

A Hands-on Tutorial in Systems Biology

From Models to Machine Learning

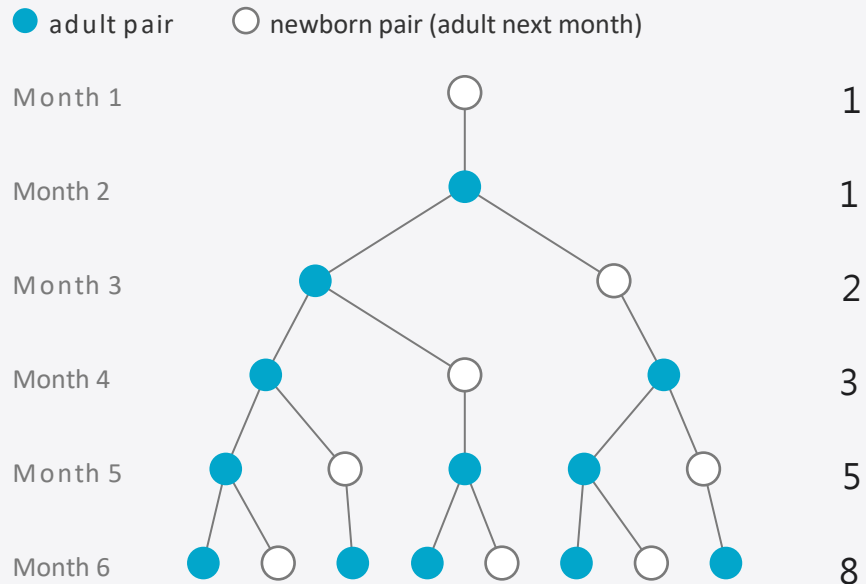
Seokjoo Chae

Department of Mathematics, Soongsil University

Mathematical Biology: From Fibonacci to Schrödinger

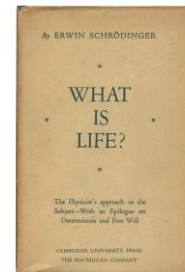
From rabbit counting in 1202 to What Is Life? in 1944, scientists have described living systems with mathematics.

Fibonacci's rabbits, Liber Abaci (1202)



Liber Abaci (1202)

manuscript page with the rabbit problem



What Is Life? (1944)

Erwin Schrödinger, Cambridge Univ. Press
Press

- 1202
Fibonacci: rabbit pairs grow as 1, 1, 2, 3, 5, 8, ...
- 1838
Verhulst: logistic growth; populations level off
- 1925–26
Lotka and Volterra: predator and prey oscillate
- 1944
Schrödinger, What Is Life?: heredity as a physical
- 1952
Turing, Hodgkin–Huxley: patterns and nerve equations

To Predict, We Need the Current State and the Dynamics

How long does it take from Tashkent to Samarkand? We use where we are now and how fast we are moving.



State now

We are in Tashkent (0 km) at 9:00. Samarkand is about 300 km away.

Dynamics

Speed is the rate of change of position, dx/dt
100 km/h, 100 km takes one hour.

Prediction

300 km at 100 km/h takes 3 hours, so we arrive around 12:00. At 80 km/h it takes 3 h 45 min.

Biology Needs the Same Kind of Prediction

Biological data keep growing, but each experiment takes time and money. A model lets us test scenarios first.

01

Explain the data

Fit a mechanism to the measurements. The parameters then have a biological meaning, such as growth, killing or clearance. clearance.

Today: seven parameters per growth curve.

02

Run fewer experiments

Test doses, timings and drug combinations in combinations in simulation, and run only the only the promising ones in the lab.

Today: 30,000 simulated experiments.

03

Predict untested experiments

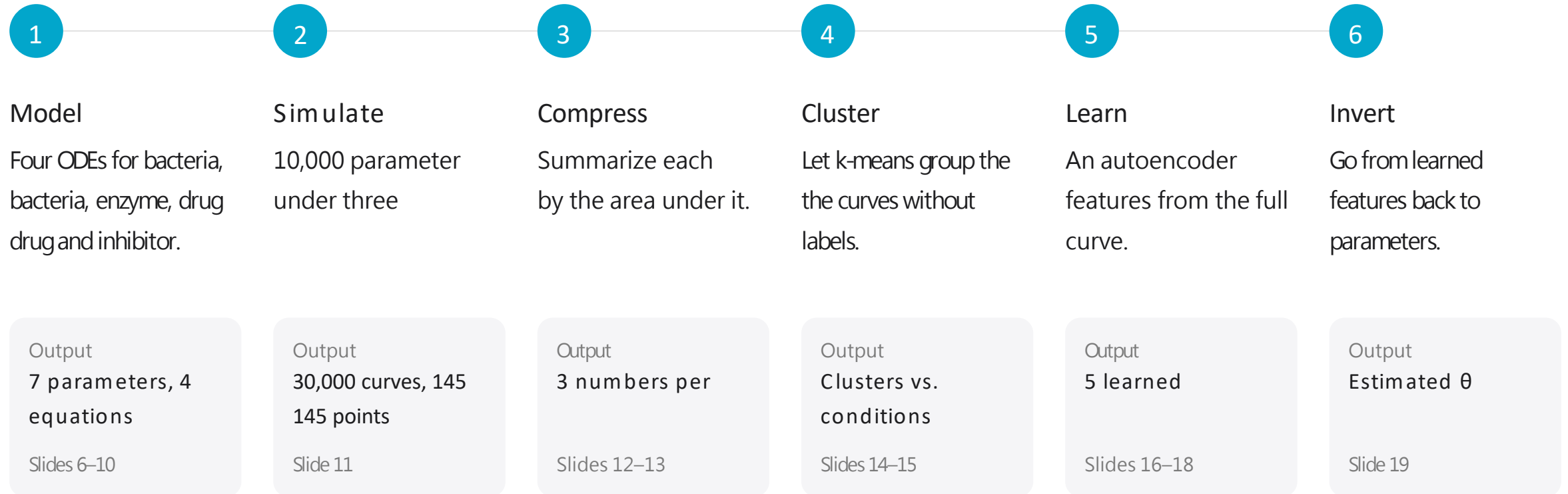
Look at conditions nobody has tried yet, as new drug combinations or resistant

Today: antibiotic with and without an inhibitor.

| Car: position + speed → arrival time. Cells: population + drug dynamics → survival or extinction.

Today's Plan: From a Small ODE Model to Machine Learning

We build a bacteria–antibiotic model, simulate it, and analyze the simulated curves with clustering and an autoencoder.



Forward: parameters $\rightarrow$ curves. Reverse: curves $\rightarrow$ conditions and parameters.

Math Bio 101: Reaction Rates as Differential Equations

In chemical kinetics a reaction rate is written as an ODE. The same form describes cell numbers and drug concentrations.

