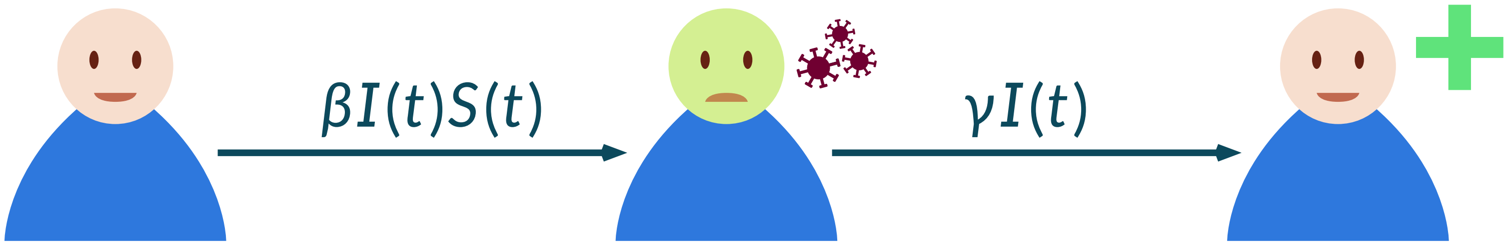
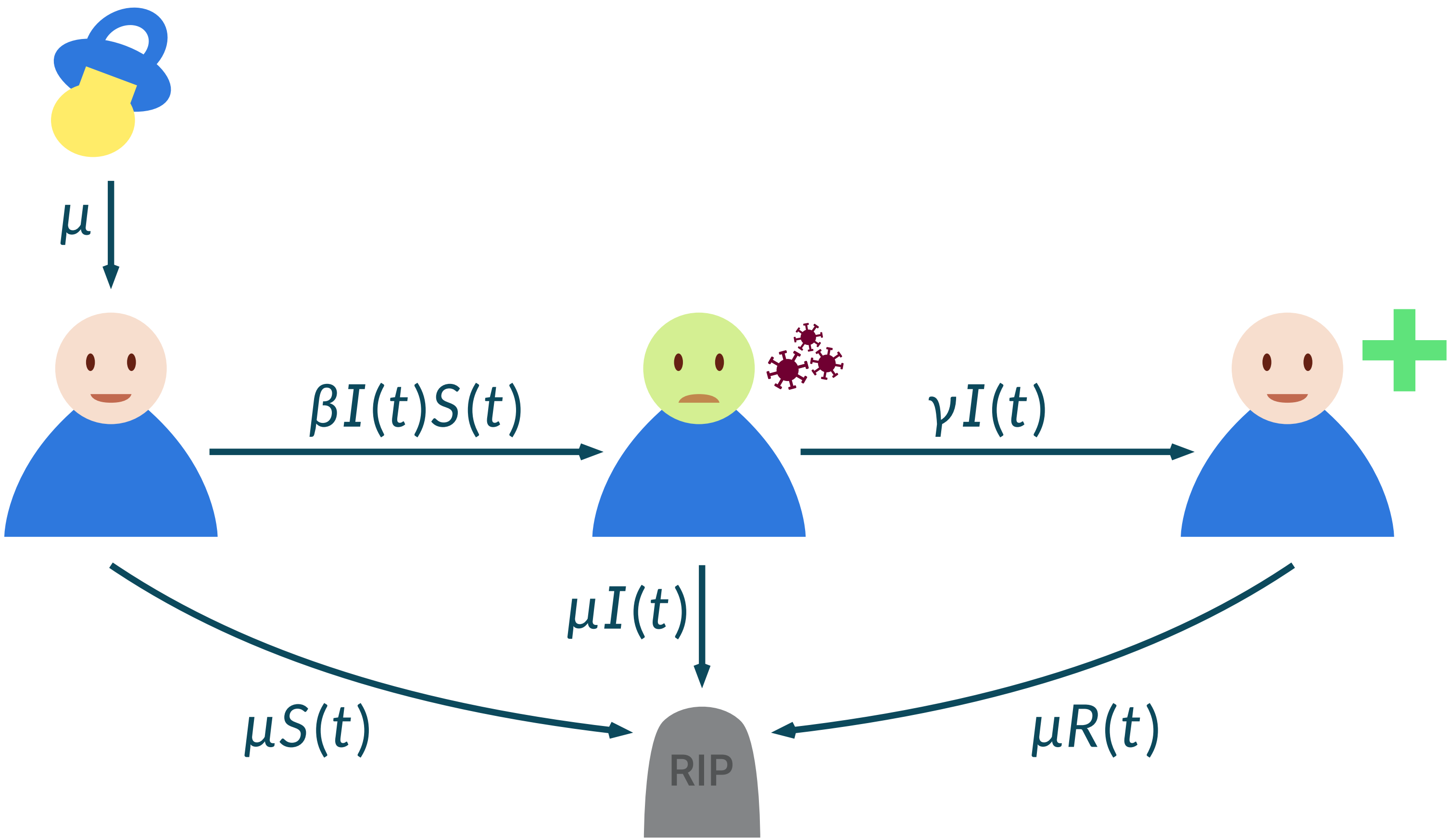
KU LEUVEN

Exploration in Infinite Dimensions:
The Basins of Attraction of Dynamical Systems with Delay

Evert Provoost (KU Leuven) Francesca Scarabel (U. Leeds)


