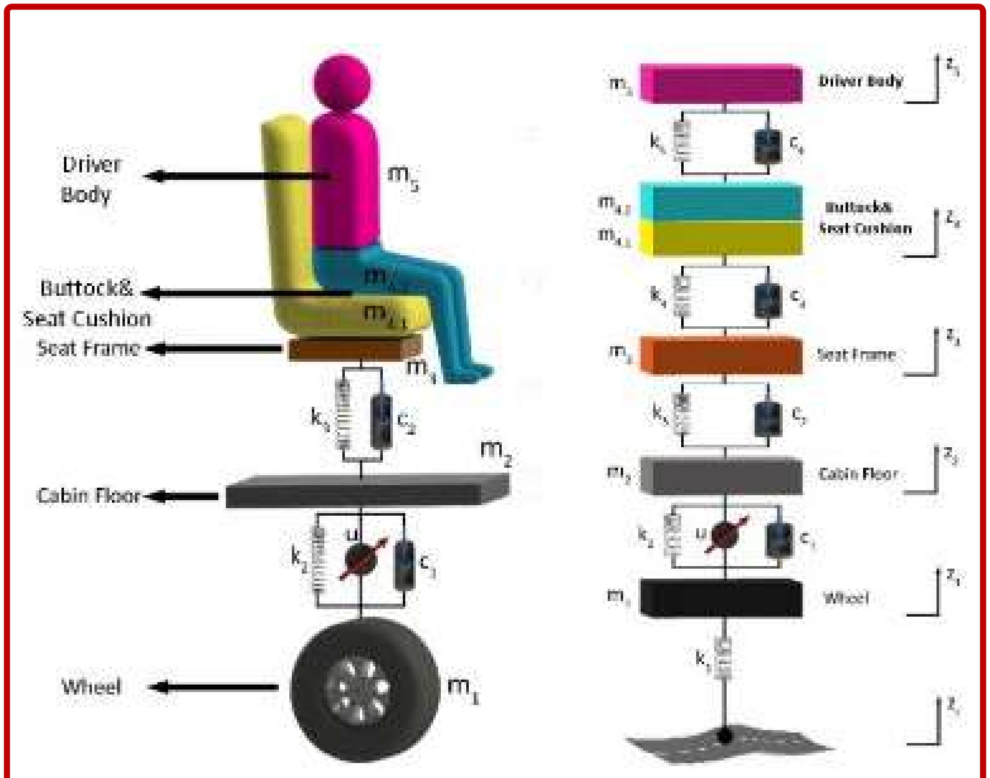
Evidence theory

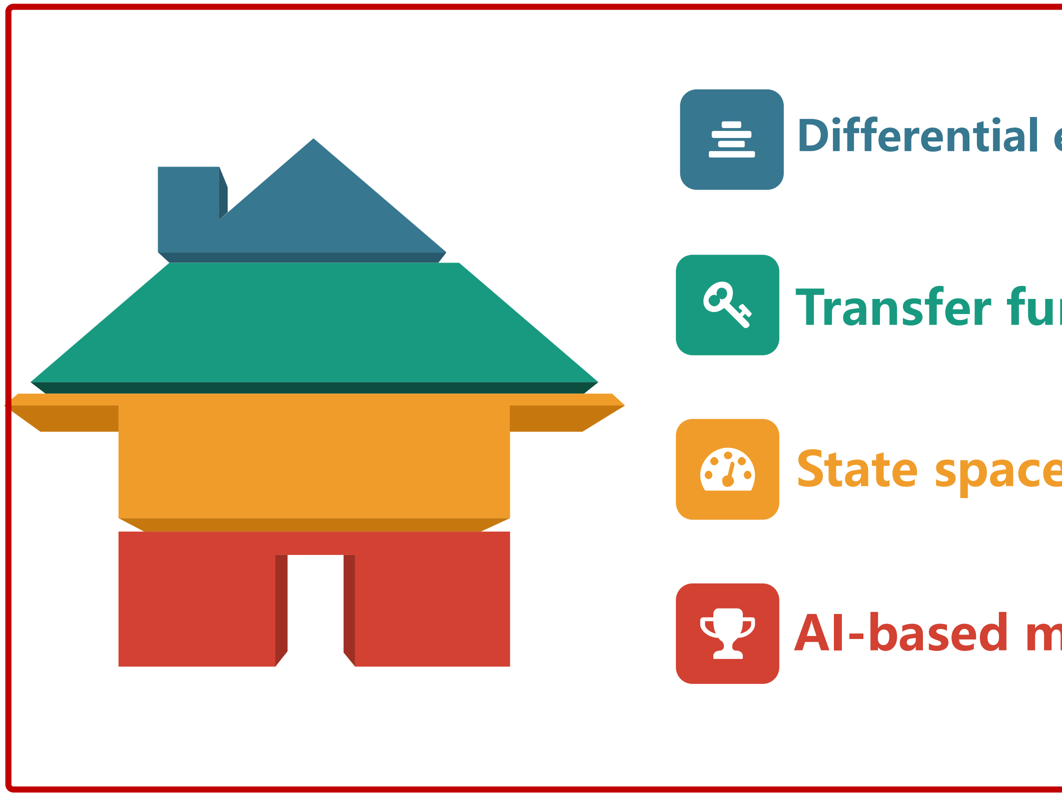
shen.yin@ntnu.no

Content

Deterministic modeling

The concepts of deterministic modeling





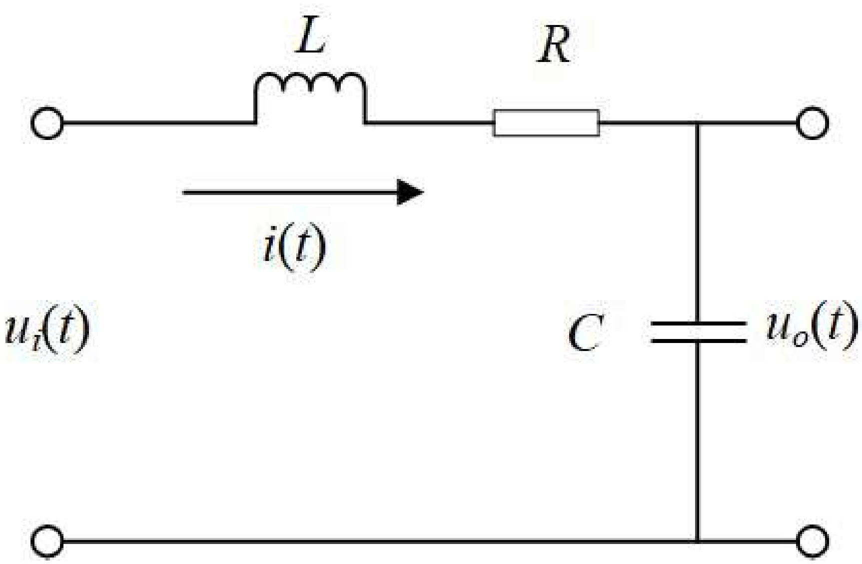
What is deterministic modeling ?

- A **mathematical model** of a system describes the **relatio** internal **physical quantities (or variables)** using mathemati
- In system modeling, deterministic models refer to mod contain any **random components**.

Deterministic modeling

Differential equation modeling — System differential equations

➢ **Example 2-1: The following figure shows a passive network composed of resisto and capacitor C. Please write down the network's differential equation with u_i and $u_o(t)$ as the output.**



RLC passive circuit network

Kirchhoff's voltage law:

$$L\frac{di(t)}{dt}+\frac{1}{C}\int i(t)dt+Ri(t)=u_i(t)$$

$$u_o(t)=\frac{1}{C}\int i(t)dt$$

Eliminating the intermediate variable $i(t)$:

$$LC\frac{d^2u_o(t)}{dt^2}+RC\frac{du_o(t)}{dt}+u_o(t)=u_i(t$$

Deterministic modeling

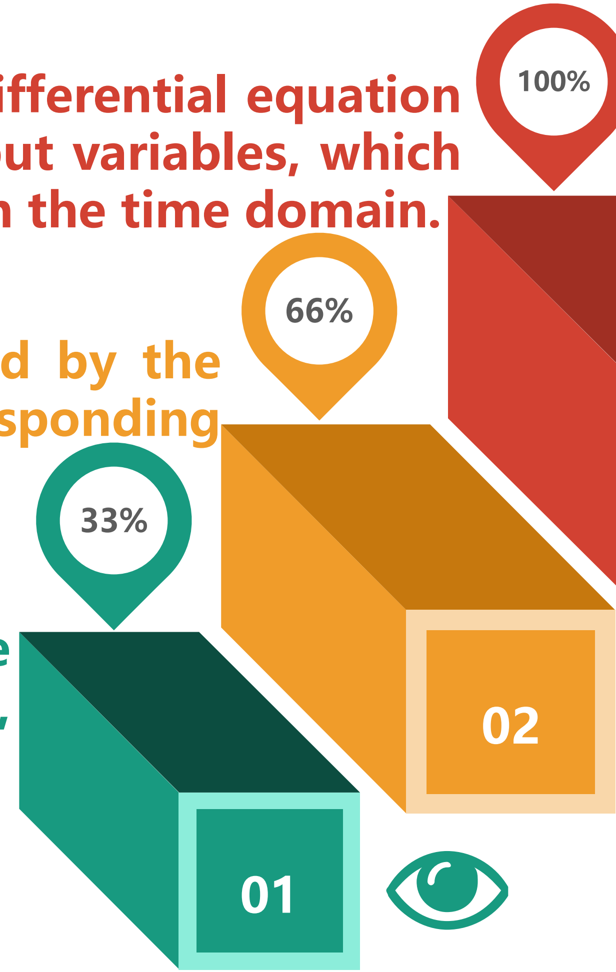
■ Differential equation modeling — System differential equations

➢ **The steps for formulating component differential equations**

Step 3: Eliminate intermediate variables to obtain the differential equation describing the relationship between the output and input variables, which constitutes the mathematical model of the component in the time domain.

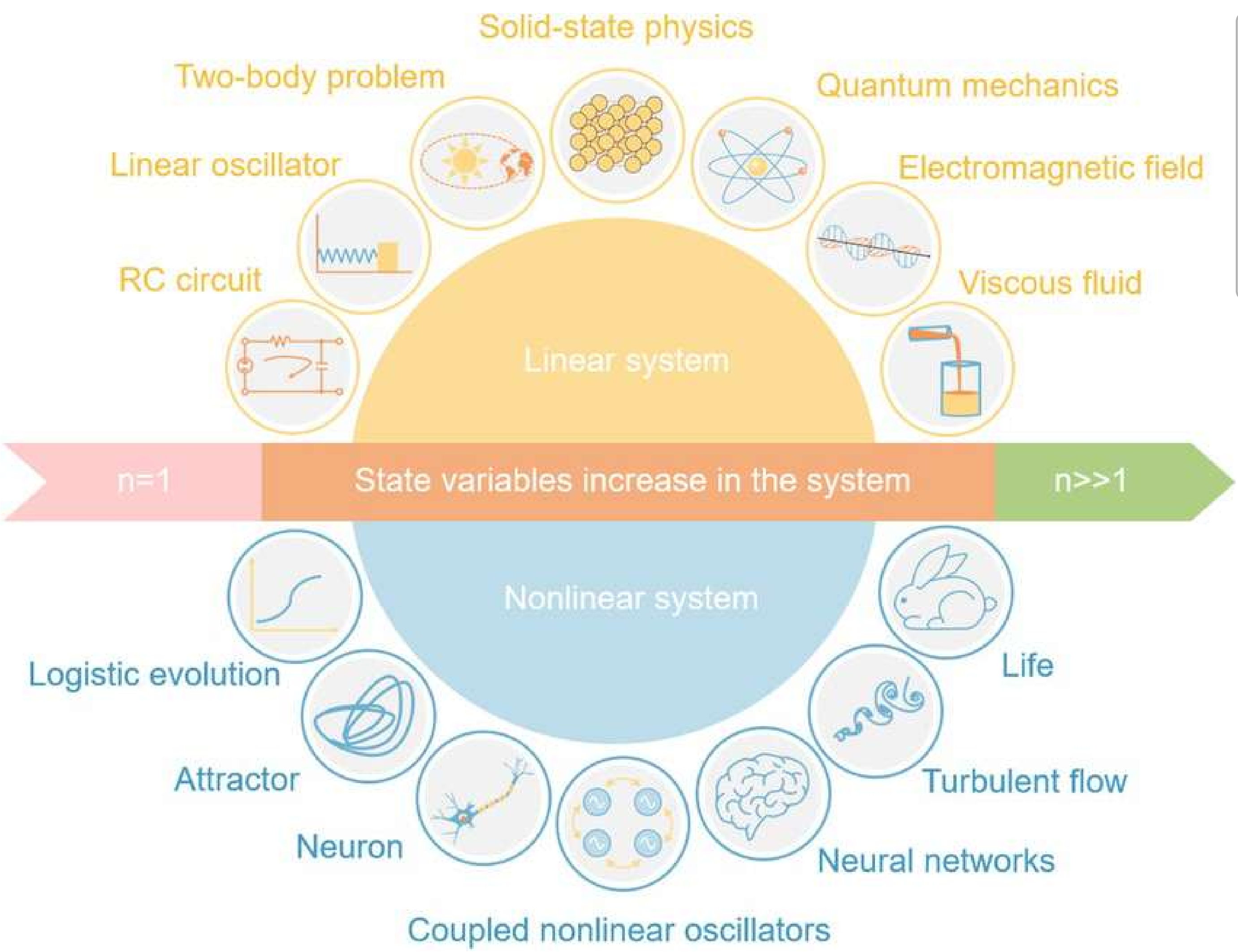
Step 2: Analyze the physical or chemical laws followed by the components in operation, and formulate the corresponding differential equations.

Step 1: Based on the operational principles of the components and their roles in control systems, determine their input and output variables.

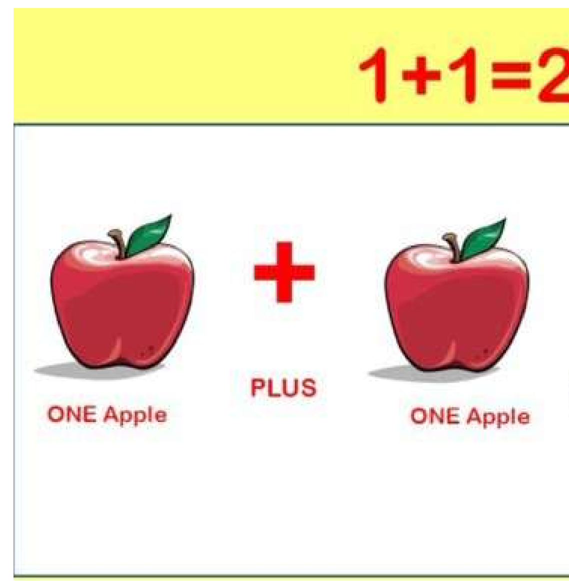


Deterministic modeling

■ Differential equation modeling — The basic characteristics of linear systems



- **What is linear system:** systems described by li equations are referre components or linear s



Superposition P

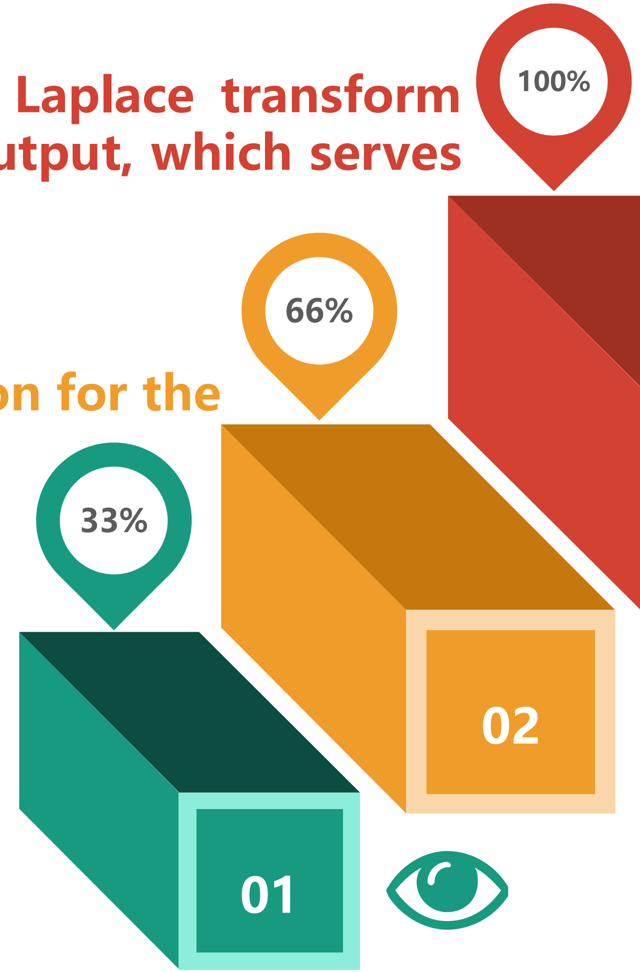
Deterministic modeling

■ Differential equation modeling — The solution of linear time-invariant differentia

Step 3: Perform the inverse transform on the output Laplace transform function to obtain the time-domain expression of the output, which serves as the solution to the given differential equation.

