Applied Computational Mathematics Lab

EPIDEMIC MODELING · MODULE 8

Parameter estimation and calibration

Fitting models to data, and knowing when you can't



6 EPISODES

1. Fitting by least squares
2. Search strategies
3. Gradients and derivative-free refinement
4. Simulation-based and state-space inference
5. Targets, observations and loss design
6. Identifiability and diagnostics

Based on Chapter 8 of T. Lazebnik, Epidemic Modeling: Principles, Algorithms, and Applications (Academic Press, 2026)

acml.teddylazebnik.com

BEFORE WE START

Warm-up: true or false?

CHECK YOURSELF

(a) If a model fits the data closely, its parameters must be close to the truth.

(b) Fitting the cumulative case curve gives trustworthy error bars.

(c) A finer grid is always the best way to search.

(d) A model you can only simulate, with no formula for its likelihood, can still be fitted.

HOW TO USE THIS

Guess now; the module settles each one. Answers at the end.

WHY THIS MODULE

Every model in this course has parameters. This module is about choosing their values from data, and about knowing when the data cannot decide.

BEFORE WE START

This module's map

PART 1

Fitting

Least squares, weights, search strategies, gradients and derivative-free refinement, noisy objectives.

PART 2

Without a likelihood

Approximate Bayesian computation and particle filters for models you can only simulate.

PART 3

Calibration and trust

Targets, observation mappings, losses, constraints, identifiability, misspecification and diagnostics.

GOALS

By the end you should be able to fit a model by weighted least squares, choose a search method for the problem, use simulation-based inference when no likelihood exists, design a calibration around its purpose, and check identifiability with profiles and stress tests.

ONE RULE

Every number in this deck comes from one **simulated outbreak** with a known truth (and, in Part 3, a richer simulated truth), so each method can be checked. All values are made up to show mechanisms.

Applied
Computational
Mathematics
Lab

PART 1

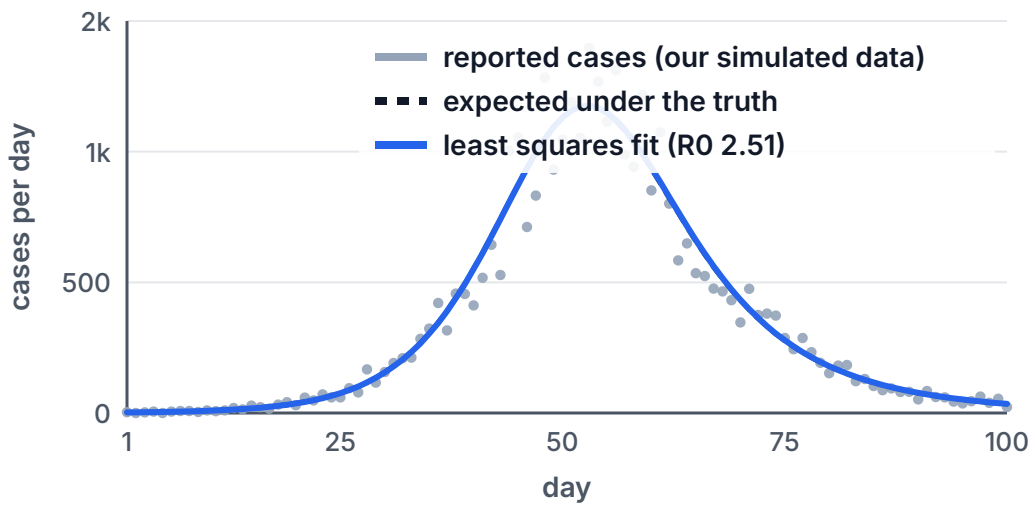
Fitting

What to minimize, and how to search.



PART 1 · LEAST SQUARES

One outbreak with a known truth



Our simulation: 34,851 cases reported over 100 days; peak on day 53.

OUR SIMULATED OUTBREAK (MADE UP)

model	SIR, 20 first infections
town	100,000
transmission rate beta	0.25 per day
recovery rate gamma	0.1 per day (10 days infectious)
R0	2.5
reported	40% of infections
noise	negative binomial, k = 50

IN PLAIN WORDS

We pretend not to know beta and gamma, fit them, and compare with the truth.

The least squares fit

$$\hat{\theta} = \arg\min_{\theta} \sum_{k=1}^{K} w_k \left(y_k - \hat{y}_k(\theta)\right)^2$$

- For any θ = (β, γ), run the model, compare with the data day by day, square the misses and add them up (optionally weighted).
- If the noise is normal with known variances σ_k^2, the weights $w_k = 1/\sigma_k^2$ make this the maximum likelihood estimate.
- Counts are not normal and their noise grows with the count, so the weights matter: divide by the expected noise, or fit on the log scale.

TRY IT

A day with 900 expected cases and Poisson noise has a standard deviation of 30. What weight does weighted least squares give it, compared with a day of 9 expected cases?

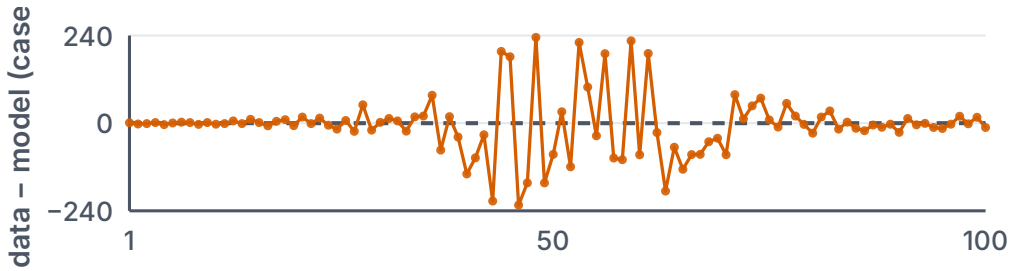
FIT TO	BETA	GAMMA	R0
daily counts	0.250	0.100	2.50
daily counts, weighted	0.249	0.099	2.51
log of daily counts	0.248	0.098	2.54
cumulative counts	0.254	0.104	2.43

Our fits to the same data (truth: 0.25, 0.10, 2.50). All four point estimates are close; their error bars are not (two slides on).