$\beta I(t)S(t)$
$\gamma I(t)$

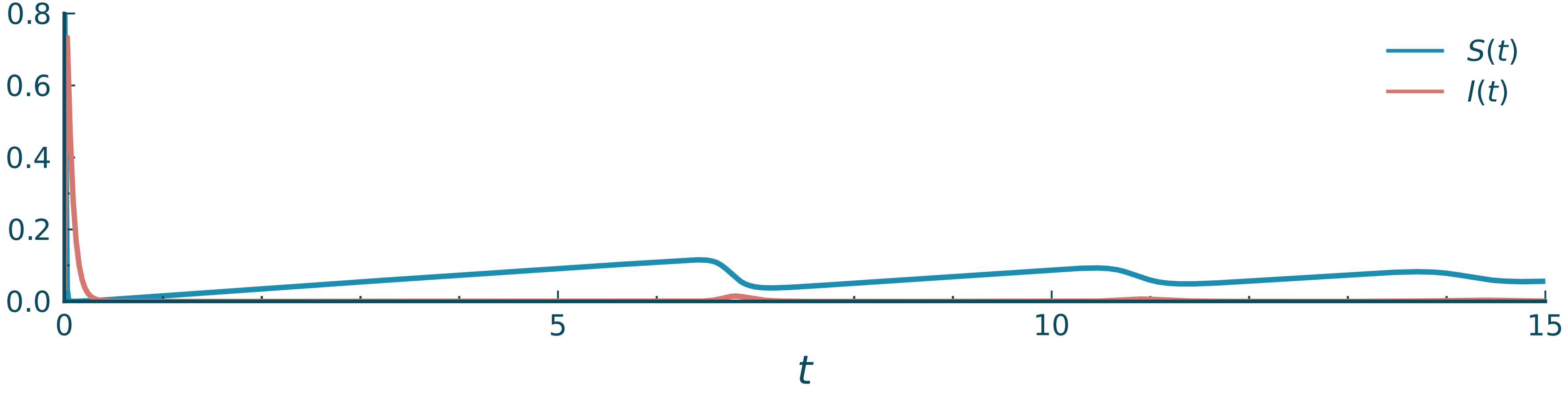

μ
βI(t)S(t)
γI(t)
μI(t)
μS(t)
μR(t)
RIP

SIR model

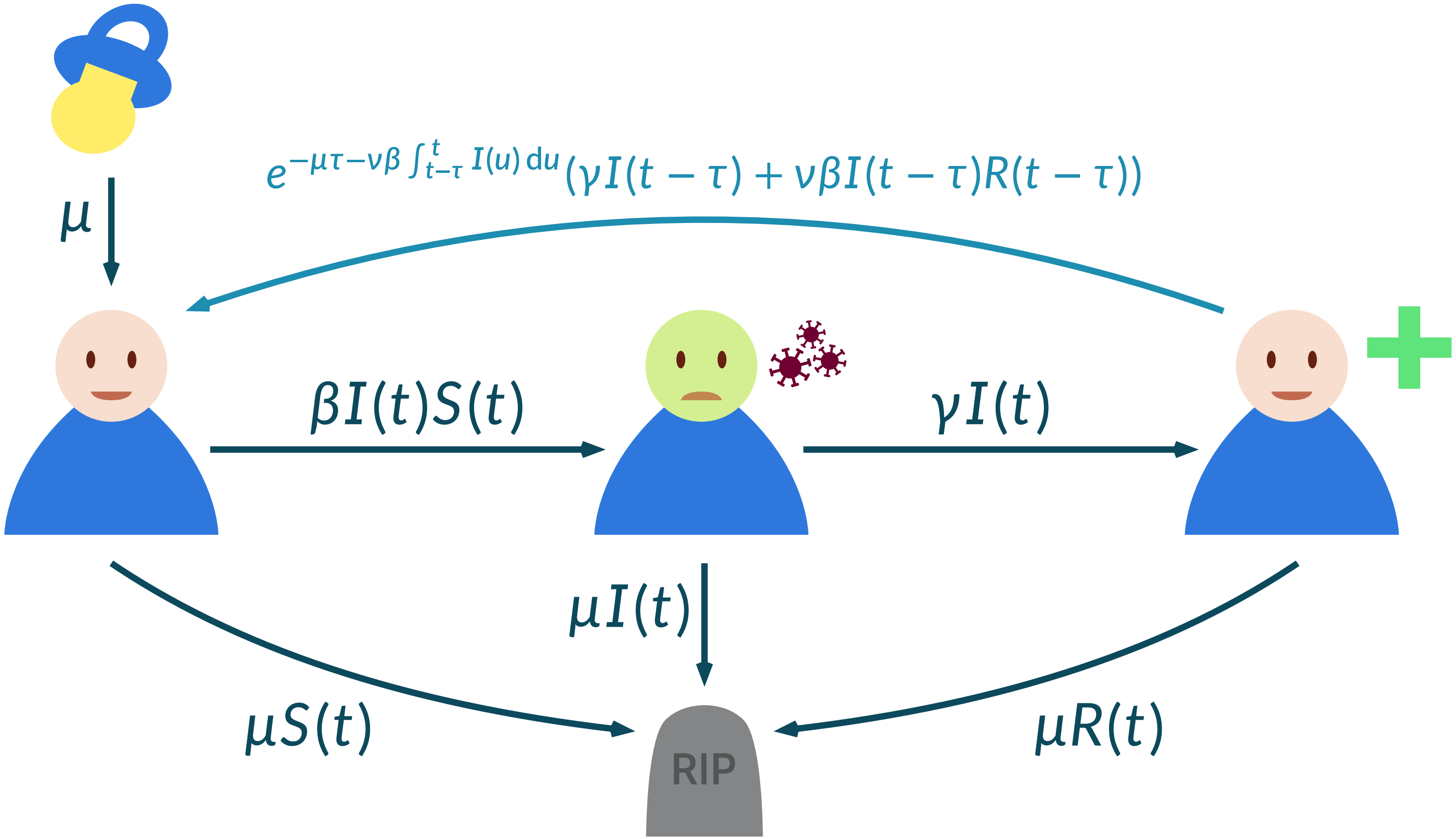
$$\dot{S}(t) = \mu - \mu S(t) - \beta I(t)S(t)$$

$$\dot{I}(t) = \beta I(t)S(t) - (\gamma + \mu)I(t)$$

$$R(t) = 1 - S(t) - I(t)$$



Example solution for $\mu = 0.02$, $\beta = 255.3$, and $\gamma = 17$.

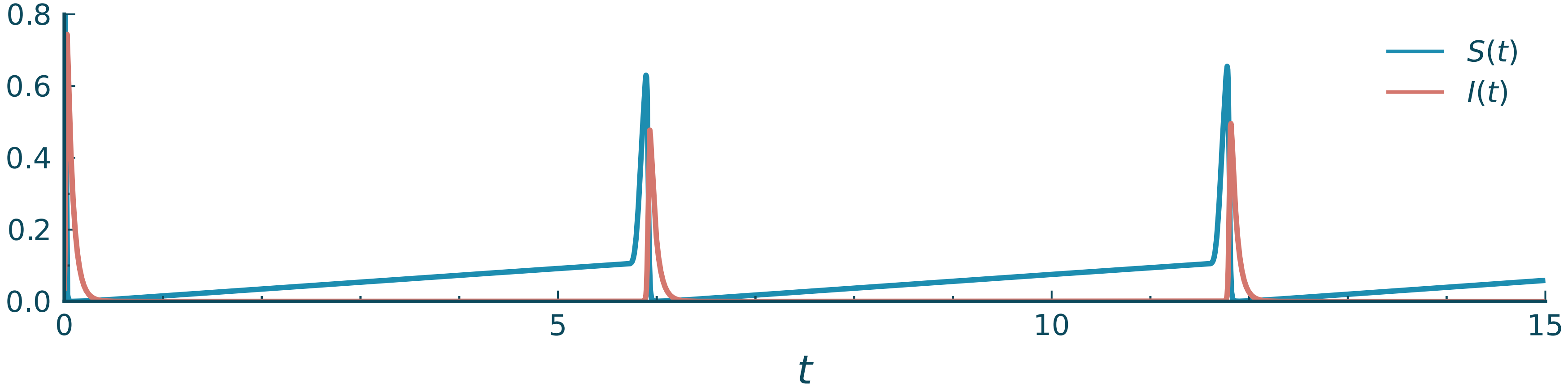

$e^{-\mu\tau-\nu\beta\int_{t-\tau}^{t} I(u)\,\mathrm{d}u}(\gamma I(t-\tau)+\nu\beta I(t-\tau)R(t-\tau))$
μ
$\beta I(t)S(t)$
$\gamma I(t)$
$\mu I(t)$
$\mu S(t)$
$\mu R(t)$
RIP

SIR model with waning and boosting of immunity

$$\dot{S}(t) = \mu - \mu S(t) - \beta I(t)S(t) + e^{-\mu\tau - \nu\beta \int_{t-\tau}^{t} I(u)\,du} (\gamma I(t-\tau) + \nu\beta I(t-\tau)R(t-\tau))$$

$$\dot{I}(t) = \beta I(t)S(t) - (\gamma + \mu)I(t)$$

$$R(t) = 1 - S(t) - I(t)$$



Example solution for $\mu = 0.02$, $\beta = 255.3$, $\gamma = 17$, $\tau = 5.6$, and $\nu = 3.2$.