Step 2: From the algebraic equation, derive the expression for the Laplace transform function of the output.

Step 1: Considering the initial conditions, perform the Laplace transform on each term of the differential equation to convert the differential equation into an algebraic equation in terms of the variables.



shen.yin@ntnu.no

Deterministic modeling

Transfer function modeling

What is Transfer function modeling ?

- When using the **Laplace transform method** to solve equations of linear systems, the mathematical model system in the complex domain can be obtained.
- The transfer function not only characterizes the sy performance, but also can be used to study the eff structure or parameter changes on **system performance**.

A linear time-invariant n-order system:

$$a_0\frac{d^n c(t)}{dt^n}+a_1\frac{d^{n-1}c(t)}{dt^{n-1}}+\ldots+a_{n-1}\frac{dc(t)}{dt}+a_n c(t)$$
$$=b_0\frac{d^m r(t)}{dt^m}+b_1\frac{d^{m-1}r(t)}{dt^{m-1}}+\ldots+b_{m-1}\frac{dr(t)}{dt}+b_m r($$

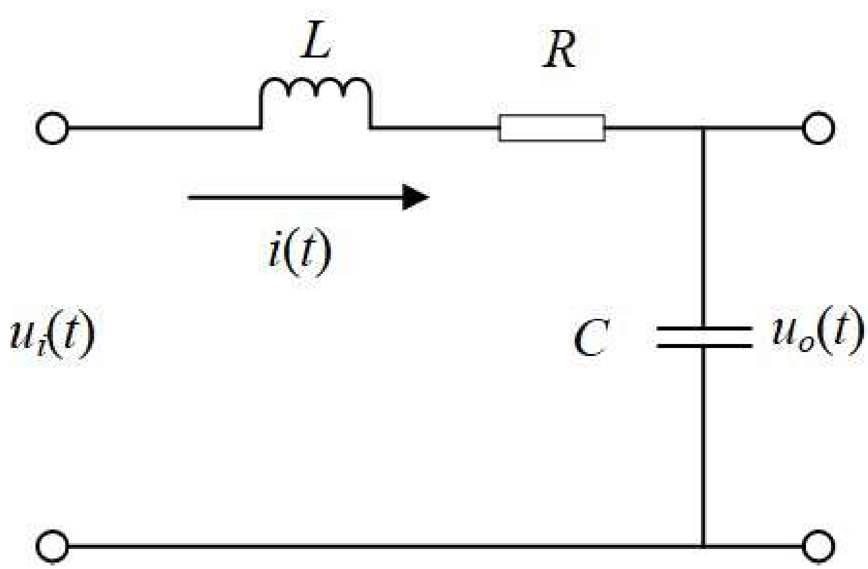
Laplace transform: $\left[a_0 s^n+a_1 s^{n-1}+\ldots+a_{n-1}s+a_n\right]C(s)=\left[b_0 s^m+b_1 s^{m-1}+\ldots+b_{m-1}s+b_m\right]R$

Transfer function: $G(s)=\dfrac{C(s)}{R(s)}=\dfrac{b_0 s^m+b_1 s^{m-1}+\ldots+b_{m-1}s+b_m}{a_0 s^n+a_1 s^{n-1}+\ldots+a_{n-1}s+a_n}$

Deterministic modeling

Transfer function modeling

➢ **Example 2-2: Let's find the transfer function $U_o(s)/U_i(s)$ for the RLC passive net in Example 2-1.**



RLC passive circuit network

The differential equation for the RLC networ

$$LC\frac{d^2u_o(t)}{dt^2}+RC\frac{du_o(t)}{dt}+u_o(t)=u_i($$

The transfer function:

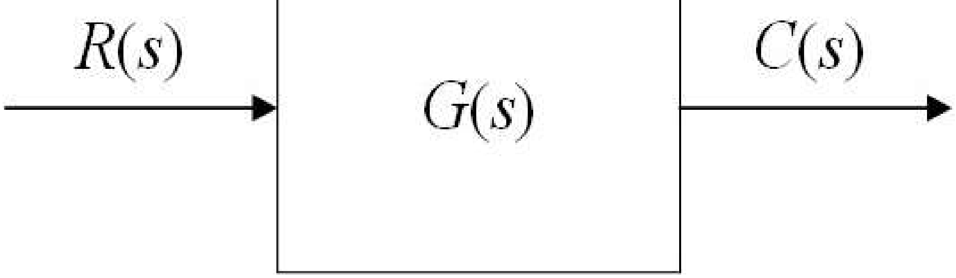
$$G(s)=\frac{U_o(s)}{U_i(s)}=\frac{1}{LCs^2+RCs+1}$$

Deterministic modeling

■ Transfer function modeling

Property 1: The transfer function is a rational function of the complex variable s.

Property 2: function is that relates the input system para

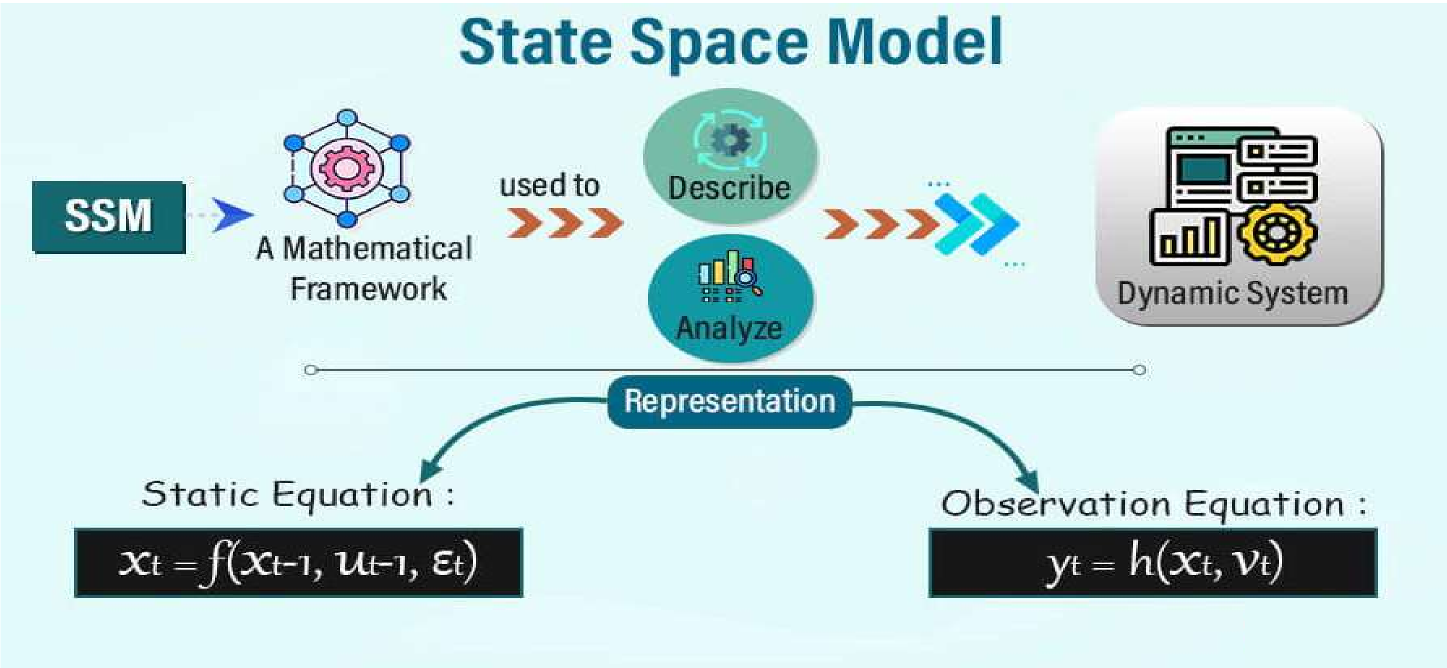


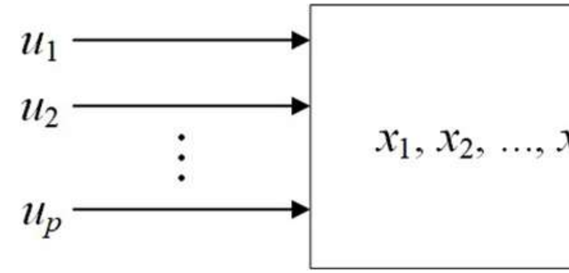
Property 3: The transfer function analogous to a differential equatio

The graphic representation of transfer function

Deterministic modeling

State space modeling





Input vectors: u

Output vectors: y

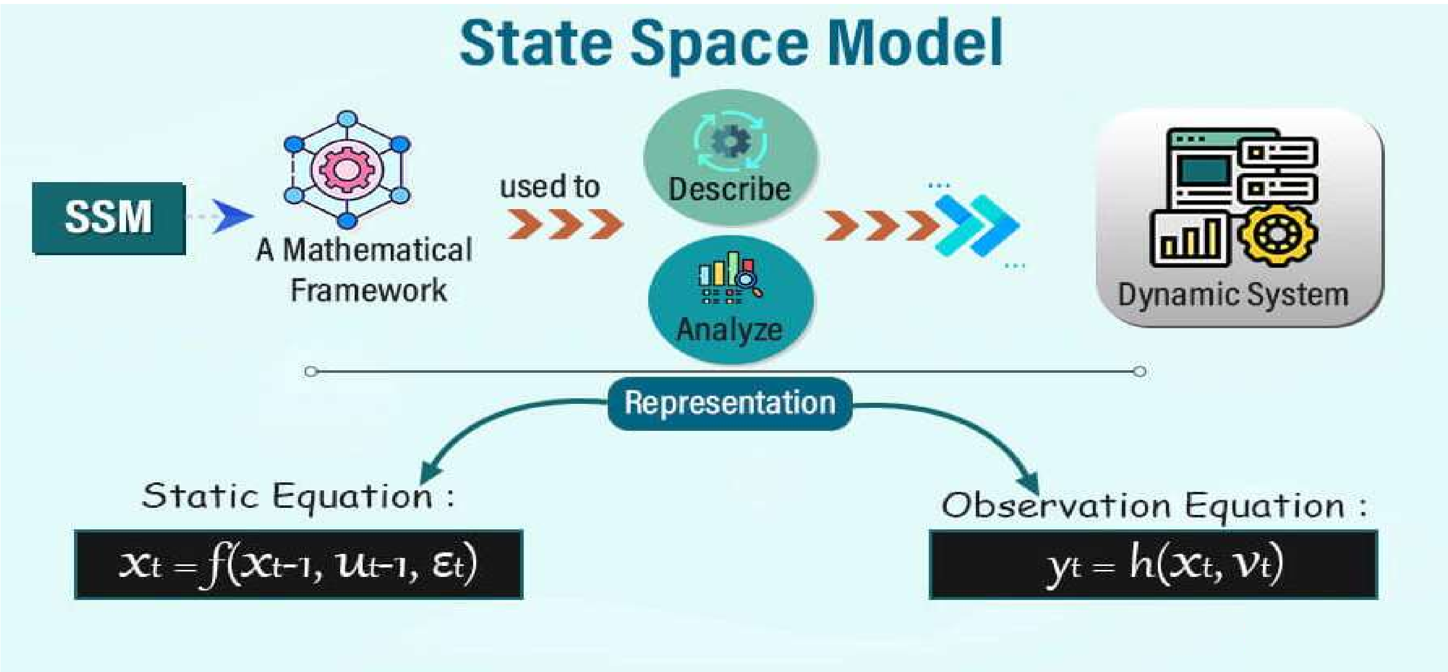
Internal variables: x

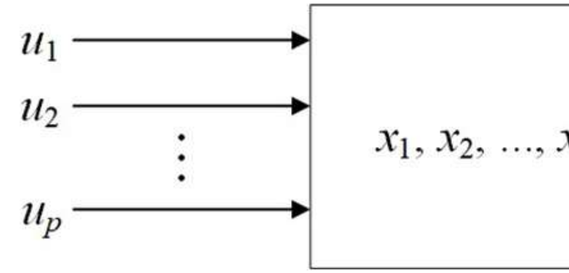
What is state space modeling ?

- **State-space modeling** is a mathematical framework for behavior of dynamic systems.
- The state space represents the system through a set of and their **changing relationship** over time, and links the in **state variables** of the system.