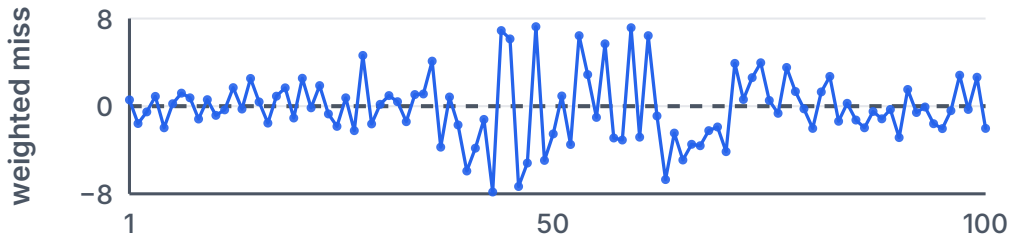
HOW IT IS SOLVED

Solvers such as Gauss–Newton and Levenberg–Marquardt use the structure of a sum of squares: they linearize the model around the current guess and solve a small least squares problem for the step.

Weights and the log scale



Our fit: plain misses grow with the count, so the peak dominates a plain sum of squares.



The same misses divided by the noise expected that day (square root of the model value): more even.

- Count noise grows with the count: for Poisson the variance equals the mean; for negative binomial it is mean + mean²/k.
- Weighted least squares divides each squared miss by the expected variance, so a miss of 30 on a day of 900 counts like a miss of 3 on a day of 9.
- Fitting log(y + 1) instead makes a 10% miss count the same at any size; it also down-weights the peak, and needs care with zeros.
- Best of all: write the likelihood of the noise model you believe (Poisson, negative binomial) and maximize it.

IN PLAIN WORDS

The weights encode what you believe about the noise. Wrong weights still give a fit, but its error bars will be wrong.

PART 1 · LEAST SQUARES

The cumulative trap

FIT TO	MEAN R0	CLAIMED ±	REAL SPREAD	95% INTERVALS HELD THE TRUTH
daily counts	2.50	0.083	0.158	74%
daily, weighted	2.56	0.104	0.156	82%
log of daily	2.66	0.196	0.374	88%
cumulative	2.50	0.004	0.149	5%

Our check: 200 new noisy datasets from the same truth, each fitted four ways. Claimed: the median standard error from each fit; real: the spread of the 200 estimates.

- Each day of a cumulative curve contains all earlier days' errors, so its residuals are strongly correlated. The usual error formula assumes independent residuals and reports far too little uncertainty.
- Here the cumulative fit claimed ±0.004 on R0; the estimates really spread by ±0.15.
- The daily fits were closer, but still too confident: the negative binomial noise is wider than the Poisson-like weights assume.

IN PLAIN WORDS

Fit incidence, not cumulative counts, and use a noise model that matches the data (negative binomial, or a likelihood with an overdispersion parameter). Check error bars by simulation.

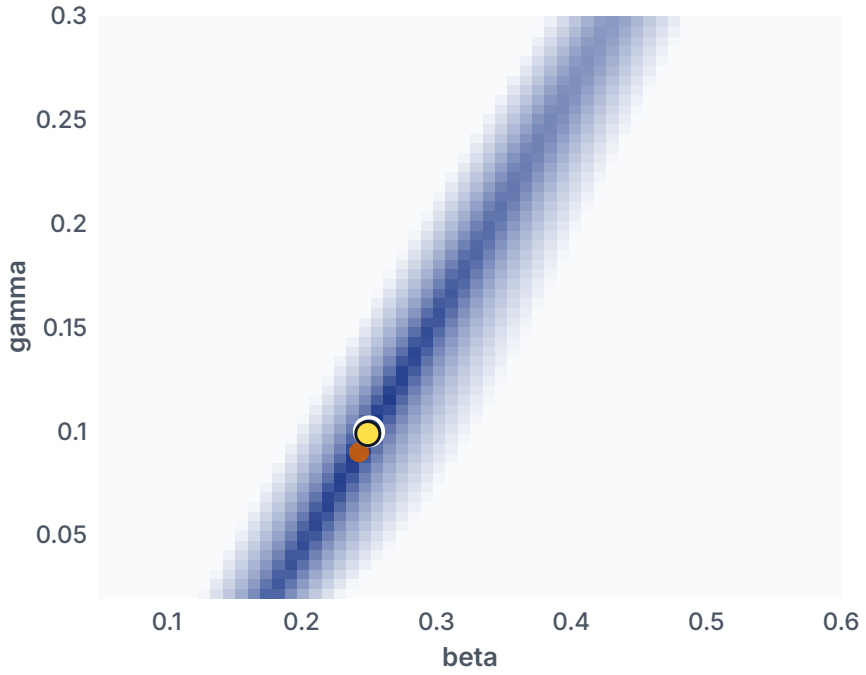
A PRACTICAL CHECK

Parametric bootstrap: simulate new datasets from the fitted model, refit each one, and use the spread of the refits as the uncertainty. It is exactly the check we ran above.

TRY IT

Why would smoothing daily counts with a 7-day moving average before fitting cause a similar (smaller) problem?

Grids, and the curse of dimensionality



Our loss map: weighted least squares over beta and gamma (log scale; dark is small). Star: best fit; orange: best point of a 41 × 41 grid; ring: truth.

SEARCH	MODEL RUNS	RESULT
41 × 41 grid	1,681	7% above the best loss
coarse to fine (3 zooms of 11 × 11)	363	1.0% above

PARAMETERS (10 VALUES EACH)	MODEL RUNS	TIME AT 8.69 MS PER RUN
2	100	0.9 s
4	10,000	1.4 min
6	1,000,000	2.4 hours
8	100,000,000	10 days

- A grid is transparent and parallel, and it maps the landscape: valleys, flat directions and second minima show at a glance.
- Its cost grows as (values per parameter) to the power of the number of parameters.

Bayesian, genetic and swarm search

SURROGATE

Bayesian optimization

A Gaussian process models the loss from the runs so far, with uncertainty; an acquisition rule (here expected improvement) picks the next run.

POPULATION

Genetic algorithm

Select the better candidates (tournaments), cross them (blends) and mutate them; keep the best (elitism).

