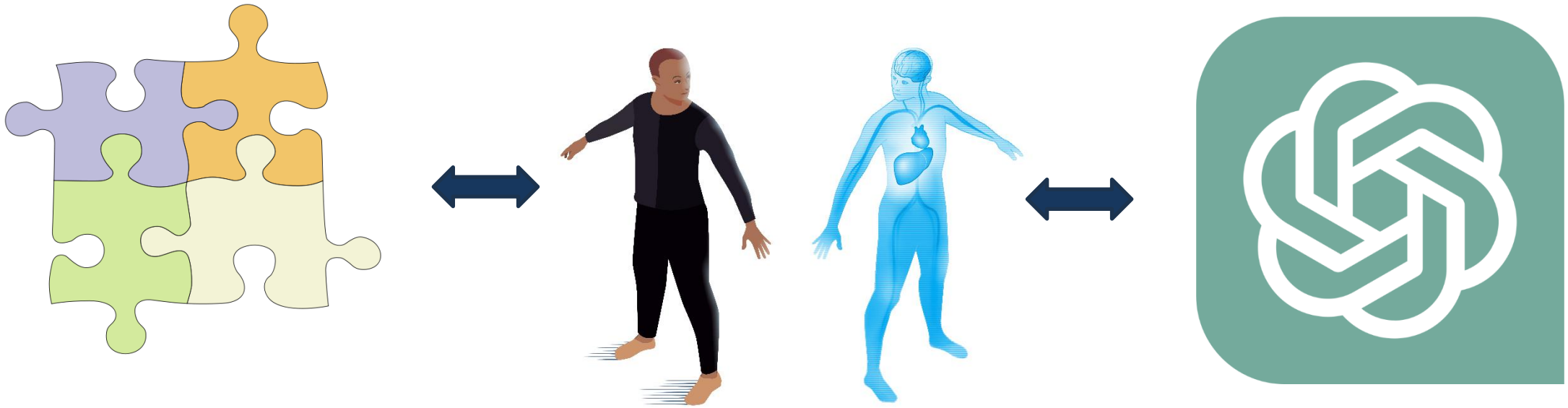
Systems Biology, Digital Twins, and AI – overview of the core concepts



Lecture 2
2023-09-01

Gunnar Cedersund,
Biomedical Engineering (IMT)

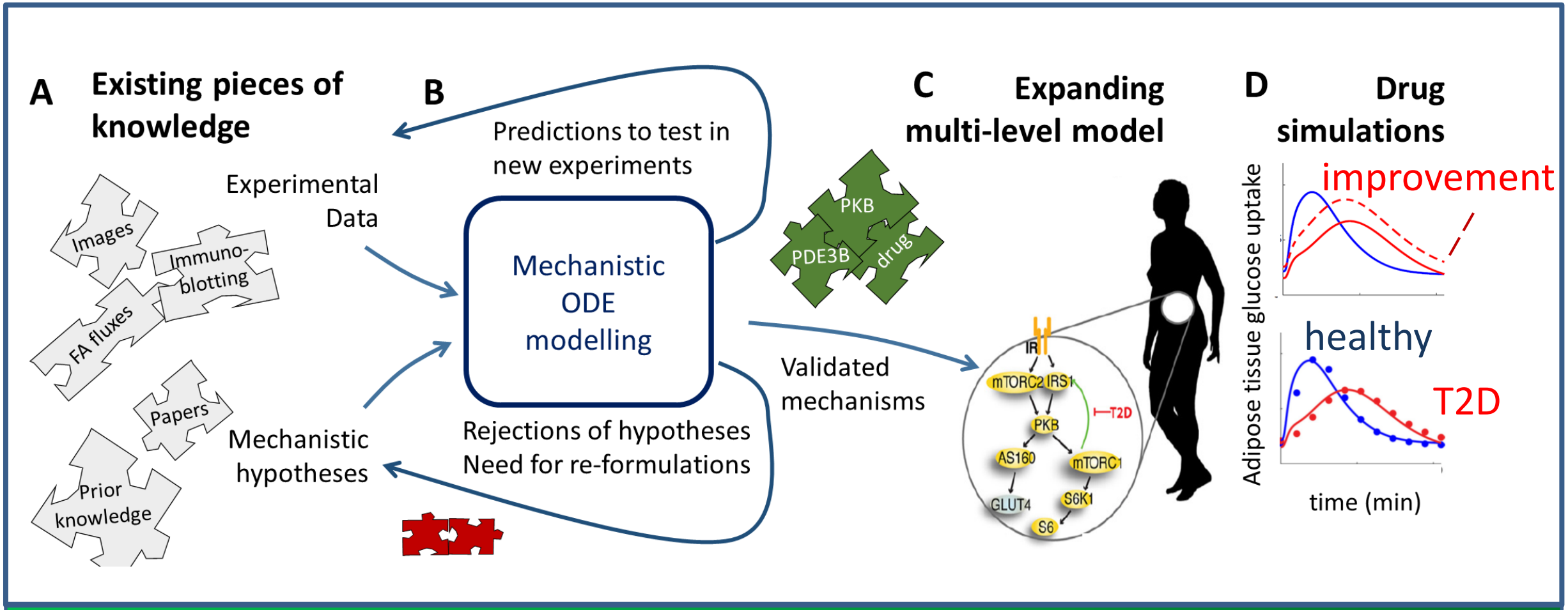
Overview of this lecture

Each now block in the lecture is marked with a cloud

- Systems biology and various formalisms for model formulation (block 1)
- Hands-on example with formulation of biochemical models
- Remaining blocks in the course:

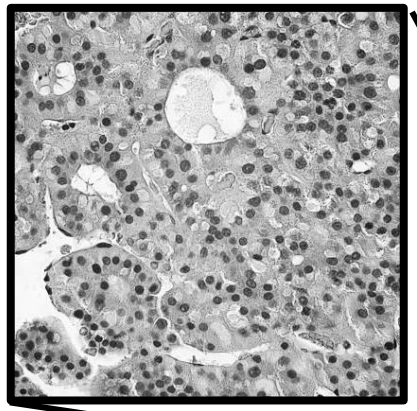
- Adding data, and estimation parameters and predictions with uncertainty
- Integrating models into a personalized digital twin model
- Hybrid models and AI

Systems biology is the art of integrating pieces of knowledge into useful models



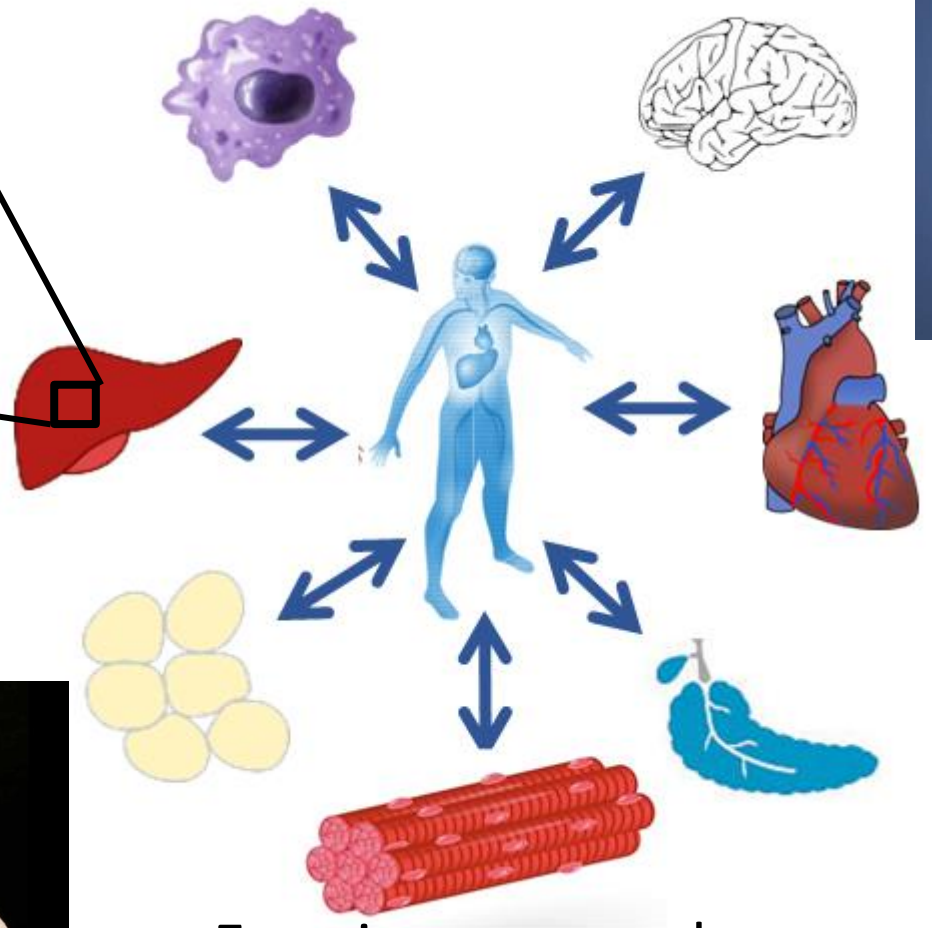
Mechanistic insights (systems biology) faster Making a difference (companies, eHealth)

Overview of our digital twins



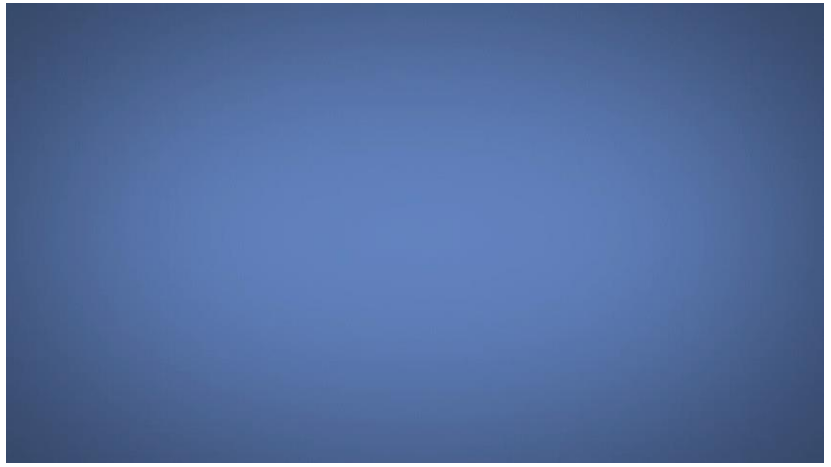
Cellular processes in fat and liver tissue