First-order kinetics

$$\frac{dy}{dt} = -ky$$

$$y(t) = y_0 e^{-kt}$$

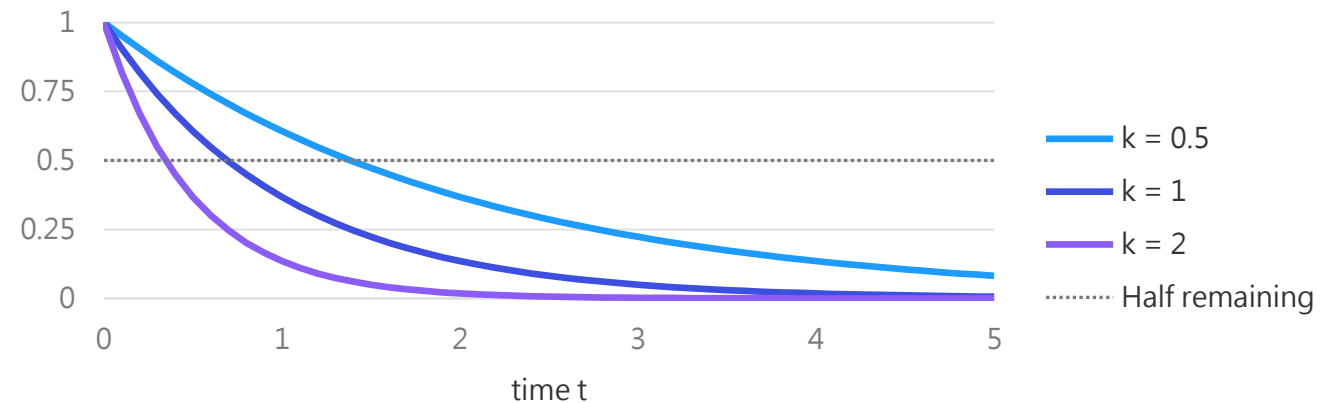
$$t_{1/2} = \frac{\ln 2}{k}$$

k: rate constant (per unit time)

y_0 : amount at $t = 0$

$t_{1/2}$: half-life, the time to halve

Fraction remaining, $y(t) / y_0$



Where it appears

y is

k is

Chemistry: $A \rightarrow B$

concentration of A

rate constant

Pharmacology

drug concentration in plasma

elimination rate

Cell biology

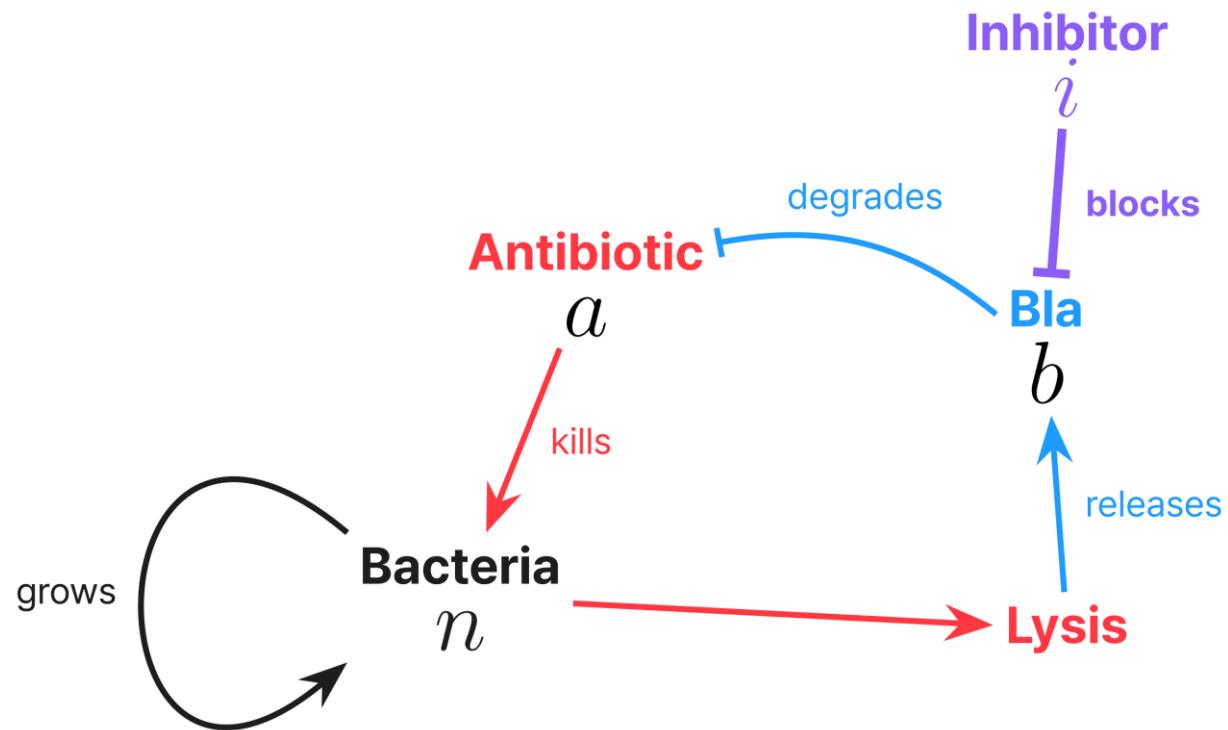
cells under lethal stress

death rate

The rate of change is proportional to the amount, which gives exponential decay with half-life $\ln 2 / k$.

A Bacteria–Antibiotic Model With β -Lactamase

Lysed cells release β -lactamase (Bla), which degrades the antibiotic. An inhibitor blocks the enzyme.



What is β -lactamase (Bla)?

An enzyme that cuts the β -lactam ring of penicillins, so they no longer kill.

Why lysis matters

Lysed cells release the enzyme into the medium, which protects the remaining cells.

β -lactamase inhibitors

Clavulanic acid, sulbactam and tazobactam block the amoxicillin + clavulanate.

Typical dynamics: growth, collapse, and regrowth once the enzyme has cleared the drug.