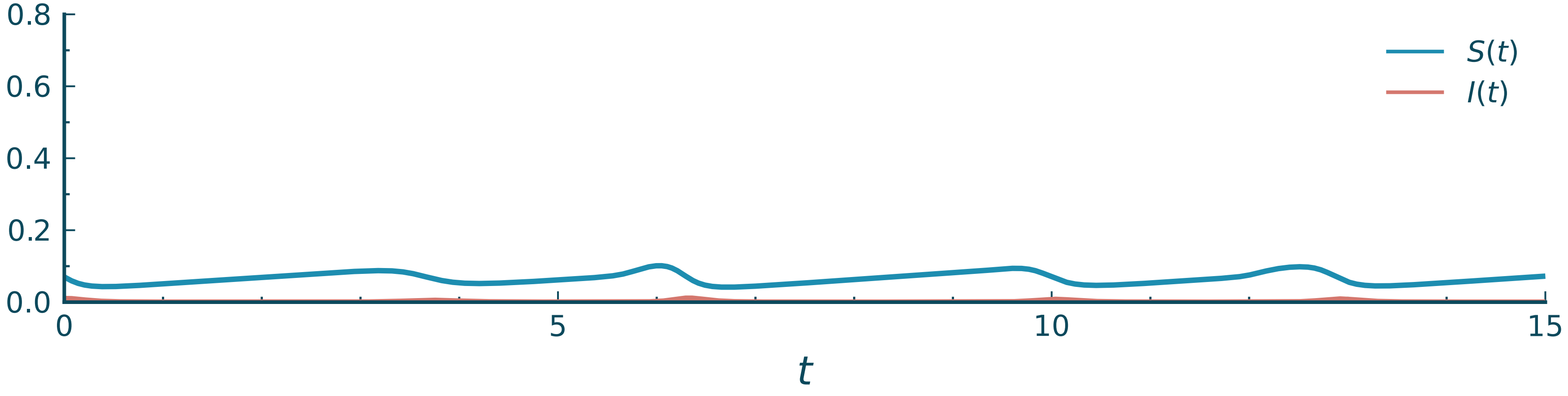
See Scarabel et al. (2025).

SIR model with waning and boosting of immunity

$$\dot{S}(t) = \mu - \mu S(t) - \beta I(t)S(t) + e^{-\mu\tau - \nu\beta\int_{t-\tau}^{t} I(u)\,\mathrm{d}u}\left(\gamma I(t-\tau) + \nu\beta I(t-\tau)R(t-\tau)\right)$$

$$\dot{I}(t) = \beta I(t)S(t) - (\gamma + \mu)I(t)$$

$$R(t) = 1 - S(t) - I(t)$$



Same parameters, different initial condition.

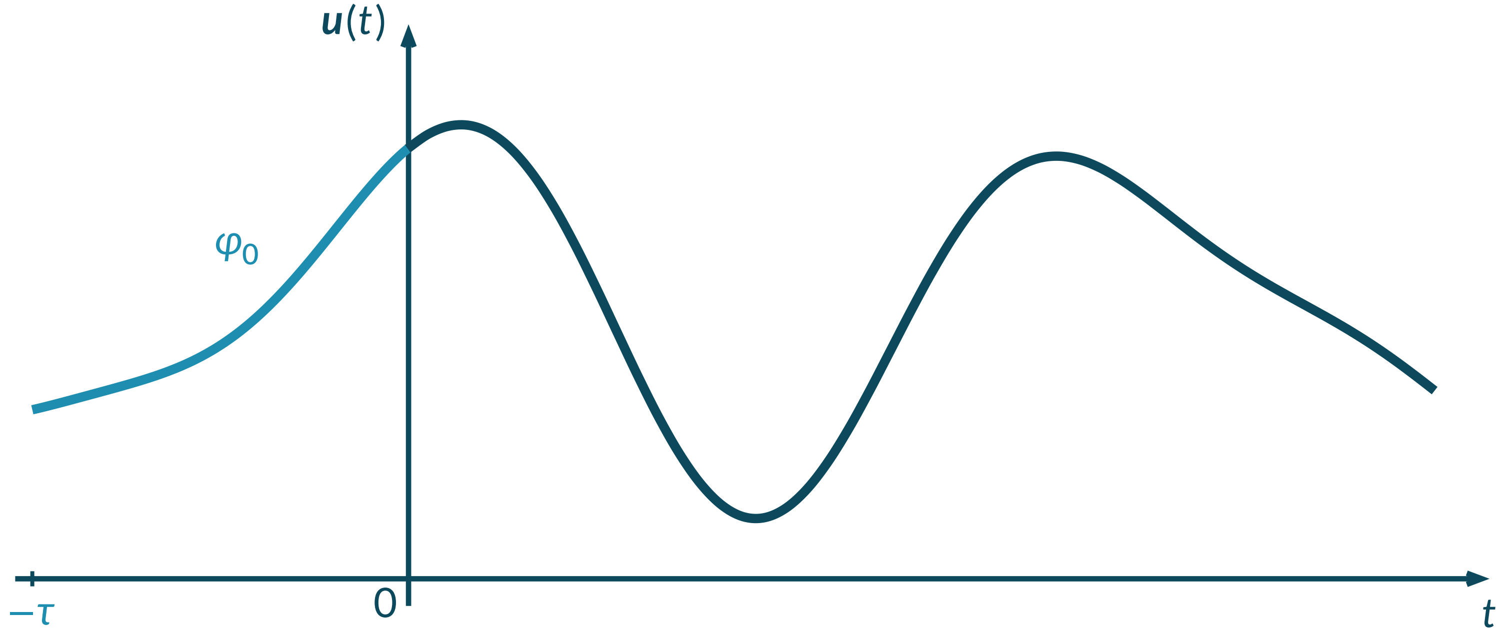
See Scarabel et al. (2025).

What do we want to know?

Which attractors are there?

How likely is a given attractor?

Infinite dimensional system



What do we want to know?

How to sample initial conditions?

Which attractors are there?

What part of the initial condition is important?

How likely is a given attractor?

How to tackle sampling?