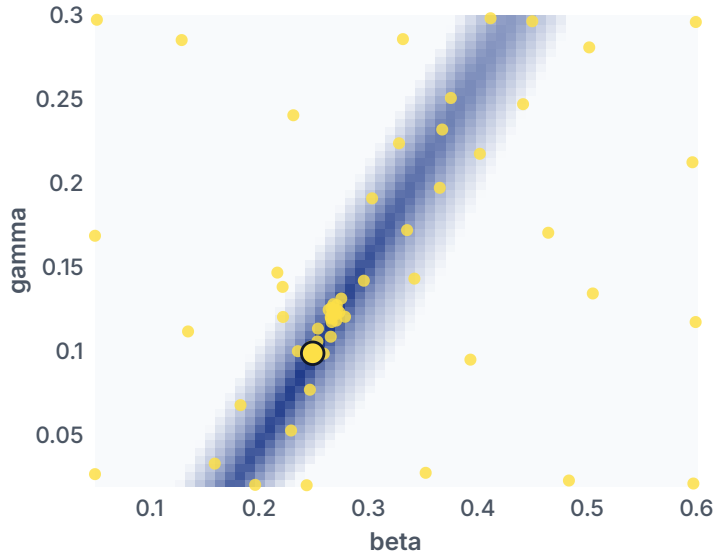
FLOCK

Particle swarm

Each particle has a velocity pulled toward its own best position and the swarm's best, with some inertia.

METHOD (20 TRIES EACH)	BUDGET	RUNS TO WITHIN 10% (MEDIAN)	TRIES WITHIN 1%
Bayesian optimization	60	47	6 of 20
genetic algorithm	452	182	9 of 20
particle swarm	450	209	10 of 20
random guessing	600	not in 600	0 of 20

Our runs on the outbreak above. Within x%: the best loss found so far is at most x% above the best fit.

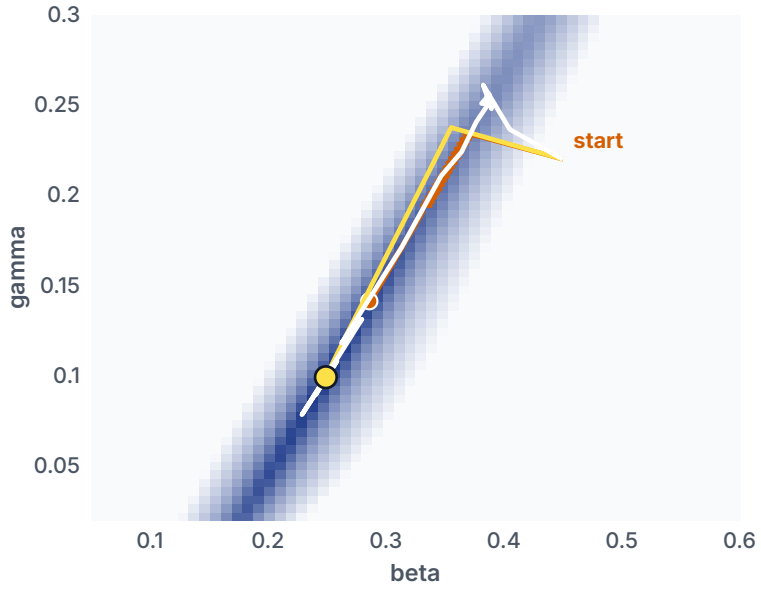


The 60 runs of one Bayesian optimization: most land in the valley.

IN PLAIN WORDS

Global searches find the right valley quickly but polish slowly. Hand the best point to a local method (next slides).

Gradients, curvature and Nelder–Mead

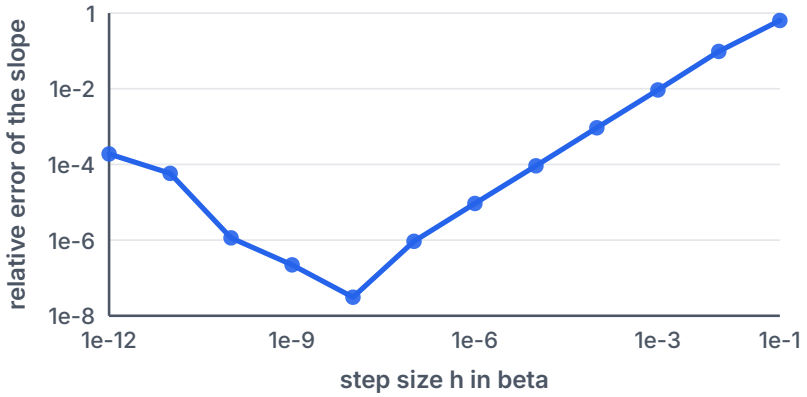


Our paths from (0.45, 0.22): gradient descent (orange) zigzags and stalls; BFGS (yellow) and Nelder–Mead (white) reach the best fit.

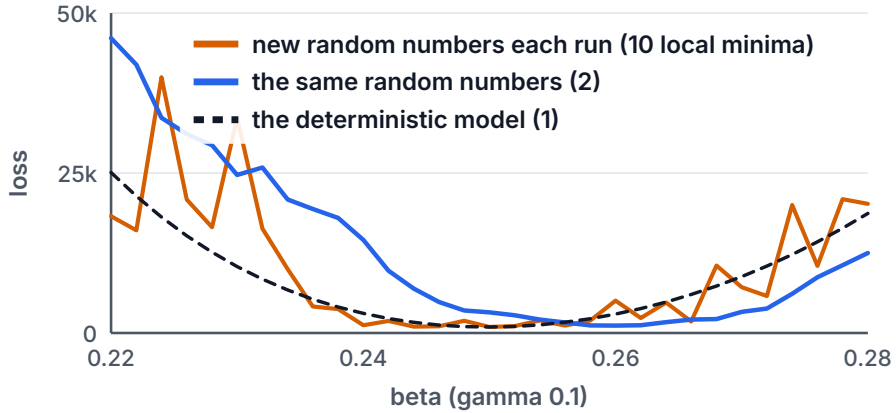
METHOD	MODEL RUNS	RESULT
gradient descent (finite differences)	1,502	still 48% above after 300 steps
Gauss–Newton (Levenberg–Marquardt)	21	the best fit
BFGS (finite differences)	22 to within 1%	127 in all
Nelder–Mead	50 to within 1%	123 in all

- Gradients: finite differences (one extra run per parameter), sensitivity equations solved with the model, or automatic differentiation.
- In a long curved valley the steepest direction points across the valley, not along it. Methods that use curvature (Gauss–Newton, quasi-Newton such as BFGS) take steps along it.
- Nelder–Mead needs only loss values: it reflects, expands, contracts or shrinks a simplex of p + 1 points.

