UNIVERSITÀ DEGLI STUDI DI BERGAMO
Dipartimento di Ingegneria Gestionale, dell'Informazione e della Produzione

Control and Modeling of Biological Systems

L1: Introduction

MASTER DEGREE IN MEDICAL ENGINEERING

TEACHERS
Prof. Antonio Ferramosca
Ing. Beatrice Sonzogni

PLACE
University of Bergamo

Who I am

Name: Antonio Ferramosca

Studies: Ph.D. Engineering (Control Systems) at *University of Seville* (Spain). Master Degree Computer Science Engineering at *University of Pavia*

Research topics: Model Predictive Control (MPC), Economic MPC, Artificial Pancreas

Teaching:
1. Dynamic system identification (6 cfu)
2. Advanced Multivariable Control (6 cfu)
3. Data analysis lab (Technological and Management lab) (3 cfu)

Contact details:

- antonio.ferramosca@unibg.it
- http://www.antonioferramosca.com/
- http://cal.unibg.it/ **CAL research laboratory**
- @ControlAutomationLabUnibg

Course content

Part I: Dynamical systems

1. **Foundations of dynamical systems**
   - 1.1 Movements
   - 1.2 Equilibria
   - 1.3 Stability
   - 1.4 Continuous time VS Discrete time
2. **Transforms**
   - 2.1 Discrete Fourier Transform
   - 2.2 Z-transform and transfer function

Part II: Biological systems

3. **Nonnegative systems**
   - 3.1 Compartmental systems
4. **Typical biological systems**
   - 4.1 Epidemiological systems
   - 4.2 Pharmacokinetics models
   - 4.3. Blood Glucose – Insuline model
   - 4.4 Anesthesia model
5. **Identification**
   - 5.1 Output-error technique
   - 5.2 Validation
6. **Control**
   - 6.1 Classic control strategies

Pre-requirements (strongly suggested)

- Calculus 1
- Fundamentals of linear algebra
- Fundamentals of statistics

Pre-requirements (optional)

- Control and automation
- Calculus 2

Evaluation

- Written exam – 2 hours
- Theoretical open questions (most likley 2) and exercises (most likely 3)
- Materials: https://cal.unibg.it/courses/control-biological-systems/

(Link to the MS Team of the course in there).

Thesis opportunities

- Artificial pancreas control and fault diagnosis, anesthesia control
- Control and data science activities, see the webpage https://cal.unibg.it/theses/

Outline

1. Introduction
2. Static systems
3. Dynamical systems
4. Biological systems

Outline

1. **Introduction**

2. Static systems

3. Dynamical systems

4. Biological systems

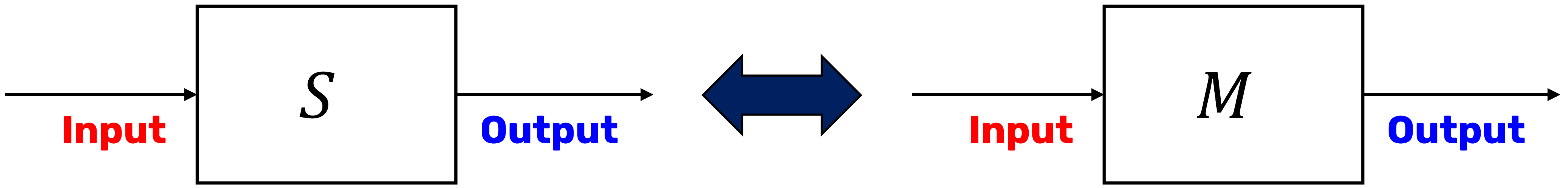
Introduction

In this course we will talk about **mathematical models** for describing natural phenomena or **systems**

System: abstract mechanism that transforms **inputs** (causes) to **outputs** (effects)

Model: mathematical description of a system

- Find a relationship expressed via a mathematical formula for relating the inputs to the outputs, e.g. $V = R \cdot I$ (Ohm's law) or $F = m \cdot a$ (Newtons' 2^{nd} law of dynamics)



Introduction

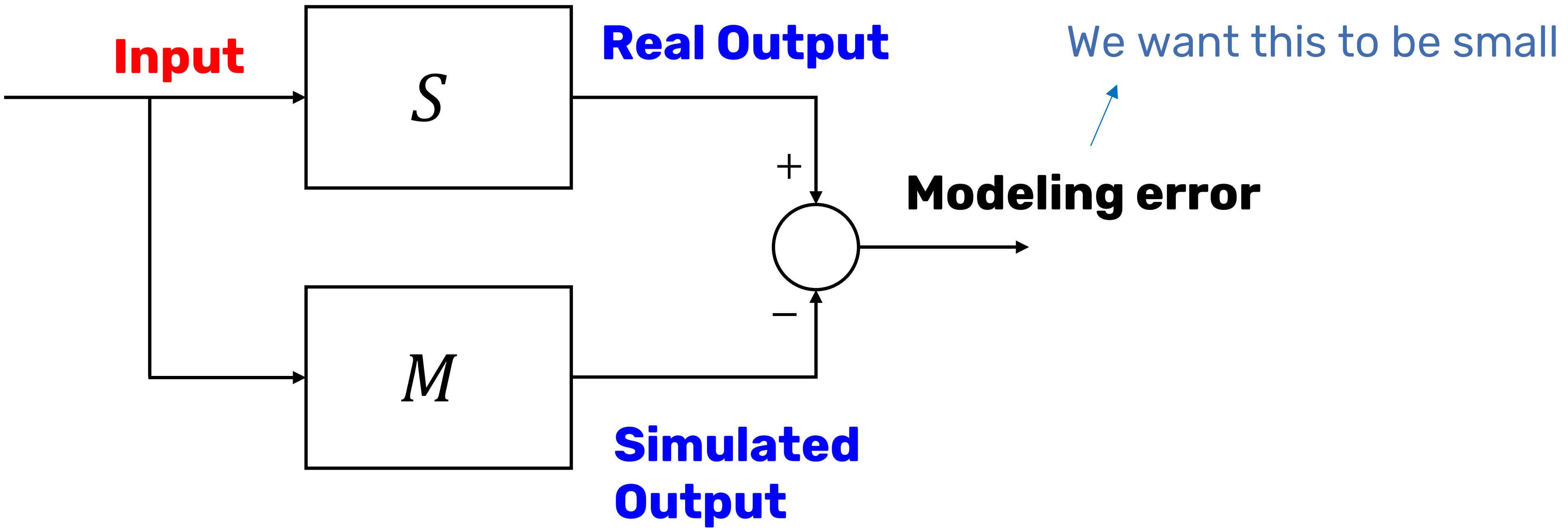
We want a model that is a **good enough representation** of the real system (for our purposes)

To construct a good model for a certain system, we need to have some knowledge on the system (process) we want to model.

One way to assess the goodness of a model is to:

1. Perform an **experiment** on the system. **Measure** its inputs and outputs
2. **Run** the model and get its output, given the measured input
3. **Compare** the real measured output with the model output