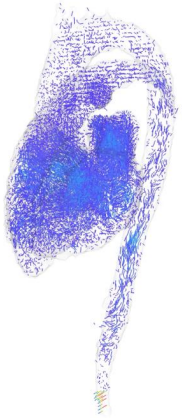
Immunology and the X-HiDE consortium



Exercise, yoga and biomechanics

A realistic brain and face – Catalyst project





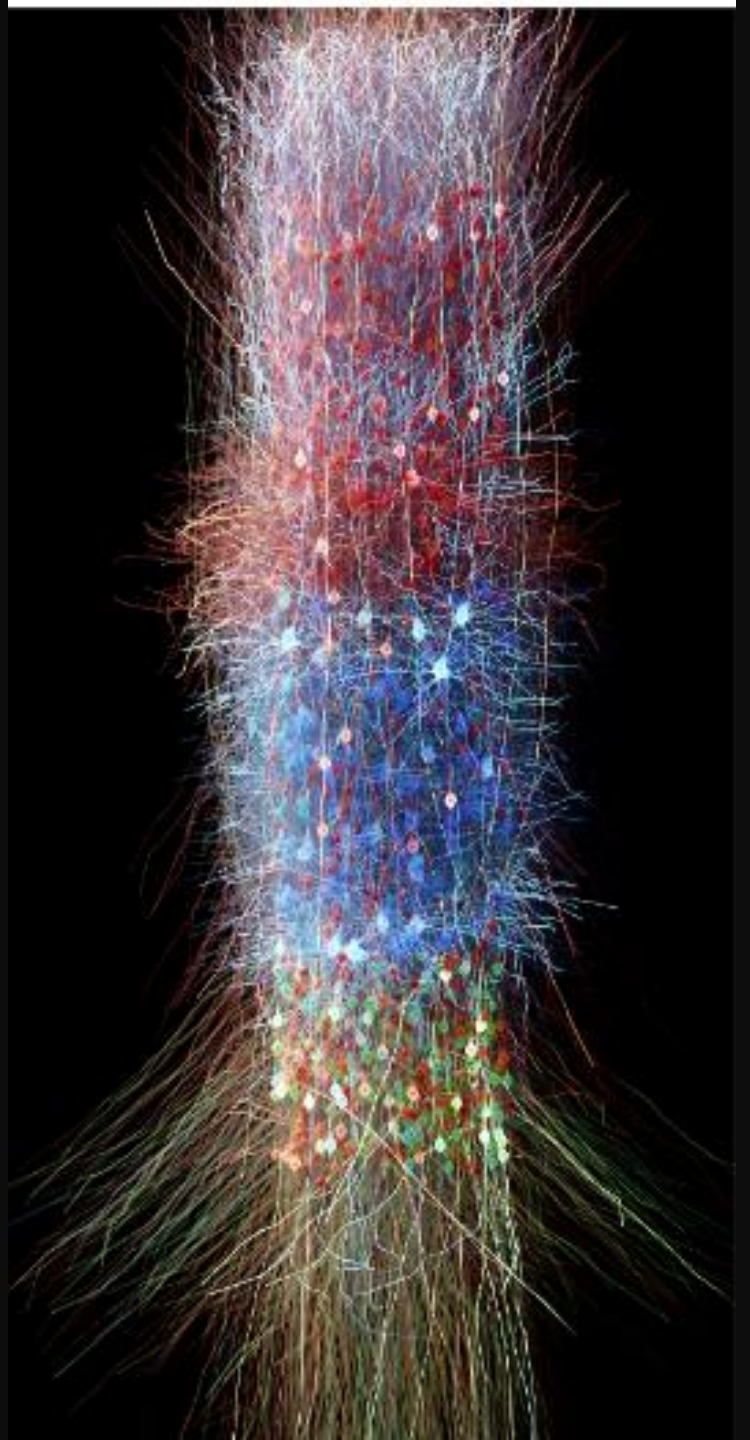
Blood flow and blood pressure, based on advanced MRI

Brain and head models

- This example is from the commercial application by Soul Machines
- This is already used at e.g. airports, to dispense information in a more human-like way
- The language part is driven by a language-based AI (similar to ChatGPT)
- Some versions of this software has a learning component, where different functions are driven by different machine-learning neural network, corresponding to different brain areas
- On top of this, the face is driven by a biomechanical model, incorporating all of the models underlying the facial movements
- There are good Ted Talks and youtube videos which give more details

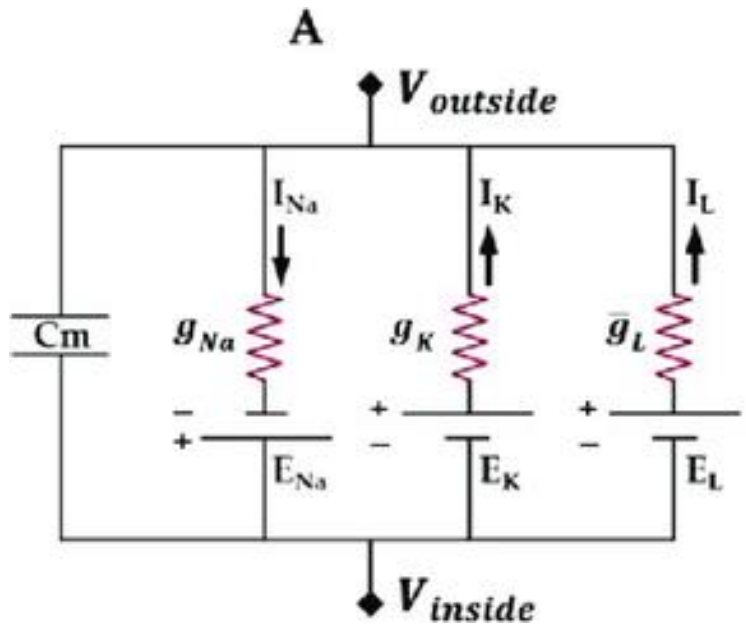
Neuronal network models

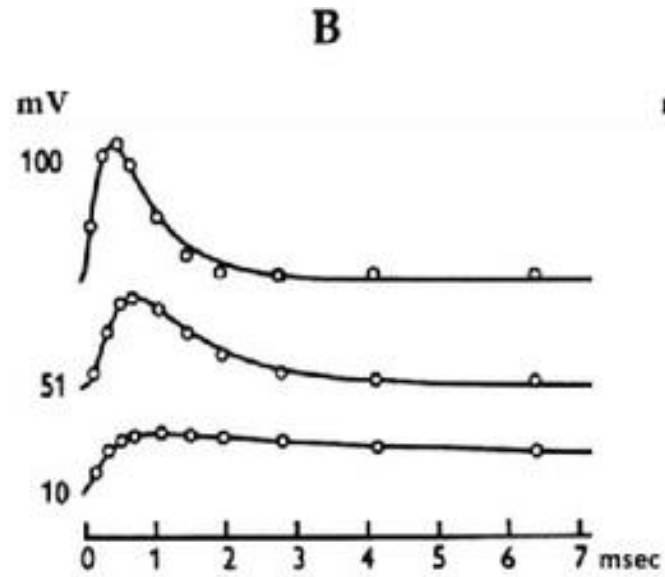
- Each node in these type of networks describe the electrophysiology of a neuron or nerve cell
- The architecture is based on the real anatomical layout of the cells, in a region of the brain
- These simulations are typically done in GUI-based software packages such as NEURON, where one does not explicitly formulate all equations
- The basic equations for each cell are based on the famous Hodgkin Huxley equations (analysed in computer lab)