Step sizes and noisy objectives



Our check: slope of the cases reported in the first 60 days with respect to beta, by forward differences. Too big a step measures a stretch; too small, rounding.



A stochastic SIR (binomial chain, quarter-day steps): the loss along beta with new random numbers at every run, with the same random numbers, and for the deterministic model.

- Same parameters, 20 runs with new random numbers: the loss varied with a standard deviation of 4,773; raising beta by 5% moved its average by only 297.
- Average replicates, reuse random numbers (common random numbers, Module 6), or use methods built for noise.

Check yourself: Part 1

CHECK YOURSELF

1. Show that for normal noise with known standard deviations σ_k, maximizing the likelihood is the same as weighted least squares. What are the weights?

2. A model has 5 parameters. A grid with 20 values each takes how many runs? At 10 ms per run, how long?

3. Why can plain gradient descent be slow in a long curved valley, and what does a quasi-Newton method do differently?

TIP

Answers are at the end of the deck. Try them before you look.

Applied Computational Mathematics Lab

PART 2

Without a likelihood

Fitting models you can only simulate.



PART 2 · ABC

Approximate Bayesian computation

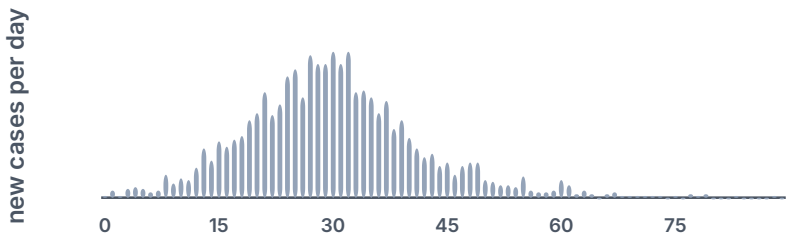
ABC REJECTION

1. draw θ from the prior
2. simulate data with θ
3. compute summaries s(simulated) and a distance to s(observed)
4. keep θ if the distance is among the smallest (or below a tolerance ε)
5. the kept θ approximate the posterior

$$p_\varepsilon(\theta \mid y) \propto p(\theta)\ \Pr\left(d(s(y^{\text{sim}}), s(y)) \le \varepsilon \mid \theta\right)$$

IN PLAIN WORDS

A smaller tolerance ε gives a better approximation and needs more simulations.



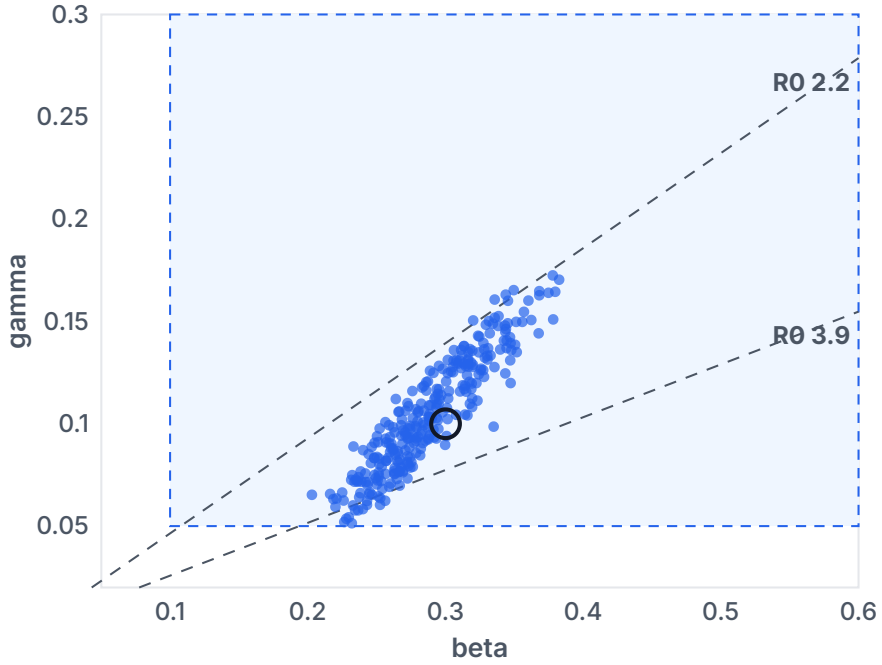
The "observed" data: one random outbreak in a town of 2,000 (5 first cases, R0 3): 1,886 cases, peak of 83 on day 30.

OUR ABC SETUP

prior	beta 0.1–0.6, gamma 0.05–0.3 (uniform)
simulations	50,000 (binomial chain, daily steps)
summaries	total cases, peak day, peak height
distance	scaled Euclidean
kept	closest 500 (1%)
fizzled	17% of simulations

PART 2 · ABC

What ABC learned



Our kept draws (300 of 500 shown) inside the prior box (dashed); dashed lines: the 90% range of R0; ring: the truth.

POSTERIOR (KEPT DRAWS)	5%	MEDIAN	95%
beta	0.24	0.29	0.37
gamma	0.06	0.11	0.17
R0	2.15	2.71	3.88

Truth: beta 0.3, gamma 0.1, R0 3. Correlation of beta and gamma among the kept draws: 0.93.

- R0 is well determined; beta and gamma separately are not: the kept draws lie along a ray of constant ratio.
- The interval is wide partly because one small town is noisy: chance alone moves its timing and peak.

TRY IT

The kept draws stretch along a line through the origin. What does that line mean, and what kind of data would shorten it?

Tolerance and summaries

KEPT SHARE	DRAWS	R0, 90% INTERVAL
10.0%	5,000	1.7–4.0
1.0%	500	2.2–3.9
0.1%	50	2.6–3.6

Our ABC with all three summaries, out of 50,000 simulations.