Introduction

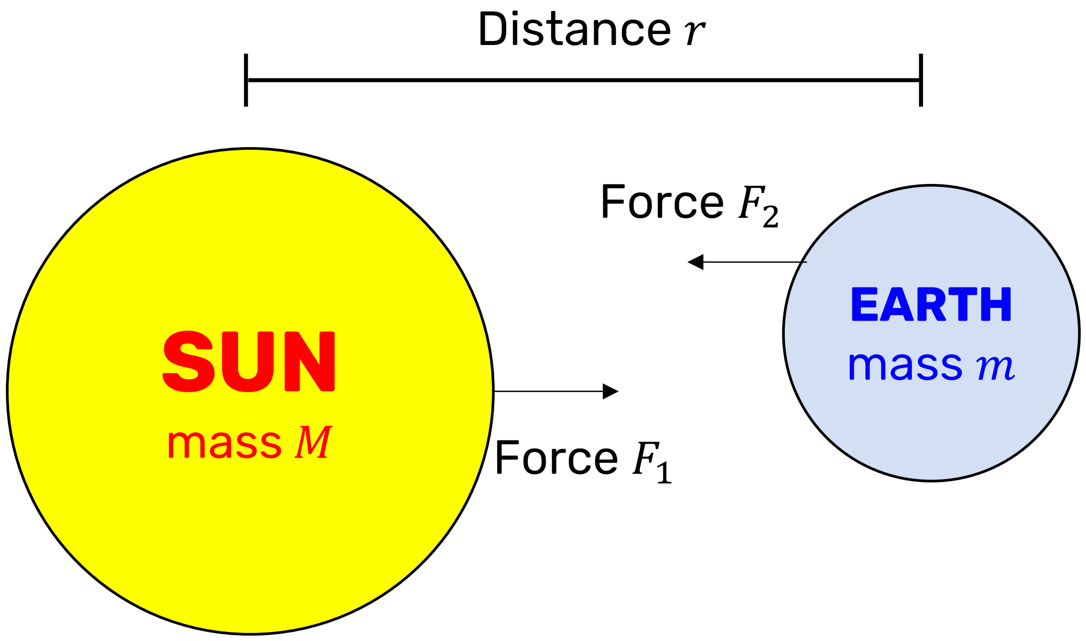


If the real (measured) and simulated (by the model) outputs are similar, then the model is able to **replicate** the real phenomenon. But **why we need models?**

Introduction

A fundamental problem in the sciences is to adequately **interpret a phenomenon** starting from its **experimental observation**

As an example, the birth of modern science corresponds to the discovery of the **universal law of gravitation**, which presents in an **abstract form** the results of a number of **experimental observations** about the motion of celestial objects



$$F_1 = F_2 = G \cdot \frac{M \cdot m}{r^2}$$

"Mathematics is the alphabet with which God has written the Universe" - Galileo

Introduction

From the second half of the nineteenth century the process of **«mathematicalization» of the knowledge** expanded to engineering disciplines (electronics, aeronautics, mechanics, bioengineering,…)

However, in passing from the world of classical physics to these new fields of application, the **phenomena** examined became so **complex** that no simple and universal "fundamental laws", such as that of gravity, can be defined

Engineers work with **uncertain** and **approximate** models, due to the fact that it is not possible to **describe mathematically** all the natural phenomena, the value of some parameter is **not known** accurately, and experiments have **noise** in the measurements

Introduction

Thus, a new discipline was born to **learn** (estimate) **models directly from experimental data**, without relying on fundamental laws of the physics

The applicative contexts of those learning methods are manifold:

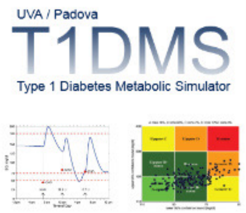
- modeling of **physical components:** electric circuit, electro-mechanical actuators, heat exchangers,…
- modeling of **economics phenomena:** forecasting the sells of a product due to an advertising campaign, study of economic cycles or seasonalities,…
- modeling of **biological phenomena:** cardiovascular system, endocrine system, respiratory system…

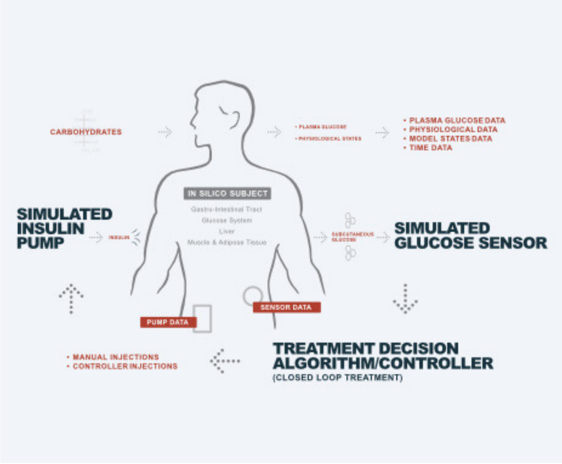
Introduction

All in all, we need a model to **better understand the phenomena** that are of our interest.

<u>Models are useful for:</u>

- **Simulation:** we can *simulate,* with a computer, the response (output) of a model due to certain inputs. By looking at the model response, we understand the behavior of the modeled system





Introduction

All in all, we need a model to **better understand the phenomena** that are of our interest.

Models are useful for:

- **Simulation:** we can *simulate,* with a computer, the response (output) of a model due to certain inputs. By looking at the model response, we understand the behavior of the modeled system

- **Control:** in control engineering (*Automatica* course) a model is used to design a *controller,* i.e. a software that *automatically* defines the system input to obtain a certain reference output

Introduction

All in all, we need a model to **better understand the phenomena** that are of our interest.

Models are useful for:

- **Decision making:** suppose that we are testing a new vaccine. We have two groups of people. We give the vaccine to the first group (test group) and a placebo to the second one (control group). We then measure some variables from patients. How can we choose if the vaccine was effective or not?

- **Communication:** a model allows to communicate to third parties the main information and results of your analysis (do you remember March/April 2020?)

Introduction

Learning models from data is the aim of the discipline called **statistical learning**

Depending on the scientific fields and modeling aims, **different names** were established for basically the **same purpose**:

- **Machine learning:** "machine learning" is the application of statistical learning tools for learning **static models** (i.e. the data do not depend on the **time**)

- **System identification:** the application of statistical learning tools for learning (identify) **dynamical models** (i.e. the data that depend on the **time**)

Outline

1. Introduction

2. **Static systems**

3. Dynamical systems

4. Biological systems

Learning of static and dynamical systems

Statistical learning refers to a vast set of tools for **understanding data.** These methods can be broadly classified as:

- **Supervised learning**: predicting an **output** based on one or more **inputs**

φ → y

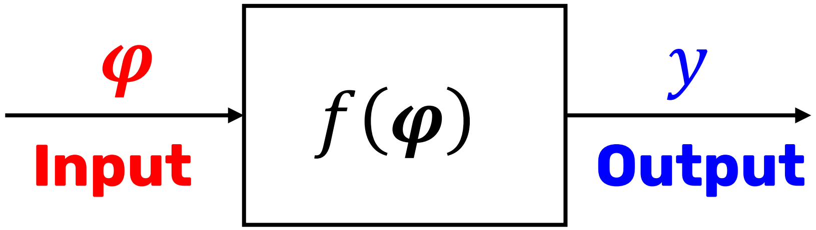
Input → **Output**

Supervised learning learns $\varphi \rightarrow y$, an **input to output mapping**

- **Unsupervised learning**: there is **no output**! The aim is to discover structures and relationships in the **inputs**

Learning of static and dynamical systems

The aim of supervised learning (*for both static and dynamical systems*) is to estimate (learn) a function $f(\boldsymbol{\varphi})$, that maps **inputs** $\boldsymbol{\varphi}$ to **outputs** y, so that $y = f(\boldsymbol{\varphi})$



The **input** is represented as a **vector** $\boldsymbol{\varphi} = [\varphi_0 \;\; \varphi_1 \;\; \cdots \;\; \varphi_{d-1}] \in \mathbb{R}^{d\times 1}$, called **features** or **regressors** vector. Each element $\varphi_0 \;\; \varphi_1 \;\; \cdots \;\; \varphi_{d-1}$ is called a feature or regressor

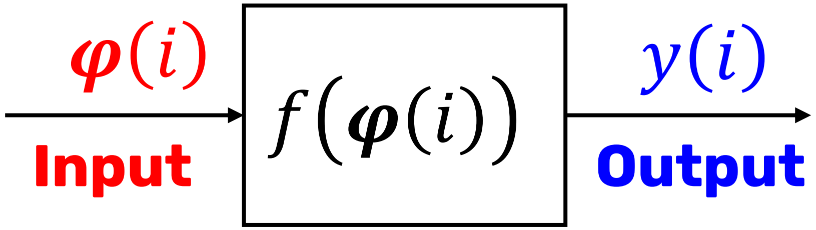
The **output** y can be

- a **number** (**continuous** output), so that $y \in \mathbb{R}$. We talk of a **regression** problem
- a **category** (**discrete** output), so that $y \in \{\text{"Cat. 1"}, \text{"Cat. 2"}, \dots \text{"Cat. } C\text{"}\}$. We talk of a **classification** problem