We can assume:

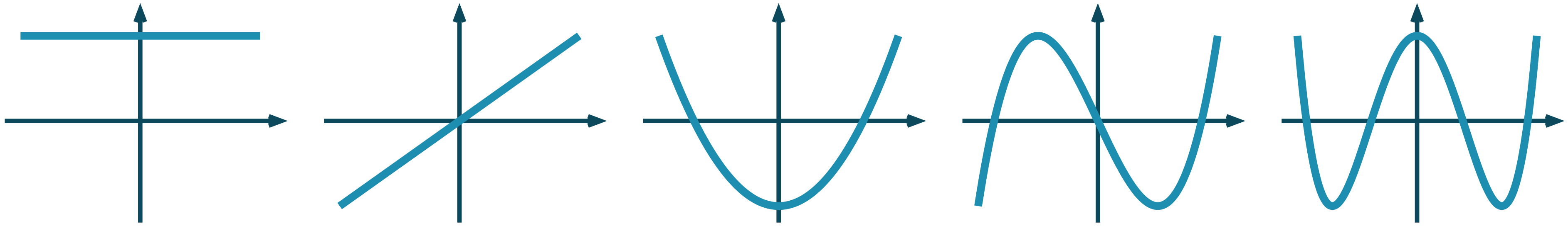
- well-behaved functions,
- with finite support,
- a typical value range.

We want:

- finite-dimensional,
- linear structure,
- truncatable.

Chebyshev series

$$p(\theta) = \sum_{i=0}^{N} c_i T_i(\theta), \quad \text{where} \quad T_i(\theta) = \cos(i \operatorname{acos}(\theta)).$$



T_0 T_1 T_2 T_3 T_4

The series' decay rate relates to the function's regularity.

Sampling initial conditions

Sample $p(\theta) = \sum_{i=0}^{N_h} c_i T_i(\theta)$ with

$$c_0 \sim N(\mu_0, \sigma_0) \quad \text{and} \quad c_i \sim N\left(0, \sigma \rho^i\right).$$

Sampling initial conditions

Sample $p(\theta) = \sum_{i=0}^{N_h} c_i T_i(\theta)$ with

$$c_0 \sim N(\mu_0, \sigma_0) \quad \text{and} \quad c_i \sim N\left(0, \sigma \rho^i\right).$$

If required to be positive, compute x_k s.t.

$$\varphi_0(\theta) = \sum_{k=0}^{2N_h} x_k T_k(\theta) = p(\theta)^2.$$

Squaring is fast using FFT (Ahmed and Fisher 1970).

Sampling initial conditions

Sample $p(\theta) = \sum_{i=0}^{N_h} c_i T_i(\theta)$ with

$$c_0 \sim N(\mu_0, \sigma_0) \quad \text{and} \quad c_i \sim N\left(0, \sigma \rho^i\right).$$

If required to be positive, compute x_k s.t.

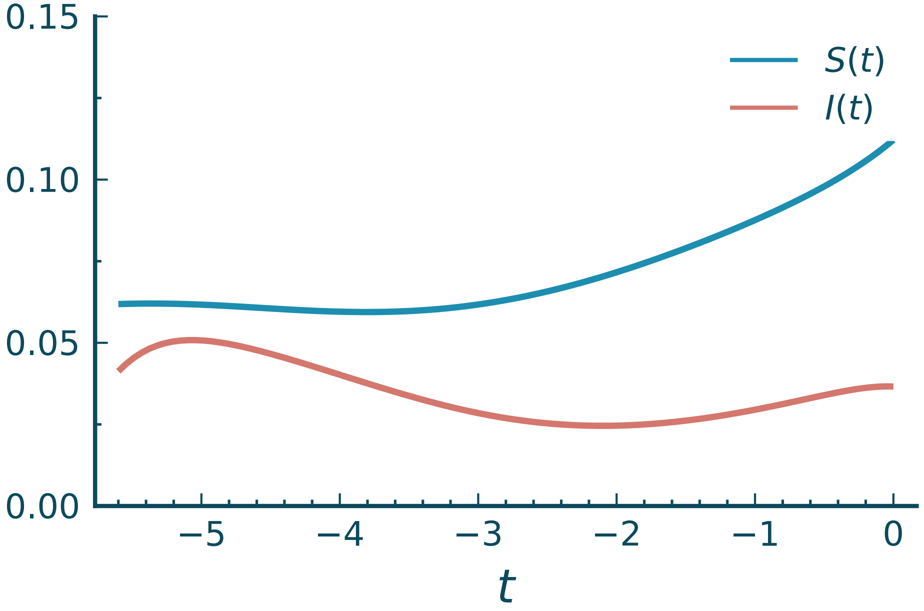
$$\varphi_0(\theta) = \sum_{k=0}^{2N_h} x_k T_k(\theta) = p(\theta)^2.$$

Repeat for every variable, yielding the initial condition $\boldsymbol{\varphi}_0$.

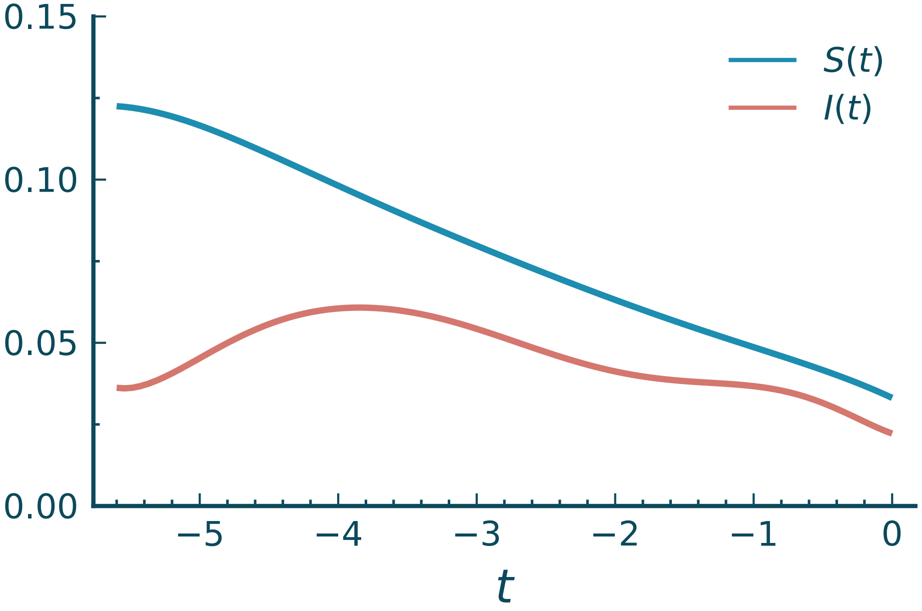
Squaring is fast using FFT (Ahmed and Fisher 1970).

Sampling for the SIR model

For S we use $\mu_0 = \sigma_0 = 0.06$, $\sigma = 0.1$, and $\rho = 0.4$, for I we use $\mu_0 = 0.02$, $\sigma = 0.05$, and $\rho = 0.6$, both with $N_h = 8$.



Example 1



Example 2

Detecting attractors

As trajectory captures entire state, the ODE approach works:

Generate n initial conditions $\boldsymbol{\varphi}_{0,i}(\theta)$.
for $i = 0, \ldots, n$ **do**
  Compute trajectory $\boldsymbol{u}_i(t)$ starting from $\boldsymbol{\varphi}_{0,i}(\theta)$.
  Extract feature vector from steady-state $\boldsymbol{X}_i = f\big(\boldsymbol{u}_i(t > t^*)\big)$.
end
Cluster feature vectors $\boldsymbol{X}_i$ using, e.g., k-means clustering.