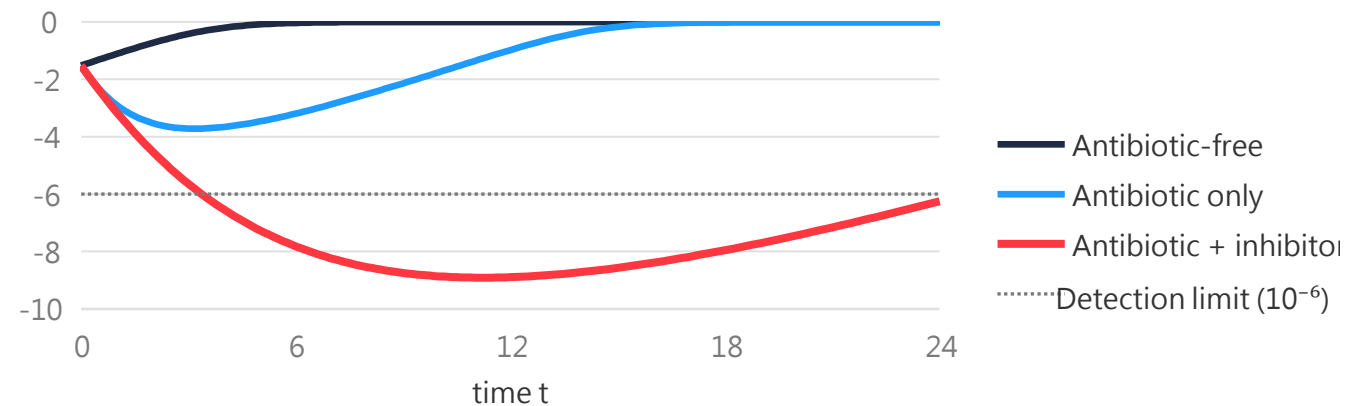
The Model as Four Coupled ODEs

Dimensionless form: time in growth units, n as a fraction of carrying capacity, a relative to the initial dose.

$$\begin{aligned}\frac{dn}{dt} &= n(1 - n) - \alpha an \\ \frac{db}{dt} &= an - \delta_b b \\ \frac{da}{dt} &= -(\delta_a + \kappa_b \varepsilon(i)b + \varphi n)a \\ \frac{di}{dt} &= -\delta_i i\end{aligned}$$

$\varepsilon(i) = 1 / (1 + i)$: the share of enzyme still working

$\log_{10} n(t)$ for one parameter set, three starting conditions



Condition	a_0	i_0	What happens
Antibiotic-free	0	0	Logistic growth to $n = 1$
Antibiotic only	1	0	Dips 2.2 decades, back to n_0 by $t \approx 10.6$
Antibiotic +inhibitor	1	4	Below 10^{-6} from $t \approx 3.4$: effectively extinct

With the same parameters, the population regrows under antibiotic alone and dies out with the inhibitor.

Simulating One Parameter Set

Given θ and the initial condition, the ODE solver returns n , b and a at 145 time points.

Parameter set θ

$\alpha = 5$
 $\kappa_b = 80$
 $\phi = 0.5$
 $\delta_a = 0.02$
 $\delta_b = 0.1$
 $\delta_i = 0.15$
 $n_0 = 0.03$

Condition

$(a_0, i_0) = (1, 0)$

antibiotic only

ODE solver

scipy solve_ivp, LSODA

rtol 10^{-6} , atol 10^{-10}

State: log n , b , a , i

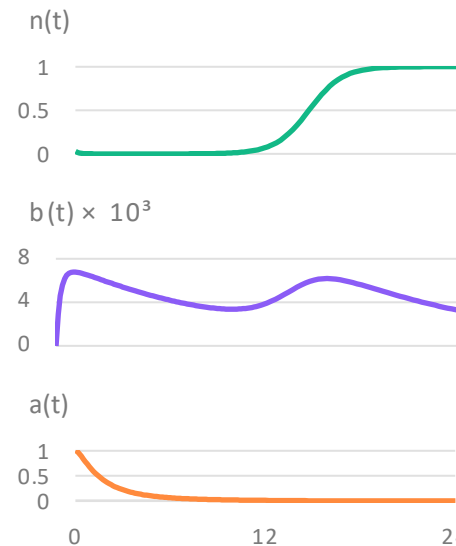
Analytic Jacobian: the system is stiff

Time grid

$0 \leq t \leq 24$, 145 points

denser before $t = 12$

Trajectory



Data array

t	n	b	a
0	3.00e-02	0.0000	1.000
0.5	4.76e-03	0.0061	0.837
1	1.25e-03	0.0068	0.637
2	2.97e-04	0.0064	0.367
4	2.19e-04	0.0053	0.138
8	3.05e-03	0.0037	0.030
12	1.05e-01	0.0038	0.009
24	1.00e+00	0.0031	0.000

8 of 145 rows shown

Size per run

$3 \times 145 = 435$ numbers

n , b , a at every time point

Each simulation takes θ and (a_0, i_0) and returns a 3×145 array.

The Simulator as a Function of the Parameters

F maps seven parameters to a 3 × 145 array. We can evaluate F for any parameter set in the prior range.

$$\frac{dn}{dt} = n(1 - n) - \alpha an$$
$$\frac{db}{dt} = an - \delta_b b$$
$$\frac{da}{dt} = -(\delta_a + \kappa_b \varepsilon(i)b + \varphi n)a$$
$$\frac{di}{dt} = -\delta_i i$$

Highlighted: the six rate constants. n_0 is the initial population.



Symbol	Meaning	Test value	Prior range
α	Killing per unit of drug	5	1 – 20
κ_b	Drug cleared by the enzyme	80	10 – 500
φ	Drug cleared by live cells	0.5	0 – 1.5
δ_a	Natural drug decay	0.02	0.003 – 0.1
δ_b	Enzyme loss	0.1	0.02 – 0.5
δ_i	Inhibitor clearance	0.15	0.03 – 0.63
n_0	Starting population	0.03	0.001 – 0.2

Evaluating F for many parameter sets gives many simulated experiments.

10,000 Parameter Sets, Three Conditions Each

Each θ drawn from the prior is simulated without antibiotic, with antibiotic only, and with antibiotic + inhibitor.