Deterministic modeling

State space modeling





Input vectors: u =

Output vectors: y =

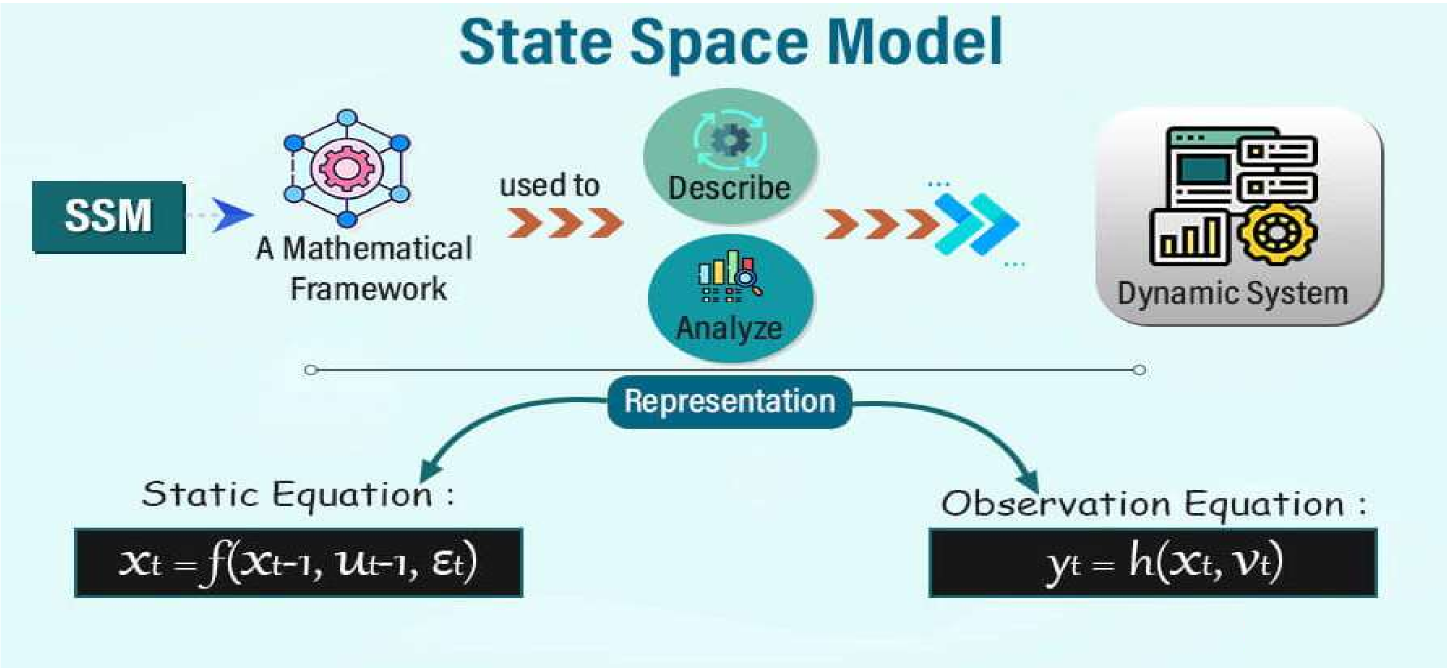
Internal variables: x

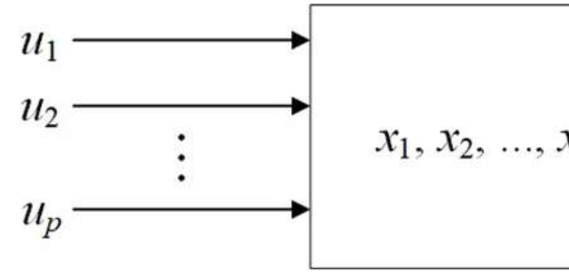
Basic concepts of state space modeling

- **State and State Variables**: The collection of information abo or motion of the system in the time domain is referred to a
- **State-space representation**: The system using a set differential (or difference) equations that relate the variables, inputs, and outputs.

Deterministic modeling

State space modeling





Input vectors: $u =$

Output vectors: y

Internal variables: x

Representation of state space modeling

$$\begin{cases} \dot{x}(t) = A(t)x(t) + B(t)u(t) \\ y(t) = C(t)x(t) + D(t)u(t) \end{cases}$$

$x(t)$: **The state vector**

$u(t)$: **The input vector**

$y(t)$: **The output vector**

A : **The sta**

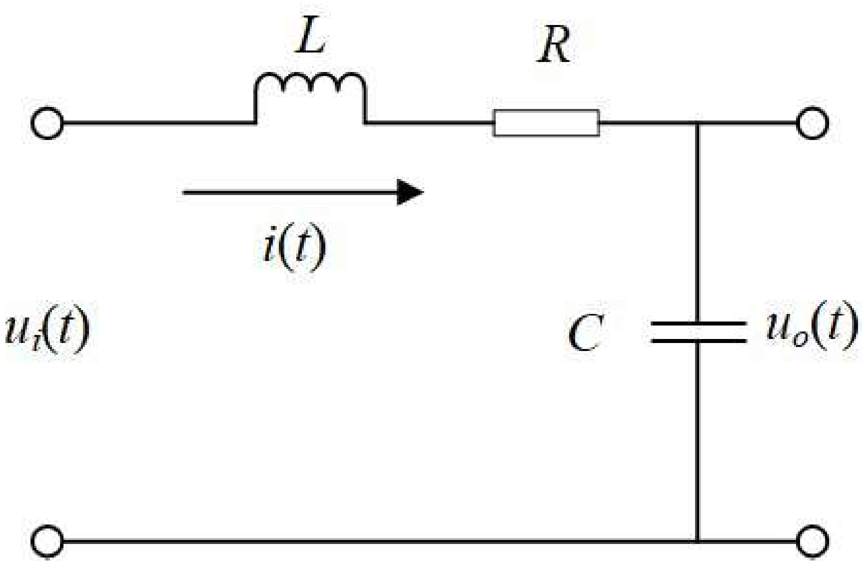
B : **The inp**

C : **The ou**

D : **The fe**

State space modeling

- **Example 2-3: Let's write down the circuit equations for the RLC network shown select a few sets of state variables, and establish the corresponding state-space Then, we'll discuss the relationship between the selected state variables.**



RLC passive circuit network

According to circuit laws, we can write th equations:

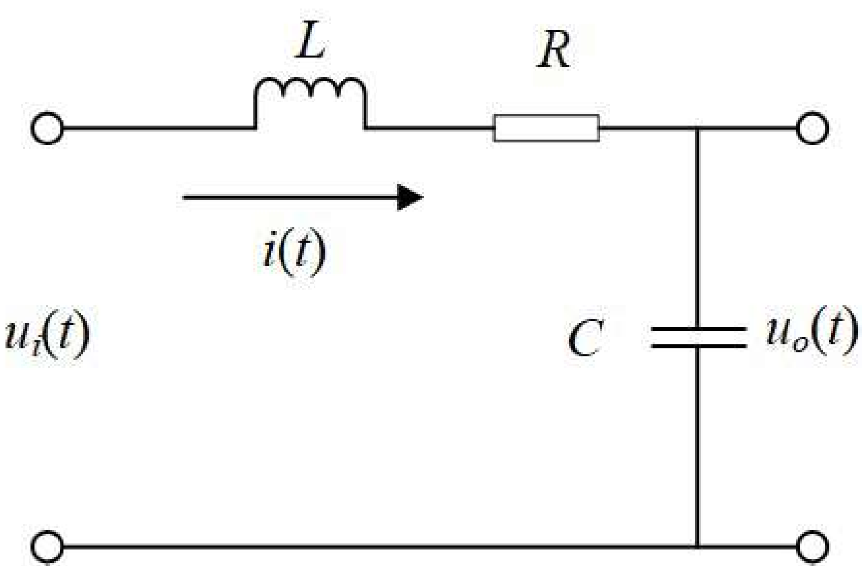
Circuit laws: $$Ri + L\frac{di}{dt} + \frac{1}{C}\int idt = u_i$$

The output of the circuit: $$y = u_c = \frac{1}{C}\int$$

Deterministic modeling

State space modeling

- Example 2-3: Let's write down the circuit equations for the RLC network shown select a few sets of state variables, and establish the corresponding state-space Then, we'll discuss the relationship between the selected state variables.



RLC passive circuit network

Solution 1: **Assume** $x_1 = i$ $x_2 = \frac{1}{C}$

$$\begin{cases} \dot{x}_1(t) = -\frac{R}{L}x_1 - \frac{R}{L}x_2 + \frac{1}{L}e \\ \dot{x}_2(t) = \frac{1}{C}x_1 \end{cases} \quad \textbf{(2-10)}$$

$$y = x_2$$

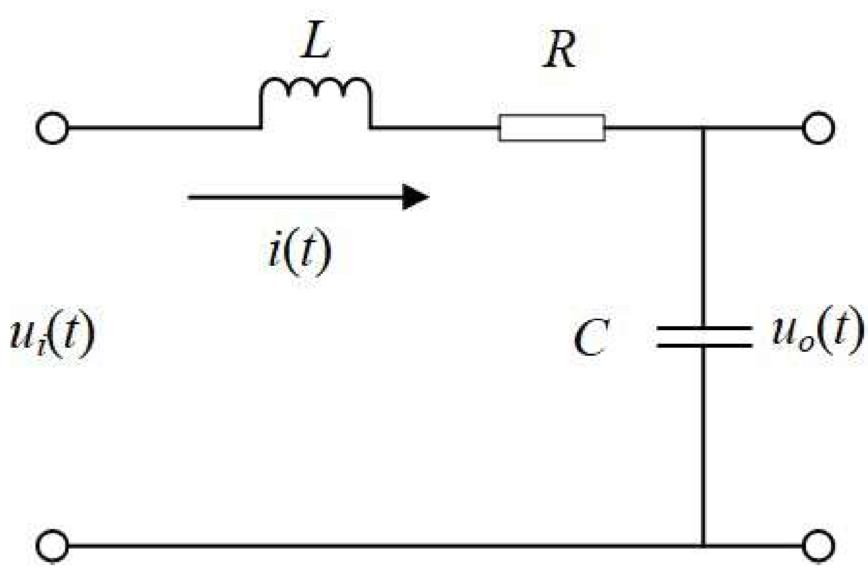
$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} -\frac{R}{L} & -\frac{1}{L} \\ \frac{1}{C} & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} \frac{1}{L} \\ 0 \end{bmatrix} e$$

$$y = \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \quad \textbf{(2-12)}$$

Deterministic modeling

State space modeling

➢ **Example 2-3: Let's write down the circuit equations for the RLC network shown select a few sets of state variables, and establish the corresponding state-space Then, we'll discuss the relationship between the selected state variables.**



RLC passive circuit network

Solution 2: **Assume** $x_1 = \frac{1}{C}\int idt + Ri \quad x_2 =$

$$\begin{cases} \dot{x}_1(t) = -\frac{R}{L}x_1 - \frac{1}{LC}x_2 + \frac{1}{L}e \\ \dot{x}_2(t) = x_1 \end{cases} \quad \textbf{(2-13)}$$

$$y = \frac{1}{C}x_2$$

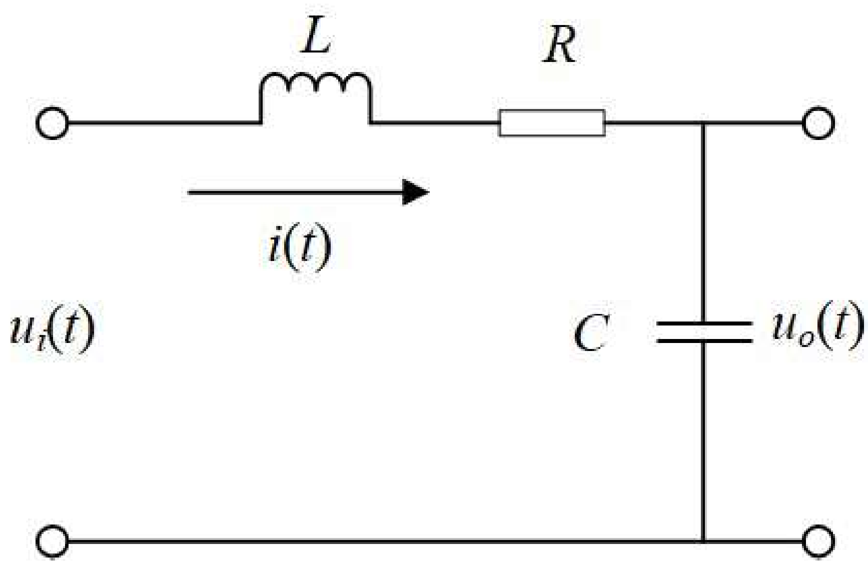
$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} -\frac{R}{L} & -\frac{1}{LC} \\ 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} \frac{1}{L} \\ 0 \end{bmatrix} e$$

$$y = \begin{bmatrix} 0 & \frac{1}{C} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \quad \textbf{(2-15)}$$

Deterministic modeling

State space modeling

- Example 2-3: Let's write down the circuit equations for the RLC network shown select a few sets of state variables, and establish the corresponding state-space Then, we'll discuss the relationship between the selected state variables.

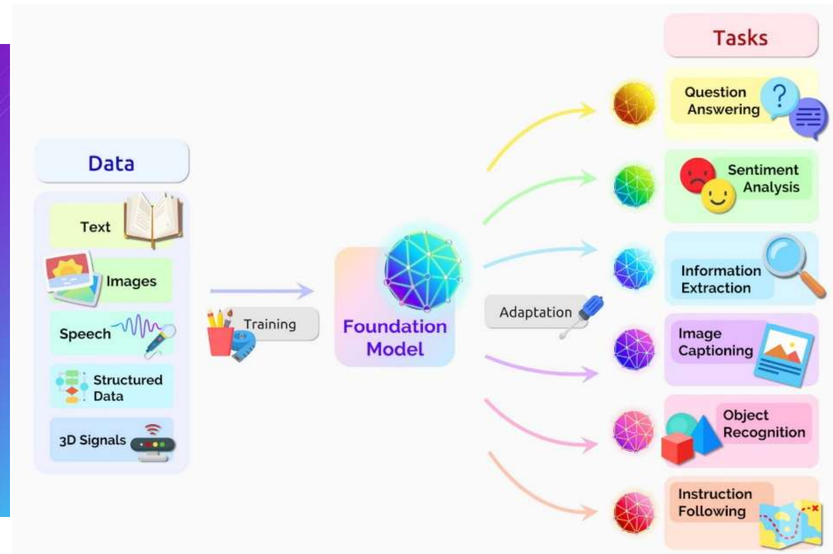


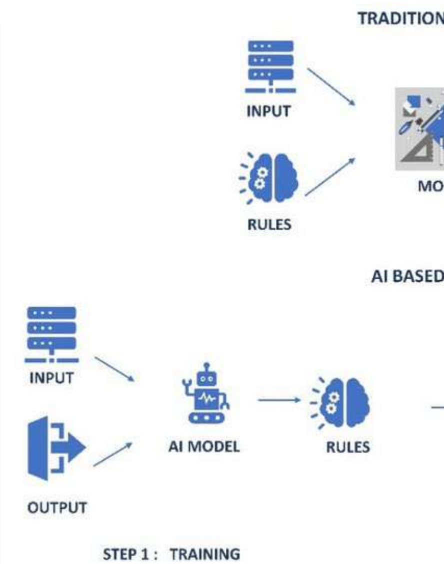
RLC passive circuit network

- **The state-space representation of the unique.** Choosing different state vari different state-space representations, describe the same system.
- It can be inferred that there exists a **transformation relationship** between c space representations describing the sam