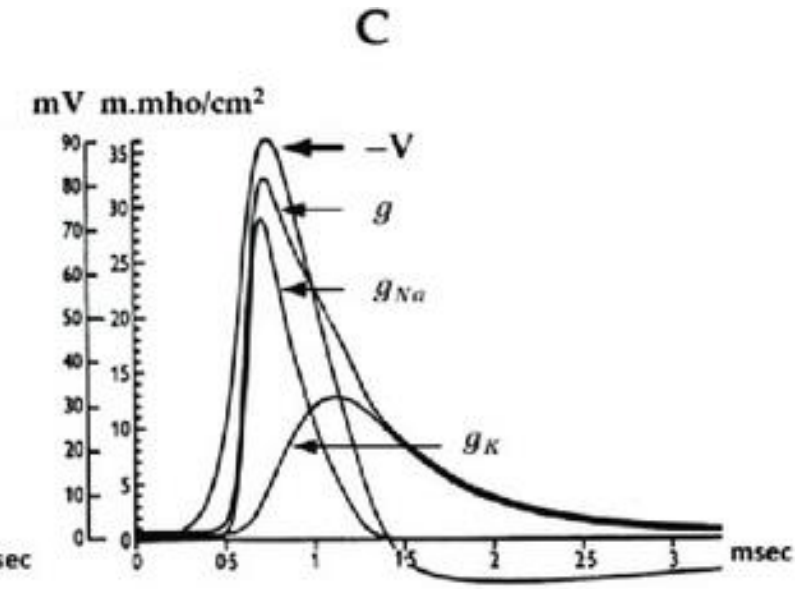


Hodgkin-Huxley model

(describing the electrophysiology of a cell, understood as an electric circuit)







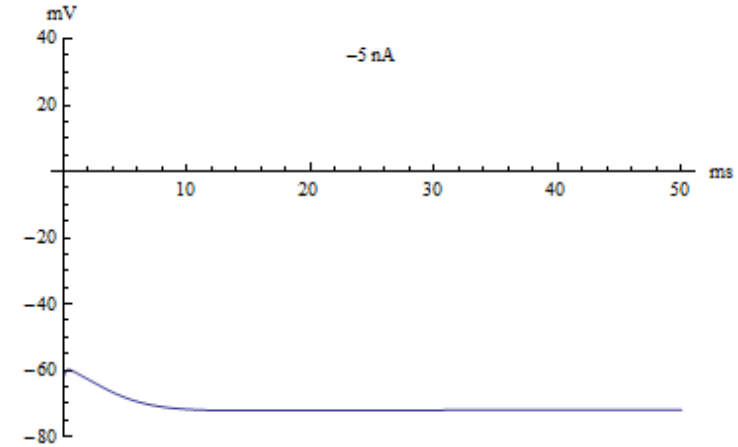
$$I = C_m \frac{\mathrm{d}V_m}{\mathrm{d}t} + \bar{g}_{\mathrm{K}} n^4 (V_m - V_K) + \bar{g}_{\mathrm{Na}} m^3 h (V_m - V_{Na}) + \bar{g}_l (V_m - V_l),$$

$$\frac{dn}{dt} = \alpha_n(V_m)(1-n) - \beta_n(V_m)n$$

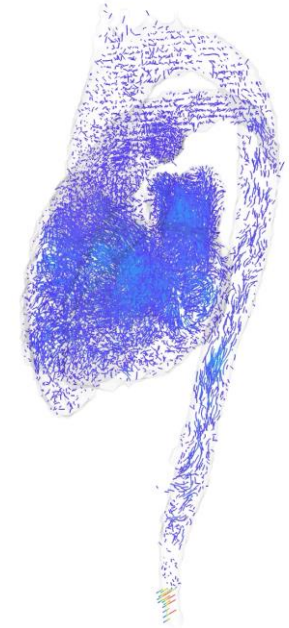
$$\frac{dm}{dt} = \alpha_m(V_m)(1-m) - \beta_m(V_m)m$$

$$\frac{dh}{dt} = \alpha_h(V_m)(1-h) - \beta_h(V_m)h$$

$$\alpha_n(V_m) = \frac{0.01(10-V_m)}{\exp\left(\frac{10-V_m}{10}\right)-1}$$



Blood flow models



- The most detailed way of simulating blood flow is to use the Navier Stokes equation
- This is based on partial differential equations, giving time-varying 3D-simulations, as seen in the animation
- These equations are solved in e.g. FEniCS, and take a long time
- A simpler and faster approach, more suitable to parameter estimation, is to study the simplified Windkessel equations (studied in computer lab)

Electric circuits to formulate blood flow models
Differential-Algebraic equations (DAEs)