10,000

parameter sets drawn from the prior

3

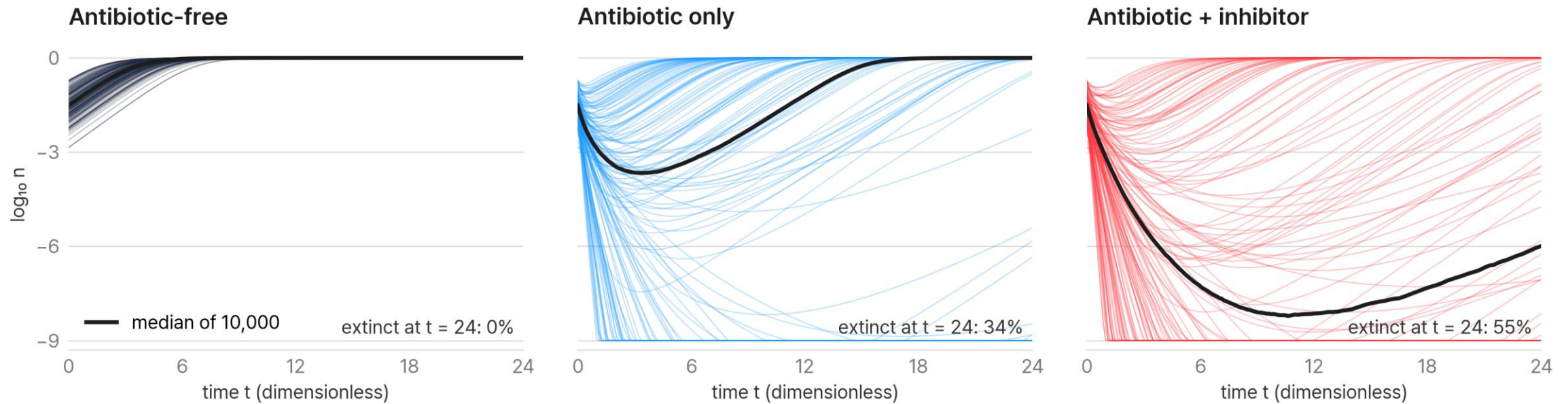
conditions per parameter set

30,000

simulated time courses

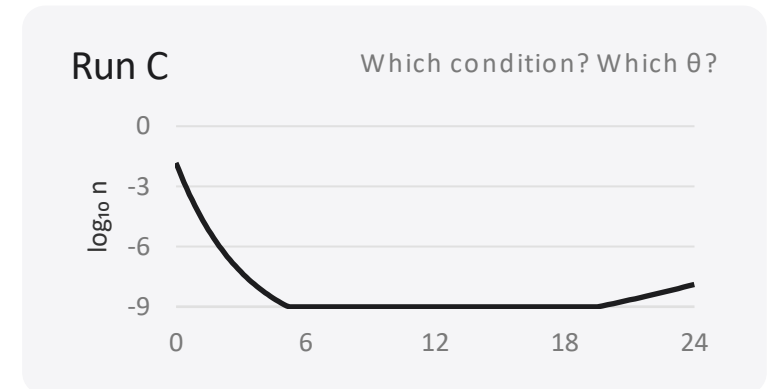
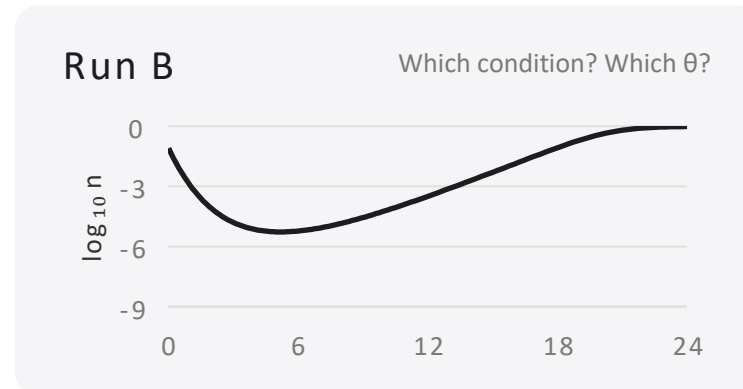
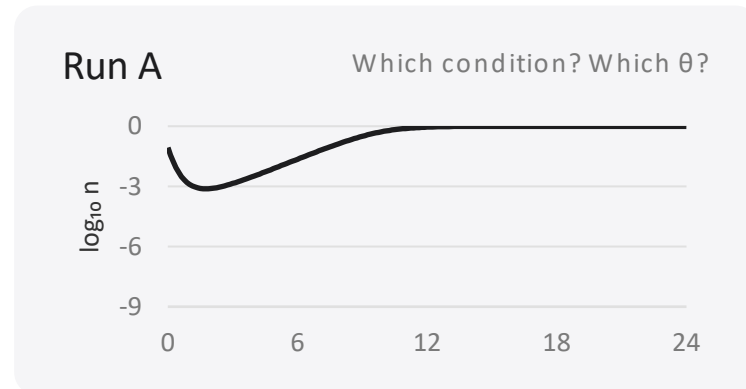
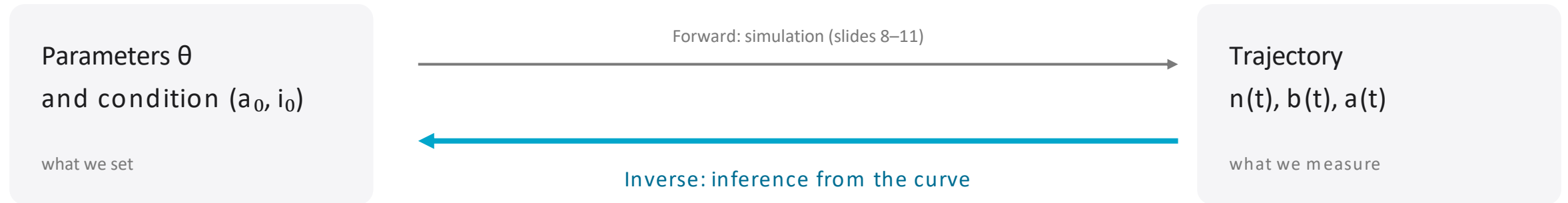
145