Deterministic modeling

AI-based modeling



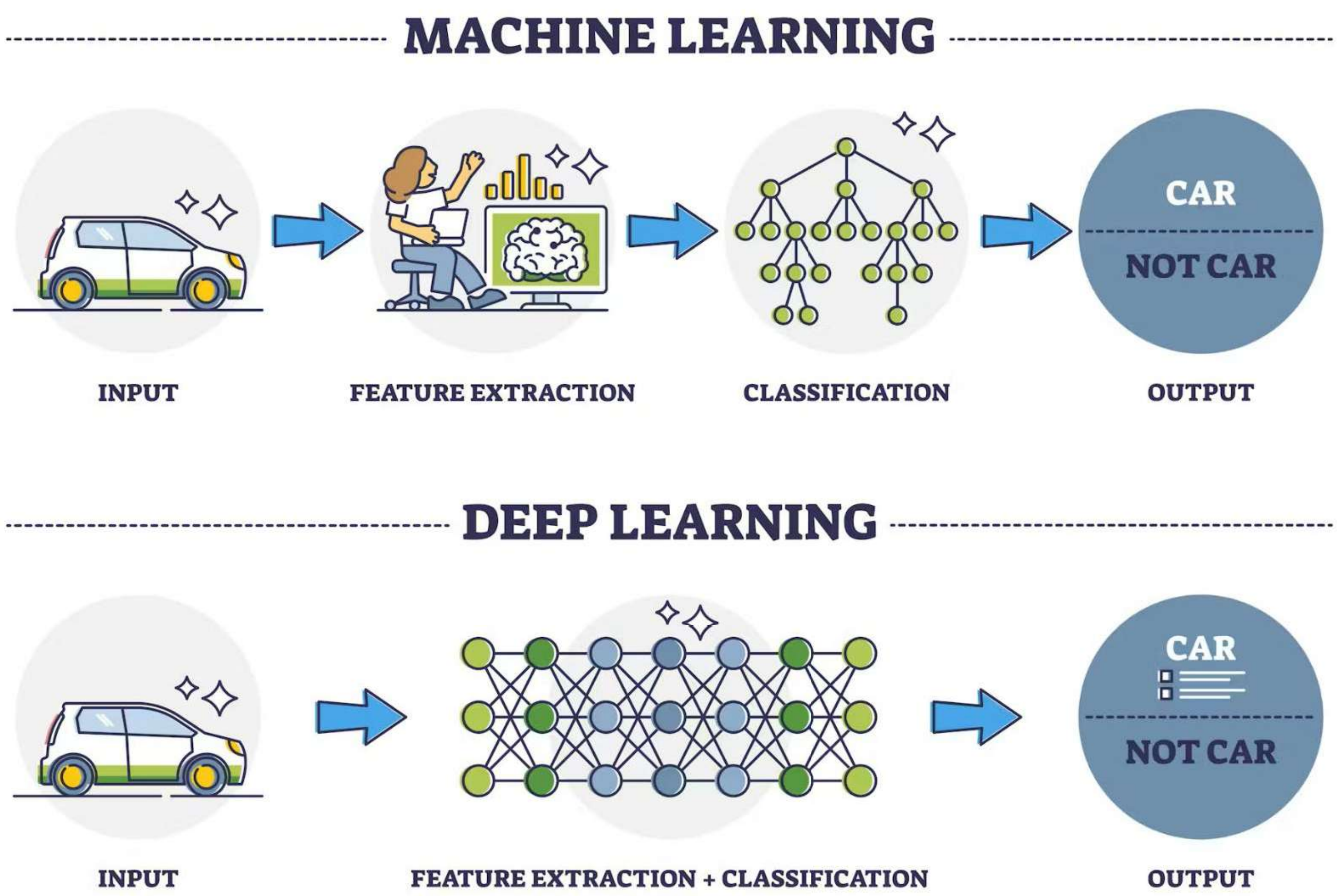


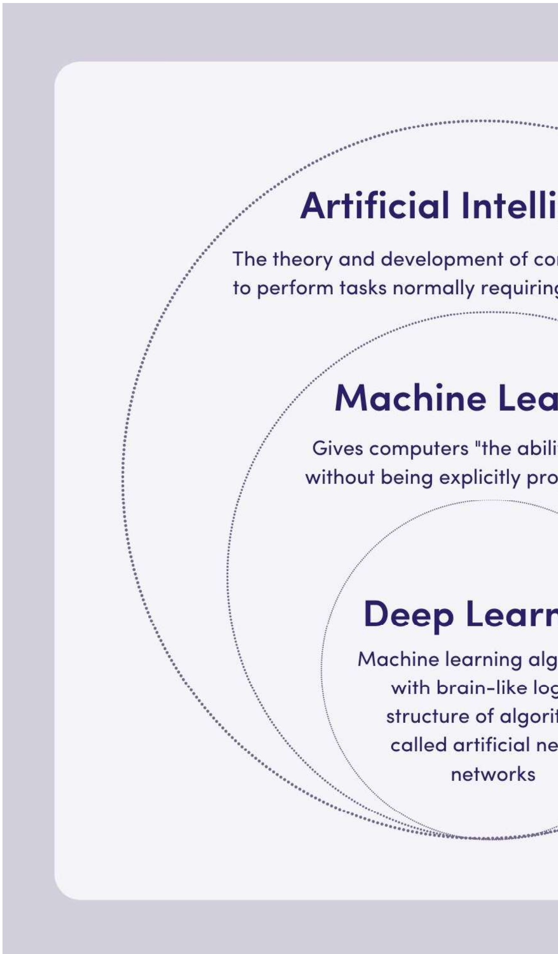
What is AI-based modeling ?

- **AI-based modeling: Construct learning-based algorithn establish the mapping relationship between input dat output for direct end-to-end prediction.**
- **Without the need for any prior knowledge, AI-based sys methods can learn feature representations from the inhere the system.**

Deterministic modeling

■ AI-based modeling

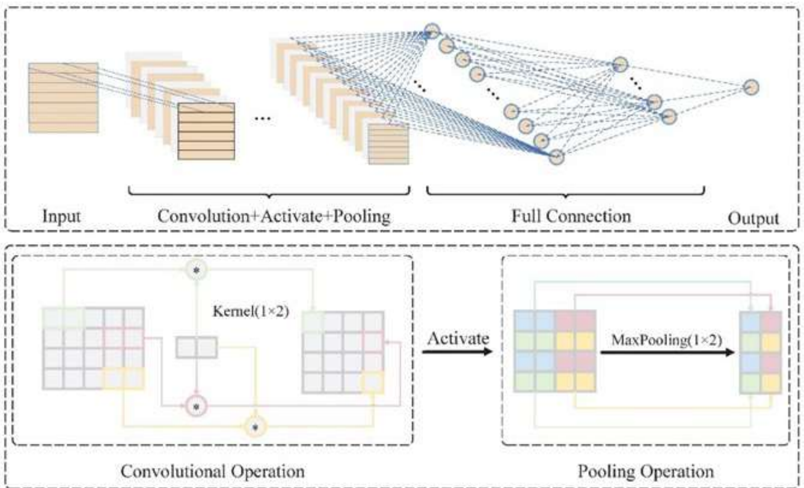




Deterministic modeling

AI-based modeling—Neural networks

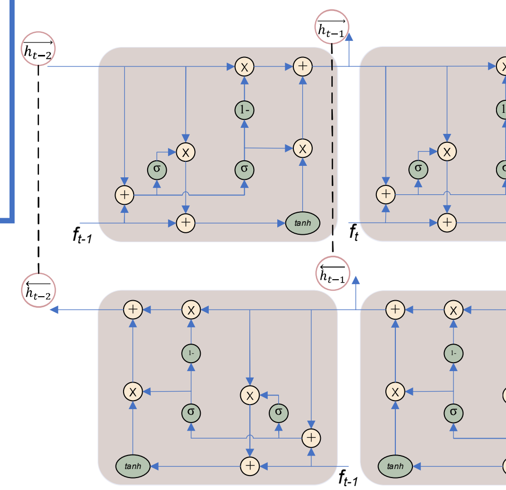
Convolutional Neural Network



Deep networks exploit

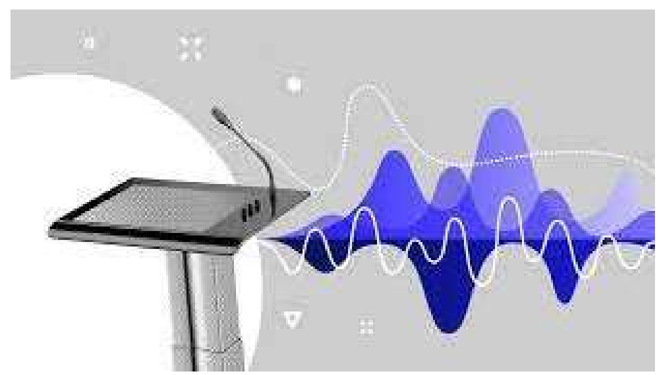
- Translation equivariance (weight sharing)
- Hierarchical compositionality

Recurrent Neural



- **Image data**

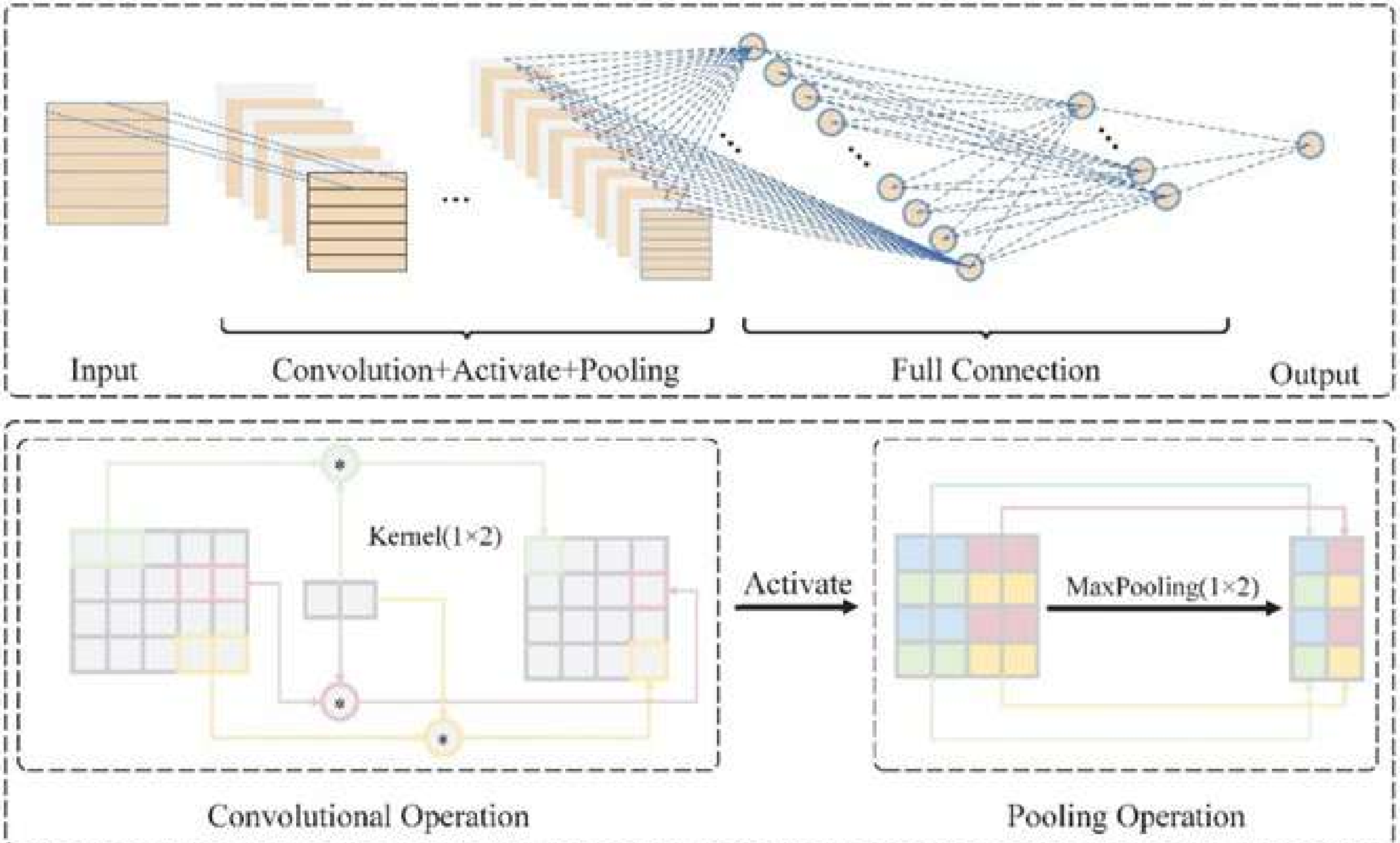
- **Speech data**



- **Natural langu**

Deterministic modeling

■ AI-based modeling—Convolutional neural network



Assuming:
Convolution kernel
Input matrix : $X = [$

At the coordinates (
the input matrix, th
output of the
convolution operati
y **shown as follows:**

$$y = f\left(\sum_{p=1}^{t}\sum_{q=1}^{t} k_{pq} \cdot x_{i+p-}\right.$$

where f **is the ReLU**
function, b **is the of**

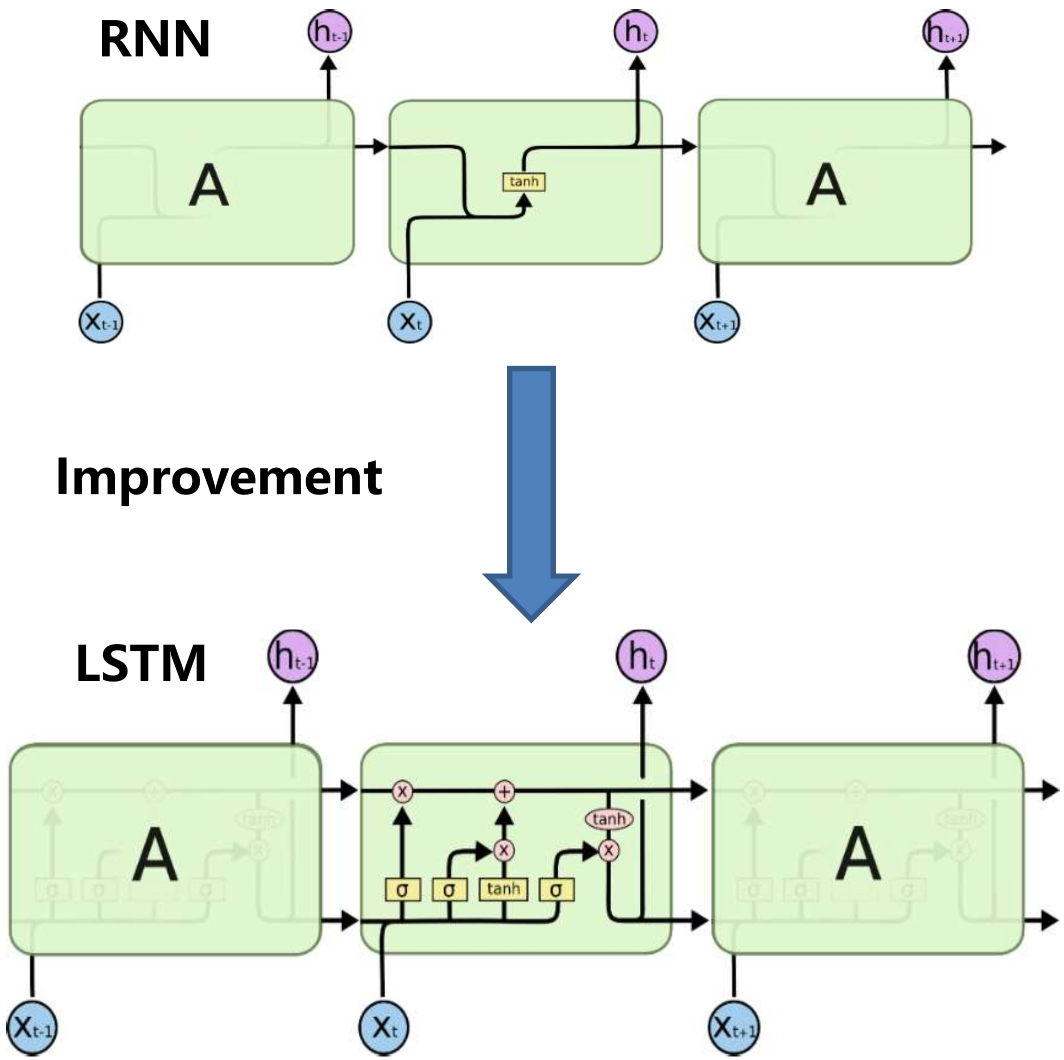
■ **Advantages**

- ◻ **Automatic feature extraction**
- ◻ **Weight sharing**

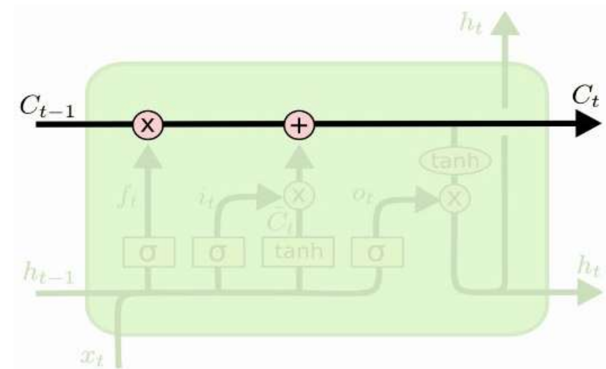
- ◻ **Convolution operation**
- ◻ **Pooling operation**