Blood flow ⟷ electronics

$\Delta P = R * Q$ ⟷ $U = RI$

Resistance *Resistor*

$C = \frac{\Delta V}{\Delta P}$ ⟷ $I = C\frac{dU}{dt}$

Compliance *Capacitor*

$\Delta P = L\frac{dQ}{dt}$ ⟷ $U = L\frac{dI}{dt}$

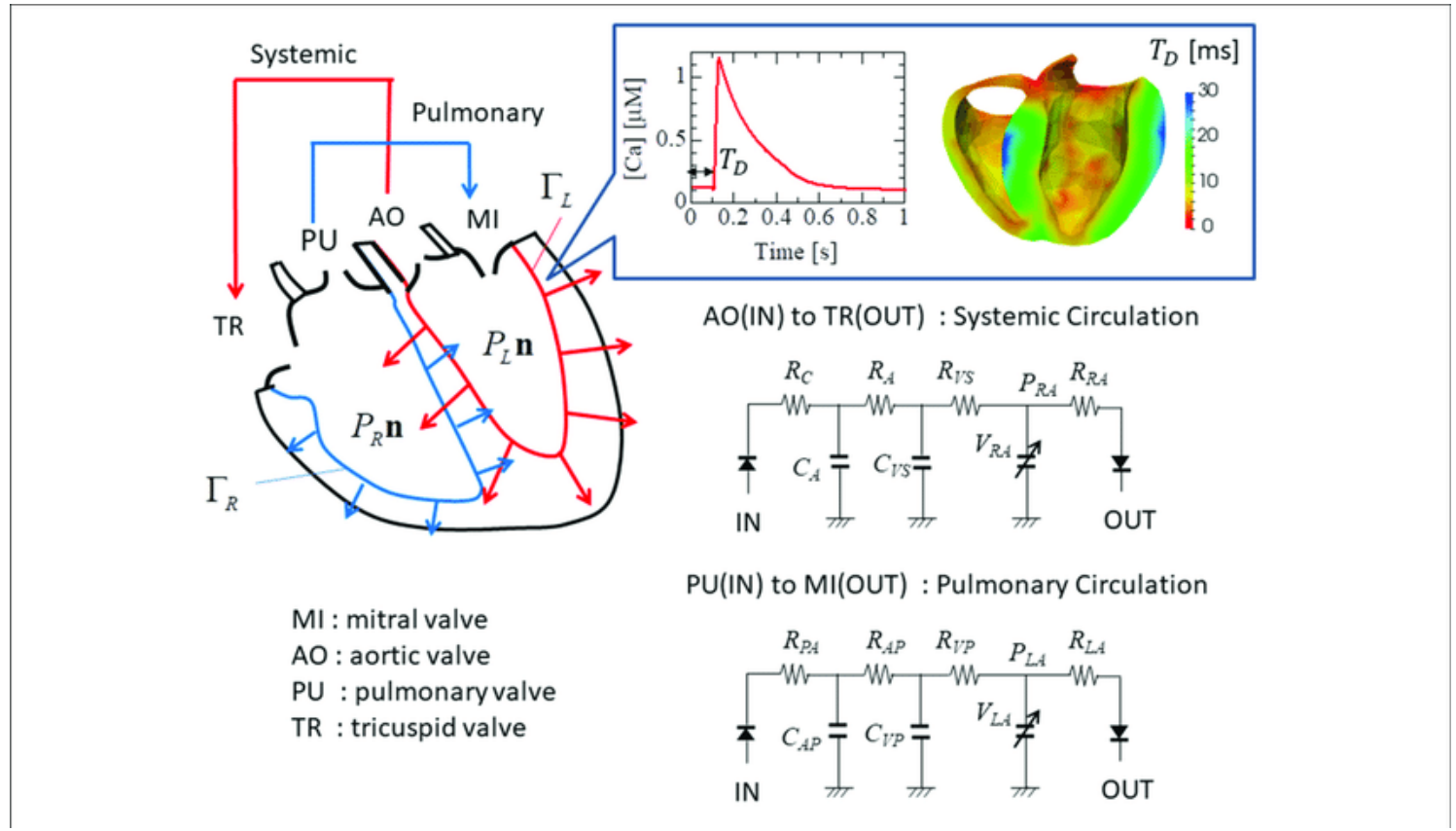
$L = \frac{\rho * l}{A}$

Inertance *Inductance*

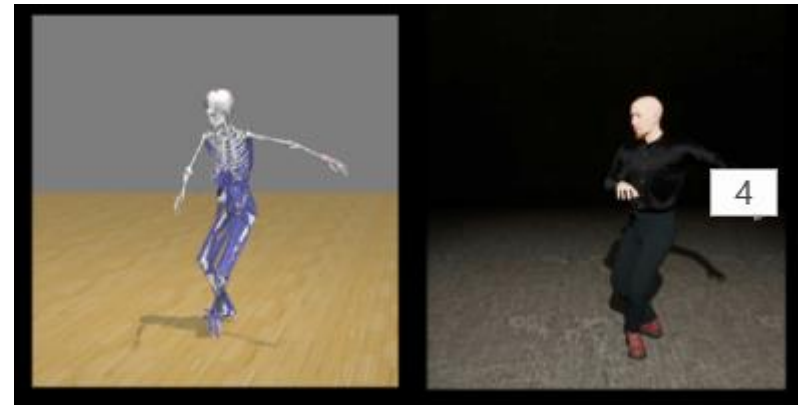
$KCL: \sum_{in} Q_i - \sum_{out} Q_i$

$KVL: \sum_{loop} \Delta Pi = 0$

Kirchoff's laws carry over directly



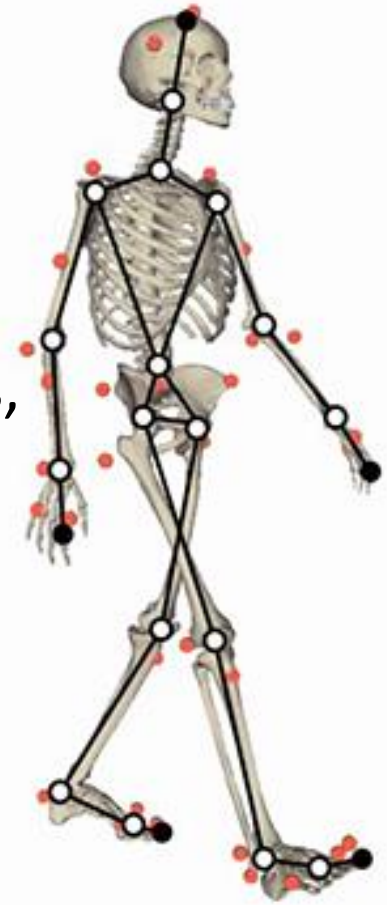
Biomechanical models

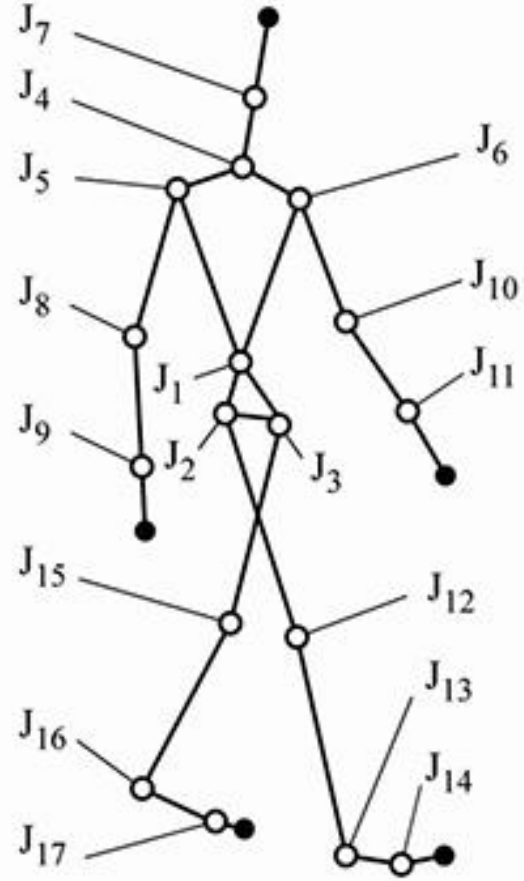


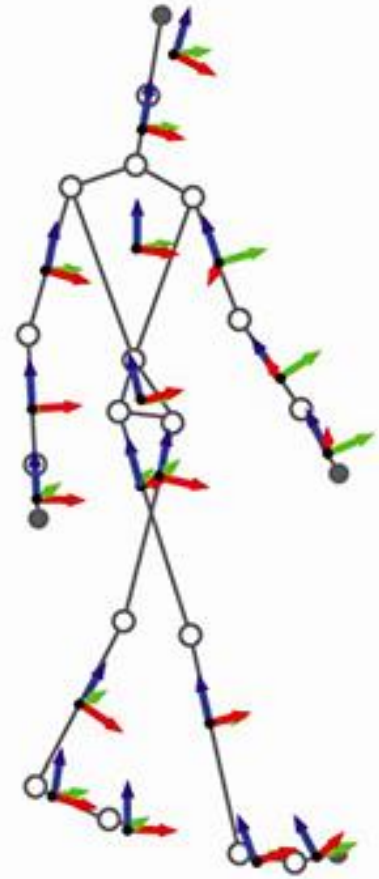
By defining which joints to include in the body, you create a system of interconnected biomechanical equations, each given by Newton's laws

Each sub-system has local coordinates, and you therefore usually do not formulate the equations by hand, but using e.g. OpenSim

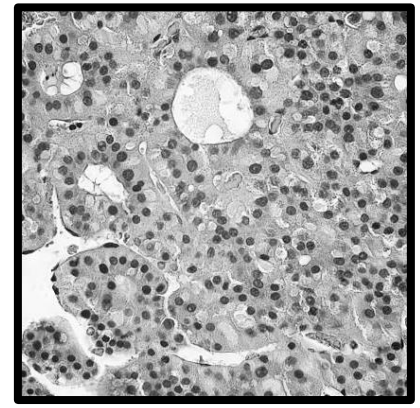
The models can be used to e.g. analyse motion-capture data







Biochemically based models