time points per curve



From Time Series Back to Conditions and Parameters

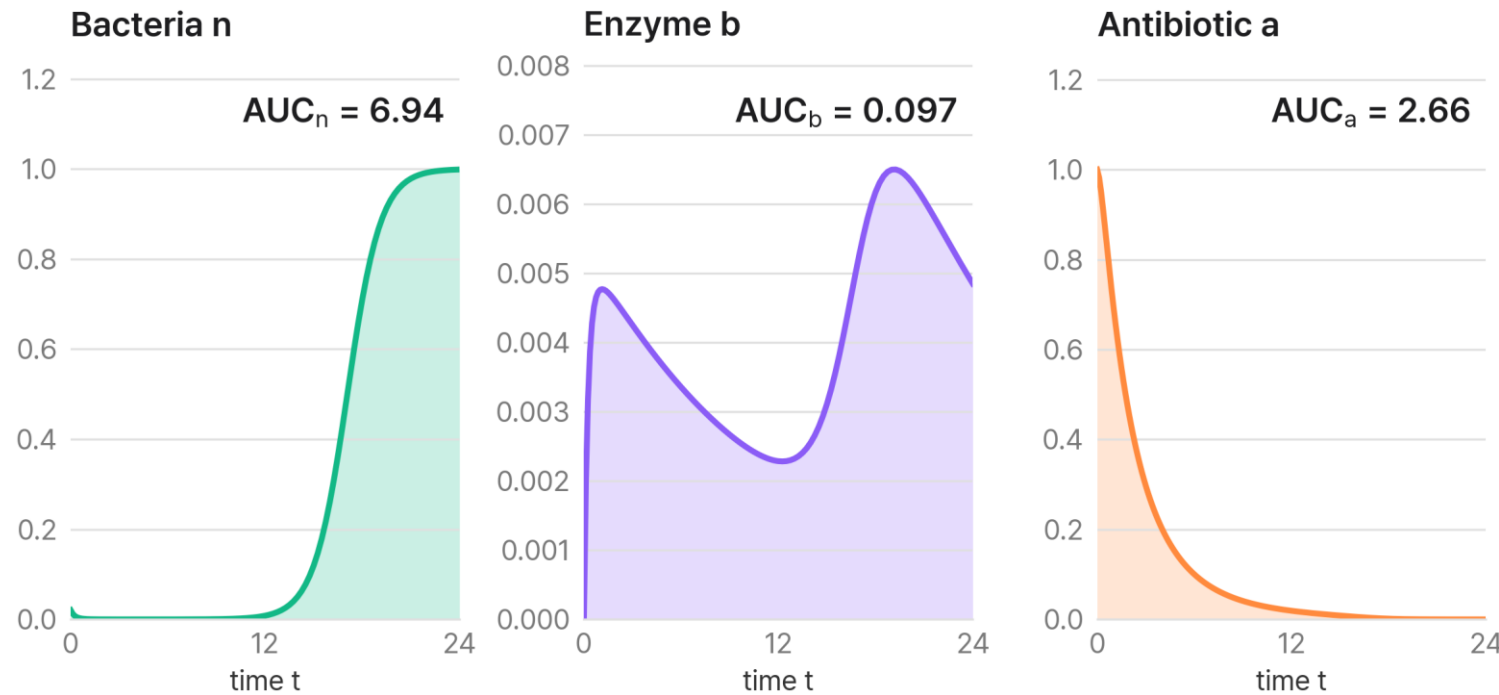
If we only have the time series and not θ , can we tell which condition produced it? Can we recover θ ?



Let's try it, first with a short summary of each curve and then with the full curve.

Compressing Each Curve With the AUC

The area under each curve reduces a trajectory to three numbers: the AUC of bacteria, enzyme and antibiotic.



435 → 3

numbers per run (3 species × 145 points → 3 areas)

Definition

$AUC_x = \int_0^{24} x(t) dt$ for $x = n, b, a$, computed with the trapezoid rule.

What is lost

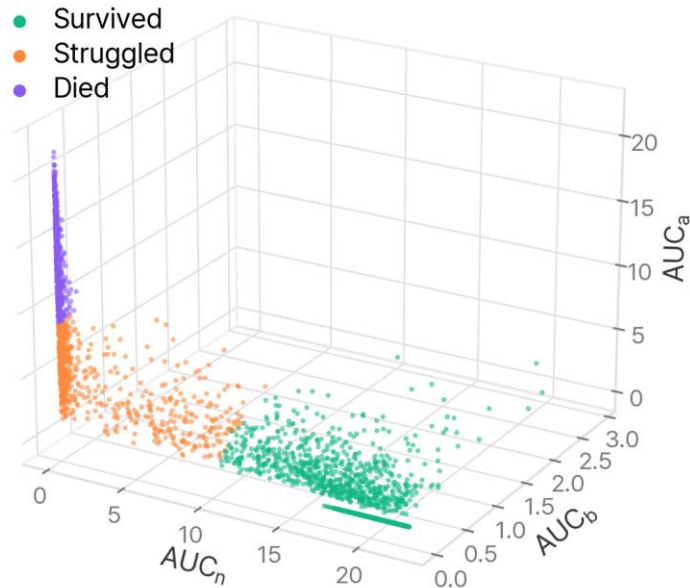
The timing of the dip. A shallow long dip and a short dip can have the same AUC.

The AUC is the simplest summary. Next we check what it can and cannot separate.

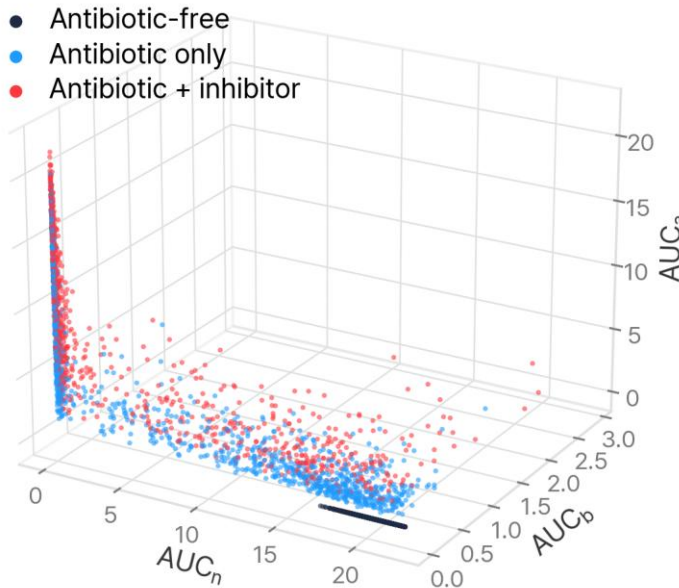
k-Means Clustering in AUC Space

Each of the 30,000 runs is a point (AUC_n , AUC_b , AUC_a). We apply k-means with $K = 3$ without using labels.

k-means clusters ($K = 3$)



True condition



Cluster centres (raw AUC)

Cluster	AUC_n	AUC_b	AUC_a
Survived	18.78	0.127	0.59
Struggled	2.26	0.106	5.06
Died	0.01	0.028	13.73