Learning of static and dynamical systems

In order to learn $f(\cdot)$ from data, we have at disposal a **dataset** $\mathcal{D} = \{\boldsymbol{\varphi}(i), y(i)\}_{i=1}^{N}$ composed by N observations of the quantities $\boldsymbol{\varphi}$ and y

The i-th observation is the couple $\{\boldsymbol{\varphi}(i), y(i)\}$, where $i = 1, \ldots, N$



Our model should **estimate the output** $y(i)$ that corresponds to the input vector $\boldsymbol{\varphi}(i)$

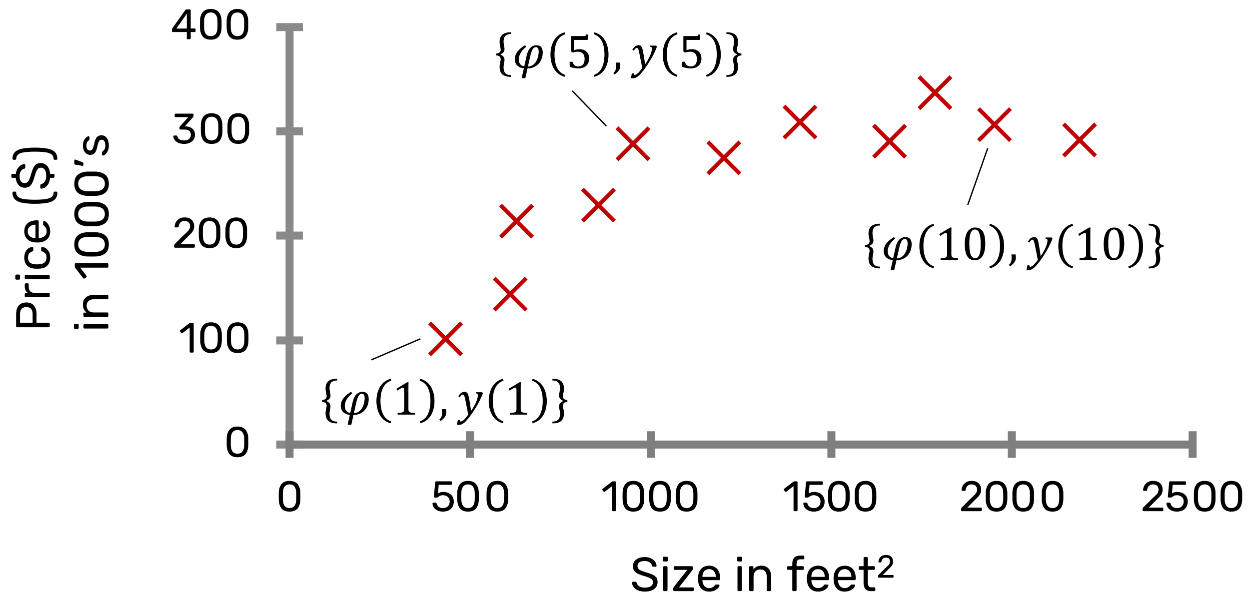
Example: house prices regression

Suppose we want to **predict the price of the houses** in Boston based on their area

We want to **learn the relation** $y = f(\varphi)$ between:

- φ: house size (feature or regressor)
- y: house price (output)

Given the data points



Example: house prices regression

Suppose we have **two regressors**, so that $\boldsymbol{\varphi} = [\varphi_1, \varphi_2\,]^\top \in \mathbb{R}^{2\times 1}$

House area (feet^2)	**# bedrooms**	**Price** $(1000\$)$
523	1	115
645	1	150
708	2	210
1034	3	280
2290	4	355
2545	4	440

- **AIM:** predict house prices
- The output y is **continuous**

Regression

$\varphi \in \mathbb{R}$ $\quad y$ → Learn the relation **from** House area **to** Price

$\boldsymbol{\varphi} \in \mathbb{R}^{2\times 1}$ $\quad y$ → Learn the relation **from** House area **AND** #bedrooms **to** Price

Example: house prices regression

Suppose we have **four regressors**, so that $\boldsymbol{\varphi} = [\varphi_1, \varphi_2, \varphi_3, \varphi_4]^\top \in \mathbb{R}^{4\times 1}$

Single feature φ_3

Output variable y

Size $[\text{feet}^2]$	Number of bedrooms	Number of floors	Age of home [year]	Price [$]
2104	5	1	45	$4.60 \cdot 10^5$
1416	3	2	40	$2.32 \cdot 10^5$
1534	2	1	30	$3.15 \cdot 10^5$
⋮	⋮	⋮	⋮	⋮
↓	↓	↓	↓	↓
φ_1	φ_2	φ_3	φ_4	y

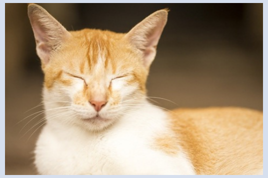
Number of observations N

Single observation (feature vector) $\boldsymbol{\varphi}$

The i-th observation is the vector $\boldsymbol{\varphi}(i) = [\varphi_1(i)\ \ \varphi_2(i)\ \ \varphi_3(i)\ \ \varphi_4(i)]^\top \in \mathbb{R}^{4x1}$

Each feature vector $\boldsymbol{\varphi}(i)$ has associated a response $y(i) \in \mathbb{R}$

Example: image classification

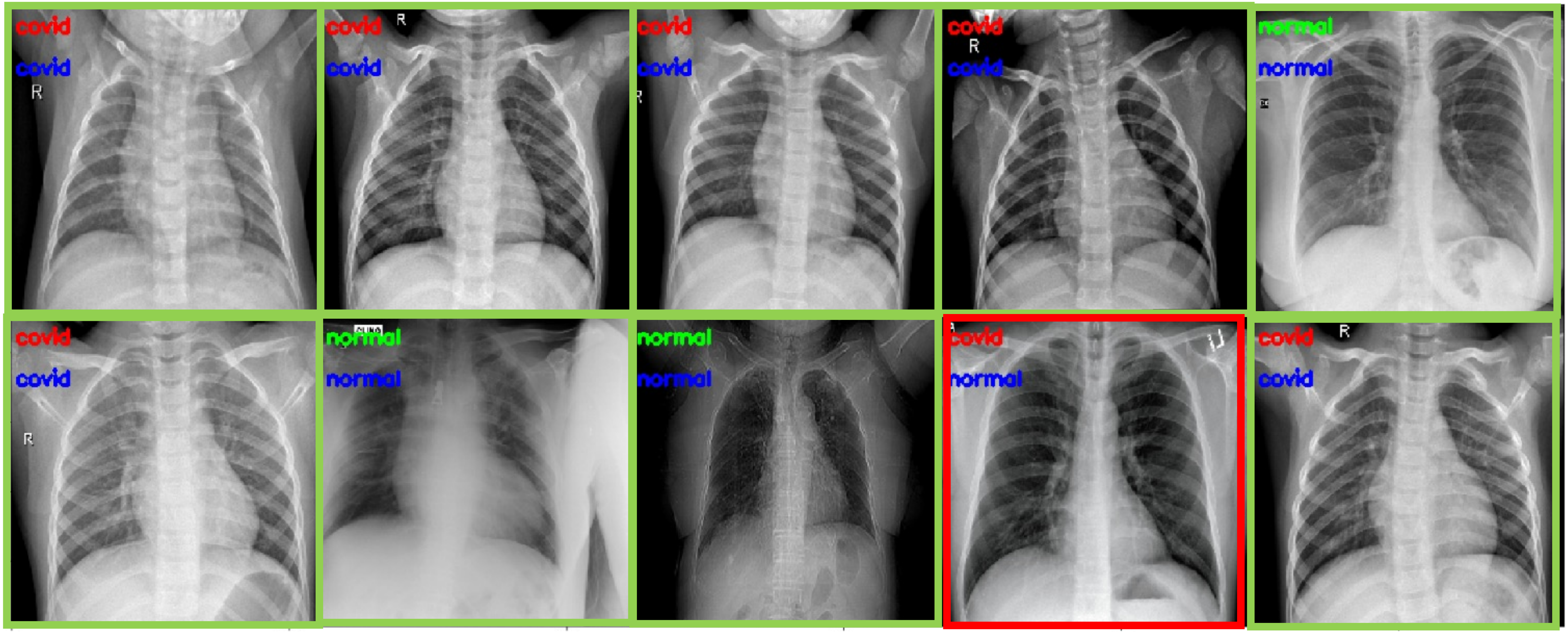
Image	Output label
	Cat
	Not cat
	Cat
	Not cat