- Models that are based on reactions can be described using basic principles from biochemistry
- Each reaction is assumed to be governed by a kinetic rate expression
- The rate of change for each substance is given by the sum of the ingoing reactions minus the sum of the outgoing reactions
- Remember: you need to compensate for the stoichiometry and the volumes (if you have transport between different compartments)

Model formulation, reaction-based

$$\xrightarrow{v_j} \quad \ldots + N_{ij} S_i + \ldots \qquad (N_{ij} > 0)$$

$$\ldots + N_{ij} S_i + \ldots \quad \xrightarrow{v_j} \qquad (N_{ij} < 0)$$

$$[\dot{S}_i] = f(x, p) = \sum_j N_{ij} v_j$$

Model formulation, reaction-based

$$N_1 S_1 + N_2 S_2 + \ldots N_r S_r \rightarrow products$$

Mass-action kinetics $$v = k \prod_{i=1}^{r} [S_i]^{N_i}$$

Michaelis-Menten kinetics ($|N_i|$ = 1, for all i): $$v = V_{\max} \cdot \prod_{i=1}^{r} \frac{[S_i]}{K_{\mathrm{M,i}} + [S_i]}$$

Hill-kinetics ($|N_i| = 1$, for all i): $$v = V_{\max} \cdot \prod_{i=1}^{r} \frac{[S_i]^{h_i}}{K_{\mathrm{M,i}}^{h_i} + [S_i]^{h_i}}$$