Who is in each cluster

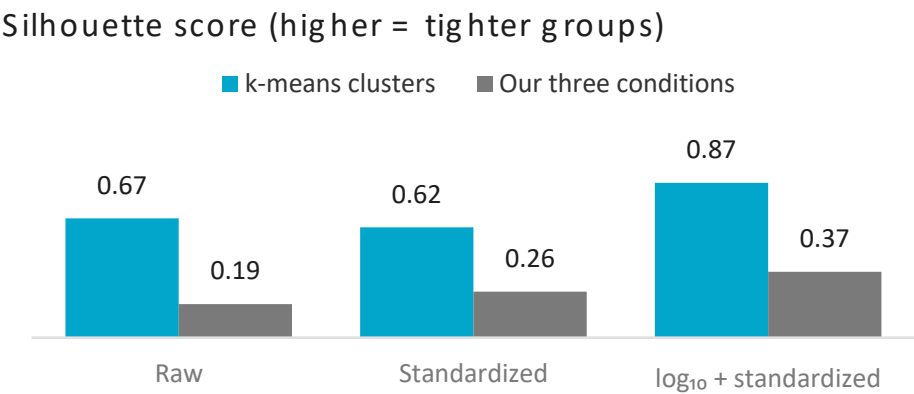
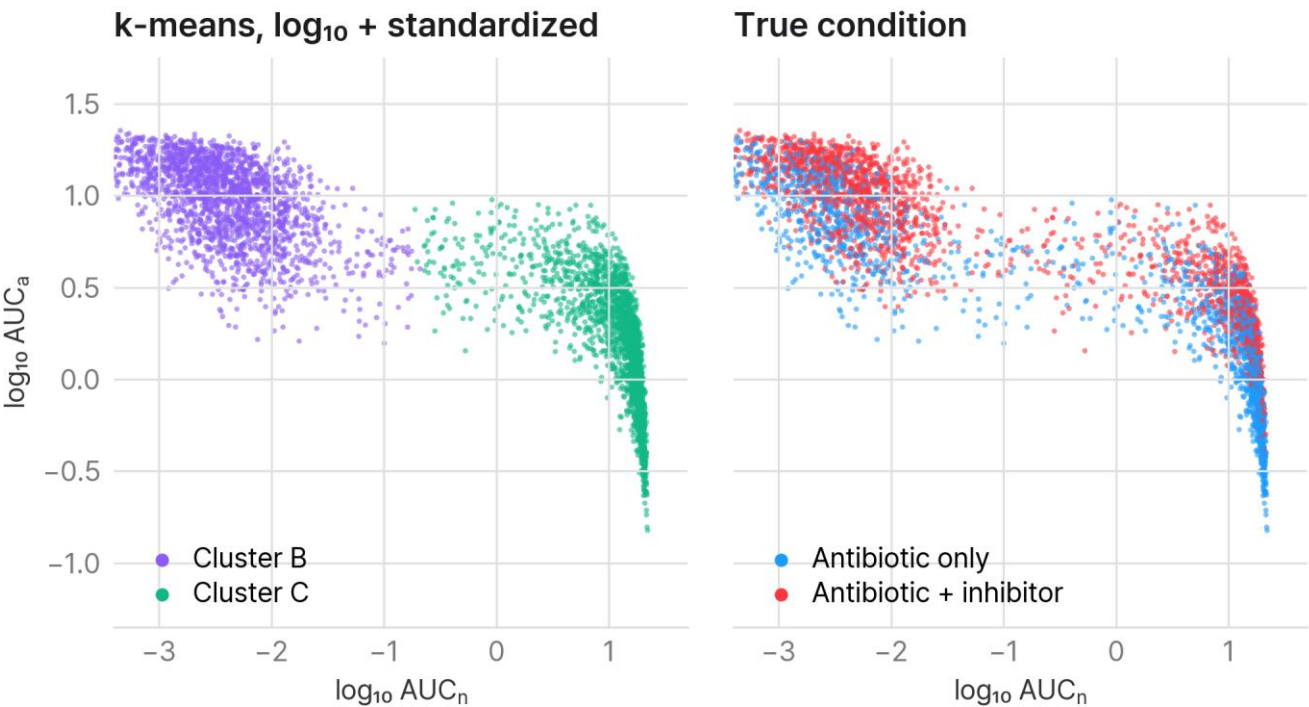
Cluster	Free	Antibiotic	+ Inhib.
Survived	10,000	4,793	2,513
Struggled	0	3,431	3,563
Died	0	1,776	3,924

Each cluster contains both antibiotic-only and inhibitor runs. The clusters follow the outcome. outcome.

The three clusters correspond to outcomes (survived, struggled, died) rather than to the conditions.

Clustering After log₁₀ Transform and Standardization

The silhouette rises to 0.87, but both drug clusters still mix antibiotic-only and inhibitor runs at about 40 : 60.



Who is in each cluster (log₁₀ + standardized)

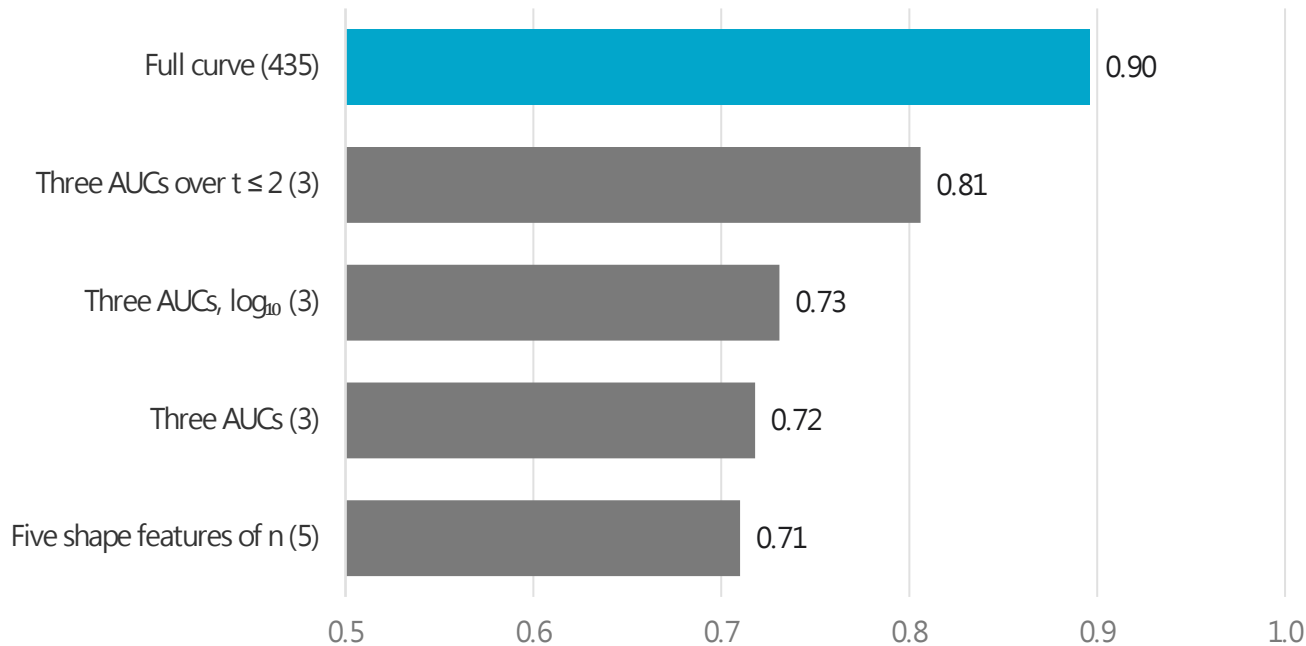
Cluster	Free	Antibiotic	+ Inhibitor
A	10,000	0	0
B	0	3,819	5,991
C	0	6,181	4,009

Clustering separates the antibiotic-free runs but cannot distinguish antibiotic only from + inhibitor.

Which Feature Separates Antibiotic Only From + Inhibitor?