- **AIM:** develop an application that recognize cats in images
- Learn the map **from** an image **to** a “membership class”
- The output y is a **category** (Cat or Not cat)

Classification

Example: image classification

Predicted covid label
Predicted healthy label
True label



Example: house prices classification

Suppose that instead of the price value in dollars, we want to classify houses as **expensive** *(class* $y = 1$*)* or **cheap** *(class* $y = 0$*)*

The features are the same, but the output is now a **category** and not a real value (it is always represented as number in the computer, but it is not treated as such by algorithms)

Size $[\text{feet}^2]$	Number of bedrooms	Number of floors	Age of home $[\text{year}]$	Price $[\text{class}]$
2104	5	1	45	1
1416	3	2	40	0
1534	2	1	30	1
⋮	⋮	⋮	⋮	⋮
↓	↓	↓	↓	↓
φ_1	φ_2	φ_3	φ_4	y

Static systems

A system is said to be **static** if its output does not depend on some values of the output itself, i.e. if the features vector $\boldsymbol{\varphi}$ does not contain values of the output as regressors

In static systems, the values of an observation $\{\boldsymbol{\varphi}(i), y(i)\}$ does not depend on the values of another observation $\{\boldsymbol{\varphi}(j), y(j)\}$, with $j \neq i$. We say that the observations are **independent**

It is also commonly assumed that observations have the **same data distribution**. This is very important for reliably learning and evaluating statistical learning models

These data are called **Independent and Identically Distributed** (i.i.d.)

Outline

1. Introduction
2. Static systems
3. **Dynamical systems**
4. Biological systems

Dynamical systems

A system is said to be **dynamical** if its output depend on some values of the output itself, i.e. if the features vector $\boldsymbol{\varphi}$ contains values of the output as regressors

Dynamical models are mathematical models that allow to describe the future evolution of the variables involved as a **function of their past trend** and external variables

Dynamical systems usually involve the **time:** the output $y(t)$ at a certain time t **depends on the output at previous times** $y(t-1), y(t-2), \dots y(t-m)$

This dependence from the past endows the model with a **«memory»** (i.e. the dynamics), of past behaviour

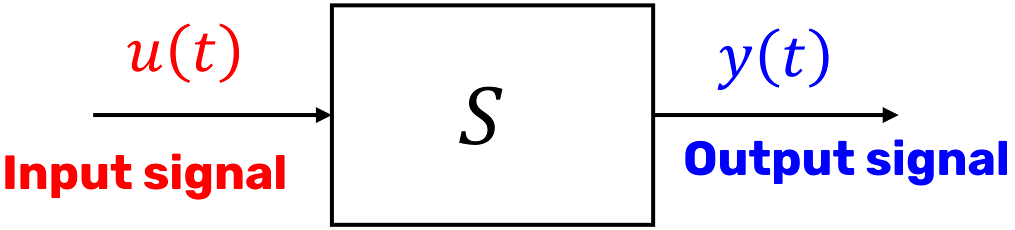
Dynamical systems

Most physical and natural systems are dynamical!

- In an **electromechanical motor**, the relation between the motor current and the motor speed can be described by a dynamical model
- The force generated by a **skeletal muscle** contraction will depend by the viscous damping given by the tissue and on the elastic storage properties by the tendons
- The regulation of the **blood-glucose** level done by the pancreas is dynamical
- The **flow equation** of the blood through the vessels depend on pressures dynamics

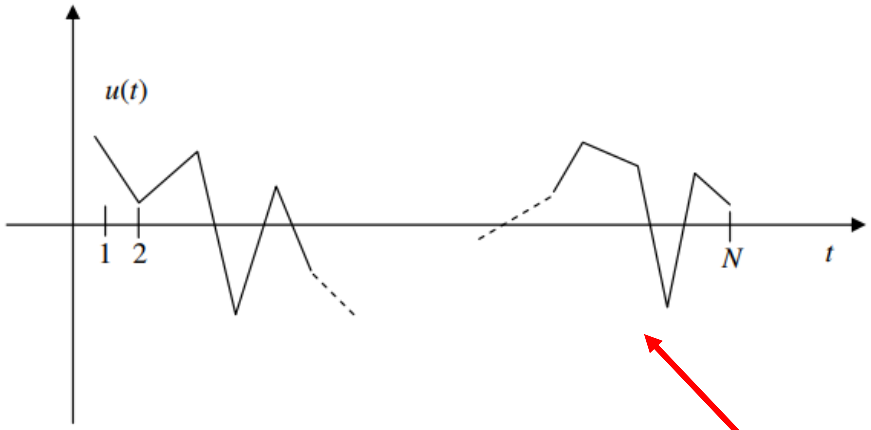
Dynamical systems

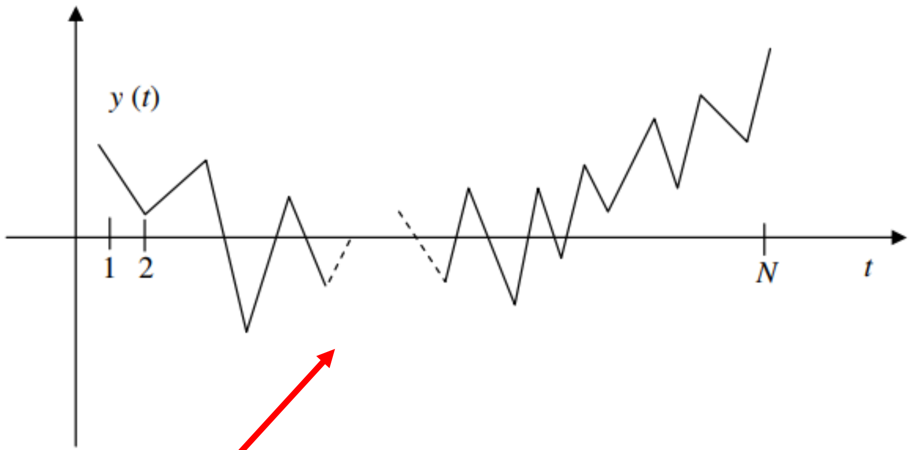
Dynamical systems, due to the presence of the time variable, are used to model relations between **input** and **output signals:**



Two sets of N data are collected, **sampling the signals** at $t = 1,2, \ldots N$ time instants

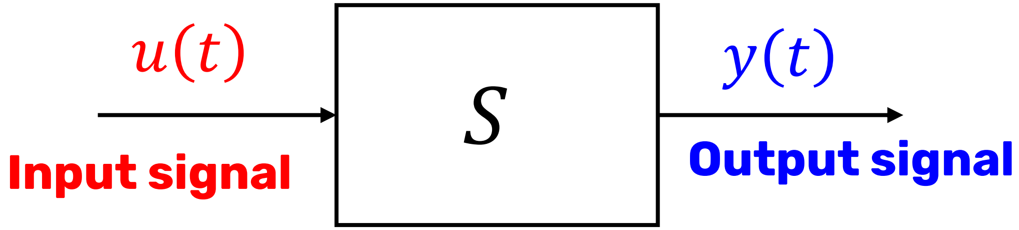
- Input data $\{u(1), u(2), \ldots, u(N)\}$ }
- Output data $\{y(1), y(2), \ldots, y(N)\}$ }