See bSTAB by Stender and Hoffmann (2022).

Detecting attractors

As trajectory captures entire state, the ODE approach works:

```
Generate n initial conditions φ_{0,i}(θ).
for i = 0, …, n do
    Compute trajectory u_i(t) starting from φ_{0,i}(θ).
    Extract feature vector from steady-state X_i = f(u_i(t > t*)).
end
Cluster feature vectors X_i using, e.g., k-means clustering.
```

Finally, use some clustering quality metric to validate result.

See bSTAB by Stender and Hoffmann (2022).

Attractors of the SIR model

Using the sampling of before and S_{min}, S_{max}, I_{min}, and I_{max} as features.

Attractors of the SIR model

Using the sampling of before and S_{min}, S_{max}, I_{min}, and I_{max} as features.

For $n = 1000$ we find:

Cluster 1 $I_{\text{min}} \approx I_{\text{max}} \approx 1.2 \times 10^{-3}$

Cluster 2 $I_{\text{min}} \approx 0, \quad I_{\text{max}} \approx 9 \times 10^{-3}$

Cluster 3 $I_{\text{min}} \approx 0, \quad I_{\text{max}} \approx 0.49$ (*new*)

Attractors of the SIR model

Using the sampling of before and S_{min}, S_{max}, I_{min}, and I_{max} as features.

For $n = 1000$ we find:

Cluster 1 $I_{\text{min}} \approx I_{\text{max}} \approx 1.2 \times 10^{-3}$

Cluster 2 $I_{\text{min}} \approx 0, \quad I_{\text{max}} \approx 9 \times 10^{-3}$

Cluster 3 $I_{\text{min}} \approx 0, \quad I_{\text{max}} \approx 0.49$ (*new*)

With silhouette score 0.9999.

See Rousseeuw (1987).

Selecting important features

Find

$$T = \underset{T \in \mathbb{R}^{p \times p}}{\arg\max}\, \mathrm{tr}\Big((T^T S^w T)^{-1}\, T^T S^b T \Big),$$

where

$$S^w = \frac{1}{2} \sum_{y_i = y_j} \Delta_{i,j}, \quad \text{and}$$

$$S^b = \frac{1}{2} \sum_{y_i \neq y_j} \Delta_{i,j},$$

with $\Delta_{i,j} = (\boldsymbol{x}_i - \boldsymbol{x}_j)(\boldsymbol{x}_i - \boldsymbol{x}_j)^T$.

Affinity

Samples $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ have an affinity

$$a_{i,j} = \exp\left(-\frac{\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2}{\sigma_i \sigma_j}\right),$$

See Zelnik-Manor and Perona (2005).

Affinity

Samples $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ have an affinity

$$a_{i,j} = \exp\left(-\frac{\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2}{\sigma_i \sigma_j}\right),$$

Where σ_i is the distance from $\boldsymbol{x}_i$ to its seventh nearest neighbour.

See Zelnik-Manor and Perona (2005).

Selecting important features (cont'd)

Find

$$T = \arg\max_{T \in \mathbb{R}^{p \times p}} \operatorname{tr}\Big((T^T S^w T)^{-1} T^T S^b T\Big),$$

where

$$S^w = \frac{1}{2} \sum_{y_i = y_j} \Delta_{i,j}, \quad \text{and}$$

$$S^b = \frac{1}{2} \sum_{y_i \neq y_j} \Delta_{i,j},$$

with $\Delta_{i,j} = (\boldsymbol{x}_i - \boldsymbol{x}_j)(\boldsymbol{x}_i - \boldsymbol{x}_j)^T$.

Selecting important features (cont'd)

Find

$$T = \underset{T \in \mathbb{R}^{p \times p}}{\arg\max} \operatorname{tr}\Big((T^T S^w T)^{-1} T^T S^b T \Big),$$

where

$$S^w = \frac{1}{2} \sum_{y_i = y_j} a_{i,j} \quad \Delta_{i,j}, \quad \text{and}$$

$$S^b = \frac{1}{2} \sum_{y_i \neq y_j} \qquad \Delta_{i,j},$$

with $\Delta_{i,j} = (\boldsymbol{x}_i - \boldsymbol{x}_j)(\boldsymbol{x}_i - \boldsymbol{x}_j)^T$.

LFDA by Sugiyama (2007).

Selecting important features (cont'd)

Find

$$T = \underset{T \in \mathbb{R}^{p \times p}}{\arg\max}\, \mathrm{tr}\Big((T^T S^w T)^{-1} \, T^T S^b T \Big),$$

where

$$S^w = \frac{1}{2} \sum_{y_i = y_j} a_{i,j} \quad \Delta_{i,j}, \quad \text{and}$$

$$S^b = \frac{1}{2} \sum_{y_i \neq y_j} \nu a_{i,j} \ \ \Delta_{i,j},$$

with $\Delta_{i,j} = (\boldsymbol{x}_i - \boldsymbol{x}_j)(\boldsymbol{x}_i - \boldsymbol{x}_j)^T$.

A slight variation of LFDA by Sugiyama (2007).

Selecting important features (cont'd)

Find

$$T = \underset{T \in \mathbb{R}^{p \times p}}{\arg\max}\, \mathrm{tr}\Big((T^T S^w T)^{-1}\, T^T S^b T \Big),$$

where

$$S^w = \frac{1}{2} \sum_{y_i = y_j} a_{i,j} \frac{1}{n_{y_i}} \Delta_{i,j}, \quad \text{and}$$

$$S^b = \frac{1}{2} \sum_{y_i \neq y_j} \nu a_{i,j} \frac{1}{n} \Delta_{i,j} + \frac{1}{2} \sum_{y_i = y_j} a_{i,j} \left(\frac{1}{n} - \frac{1}{n_{y_i}} \right) \Delta_{i,j},$$

with $\Delta_{i,j} = (\boldsymbol{x}_i - \boldsymbol{x}_j)(\boldsymbol{x}_i - \boldsymbol{x}_j)^T$.

A slight variation of LFDA by Sugiyama (2007).

Selecting important features (cont'd)

Find

$$T = \underset{T \in \mathbb{R}^{p \times p}}{\arg\max}\, \mathrm{tr}\Big((T^T S^w T)^{-1}\, T^T S^b T \Big),$$

where

$$S^w = \frac{1}{2} \sum_{y_i = y_j} a_{i,j} \frac{1}{n_{y_i}} \Delta_{i,j}, \quad \text{and}$$

$$S^b = \frac{1}{2} \sum_{y_i \neq y_j} \nu a_{i,j} \frac{1}{n} \Delta_{i,j} + \frac{1}{2} \sum_{y_i = y_j} a_{i,j} \left(\frac{1}{n} - \frac{1}{n_{y_i}} \right) \Delta_{i,j},$$

with $\Delta_{i,j} = (\boldsymbol{x}_i - \boldsymbol{x}_j)(\boldsymbol{x}_i - \boldsymbol{x}_j)^T$.