- Tighter tolerance: closer to the true posterior, but fewer draws, so more simulations are needed for the same number.
- Refinements: regression adjustment of the kept draws, and sequential ABC that narrows the prior step by step (ABC-SMC).

SUMMARIES USED (1% KEPT)	R0, 90% INTERVAL	BETA, 90% INTERVAL
all three	2.2–3.9	0.24–0.37
total cases only	2.6–3.1	0.20–0.59
total and peak height	2.4–4.0	0.22–0.33
peak day and height	1.6–3.5	0.24–0.47

The same simulations, compared on different summaries.

- Total cases alone pin R0 tightly: the final size depends on R0 only (Module 3). But they say almost nothing about beta: speed needs timing or height.
- Noisy summaries (here the peak day in a small town) can widen the result. Choose summaries that carry information about the parameters you need.

State-space models and particle filters

$$x_t \sim p_\theta(x_t \mid x_{t-1}), \qquad y_t \sim p_\theta(y_t \mid x_t)$$

BOOTSTRAP PARTICLE FILTER (EACH DAY)

1. move every particle x with the model (here S, I and log R, which follows a random walk)
2. weight it by the likelihood of today's count
3. resample particles in proportion to their weights
4. the weighted particles approximate the hidden state, and R

IN PLAIN WORDS

The filter never needs the likelihood of the whole series in closed form: it only simulates forward and scores one day at a time.

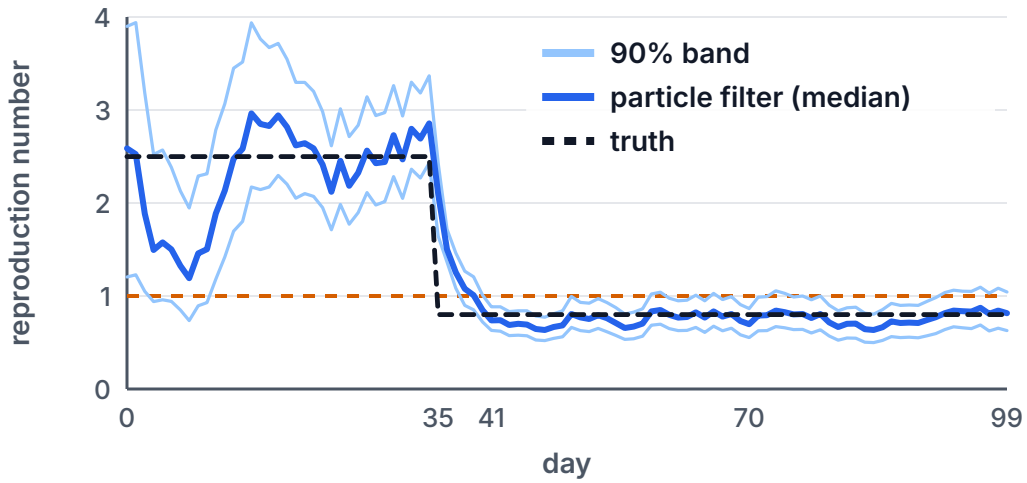
JOINT STATE AND PARAMETERS

Unknown fixed parameters can be added to the state with a small random walk, estimated by iterated filtering (repeated filtering with shrinking random walks), or sampled with particle MCMC, which uses the filter's likelihood estimate inside a Metropolis–Hastings sampler.

WATCH THE WEIGHTS

Effective sample size: 1 / sum of squared weights, from 1 (all weight on one particle) to the number of particles. A low value means the filter is relying on very few guesses.

Tracking R through a lockdown



Our run: the filter's estimate of R with its 90% band, and the truth.

- Our test: 3,000 particles, negative binomial reports (40% reported); R drops from 2.5 to 0.8 on day 35; log R follows a random walk in the filter.
- The whole 90% band fell below 1 on day 41, 6 days after the change.
- The bands held the truth on 88% of days; the average effective sample size was 2,074 of 3,000.
- Early on, with few cases, the band is wide: little data, much uncertainty.

TRY IT

Why would a random walk with a larger step make the filter react faster to the lockdown, and what would it cost?

Check yourself: Part 2

CHECK YOURSELF

4. In ABC, what happens to the kept draws as the tolerance goes to zero, and what is the practical price?

5. Total cases alone pinned R0 but not beta. Explain using the final-size relation.

6. In a particle filter, all weight ends up on a few particles. What is this called, and what helps?

TIP

Answers are at the end of the deck.

Applied Computational Mathematics Lab

PART 3

Calibration and trust

Fit for a purpose, and checked.



Fit for a purpose

$$\hat{\theta}_{\text{cal}} = \arg\min_{\theta} \sum_{k} \rho_k \, \ell\big(z_k, \hat{z}_k(\theta)\big) + \lambda \, \mathcal{R}(\theta)$$

- z_k: calibration targets made from the data (daily cases, peak timing, bed occupancy ...); $\hat{z}_k(\theta)$: the same quantity from the model, through the observation mapping.
- ℓ: the loss; ρ_k: weights; $\mathcal{R}$: a penalty (regularization, priors).
- Estimation asks which parameters the data support. Calibration asks how to tune a model, usually imperfect, so that it serves a decision.