The signals are not random! They show a **temporal correlation**

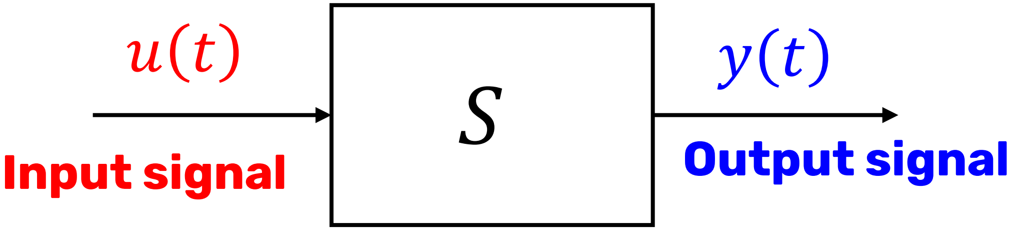
Dynamical systems



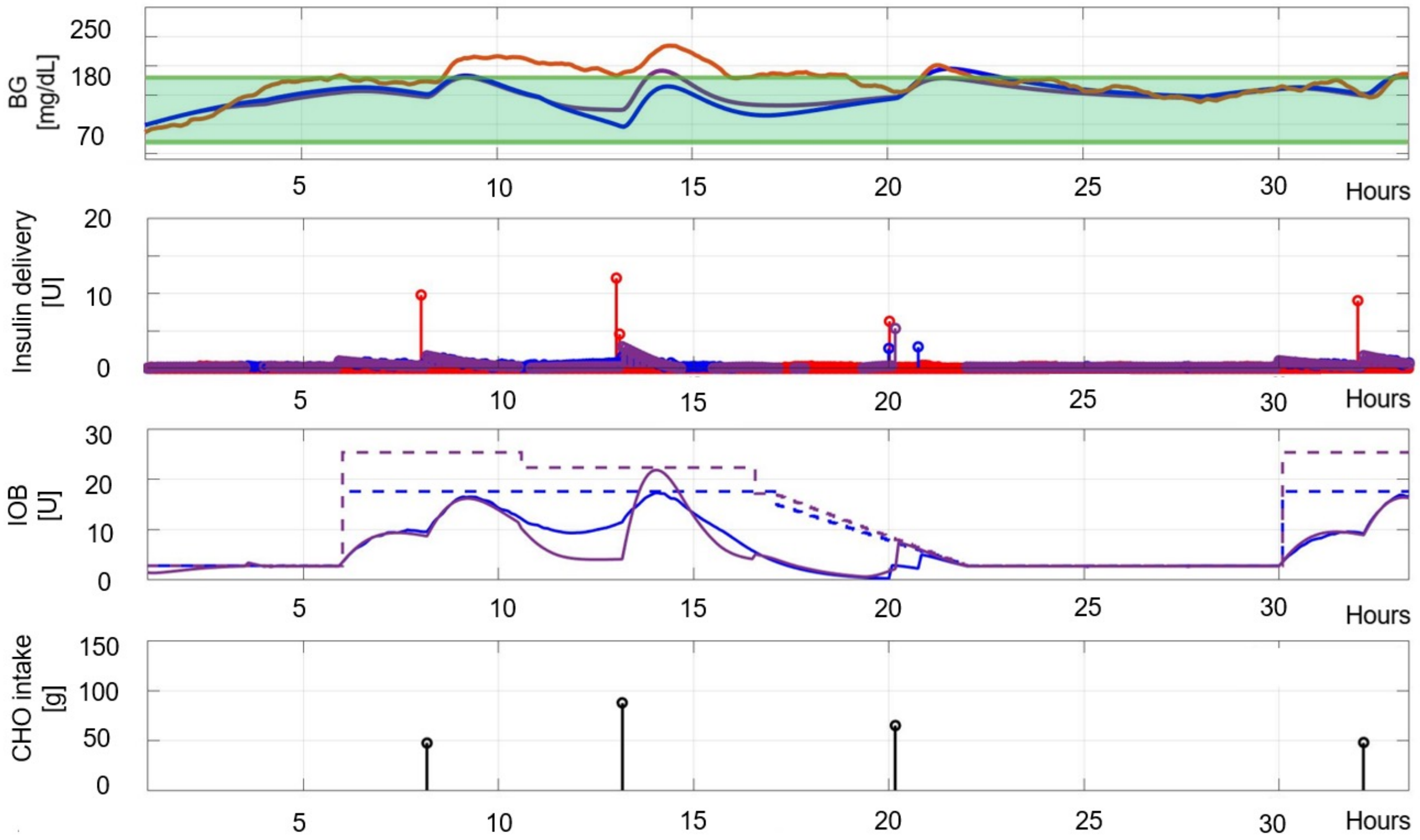
Examples of input/output dynamical systems:

Input (explicative variable)	Output (explained variable)
Audio signal (before transmission)	Audio signal (after transmission)
Current	Motor torque
Medicine amount	Hormone concentration
Mm of rain	Concentration of pollution
Insuline infusion	Blood-glucose level

Dynamical systems



T1 Diabetes patient model:



Output: Blood Glucose

Input: Insulin delivery

CHO Intake

The BG $y(t)$ at a certain time t **depends on its values at previous times**

Dynamical systems

Dynamical systems can be defined in **continuous-time** or **discrete-time**

Natural and physical phenomena are inherently continuous

- In this case, the system is described through **differential equations**, like

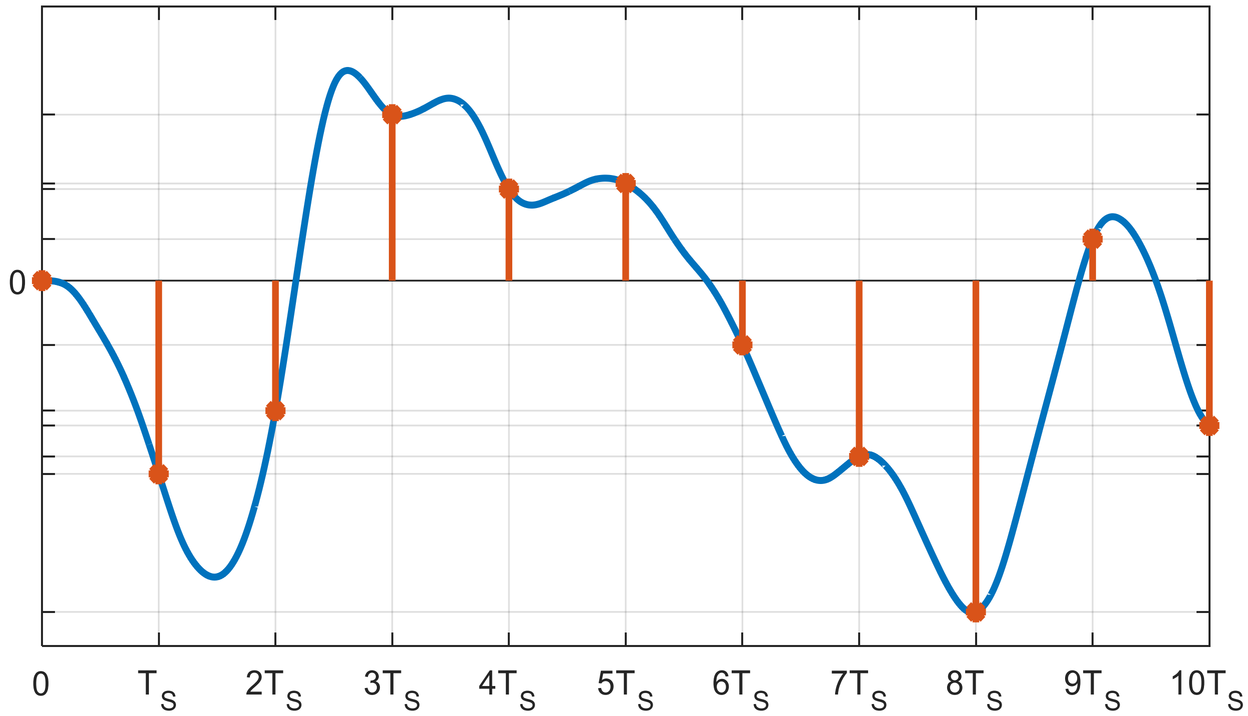
$$\frac{dy}{dt} = \dot{y}(t) = -2 \cdot y(t) + 3 \cdot u(t)$$