For large n, take $a_{i,j} = 0$ beyond, e.g., the 50th nearest neighbour.

A slight variation of LFDA by Sugiyama (2007).

Finding the optimizer

$$\underset{T \in \mathbb{R}^{p \times p}}{\arg\max} \operatorname{tr}\Big((T^T S^w T)^{-1} T^T S^b T \Big)$$

is given by

$$\begin{pmatrix} \boldsymbol{w}_1 & \boldsymbol{w}_2 & \dots & \boldsymbol{w}_p \end{pmatrix}$$

with $S^w \boldsymbol{w}_i = \lambda_i S^b \boldsymbol{w}_i$.

Finding the optimizer

$$\arg\max_{T \in \mathbb{R}^{p \times p}} \operatorname{tr}\left((T^T S^w T)^{-1} T^T S^b T \right)$$

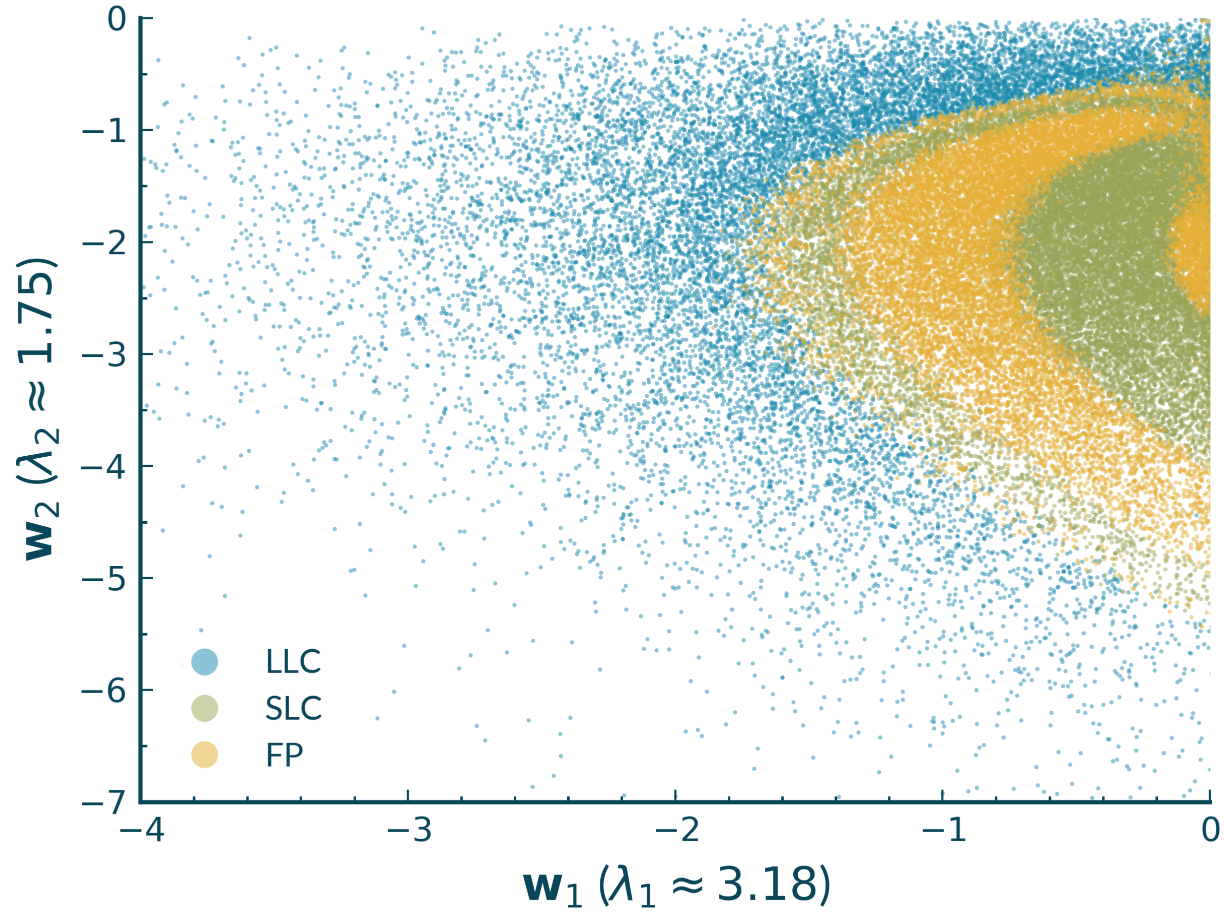
is given by

$$\begin{pmatrix} \boldsymbol{w}_1 & \boldsymbol{w}_2 & \dots & \boldsymbol{w}_p \end{pmatrix}$$

with $S^w \boldsymbol{w}_i = \lambda_i S^b \boldsymbol{w}_i$.

λ_i measures how much ordering by $\boldsymbol{x} \mapsto \boldsymbol{w}_i^T \boldsymbol{x}$ improves the criterion.

Found dimensionality reduction of the SIR model



Visualizing functionals

$$\text{Find some} \quad f \quad \text{s.t.} \quad \boldsymbol{w}^T \boldsymbol{x} = \int_{-1}^{1} f(\theta) \left(\sum_{i=0}^{N} x_i T_i(\theta) \right) \mathrm{d}\theta.$$

Visualizing functionals

Find some f s.t. $\boldsymbol{w}^T \boldsymbol{x} = \int_{-1}^{1} f(\theta) \left(\sum_{i=0}^{N} x_i T_i(\theta) \right) \mathrm{d}\theta.$

Using $\int_{-1}^{1} P_i(\theta) P_j(\theta) \,\mathrm{d}\theta = \frac{2}{2i+1} \delta_{ij}$,

Visualizing functionals

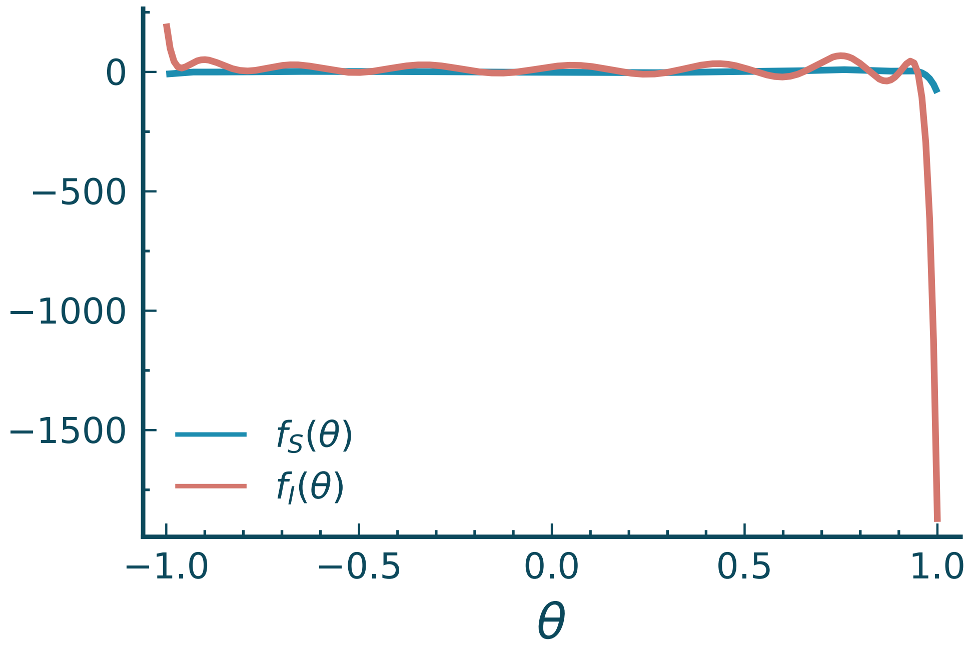
Find some f s.t. $\boldsymbol{w}^T \boldsymbol{x} = \int_{-1}^{1} f(\theta) \left(\sum_{i=0}^{N} x_i T_i(\theta) \right) \mathrm{d}\theta.$