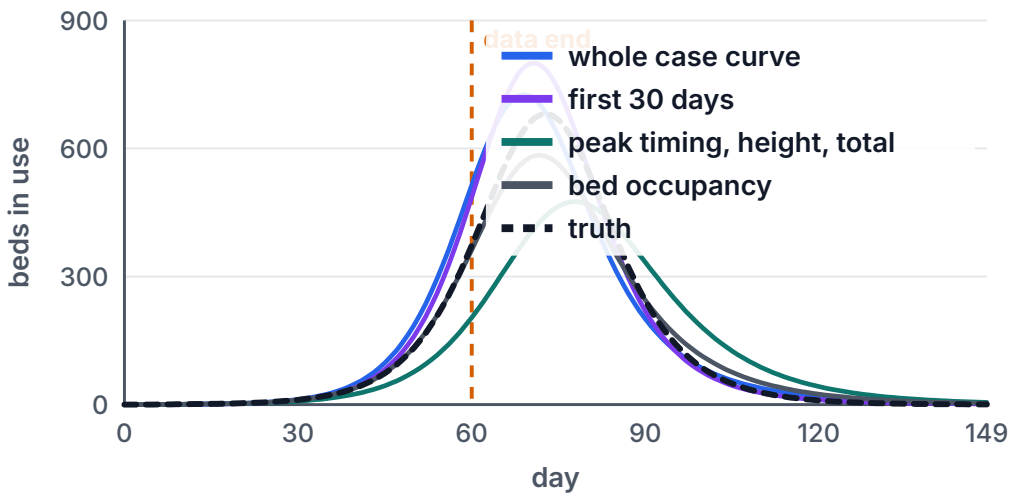
PART 3'S TEST (MADE UP)	
the truth	SEIR, 4 latent days, R0 2.5
testing	reports rise from 20% to 50% of infections over 40 days
hospital	3% of infections, 5 days later, 8 days in a bed
the analyst	SIR with a reporting share, first 60 days of cases
the question	peak beds in use

IN PLAIN WORDS

The truth is richer than the model on purpose: real calibrations always fit an imperfect model.

PART 3 · CALIBRATION

Four targets, four answers



Bed occupancy predicted by the same SIR model calibrated to four targets, and the truth.

CALIBRATION TARGET	R0	PEAK BEDS	PEAK DAY
whole case curve	3.51	726	69
first 30 days	8.22	800	71
peak timing, height, total	1.93	476	78
bed occupancy	2.26	584	72
truth	2.5	681	73

Our calibrations: same model, same 60 days, four targets.

- Same model, same data: from 476 to 800 peak beds.
- Calibrating to bed occupancy itself did not win here: the occupancy peak came after the 60 days of data.
- Choose targets for the purpose, and check the answer on the quantity that matters, on data the fit did not use.

PART 3 · CALIBRATION

The observation mapping and the loss

FIRST 40 DAYS, FITTED GROWTH RATE	PER DAY	DOUBLING
constant reporting share assumed	0.149	4.6 days
testing ramp in the mapping	0.130	5.3 days
truth	0.126	5.5 days

Our fits to the first 40 days, while testing ramped up.

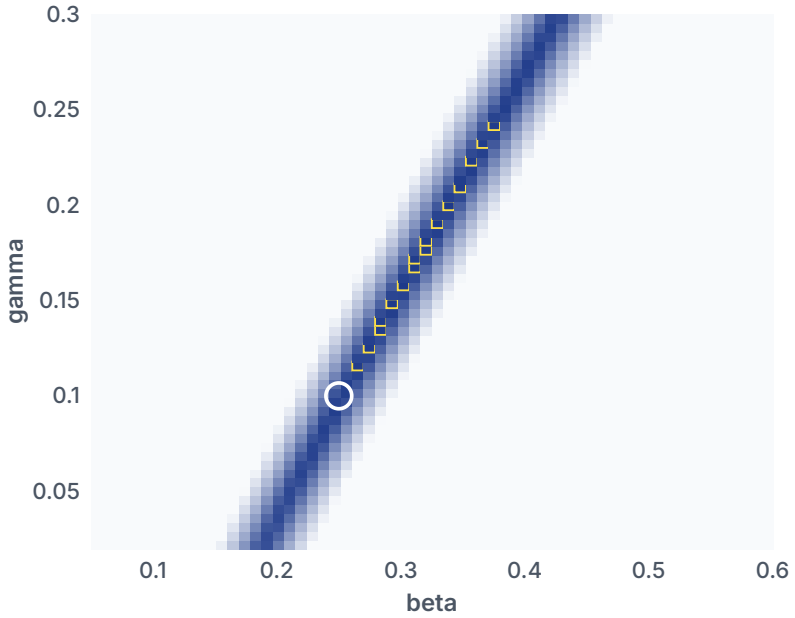
- The observation mapping turns model states into what is measured: reporting share, delays, calendar, test positivity, hospital admissions.
- If testing grows, cases grow faster than infections. Leaving that out of the mapping makes the epidemic look faster than it is.

LOSS (60 DAYS OF CASES)	R0	MISS, FIRST 25 DAYS	MISS AT THE PEAK
squared errors	2.35	107%	9%
weighted by expected noise	3.51	75%	10%
log scale	7.42	58%	17%

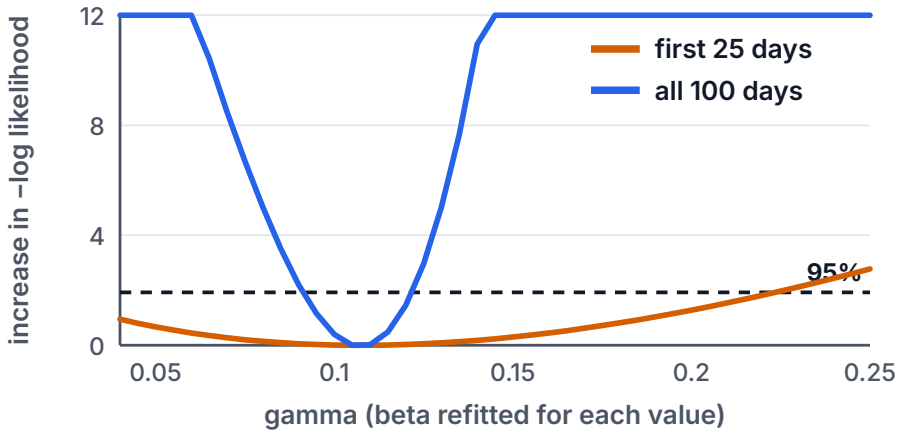
Same SIR model and data; only the loss changes. The truth was R0 2.5.

- A misspecified model cannot fit everything: the loss decides which errors it accepts, and the parameters move to compensate.
- Several of these fits pushed gamma to the lower bound we allowed (0.02, a 50-day infectious period): a warning sign, not an answer.
- Check the calibrated model on the quantity that matters, not only on the fitting target.

Valleys, profiles and outside information



Our loss map for the first 25 days only; yellow: fits within 5% of the best. Ring: truth.



Profiles: fix gamma, refit beta, record how much worse the fit gets; below the dashed line the data cannot rule a value out (95%).