The derivative is the «mathematical representetation of the **future behaviour** of a function». This notion of «future» is exactly what we need to represent the **memory** of a continuous-time dynamical system

Dynamical systems

However, the computer can only manage a **finite amount of data**. Thus, signals should be **sampled** with a sampling time T_s, such that we store a finite amount of data at **discrete times** $t \cdot T_s$, with $t = 1, \dots, N$

$$y(t) = y(t \cdot T_s)$$



In the following, we will use $y(t)$ with the meaning of $y(t \cdot T_s)$

Dynamical systems

The evolution of **discrete-time signals** (sampled from **continuous-time** ones) can be described by **discrete-time systems**:

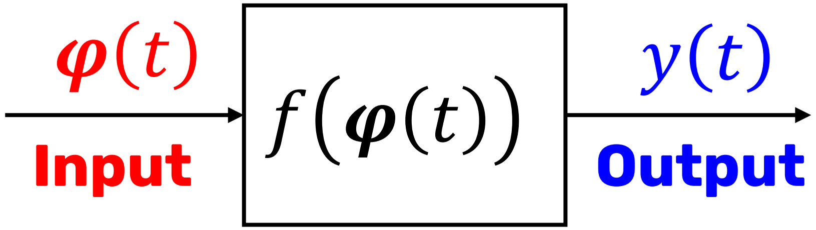
- instead of a differential equation, we have a **difference equation**

$$y(t) = -0.5 \cdot y(t-1) + 3 \cdot u(t)$$

With the difference equation, it is very clear that $y(t)$ **depends on its previous values** (and also on the input $u(t)$)

Dynamical systems

For the **purpose of learning** dynamical systems, we will cast the learning problem **just like for static systems**

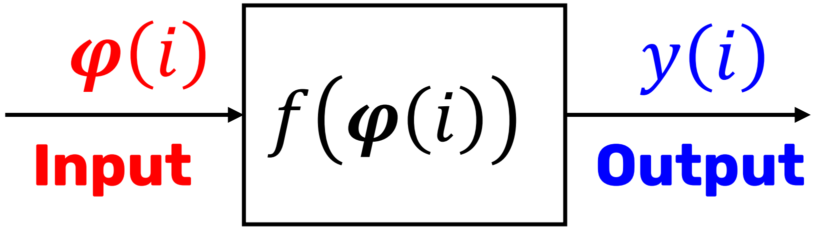


The only difference is how the **regressors vector** φ will be defined. Since the output depends on the input and output signals $u(t)$ and $y(t)$, the regressors vector $\boldsymbol{\varphi}(t)$ **at certain time** t will look like:

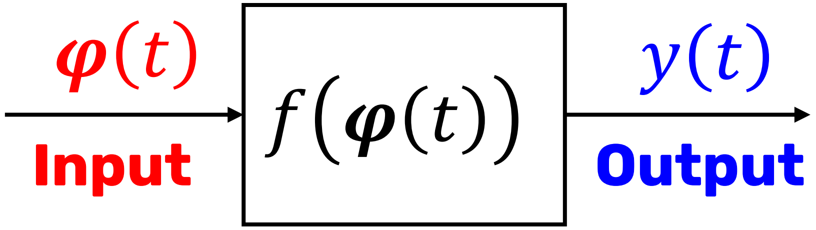
$$\boldsymbol{\varphi}(t) = [y(t-1) \cdots y(t-m) \;\; u(t) \;\cdots u(t-p)]^\top$$

Dynamical systems

Static systems



Dynamical systems



- With **static systems**, we will index the observations with the index i
- With **dynamical systems**, we will index the observations with the index t

In either cases, both model will be **to learn $f(\cdot)$ from data**

- In the dynamical case, we will talk of **«identification»**. It is a synonym of **«learn»**

Outline

Biological systems modelling

The aim of this course will be to derive mathematical model to describe in a qualitative and quantitative way certain phisiological/bio processes.

This modelling help to better understand a certain phenomenon and to give a formal description of it.

Biological model can help to understand the behaviour of certain phisical quantities that cannot be directly measured, (ex: insuline time evolution).

Models also help to design specific experiments to better characterize a certain phenomenon.

Biological systems modelling

Models can be **simulation tools** to study, for example, the effects on variables of interest, of changes in external signals and/or parts of the model (e.g. to distinguish physiological situations from pathological ones)

They can be used as **diagnostic tools** to distinguish different pathologies, different severities of the same pathology or between healthy subjects and subjects suffering from pathologies

Biological models are also particularly useful to determine a correct **therapeutic protocol** (administration schedule, dosages, etc.)

Biological systems modelling

To model a certain biological process the first thing to do is to define the purpose and objectives of the model itself.

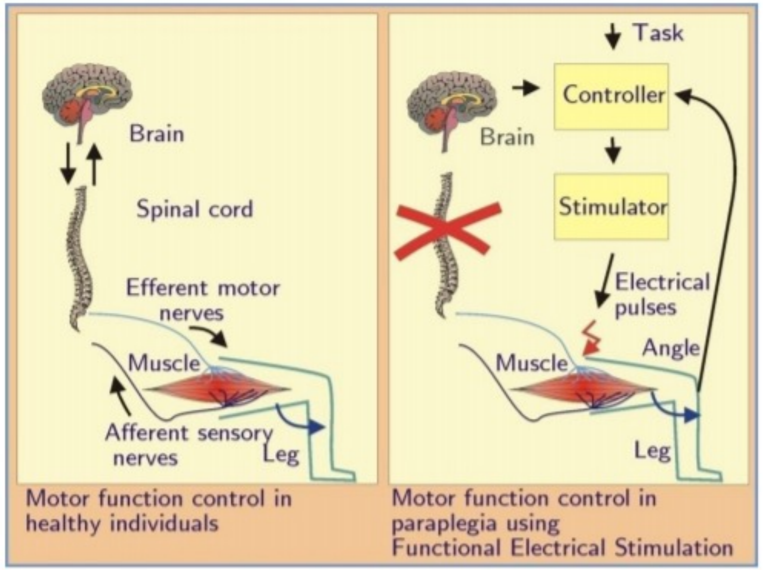
Therefore, the problem in general is not to define the correct model of a system, but the most useful model for the intended purpose.

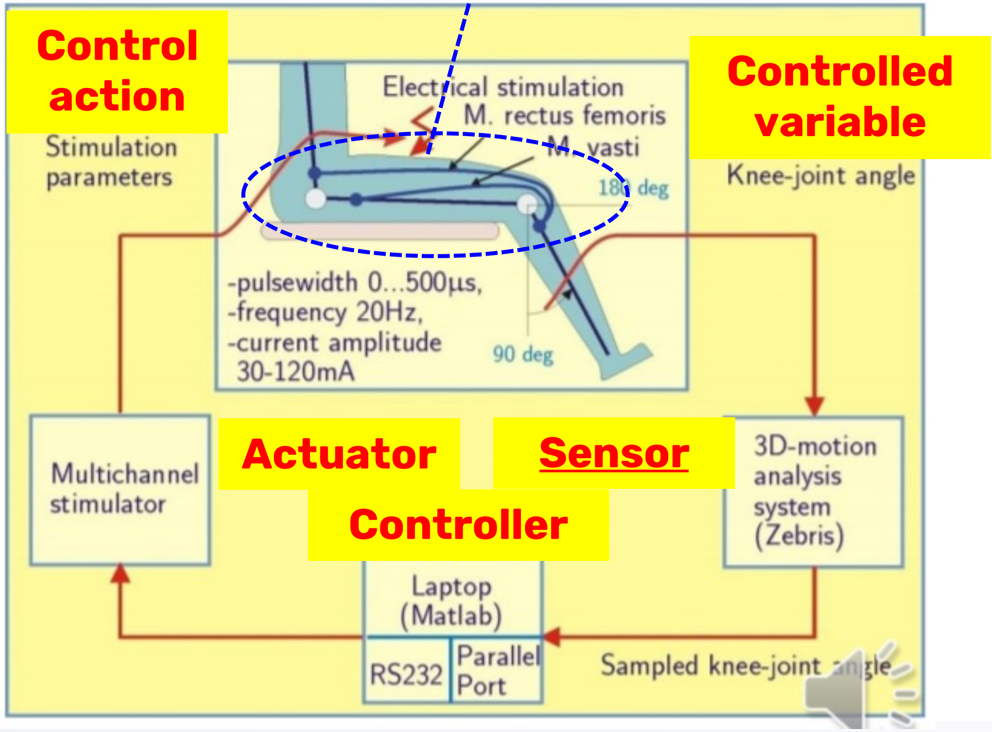
There is no "universal" model of a system that is valid in every context. On the contrary, according to a famous phrase that sums up the "belief" of all modelers, **all models are wrong, some are useful.**