Deterministic modeling

■ AI-based modeling—Long–short-term memory network



1. Cell State: Store LSTM cell inforn



2. Gated structure

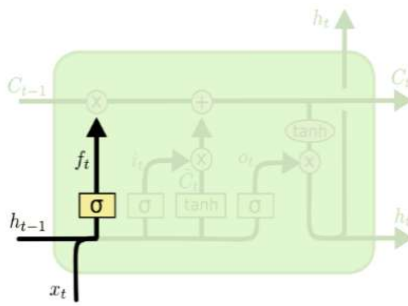
■ How to control the cell state of t

- LSTM **removes or adds** cell state inf the gate structure.
- LSTM contains **3 gate structures** to namely, input gate, forget gate and
- **Each gate structure** contains a Sigm layer and a bitwise multiplication o

Deterministic modeling

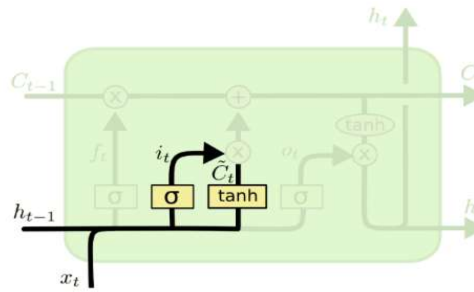
AI-based modeling—Long–short-term memory network

Forget gate:
Decide which part of the information is forgotten.



$$f_t = \sigma\left(U_f\left[h_{t-1}, x_t\right] + b_f\right) \quad \textbf{(2-17)}$$

+

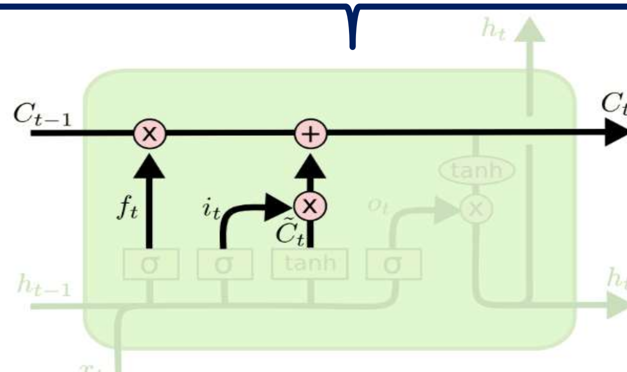


$$i_t = \sigma\left(U_i\left[h_{t-1}, x_t\right] + b_i\right) \quad \textbf{(2-18)}$$

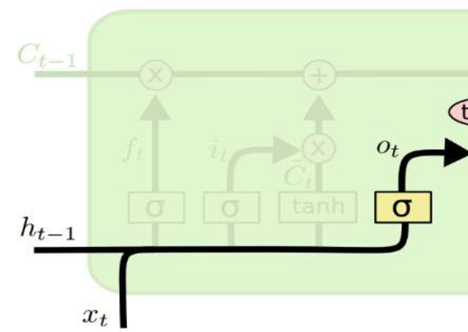
Input gate:
Determine the informa that needs to be adde the cell state in the cur sequence information

Output gate:
h(t-1),x(t),c(t) to determine l current mome

Forget gate and input gate work together to update cell state.



$$c_t = f_t * c_{t-1} + i_t * \tanh\left(U_c\left[h_{t-1}, x_t\right] + b_c\right) \quad \textbf{(2-19)}$$



$$o_t = \sigma\left(W_o\left[h_{t-1}, x_t\right] + b\right.$$

$$h_t = o_t * \tanh\left(c_t\right)$$

Content

Uncertainty modeling

The concept of uncertainty modeling

What is Uncertainty ?

- Uncertainty refers to epistemic situations involving imperfect or unknown information.
- Uncertainty is usually classified into aleatory uncertainty and epistemic uncertainty.

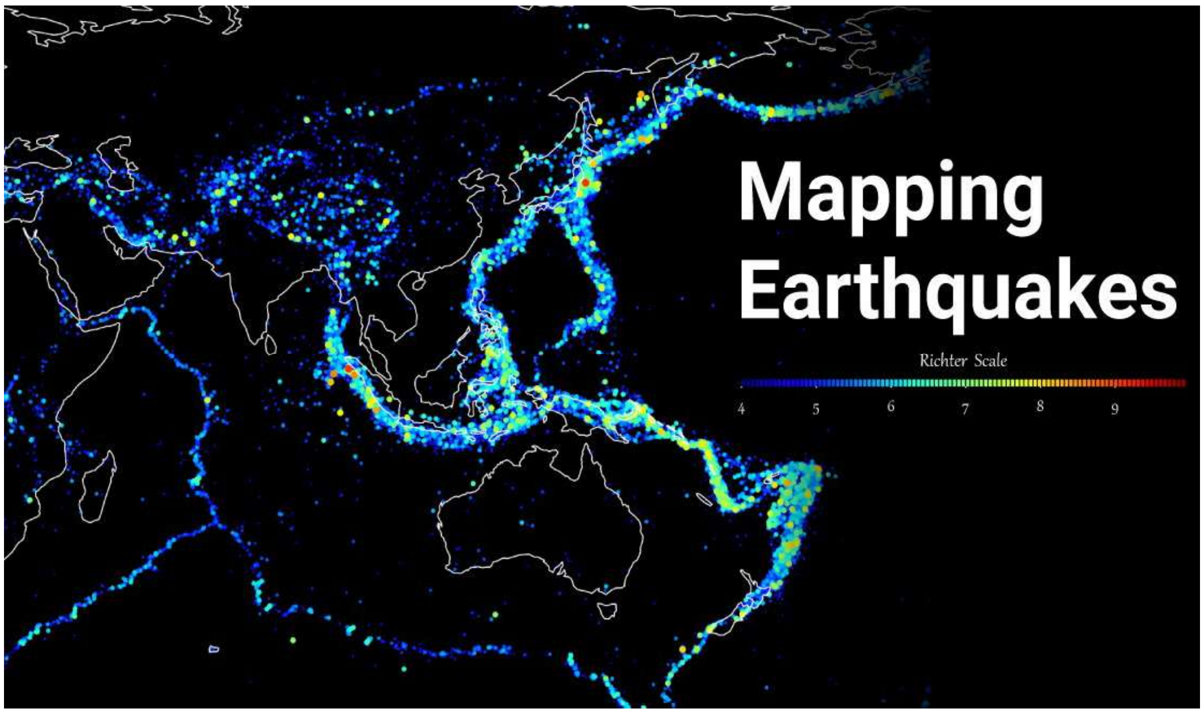
What is uncertainty modeling ?

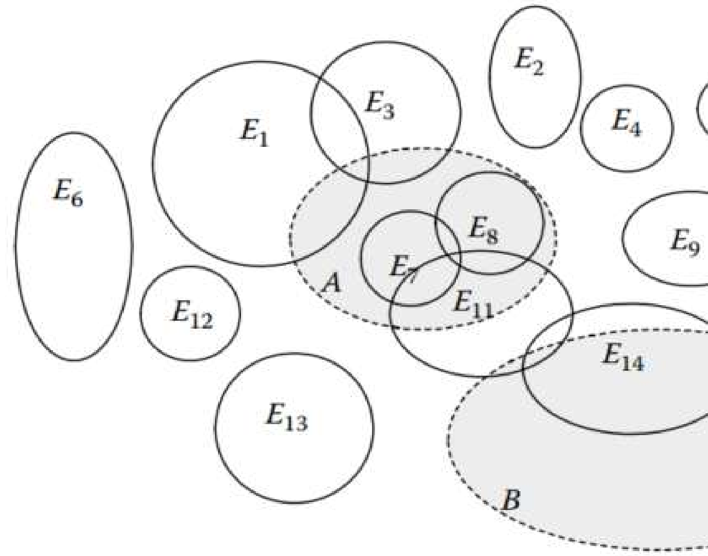
- Uncertainty modeling refers to the process of representing an
various sources of uncertainty that exist within a system or a moc
- It applies to predictions of future events, to physical measu
already made, or to the unknown.

Uncertainty modeling

Evidence theory

- Example 3-1: Assessing the potential consequences of an earthquake wi
location of its epicenter.



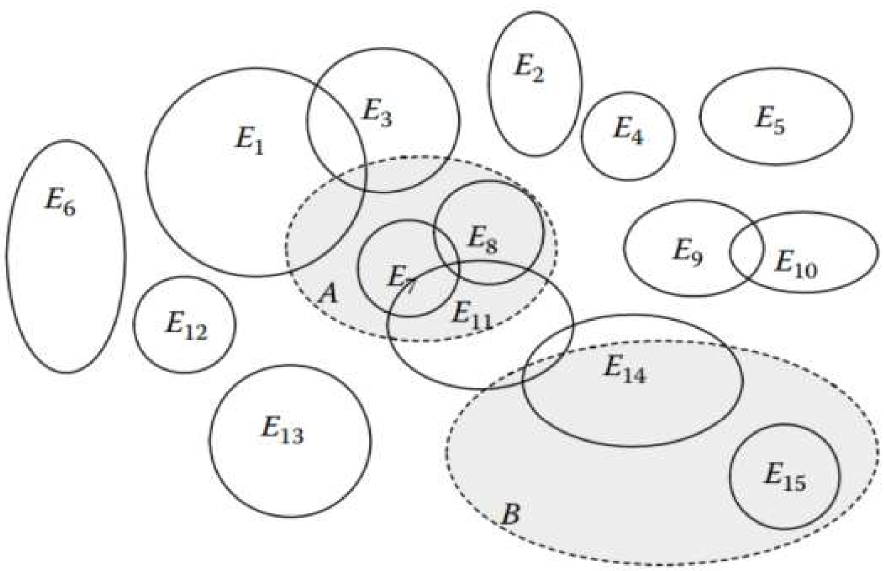


A group of 15 experts provided thei
possible location of the epicenter of t
in the form of zones

Uncertainty modeling

Evidence theory

- **Example 3-1: Assessing the potential consequences of an earthquake**
- A group of 15 experts provided their estimates of possible location of the epicenter earthquake in the form of zones



Estimates of the locations of the epicenter of an earthquake

- **The location estimates are both nonsp (i.e., provided as regions) and conflicti not in agreement) with each other.**
- **The expert opinions can be treated as of evidence in this case.**
- **This evidence body can be used to est the likelihood that the epicenter is insi particular areas of interest representin cities or densely populated areas A and**

shen.yin@ntnu.no

Uncertainty modeling

■ Evidence theory

- The theory of evidence, also called the Dempster–Shafer theory (DST) developed by Shafer and Dempster. This theory is based on belief mea and plausibility measures.

1: Belief and plausibility measures

2: Basic assignment

A basic assignment (m) can be conveniently by the following function that maps the powe the range [0, 1]:

$$m : P_X \rightarrow [0,1]$$

For any set $A_i \in P_X$, the belief measure (Bel) a (Pl) measure is

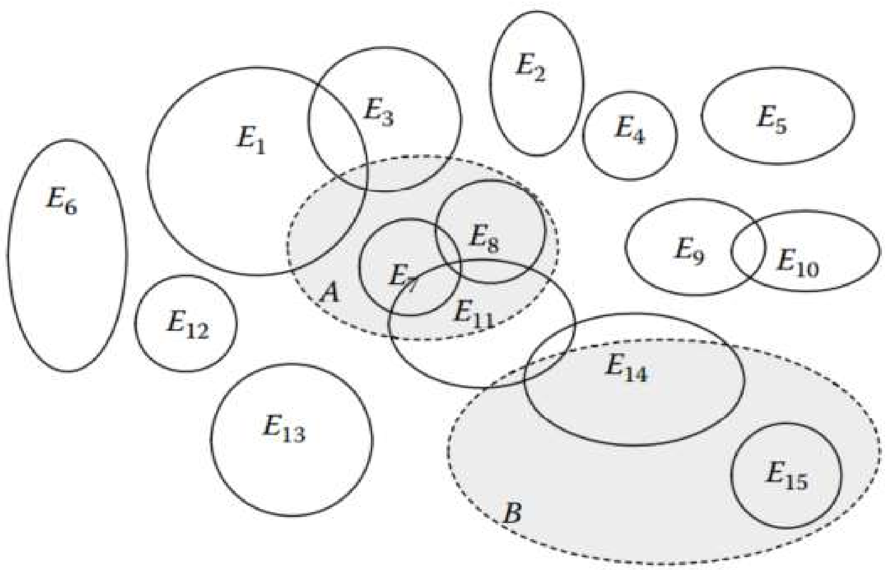
$$Bel(A_i) = \sum_{all A_j \subseteq A_i} m(A_j) \qquad Pl(A_i) = \sum_{all A_j \cap A_i \neq \varnothing}$$

The belief and plausibility functions satisfy

$$Bel(A) \leq P(A) \leq Pl(A)$$

■ Evidence theory

➢ **Example 3-1: Assessing the potential consequences of an earthquake**

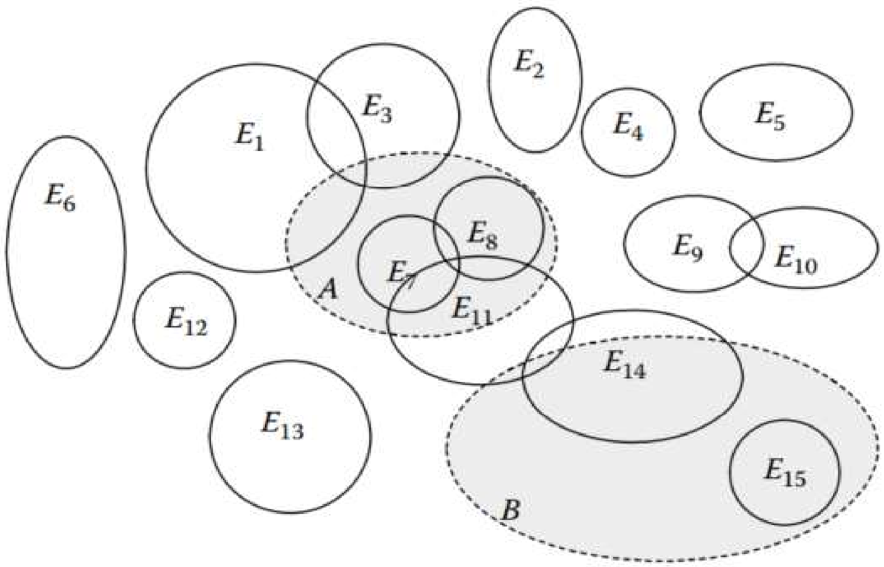


Estimates of the locations of the epicenter of an earthquake

- **The experts can be assumed for the purp example to have an equal level of credibi reliability, therefore assigning their respe regions equal weights of evidence of 1/1 words, equally credible and reliable (i.e., equivalent in their opinions).**
- **This evidence body can be used to estim likelihood that the epicenter is inside par of interest representing two cities or den populated areas A and B**

■ Evidence theory

➢ **Example 3-1: Assessing the potential consequences of an earthquake**



Estimates of the locations of the epicenter of an earthquake

1. The belief that the epicenter is in region A i

$$Bel(A) = 2/15 = 0.13$$

2. The plausibility that the epicenter is in regio

$$Pl(A) = 5/15 = 0.33$$

3. Similarly for region B, the belief and plausib

$$Bel(B) = 1/15 = 0.07 \; ; \; Pl(B) = 3/1$$

4. These belief and plausibility values can be u the following interval valued estimates of resp probabilities of A and B:

$$0.13 \le P(A) \le 0.33; \quad 0.07 \le P(B) \le$$

Uncertainty modeling

■ Bayesian theory

➢ **Example 3-2: Calculating the probability of a factory producing de components.**

How can you determine th defect rate of a batch of pro when you know the defect r each production line?