Using $\int_{-1}^{1} P_i(\theta) P_j(\theta)\, \mathrm{d}\theta = \frac{2}{2i+1} \delta_{ij}$,

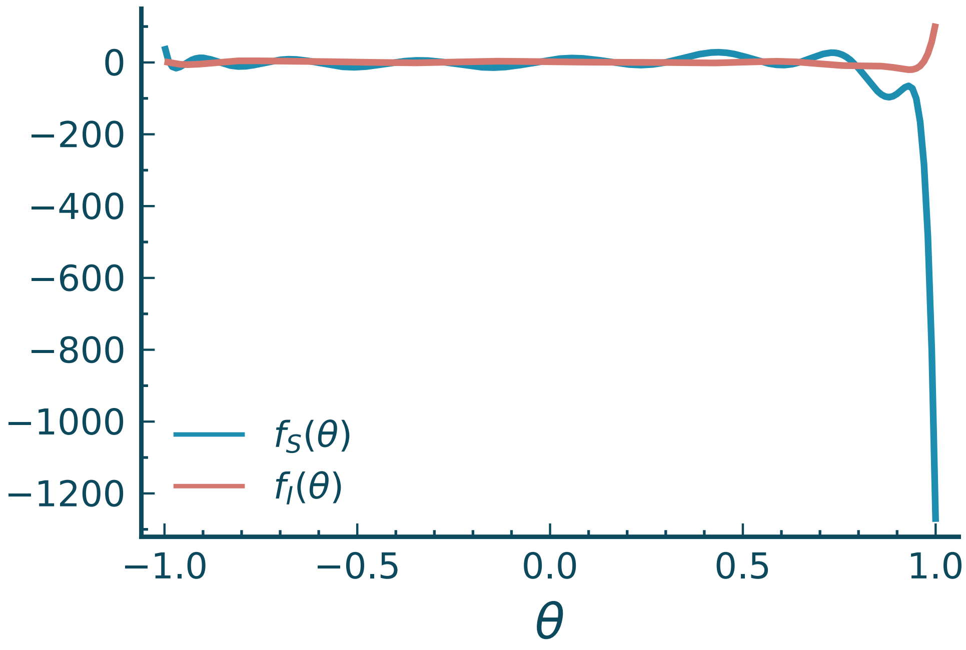
$$f(\theta) = \sum_{i=0}^{N} f_i P_i(\theta), \quad \text{with} \quad f_i = \frac{2i+1}{2} \left(C_{T \to P}^{-T} \boldsymbol{w} \right)_i$$

Methods exist to quickly compute $C_{T \to P}$ and $C_{P \to T}$ (Olver et al. 2020).