Example: functional electrical stimulation

Purpose: Rehabilitation of paraplegic subjects

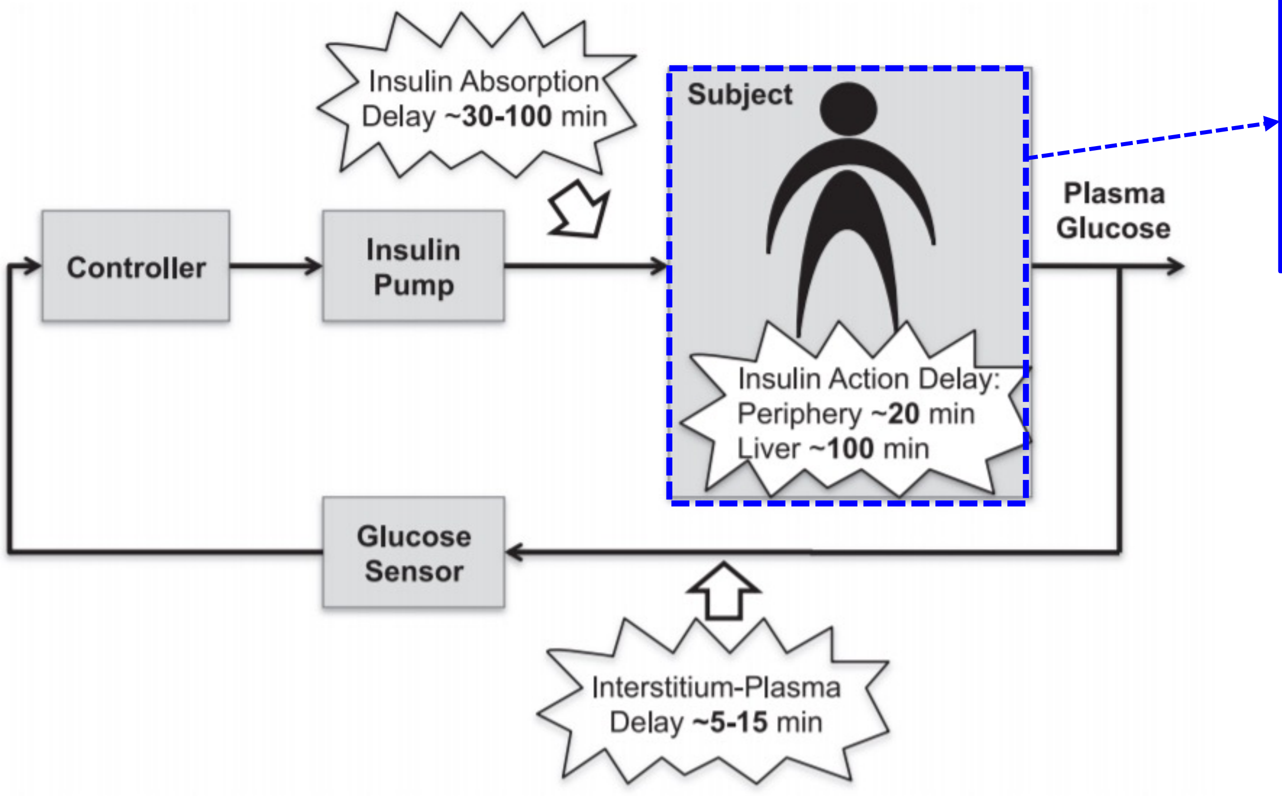
A **dynamical model of the muscles response** to electrical stimulations is estimated to design the control algorithm





Example: artificial pancreas

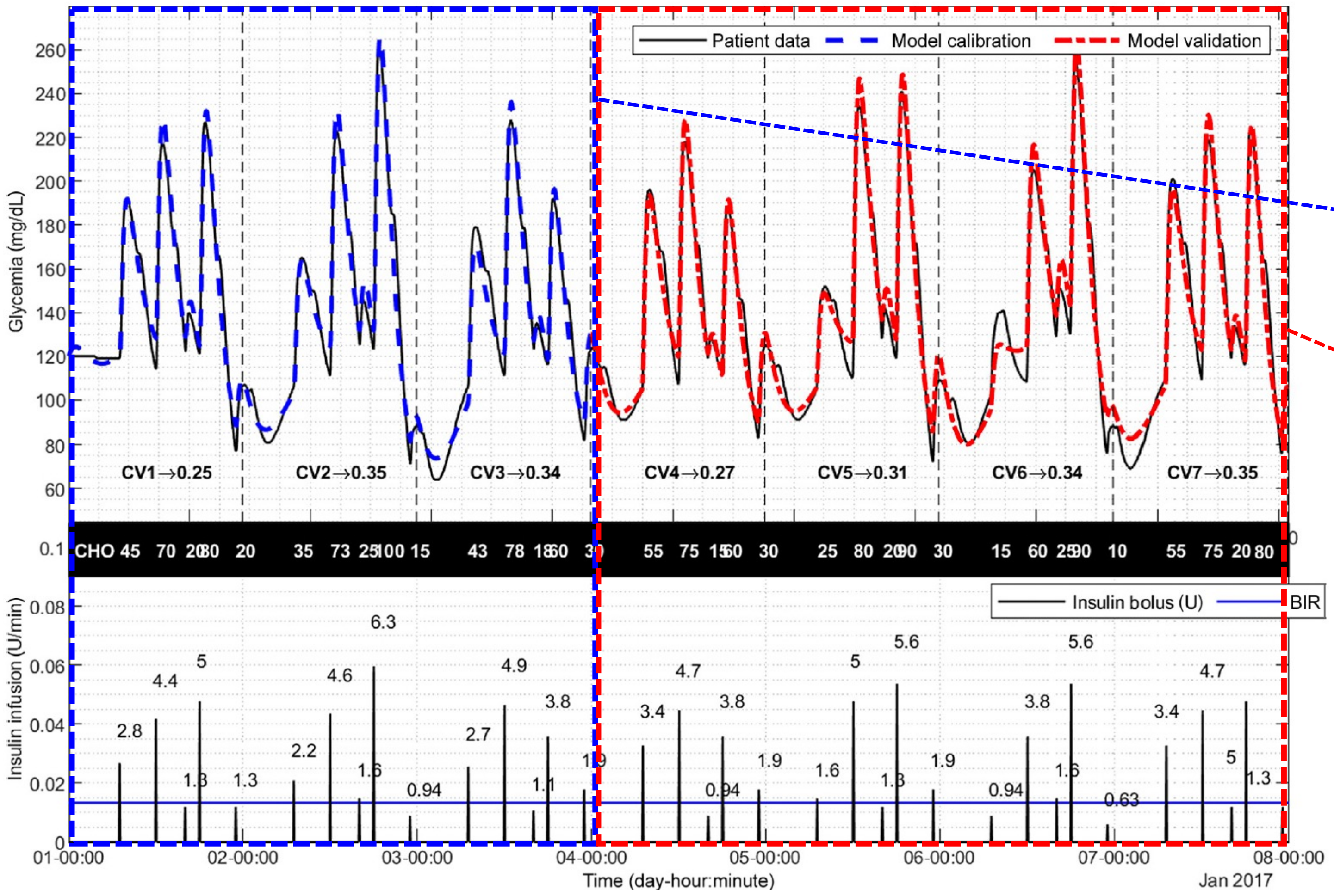
Purpose: semi-automatic insulin regulation