95% PROFILE INTERVAL	FIRST 25 DAYS	ALL 100 DAYS
gamma	0.04–0.22 (edge of the search)	0.095–0.12
R0	1.6–5.1	2.2–2.6
growth rate beta – gamma	0.136–0.163	0.148–0.150

- Early data measure the growth rate, not R0: many (β, γ) pairs share the same β − γ.

PART 3 · IDENTIFIABILITY

Breaking ties, and ties that never break

FIRST 25 DAYS	R0
no outside information (95% profile interval)	1.6–5.1
prior: infectious for 10 ± 2 days (best fit)	2.3
truth	2.5

Our fits: the prior enters as a penalty on 1/gamma.

REPORTING SHARE AND POPULATION, FITTED TOGETHER	
reporting share	0.49 (truth 0.40)
population	76,892 (truth 100,000)
their product	37,680 (truth 40,000)
correlation of the two estimates	-0.997

Our fit to all 100 days.

TWO KINDS

Structural non-identifiability: no amount of perfect data separates the parameters (the reporting share and the population here); estimate their product, or fix one from other sources and say so. Practical non-identifiability: these data cannot, but more or different data could (beta and gamma from 25 days); outside information such as clinical studies or serosurveys helps.

TRY IT

A serosurvey finds that 30% of the town has been infected by day 60. Which of the two problems does it help with, and how?

PART 3 · DIAGNOSTICS

Misspecification, stress tests and trade-offs

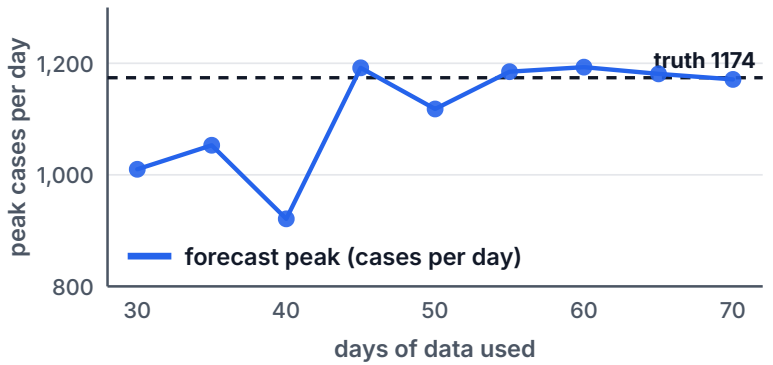
FITTED TO SEIR DATA	SIR (WRONG)	SEIR (RIGHT)
loss (weighted)	818	775
residual lag-1 correlation	0.03	-0.01
gamma	0.020 (the bound)	0.122
R0	8.1	3.2

Our fits; truth R0 2.5. The wrong model fits almost as well, with no pattern in its residuals.

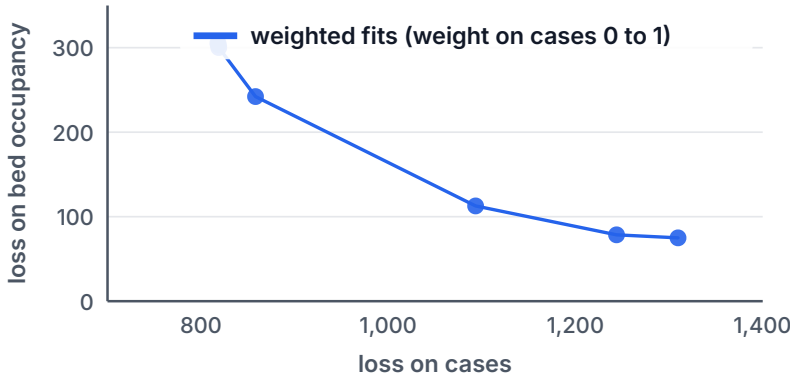
$$z_k = \hat{z}_k(\theta) + \delta_k + \varepsilon_k$$

A discrepancy term δ separates structural mismatch from noise ε, so parameters stop compensating for it.

- A good fit is not proof: check parameters against outside knowledge, test on data the fit did not see, and refit on different windows.



Stress test: refits on 30 to 70 days. R0 ranged 1.8–2.6.



Cases against bed occupancy: each point is one weighting; the curve of compromises is a Pareto front, and choosing a point is a decision.

PART 3 · PRACTICE

A calibration pipeline

ONE OBJECTIVE FUNCTION, MANY OPTIMIZERS

1. simulate: latent states from θ (seeded, versioned)
2. observation mapping: from states to predicted observables (reporting, delays, calendar)
3. targets: the same extraction for data and model
4. loss: weights, transformations, penalties
5. optimizer: global search, then local refinement
6. evaluation: held-out windows, other streams, diagnostics

IN PLAIN WORDS

Keep the pieces separate so a change in the observation mapping or the loss does not mean rewriting the model, and every optimizer sees the same objective.

LOG

Everything

Parameters, seeds, losses and versions for every run, so a result can be traced and rerun.

CHECK

Before trusting

Profiles or posteriors, residuals by phase, out-of-window tests, refits on other windows, parameters at their bounds.

TRY IT

Your calibrated model forecasts hospital beds well but needs a reporting share of 95%. What do you check, and what might you change?

Check yourself: Part 3

CHECK YOURSELF

7. With 25 days of exponential growth, why can the data pin down $\beta - \gamma$ but not $R0 = \beta / \gamma$?

8. Give one structural and one practical identifiability problem from this module.

9. Two calibrations of the same model give 584 and 726 peak beds. How would you decide which to report?

TIP

Answers are on the next slides.

Answers: warm-up and Parts 1–2

- **(a)** False: an SIR model fitted to SEIR data fitted almost as well and gave R0 8.1 (truth 2.5). **(b)** False: its 95% intervals held the truth 5% of the time. **(c)** False: cost explodes with parameters, and the optimum falls between points. **(d)** True: ABC and particle filters.
- **1.** $-\log L = \sum_k (y_k - \hat{y}_k)^2/(2\sigma_k^2) + \text{const}$, so maximizing L minimizes $\sum_k w_k (y_k - \hat{y}_k)^2$ with $w_k = 1/\sigma_k^2$.
- **2.** 20^5 = 3.2 million runs; at 10 ms, 32,000 s, about 9 hours. **3.** The steepest direction points across the valley, so steps zigzag; quasi-Newton methods build a curvature estimate and step along the valley.
- **4.** They approach the posterior given the summaries; the acceptance rate falls, so many more simulations are needed. **5.** Final size depends only on R0 ($z = 1 - e^{-R0\ z}$), so any β, γ with the same ratio gives the same total.
- **6.** Weight degeneracy (a low effective sample size): more particles, resampling each step, better proposals, or a smaller observation noise mismatch.