Uncertainty modeling

Bayesian theory

- **It is common in engineering to encounter problems with both objective and subjective types of information. Both types of information can be used to obtain solutions or make decisions.**
- **The subjective probabilities are assumed to constitute a prior knowledge about a parameter, with gained objective information (or probabilities).**

If $A_1, A_2, \cdots, A_N$ represents the pri information, or a partition of a unive $E \in X$ represents the objective ability

$$P(E) = P(A_1)P(E|A_1) + P(A_2)P(E|A_2) +$$

where $\mathrm{P}(A_i) =$ the probability of the $=$ the occurrence of E given A_i, wh

Bayes' theorem is based on the sa of partitioning. The posterior probab computed as follows:

$$P(A_i|E) = \frac{P(A_1)P(E|A_i)}{P(A_1)P(E|A_1) + P(A_2)P(E|A_2) + \cdots +}$$

Uncertainty modeling

Bayesian theory

- **Example 3-2: A factory has three production lines. The three lines manufactur and 50% of the components produced by the factory, respectively. The qualit department of the factory determined that the probability of having defectiv from lines 1, 2, and 3 are 0.1, 0.1, and 0.2, respectively.**

1. The following events were defined:

$$L_1 = \text{Component produced by line 1}$$
$$L_2 = \text{Component produced by line 2}$$
$$L_3 = \text{Component produced by line 3}$$
$$D = \text{Defective component}$$

2. The following probabilities are given:

$$P(D \mid L_1) = 0.1;$$
$$P(D \mid L_2) = 0.1;$$
$$P(D \mid L_3) = 0.2$$

3. Since these events are not independen probabilities can be determined as follow

$$P(D \cap L_1) = P(D \mid L_1)P(L_1) = 0.1(0.2) = 0.$$
$$P(D \cap L_2) = P(D \mid L_2)P(L_2) = 0.1(0.3) = 0$$
$$P(D \cap L_3) = P(D \mid L_3)P(L_3) = 0.2(0.5) = 0$$

4. The probability of a defective compon

$$P(D) = P(D \mid L_1)P(L_1) + P(D \mid L_2)P(L_2) + P$$
$$= 0.1(0.2) + 0.1(0.3) + 0.2(0.5) = 0.02$$
$$= 0.15$$

Markov Process Modeling

- **Example 3-3: A particle moves along the real axis in one step from any in... valued coordinate i either to i+1 or to M with equal probabilities. The single... are independent of each other.**

What is the absolute distribution of the position... the particle after 2 steps ?

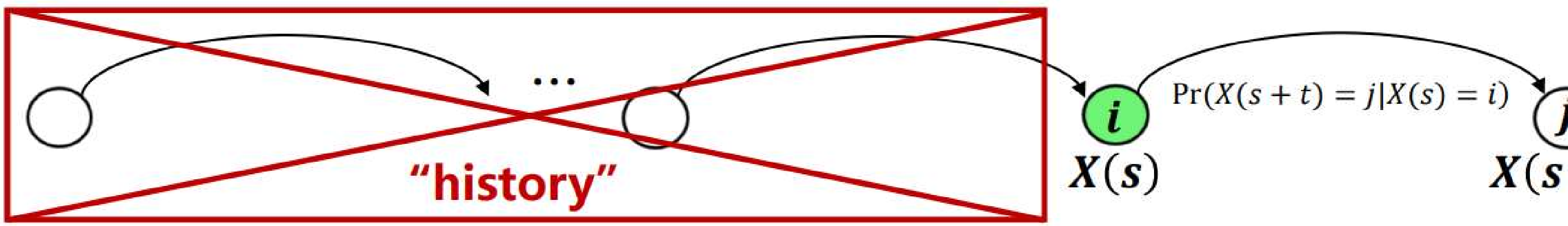
Uncertainty modeling

■ Markov Process Modeling

- **Definition 1: A process is said to have Markov property if**

$$\boxed{\Pr(X(s+t)=j|X(s)=i, X(u)=x(u), 0\le u<s)} = \boxed{\Pr(X(s+t)=j|X(s) =}$$

for all $x(u), 0 \le u < s$



The future development of a discrete Markov chain depends only on its pr
and not on the sequence of previous states.

Markov Process Modeling

- **A Markov chain is called homogeneous if it has homogeneous (stationary) increments.**

The one-step transition probabilities from state i to state j of a homogeneous Markov chain do not depend on n:

$$p_{ij}(n) = p_{ij}, \quad \text{for all}, \quad n = 0, 1, \ldots$$

The one-step transition probabilitie combined in the matrix of the one- probabilities (shortly: transition ma

$$\mathbf{P} = \begin{pmatrix} p_{00} & \cdots & p_{0j} \\ \vdots & \ddots & \vdots \\ p_{i0} & \cdots & p_{ij} \end{pmatrix}$$

- **The m-step transition or probabilities of a Markov chain are defined by**

$$p_{ij}^{(m)} = P(X_{n+m} = j | X_n = i); m = 1, 2, \cdots$$

- **The following relationships bet transition probabilities of a Ma are called Chapman-Kolmogoro**

$$p_{ij}^{(m)} = \sum_{k \in Z} p_{ik}^{(r)} p_{kj}^{(m-r)}; r = 0, 1, \ldots$$

Markov Process Modeling

- Example 3-3: A particle moves along the real axis in one step from any ir valued coordinate i either to i+1 or to M with equal probabilities 1/2. The absorbing barriers at x=0 and x=6, so that if the particle arrives at one of th can-not leave it again. Provided that $0<X_0<5$, the state space c corresponding Markov chain $(X_0, X_1, \ldots)$ is Z=(0,1,...,6) .

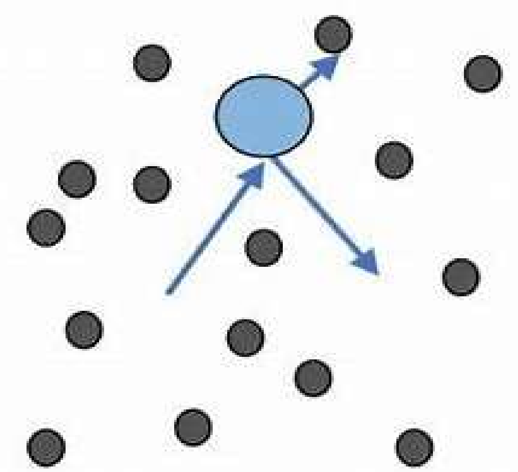


Fig 1. Random Motion. The particles in a gas have a random motion, moving in any direction at any speed.

1. The transition probabilities are

$$p_{ij}=\begin{cases}1/2 & \text{for } j=i+1 \text{ or } j=i-1 \ \text{ and } 1\\ 1 & \text{for } j=i=0 \text{ or } j=i=6\\ 0 & \text{ot}\end{cases}$$

Markov Process Modeling

- Example 3-3: The starting position of the particle X_0 is uniformly distribute {1,2, …,5} and the initial distribution is given by $p_i^{(0)} = P(X_0 = i) = 1/5; i = 1,2,\cdots,$

2. The matrices of the one and two-step transition probabilities are

$$\mathbf{P}^{(1)} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1/2 & 0 & 1/2 & 0 & 0 & 0 & 0 \\ 0 & 1/2 & 0 & 1/2 & 0 & 0 & 0 \\ 0 & 0 & 1/2 & 0 & 1/2 & 0 & 0 \\ 0 & 0 & 0 & 1/2 & 0 & 1/2 & 0 \\ 0 & 0 & 0 & 0 & 1/2 & 0 & 1/2 \\ 0 & 0 & 0 & 0 & 0 & 1/2 & 1 \end{pmatrix} \qquad \mathbf{P}^{(2)} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1/2 & 1/4 & 0 & 1/4 & 0 & 0 & 0 \\ 1/4 & 0 & 1/2 & 0 & 1/4 & 0 & 0 \\ 0 & 1/4 & 0 & 1/2 & 0 & 1/4 & 0 \\ 0 & 0 & 1/4 & 0 & 1/2 & 0 & 1/4 \\ 0 & 0 & 0 & 1/4 & 0 & 1/4 & 1/2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

3. The absolute distribution of the position of the particle after 2 steps is

$$p^{(2)} = \left\{ \frac{3}{20}, \frac{2}{20}, \frac{3}{20}, \frac{3}{20}, \frac{3}{20}, \frac{3}{20} \right\}$$

Content

Case study-System modeling and fault diagnosis

Data description

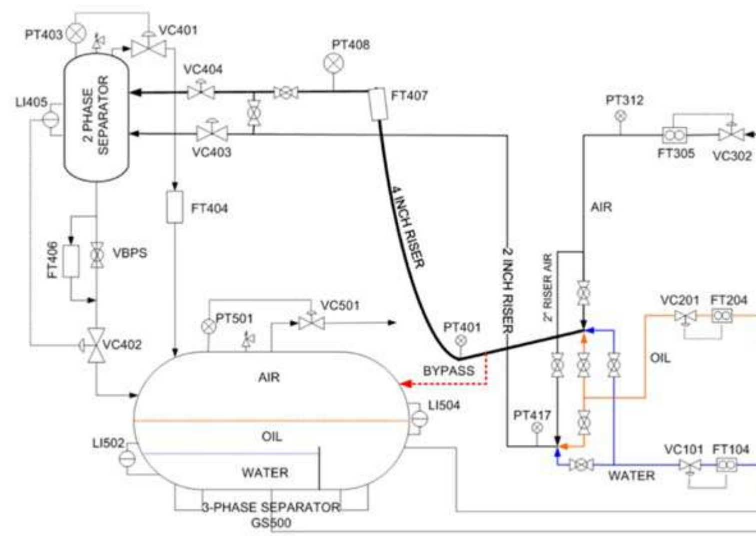
The three-phase flow equipment at Cranfield University aims to provide con measurable flow rates of water, oil, and air for pressurized systems. The industr shown in Figure 1 and can be used to achieve fault diagnosis and health monitori dataset contains 24 variables collected by sensors and six types of faults base operating conditions.

Fault description of three-phase flow process.

Faulty types	0	1	2	3
Description	Normal condition	Air line blockage	Water line blockage	Top separator input blockage

Overview of the working conditions of three-phase flow process.

Domain	Conditions	Class 1	Class 2	Class 3	Class 4
A	Water flow (kg/s)	2	2	2	2
	Air flow (m^3/s)	0.0278	0.0417	0.0278	0.0278
	No. of samples	1000	1000	1000	1000
B	Water flow (kg/s)	3.5	3.5	3.5	3.5
	Air flow (m^3/s)	0.0208	0.0208	0.0417	0.0208
	No. of samples	1000	1000	1000	1000



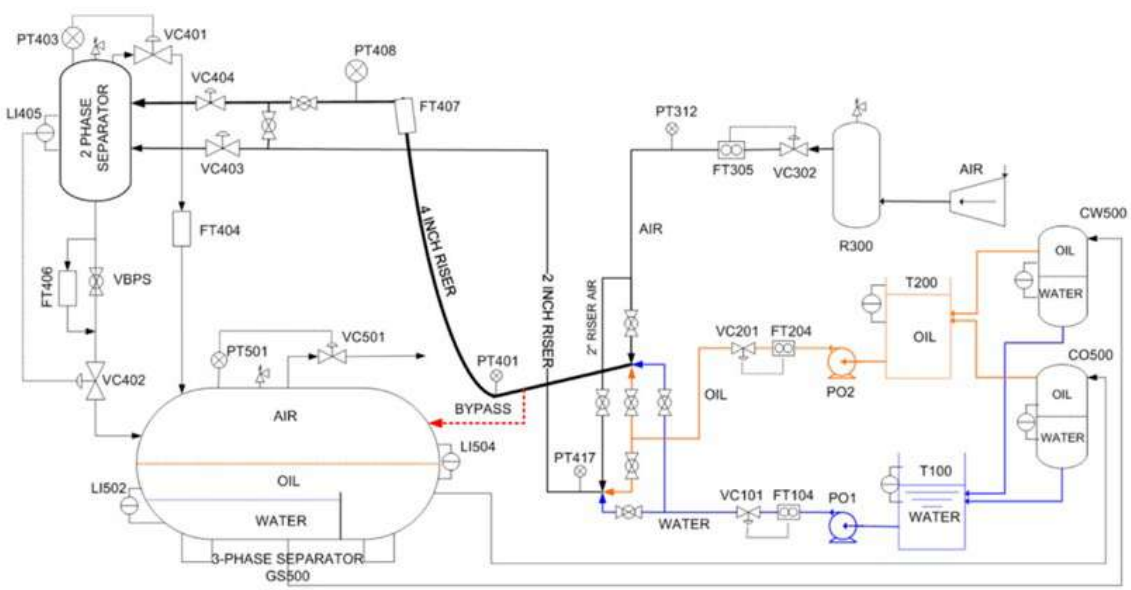
The diagram of three-phase flow in

shen.yin@ntnu.no

Case study-System modeling and fault diagnosis

Data preprocessing

- Data preprocessing can refer to manipulation, filtration or augmentation of data before it is often an important step in the data mining process and improve the quality of dat collection methods are often loosely controlled, resulting in out-of-range values, combinations, and missing values, amongst other issues.