We train an SVM with labels. Summary features give about 0.72 test accuracy; the full 435-number curve gives 0.90.

Test accuracy (axis starts at chance, 0.50)



The full curve contains it

With all 435 numbers per curve, the SVM reaches 0.90 test accuracy.

Most of it is early

76% of the linear SVM weight on $\log_{10} n$ lies in $t < 2$, the time window.

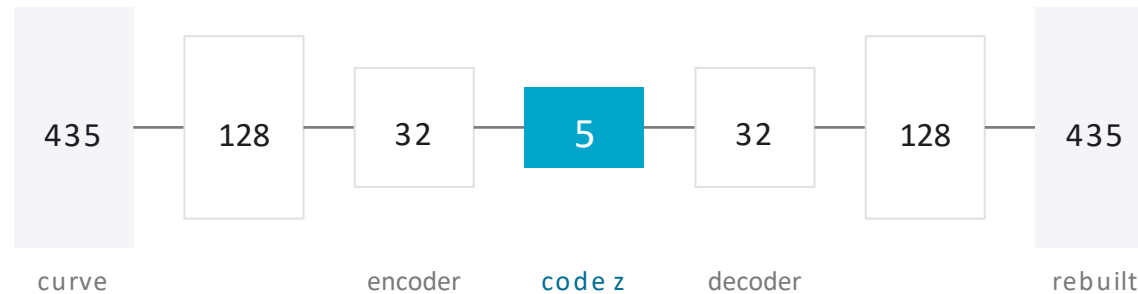
The AUC averages it out

The AUC integrates over $t = 0-24$. Restricting it to $t \leq 2$ raises the accuracy to 0.81.

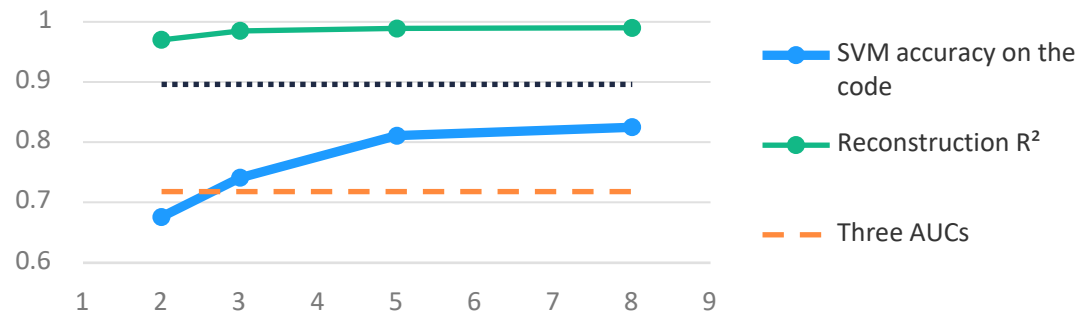
Question: can we find such features from the data, without choosing the time window ourselves?

Learning Features With an Autoencoder

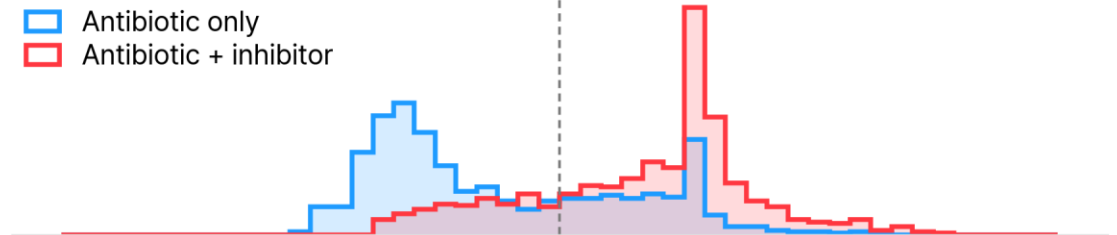
The autoencoder compresses each 435-number curve to r numbers and reconstructs it, without using labels.



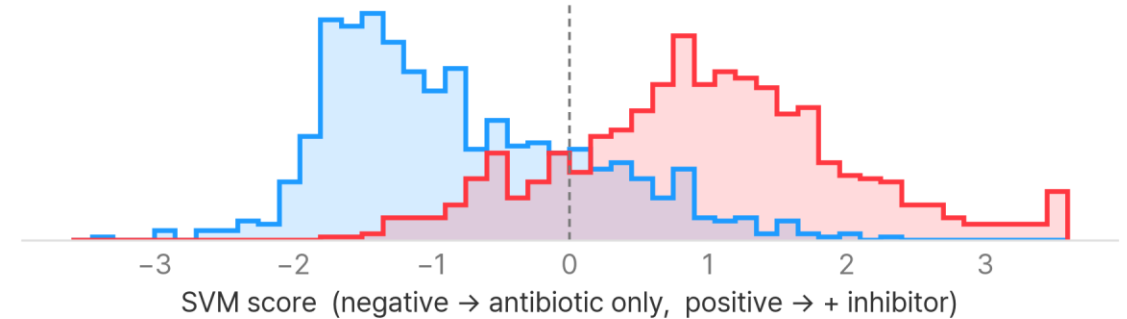
Test accuracy and reconstruction R^2 vs. latent size r



Three AUCs · test accuracy 0.72



Five learned latents · test accuracy 0.81

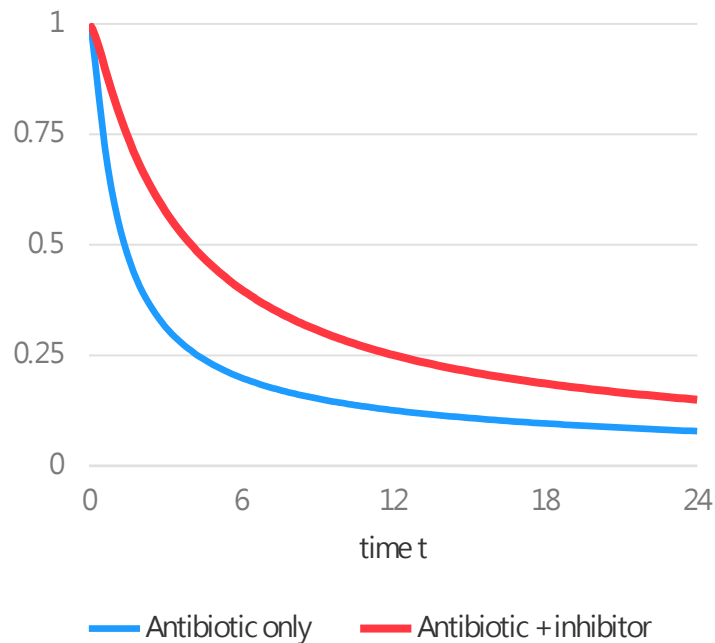


With $r = 5$, an SVM on the latent code reaches 0.81, compared with 0.72 for the three AUCs.