TRY-IT ANSWERS

Try-it answers: Poisson weight 1/900 against 1/9, so the big day counts 100 times less per squared case. Smoothing makes neighbouring residuals correlated, so error bars shrink falsely. A larger random-walk step lets R jump faster but makes the estimate noisier on stable days.

Answers: Part 3

- **7.** Early on S is close to N, so cases grow like $e^{(\beta - \gamma)t}$: every pair with the same $\beta - \gamma$ predicts the same curve. R0 needs the bend of the curve (later data) or outside information on γ.
- **8.** Structural: the reporting share and population size (only their product). Practical: β and γ from the first 25 days (more data separated them).
- **9.** Check both on held-out data or other streams for the quantity that matters (beds), look at parameter plausibility and at refits on other windows, and report the range if the doubt remains.

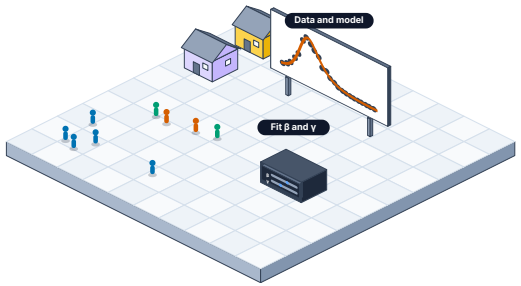
TRY-IT ANSWERS

Try-it answer: a 95% reporting share is implausible, so the fit is compensating for something: check the observation mapping (testing changes, delays), the population at risk, and the model structure (for example a latent period); refit with a plausible bound or prior and compare forecasts.

CREDITS

Sources and credits

- This deck accompanies Chapter 8 of T. Lazebnik, Epidemic Modeling: Principles, Algorithms, and Applications (Academic Press, 2026), the course textbook. It follows the chapter's topics in our own words; no text, figures, tables, exercises or code are copied from the book.
- The text, figures, examples and simulations were written, drawn and run for ACML; nothing is copied from the book. Numbers labelled "our run" come from our own code; counts from agencies and papers are credited where they appear.
- All numbers and curves come from our own simulated outbreaks and fits (production/explains/sims/epi_m08_sims.py), with made-up values chosen to show mechanisms, not to describe any real pathogen or place.
- Named methods, described in our own words: least squares and Gauss–Newton; Levenberg–Marquardt (Levenberg, 1944; Marquardt, 1963); BFGS (Broyden, Fletcher, Goldfarb and Shanno, 1970); Nelder–Mead (Nelder and Mead, 1965); particle swarm optimization (Kennedy and Eberhart, 1995); approximate Bayesian computation (for example Beaumont, Zhang and Balding, 2002); particle filters (Gordon, Salmond and Smith, 1993); iterated filtering (Ionides, Bretó and King, 2006); particle MCMC (Andrieu, Doucet and Holenstein, 2010).
- Companion videos: **Epidemic Modeling**, module 8, episodes 32–37. More from the lab: acml.teddylazebnik.com



Applied Computational Mathematics Lab

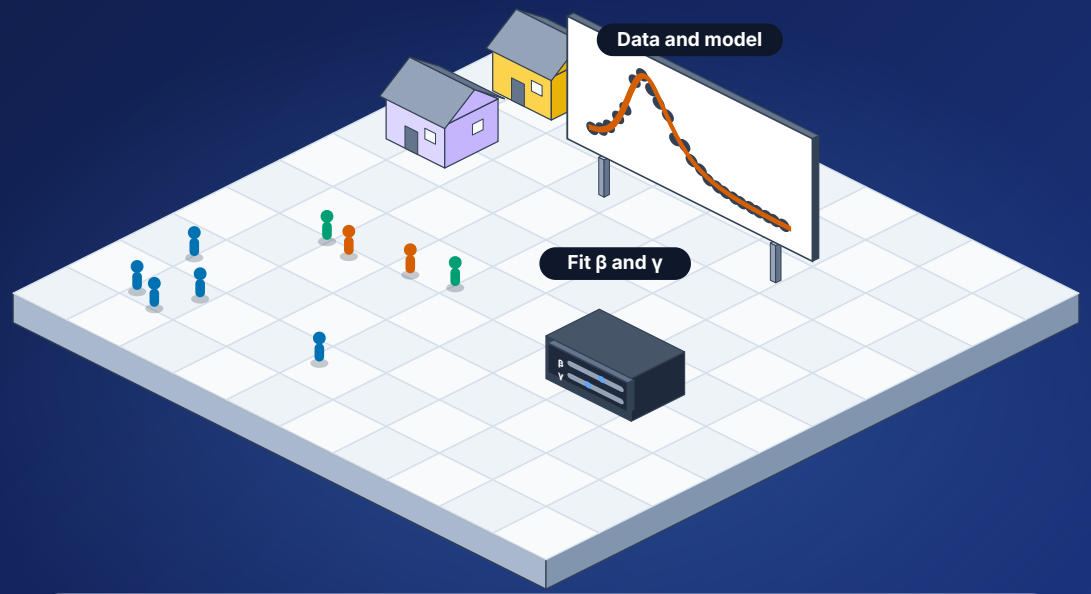
END OF MODULE 8

Parameter estimation and calibration

YOU COVERED

- Fitting by least squares
- Search strategies
- Gradients and derivative-free refinement
- Simulation-based and state-space inference
- Targets, observations and loss design
- Identifiability and diagnostics

Module 8 complete



UP NEXT · MODULE 9

Surveillance, mobility and genomic data

What the data streams see, and what they miss

Based on Chapter 8 of T. Lazebnik, Epidemic Modeling: Principles, Algorithms, and Applications (Academic Press, 2026)