Detailed steps

- **Feature analysis and selection**
  Analyze the relationship between features and select key features.
- **Outliers Removing:**
  Detect outliers in the training set and remo
- **Noise Reduction:**
  Reduce the negative impact of noise on da
- **Normalization:**
  Features are transformed into a range of 0 reduce the impact of data size imbalance c

Case study-System modeling and fault diagnosis

AI-based modeling—Decision tree

Process of decision trees training

- **Information entropy of current sample set D**

$$Ent(D) = -\sum_{k=1}^{m} p_k \log_2 p_k$$

- **Information gain with attribute a**

$$Gain(D,a) = Ent(D) - \sum_{v=1}^{V} \frac{|D^v|}{D} Ent(D^v)$$

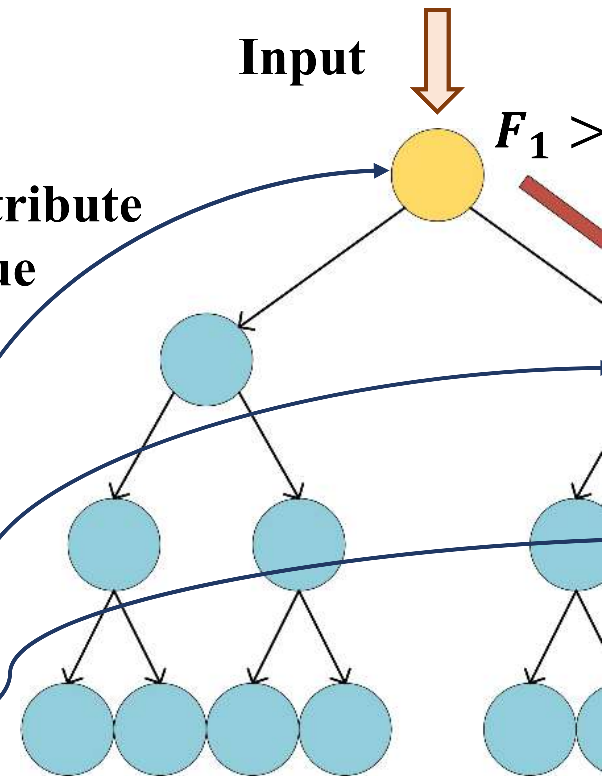
- **Gain ratio calculation**

$$IV(a) = -\sum_{v=1}^{V} \frac{|D^v|}{D} \log_2 \frac{|D^v|}{D}$$

$$Gain_{ratio}(D,a) = \frac{Gain(D,a)}{IV(a)}$$

Guide the attribute and split value

Decision trees model



Predicted faulty types by the tr

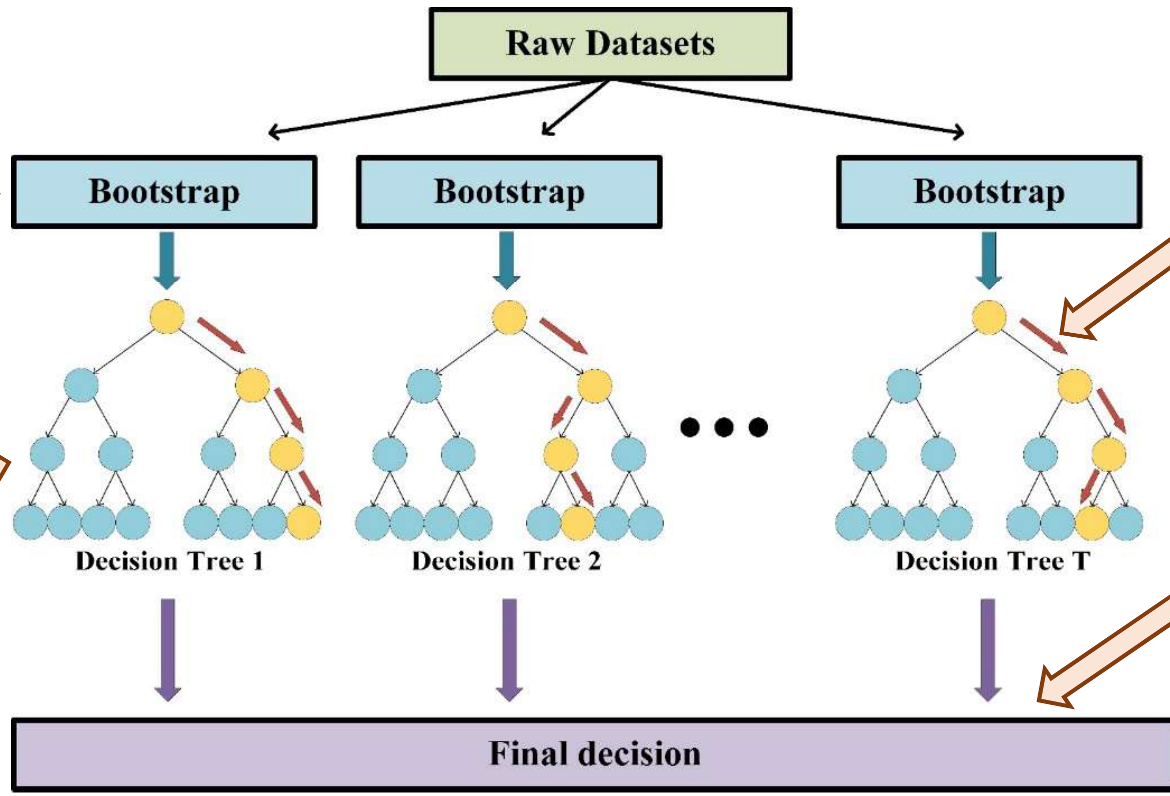
Case study-System modeling and fault diagnosis

AI-based modeling—Random forest

- Random forest is an ensemble learning method for classification, regression and that operates by constructing a multitude of decision trees at training time.

Step 1: Bootstrap sampling to obtain sub datasets

Step 2: Decision tree model training based on sub datasets



Step 3: Ensemble multiple d

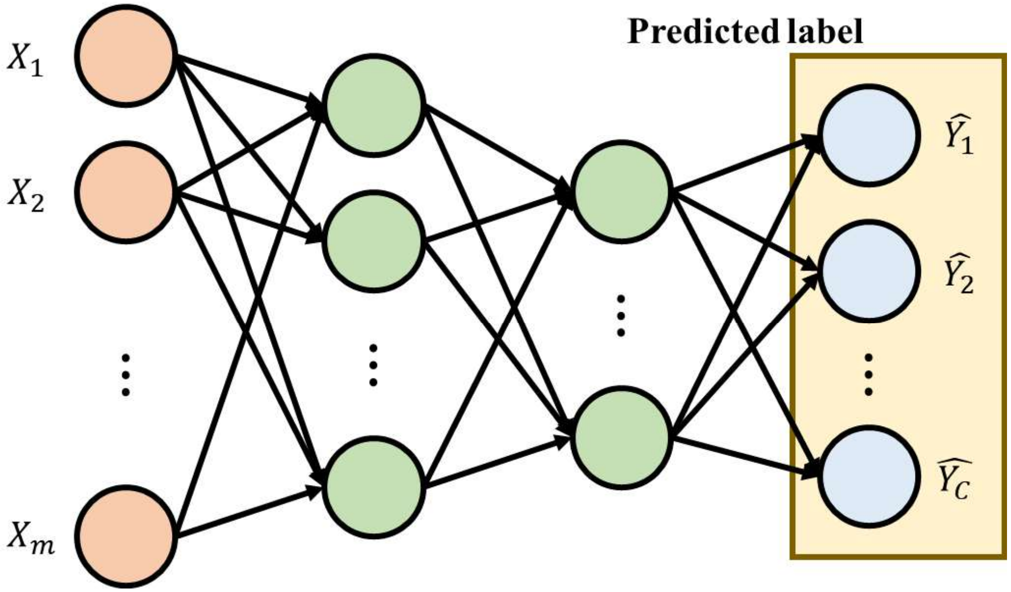
Step 4: Achieve fa

Case study-System modeling and fault diagnosis

AI-based modeling—Neural network

Step 2. Forward calculation

$$Y_{pred} = \sigma_k\left(\cdots \sigma_1\left(W_1 X_{Input} + b_1\right)\cdots\right)$$



Step 3. Gradient descent

$$L = \sum \left\| Y_{pred} - Y_{real} \right\|_2^2 \qquad \Theta_{i+1} = \Theta_i - \lambda \frac{\partial L}{\partial \Theta}$$

- **Main algorithm**
  - Step 1: Data preprocessing
  - Step 2: Forward calculation
  - Step 3: Back propagation (Gr
  - Step 4: Iterative learning on S
  - Step 5: Results validation

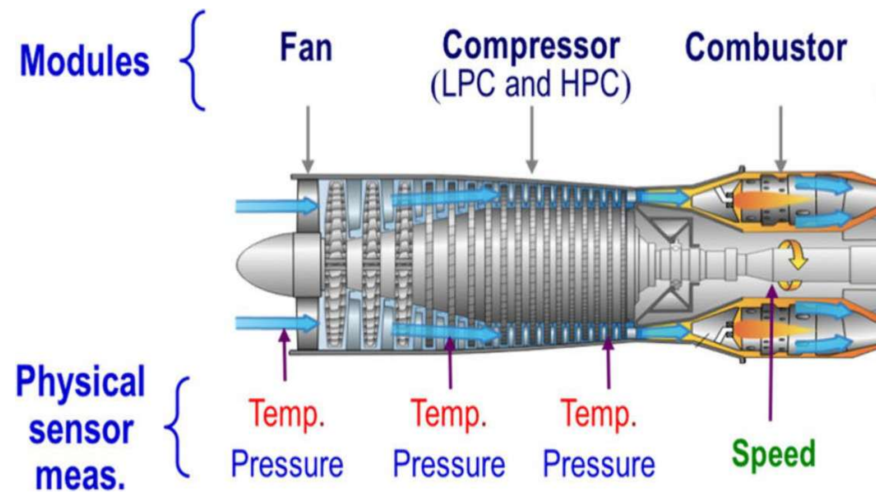
- **Advantages**
  - Strong nonlinear fitting capal
  - Good learning and generaliza
  - Relax the assumption of phys

Case study-System modeling and remaining useful life

Data description

The widely used **commercial modular aero-propulsion system simulation** (C-MAP
adopted for the evaluation of the proposed RUL prediction approach. This data
the degradation process of the aircraft engine. The engine consists of fan,
compressor (LPC), high pressure compressor (HPC), combustor, low pressure turbi
high pressure turbine (HPT). Total **21 onboard sensors**, measuring temperature,
speed, are deployed at different locations to **monitor** the condition of the engine.

Train data	The full life cycle monitoring data of 21 sensors
Test data	The monitoring data of 21 sensors of the partial life cycle, and the corresponding real value of RUL



The diagram of aircraft en

Case study-System modeling and remaining useful life

Data preprocessing

Feature selection

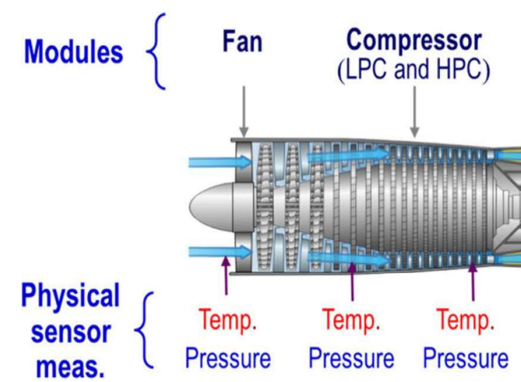
- For the 21 sensors (indices from 1 to 21 in training and testing files), the sensors with indi 18, and 19 always have constant values during the run-to-fail experiments. This means th are not related to the degradation of engines. In this sense, 14 sensors are used for the RU

Time windows acquisition

- Since the data picked up by the sensors are time series data, the data are combined in t windows , which act as the input into the RUL prediction model.

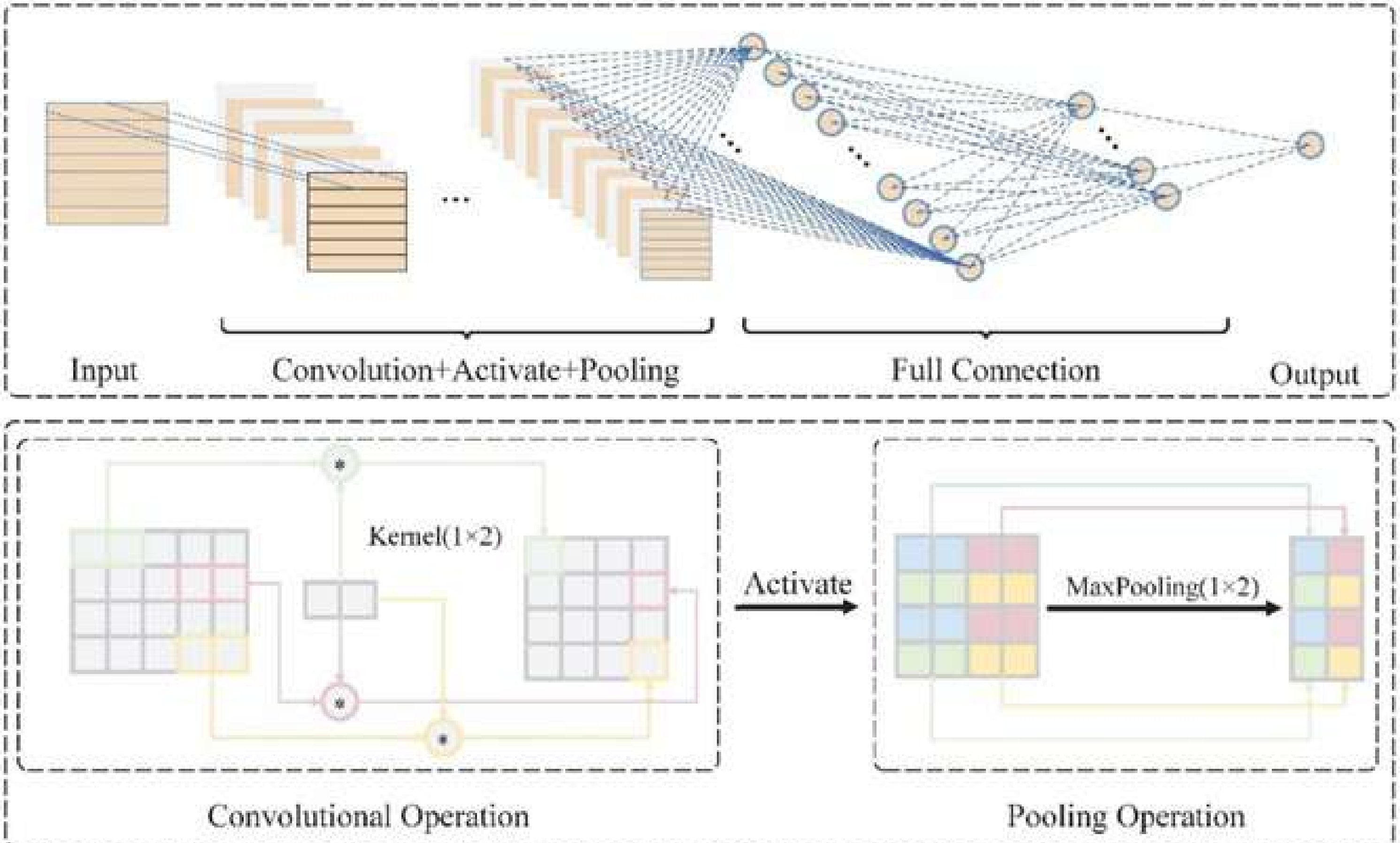
The description of aircraft turbofan engine dataset.