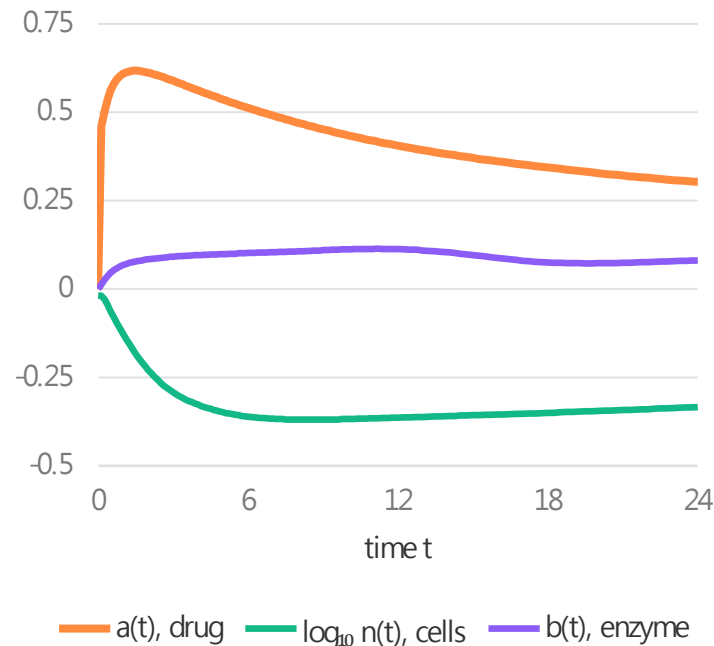
Relating the Latent Code to the Time Series

We correlate the SVM score computed from the latent code with the curves at each time point.

Mean antibiotic $a(t)$



Correlation $r(t)$ with the latent score



Antibiotic $a(t)$

Highest correlation, $r = 0.62$ at $t = 1.5$. With the inhibitor the drug stays longer.

Bacteria $\log_{10} n(t)$

$r = -0.37$ at $t = 8.5$. Fewer cells survive when lasts longer.

Enzyme $b(t)$

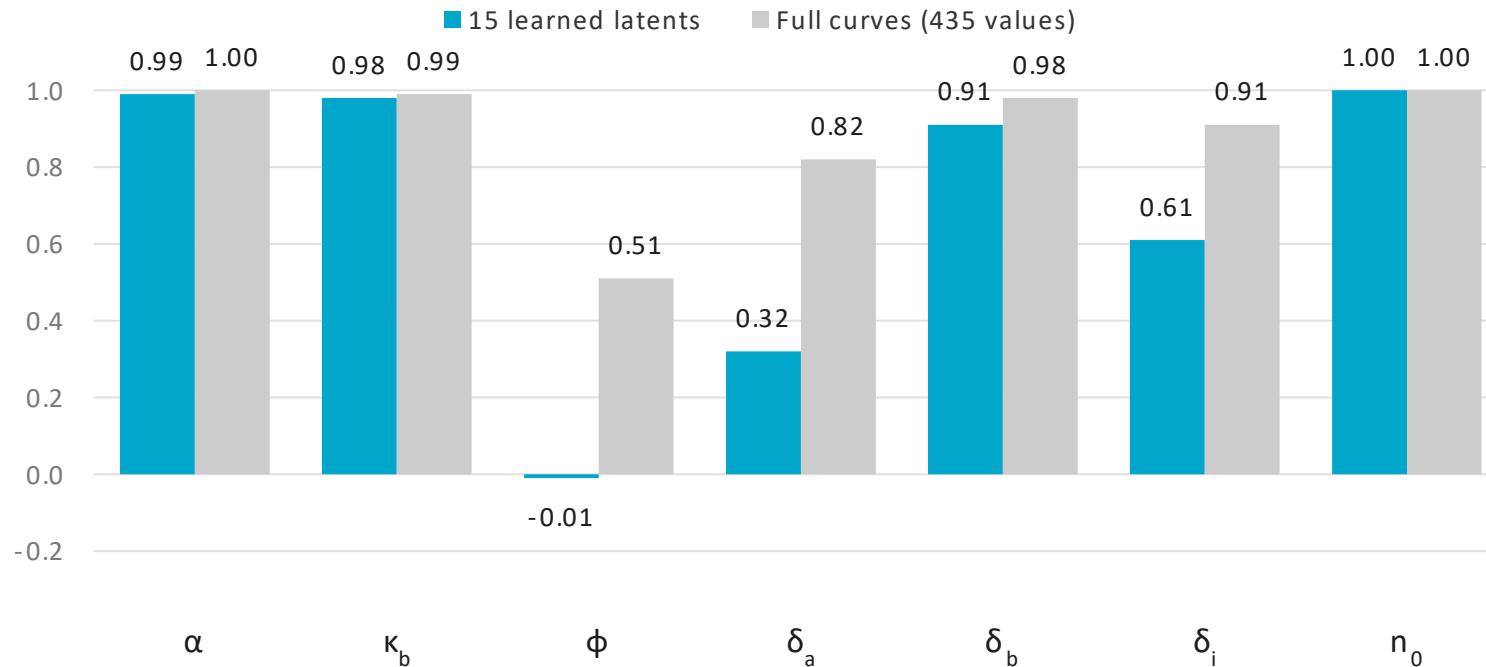
$|r| \leq 0.11$. The enzyme curves are similar in both conditions.

The latent code picks up the early part of the drug curve, the same window found from the SVM weights.

From the Latent Code Back to the Parameters

A small neural network predicts θ from the 15 latents of each parameter set (5 per condition).

R^2 on 2,000 held-out parameter sets (1 = perfect recovery)



Well recovered

$\alpha, \kappa_b, \delta_b, n_0$: R^2 from 0.91 to 1.00.

Partly recovered

δ_i 0.61 and δ_a 0.32. The full curves do better (0.91, 0.91, 0.82).

Not recovered

ϕ : -0.01 from the latents, 0.51 from the curves. effect resembles other drug-loss terms.

Parameters that cannot be recovered point to an identifiability problem in this experimental design.

Conclusion: What Mathematical Biology Does

We learn a model, simulate it, and use the simulation results to build intuition about the biology.

1

Describe the

Mathematical biology
biological processes as
equations. Growth, killing
resistance become rates.

2

Simulate the model

The model generates
experiments that are hard
run in the lab, over a whole
range of parameters and
conditions.

3

Gain intuition

The simulations show
the information is. Here, the
early part of the drug curve
separates the treatments.

4

Return to experiments

The same analysis tells us
parameters an experiment
determine and what to
next.

From Fibonacci to today the approach is the same: build a model, simulate it, and learn from the results.