A **dynamical model of the patients response to** insulin is estimated to design the control algorithm

Example: artificial pancreas

Model Identification (child): training vs validation in a 7 days test

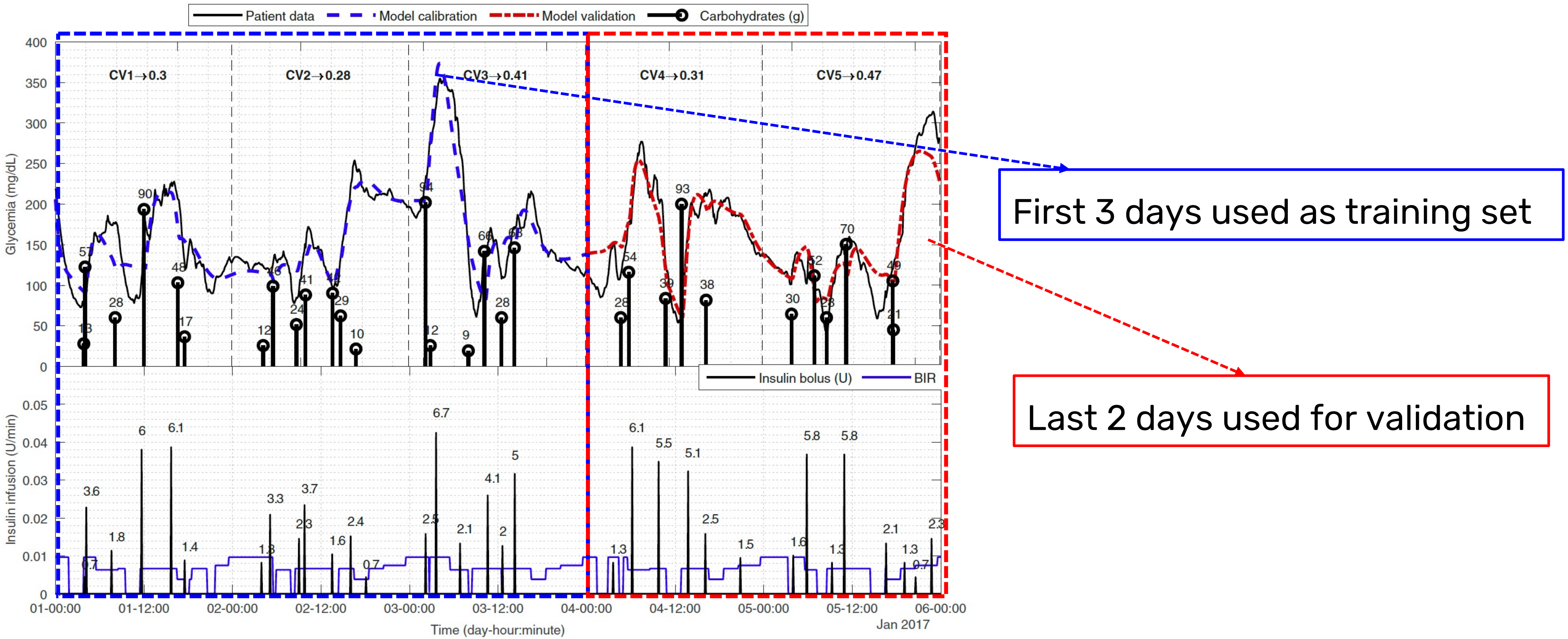


First 3 days used as training set

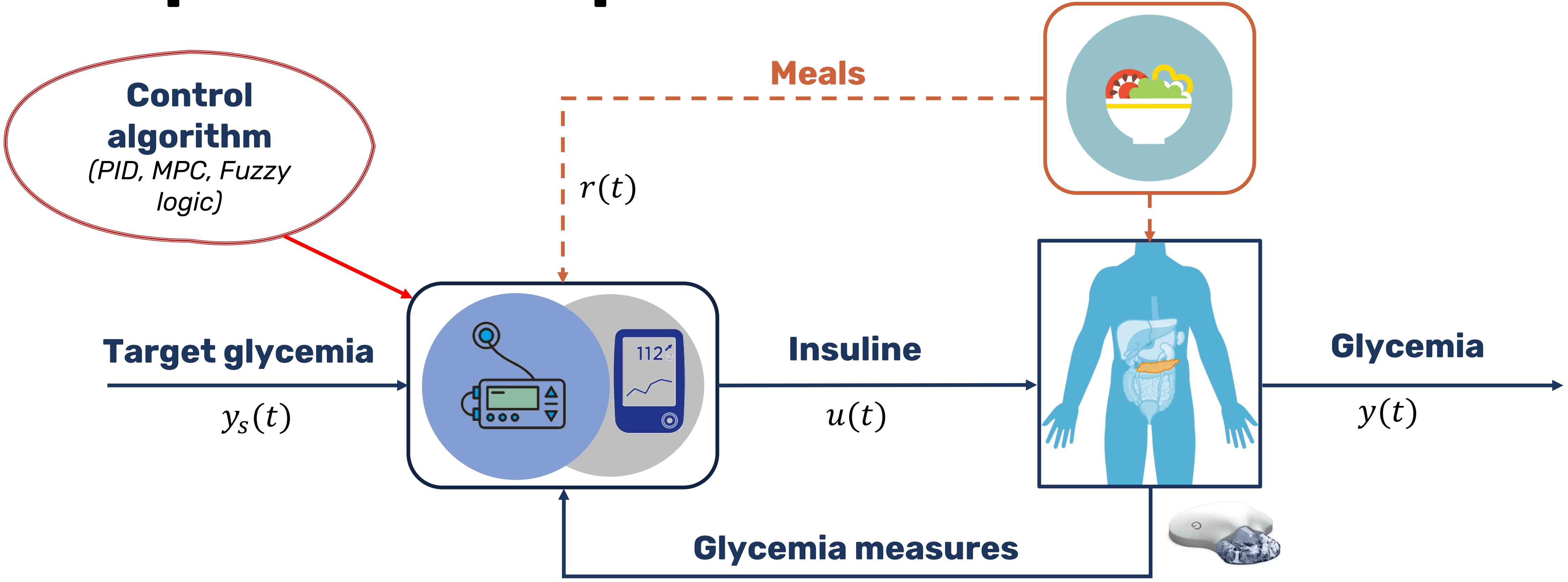
Last 4 days used for validation

Example: artificial pancreas

Model Identification (adult): training vs validation in a 5 days test



Example: artificial pancreas



From the control point of view, the process to be controlled is the patient.
The model helps to design a specific control algorithm (therapy) for the patient.

Course content

Part I: Dynamical systems

1. **Foundations of dynamical systems**
   - 1.1 Movements
   - 1.2 Equilibria
   - 1.3 Stability
   - 1.4 Continuous time VS Discrete time
2. **Transforms**
   - 2.1 Discrete Fourier Transform
   - 2.2 Z-transform and transfer function

Part II: Biological systems

3. **Nonnegative systems**
   - 3.1 Compartmental systems
4. **Typical biological systems**
   - 4.1 Epidemiological systems
   - 4.2 Pharmacokinetics models
   - 4.3. Blood Glucose – Insuline model
   - 4.4 Anesthesia model
5. **Identification**
   - 5.1 Output-error technique
   - 5.2 Validation
6. **Control**
   - 6.1 Classic control strategies

Friday, 01/03/2024 – 2:00pm
Aula A101

From control theory to clinical validation

An experience on
Automatic Insulin Delivery (AID) systems development

Prof. Fabricio Garelli

Grupo de Control Aplicado.
Instituto LEICI (CONICET-UNLP).
Facultad de Ingeniería, Universidad Nacional de La Plata, Argentina.
fabricio@ing.unlp.edu.ar
http://gca.ing.unlp.edu.ar

Profesor Titular Dedicación Exclusiva UNLP
Investigador Principal CONICET
Director Doctorado y Maestría en Ingeniería UNLP

UNIVERSITÀ DEGLI STUDI DI BERGAMO

Dipartimento di Ingegneria Gestionale, dell'Informazione e della Produzione