Hands-on example

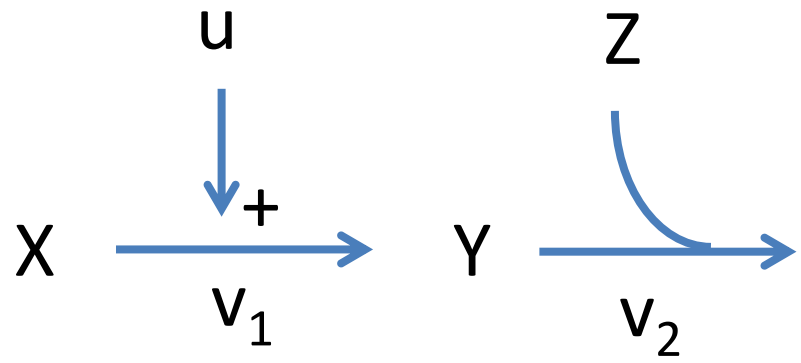
Write the state-space form of

1. The states are x_1=#X, x_2=#Y, x_3=#Z
2. The reaction rates are
$v_1=V_{max}x_1u/(K_M+u)$ (saturated w.r.t. u but not w.r.t x_1);
$v_2 = k_2x_2x_3$ (mass action kinetics)
3. The resulting ODEs are

$\dot{x}_1 = -v1$

$\dot{x}_2 = v_1 - v_2$

$\dot{x}_3 = -v_2$



1: states
2: reaction rates
3: ODEs $\dot{x} = v^{in}_1 + v^{in}_2 + \ldots - v^{out}_1 - \ldots$
4: $\hat{y}$ equation
5. $p = (p_x, x_0, p_y)$ values

4. We assume that we can measure the rate of the second reaction times an unknown constant, i.e. $\hat{y}= k_y\ v_2$
5. Parameter values $p = (p_x, x_0, p_y)$ =
(V_{max},K_M,k_2,$x_1(0)$,$x_2(0)$,$x_3(0)$, k_y) = (1,2,3,4,5,6,7)

Don’t forget – if you transport between different compartment with different volumes, the volumes will appear in the equations

$$\frac{dNG_{m,liver}(t)}{dt} = G_d(t) + V_{hep} \cdot EGP - V_{hep}\left(E_{G0} + S_I(t) \cdot \frac{NI_{m,liver}(t)}{V_{m,liver}}\right)\frac{NG_{m,liver}(t)}{V_{m,liver}} + Q \cdot \frac{NG_{m,islets}(t)}{V_{m,islets}} - Q \cdot \frac{NG_{m,liver}(t)}{V_{m,liver}} \; (mmol/h)$$

There are many different ways to do this, check carefully that your method is consistent with itself in all places!

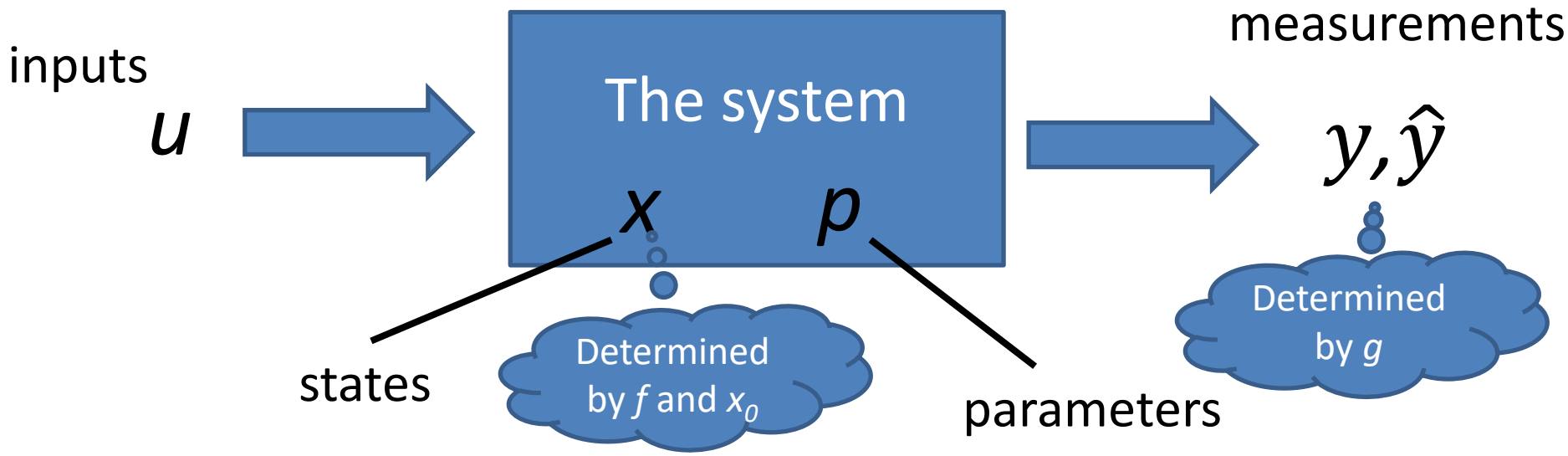
Now the remaining blocks

The four blocks of material

- Formalisms for model formulation and nonlinear dynamical systems
- Parameter estimation and model uncertainty
- Nonlinear mixed-effects modelling, and applications in drug development and personalized medicine
- Hybrid models, machine learning, and digital twins

To estimate the parameters, we use the inputs and output data

- The inputs u may vary over time, and are typically known
- The inputs are what you do to the system (e.g. add insulin or some other stimulation)
- The inputs are not affected by what happens in the system
- The measured outputs y are often varying over time and form a given set of values, independent of the model
- The simulated outputs $\hat{y}$ depend on the parameters



In the standard state-space description of a model

$$\frac{dx}{dt} = \dot{x} = f(x, p_x, u)$$
$$x(t = 0) = x(0) = x_0$$
$$\hat{y} = g(x, p_x, p_y, u)$$

The inputs are given by u

The *measured* outputs are given by y

The *simulated* outputs are given by $\hat{y}$

The *residuals r* describe the difference between measured and simulated outputs.

We sum up (square of) all residuals, weigh them with the uncertainty of each datapoint, to get the *average agreement* between data and model.