Data	Aircraft turbofan engine datasets			
	FD001	FD002	FD003	FD004
Training engines	100	260	100	249
Testing engines	100	259	100	248
Fault modes	1	1	2	2
Operational conditions	1	6	1	6



Case study-System modeling and remaining useful life

AI-based modeling—Convolutional neural network



Assuming:
Convolution kernel

Input matrix : $X = [$

At the coordinates (
the input matrix, th
output of the
convolution operati
y **shown as follows:**

$$y = f\left(\sum_{p=1}^{t}\sum_{q=1}^{t} k_{pq} \cdot x_{i+p-}\right.$$

where f **is the ReLU**
function, b **is the of**

Advantages

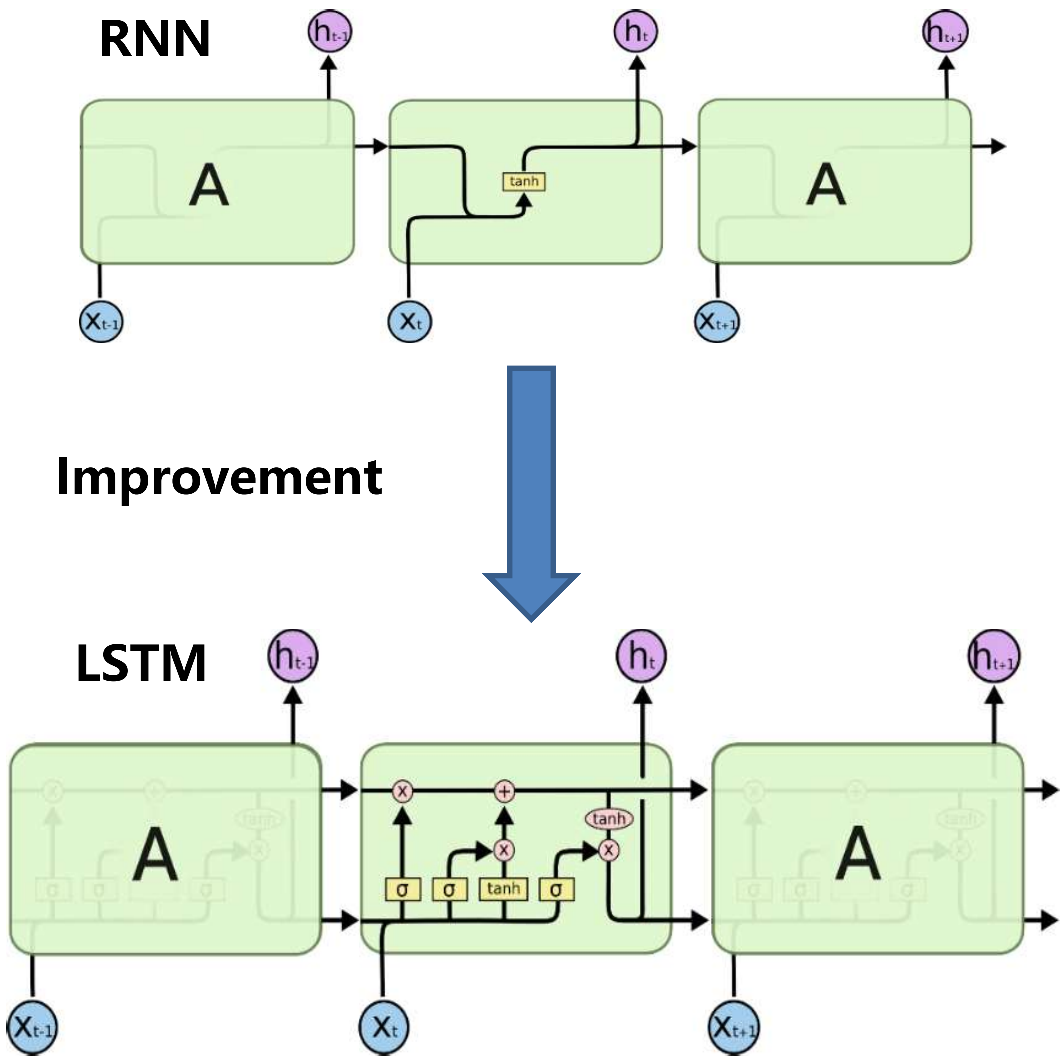
- Automatic feature extraction
- Weight sharing

←

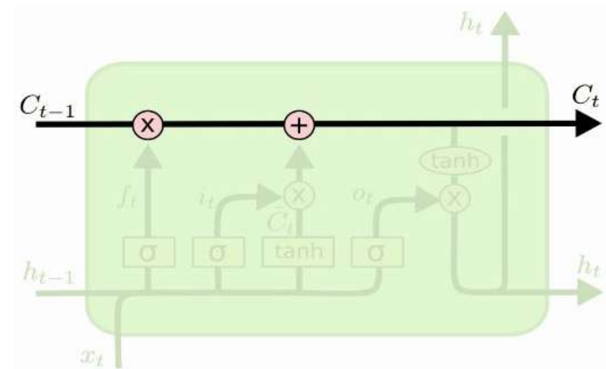
- Convolution operation
- Pooling operation

Case study-System modeling and remaining useful life

■ AI-based modeling—Long–short-term memory network



1. Cell State: Store LSTM cell inforn



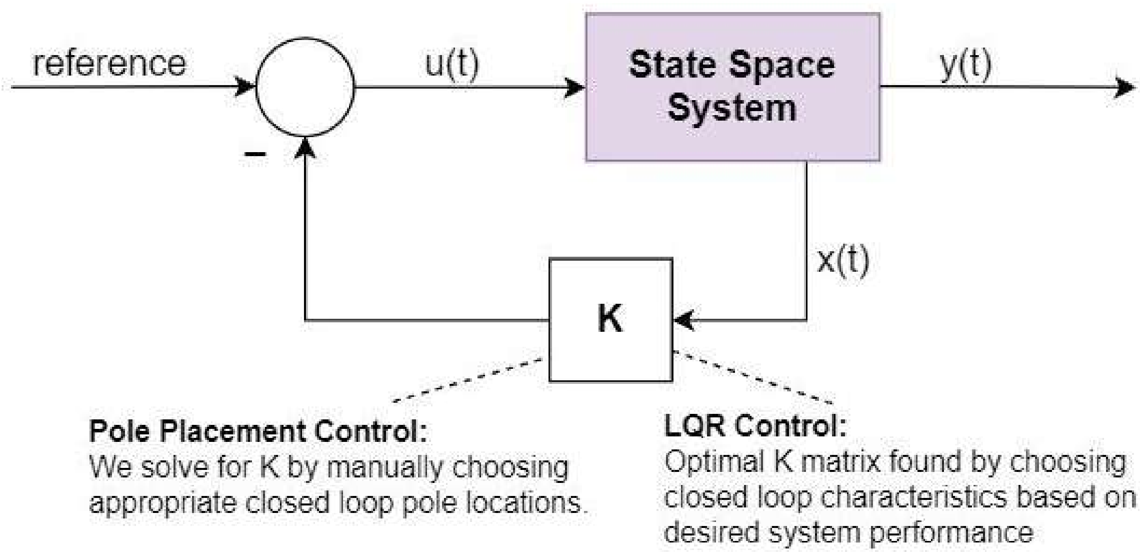
2. Gated structure

■ **How to control the cell state of t**

- □ **LSTM removes or adds cell state inf the gate structure.**
- □ **LSTM contains 3 gate structures to namely, input gate, forget gate and**
- □ **Each gate structure contains a Sigm layer and a bitwise multiplication o**

Case study-System modeling and control

State feedback control



What is state feedback

- State-feedback control is a system design technique t full state of the system to control input.
- The feedback gains are th by solving an optimizatio using various control desi such as pole placement o quadratic regulator (LQR)

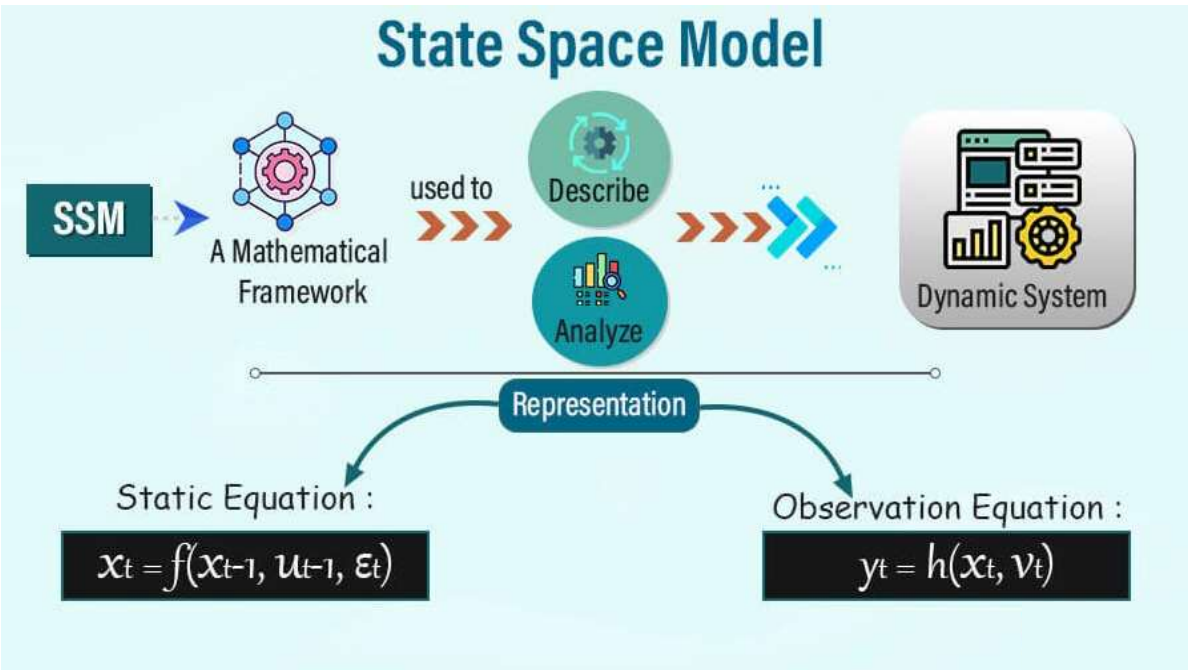
What are the benefits of using state feedback control?

- By using the full state information provided by the system model, the state control system can achieve improved performance, enhanced stability, a disturbance rejection compared to feedback control systems that rely o system output (e.g., PID control).

Case study-System modeling and control

State feedback control

Step1: Obtain the state-space model of the system



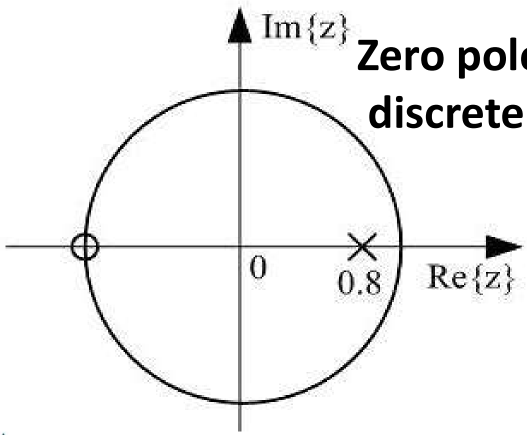
- Identify the state variable variables, and output vari system.
- Derive the state-space equ describe the dynamics of t typically in the form:

  State equation: dx/dt = Ax

  Output equation: y = Cx +

Case study-System modeling and control

State feedback control



Zero pole diagram of discrete systems

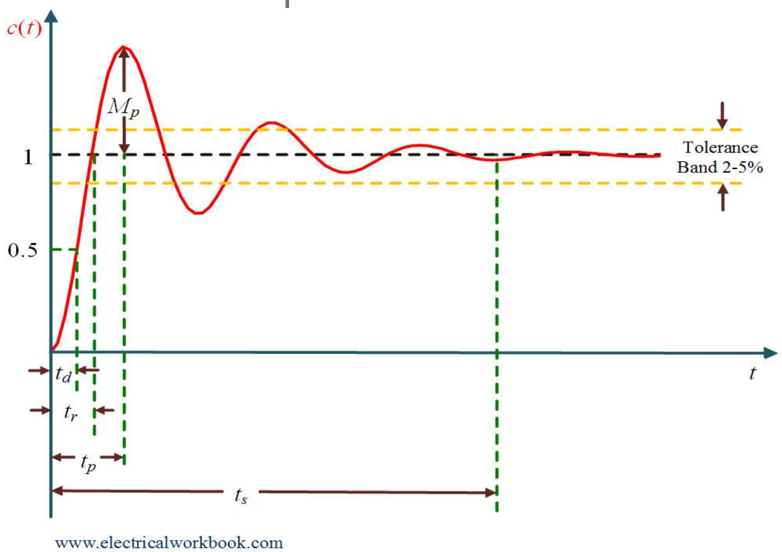


Figure 2 Step response (underdamped case) of a second order control system.

Step2: Determine the desired close system characteristics

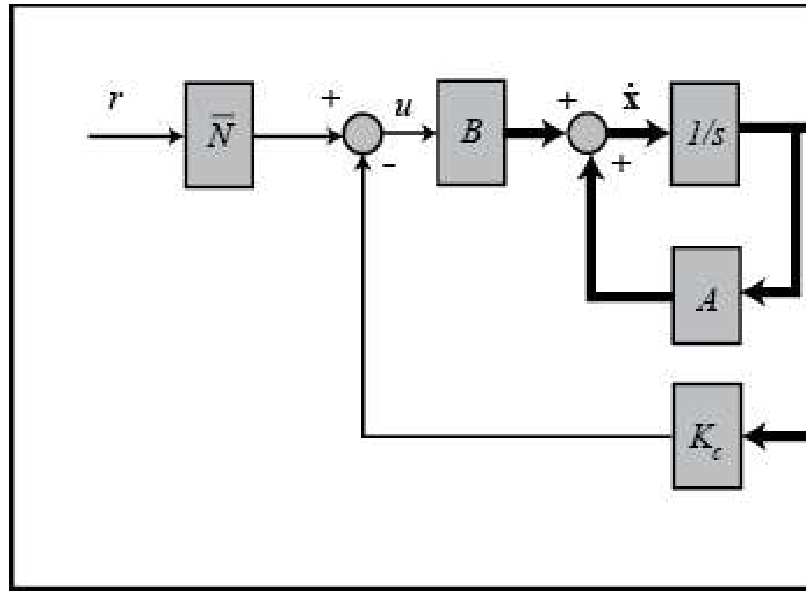
- **Specify the desired closed-loop sys performance requirements, such a stability, transient response, and st state accuracy.**
- **Identify the desired closed-loop sy poles (eigenvalues of the closed-lo system matrix).**

Case study-System modeling and control

State feedback control

Step3: Design the state-feedback control law

- Construct the state-feedback control law in the form: u = -Kx, where K is the state-feedback gain matrix.
- The state-feedback gain matrix K is designed to place the closed-loop system's poles at the desired locations, ensuring the desired closed-loop system characteristics.



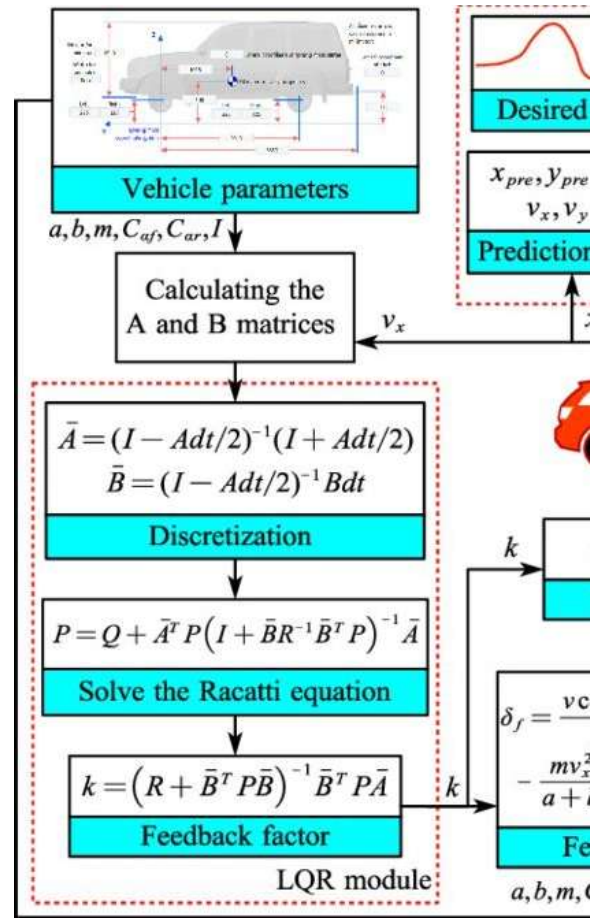
State feedback control bloc

shen.yin@ntnu.no

Case study-System modeling and control

Step4: Compute the state-feedback gain matrix K

- Use pole placement or optimal control techniques to determine the values of the state-feedback gain matrix K.
- Pole placement method: Specify the desired closed-loop poles and solve for the elements of K that place the closed-loop system's poles at the desired locations.
- Optimal control method: Formulate an optimality criterion (e.g., minimizing a performance index) and solve for the state-feedback gain matrix K that optimizes the criterion.



Thanks for your attention!