The parameter estimation then seeks to *maximize this agreement*

Basic relationships

- Residuals are given by the difference between simulated and experimental values at each time point

$$\varepsilon(t,p) := y(t) - \hat{y}(t,p)$$

- The sum of the squared residuals form the basis for the least squares cost function

optional reduction in size of V

always positive contribution

$$V(p) = \sqrt{\sum_{t=1}^{N} \frac{(y(t) - \hat{y}(t,p))^2}{\sigma(t)^2}}$$

all differences considered together

data-points with low uncertainty contribute more

We will learn

- Various approaches to the optimization step
- How to deal with the fact that the optimum often is not well-determined, i.e. there are many parameter sets that give the same agreement with data
- Approaches to find the uncertainty of predictions: Monte Carlo Sampling, Profile Likelihood, etc

Block 3

Nonlinear mixed-effects models NLME

(how to combine patient-specific and population data)

just as usual

state for individual *i*

input for individual *i*

parameters for individual *i*

$$\dot{x}^i = f\left(x^i, u^i, \phi^i\right)$$

new part in NLME

Typical population value (fixed effects)

unique random modification of population value (random effects)

$$\phi^i = g\left(\Theta, \eta^i, Z^i\right)$$

$$\eta^i \in N(0, \Omega)$$

covariate = known modification of value

Block 4

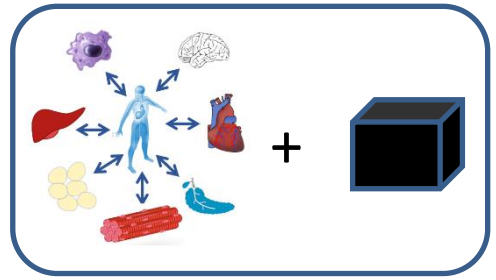
Hybrid models, combining AI and mechanistic models

Generative models, and language models

Simulated risk

Classification models

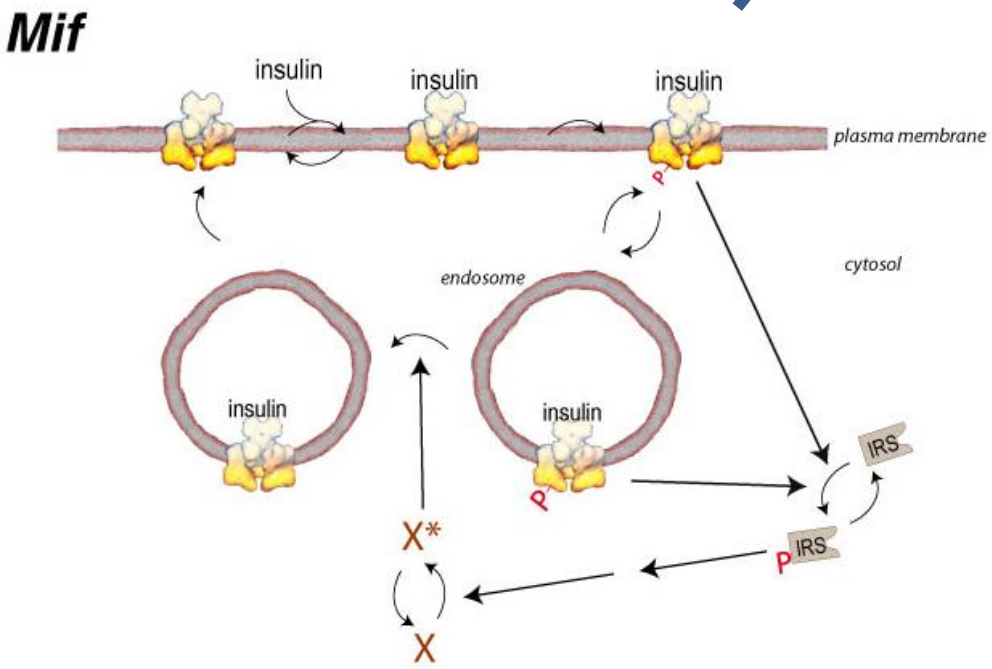
Simulations & biomarkers



Modules & biomarkers

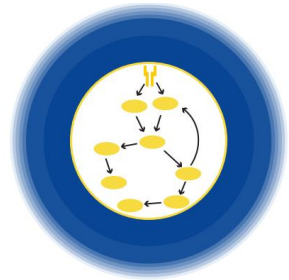
Hybrid digital twins

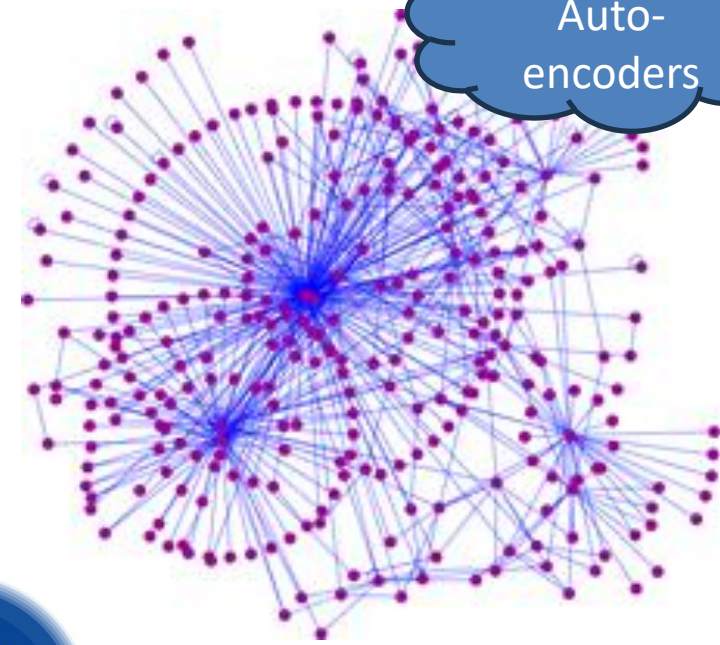
Variational Auto-encoders



Mechanistic models

Automated modelling to omics-level





Bioinformatics network models