Functionals of the SIR model

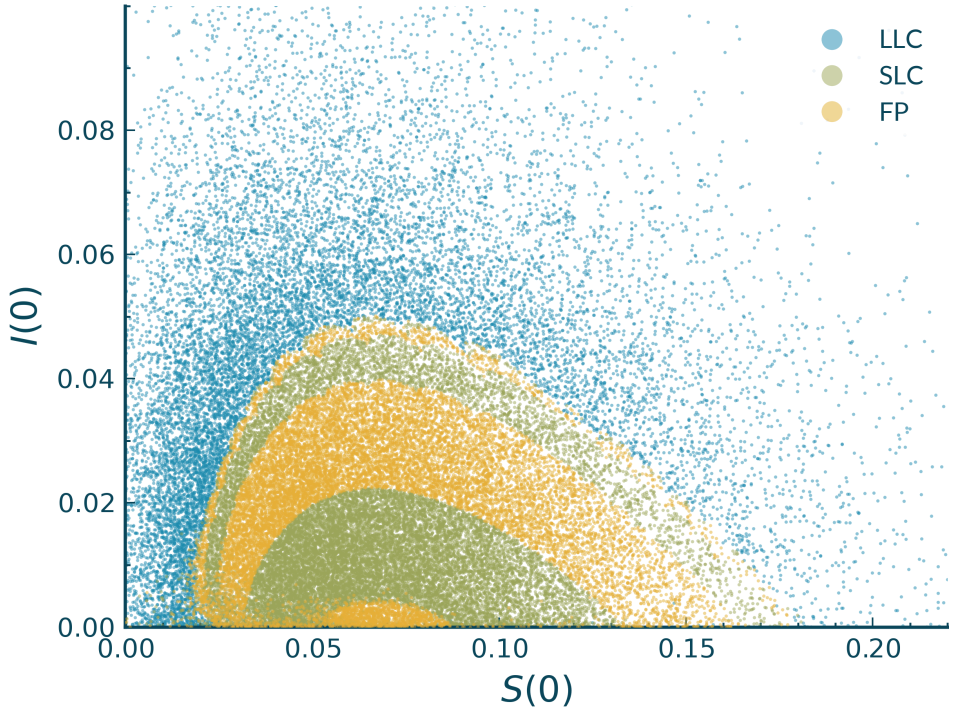


Functionals of $\boldsymbol{w}_1$

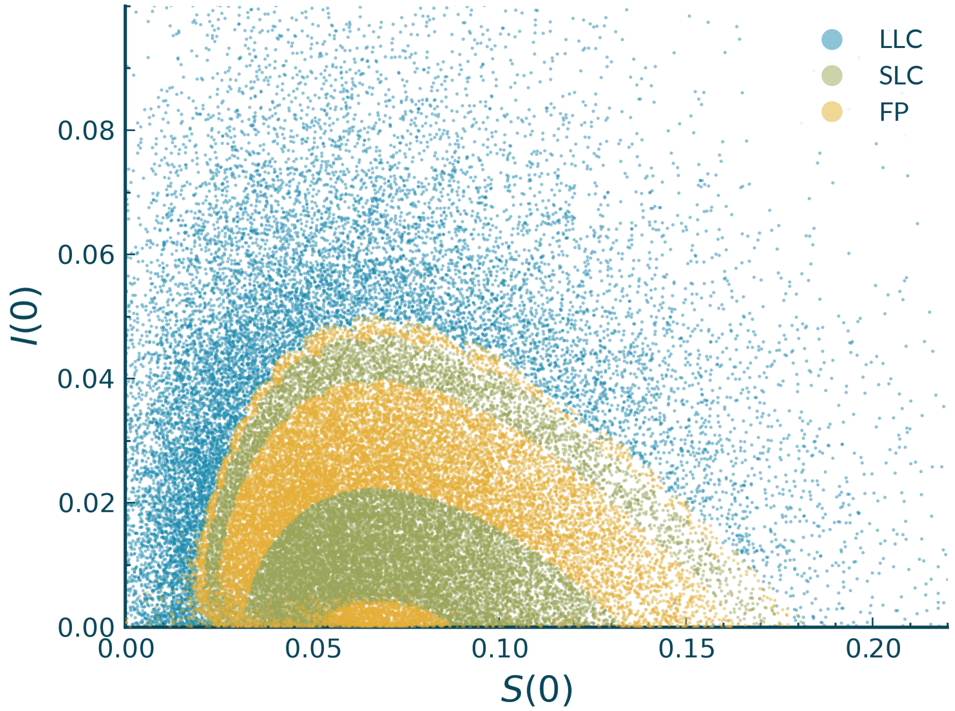


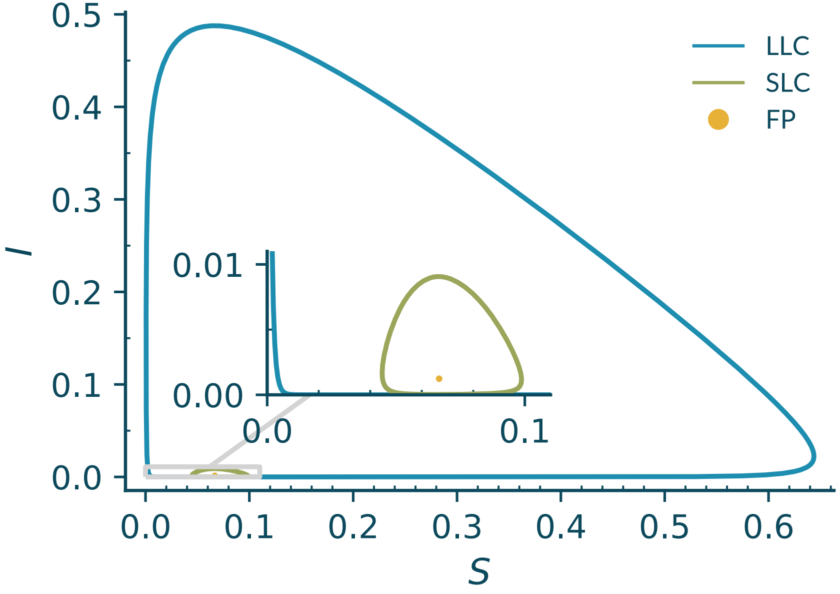
Functionals of $\boldsymbol{w}_2$

Interpretable dimensionality reduction of the SIR model

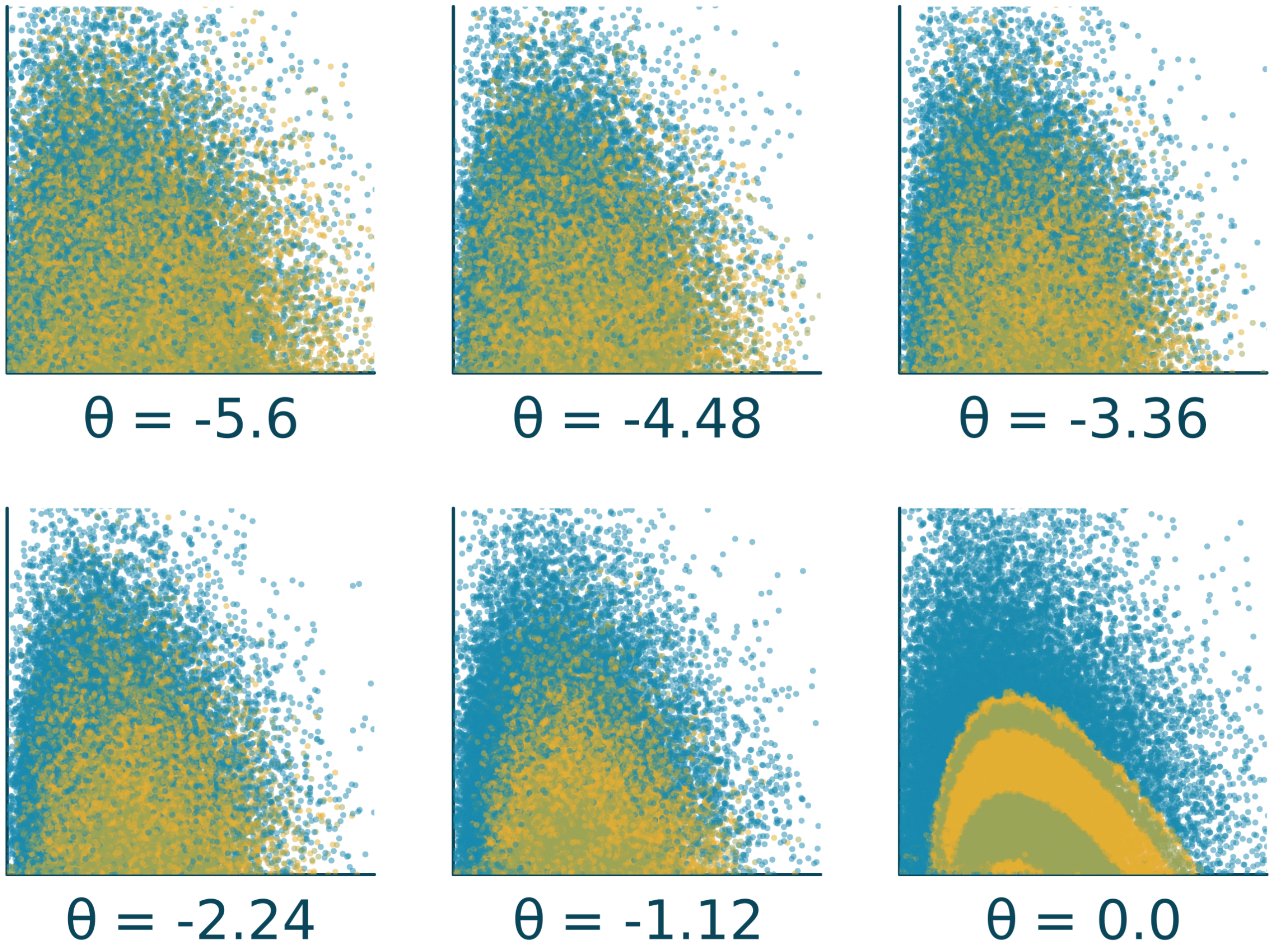


Interpretable dimensionality reduction of the SIR model





Sanity check



Estimating likelihoods

Sample $n \gg 1$ from a distribution and tally results.

Estimating likelihoods

Sample $n \gg 1$ from a distribution and tally results.

ODEs Sampling hypercubes allows the reporting of volumes.

Estimating likelihoods

Sample $n \gg 1$ from a distribution and tally results.

ODEs Sampling hypercubes allows the reporting of volumes.

DDEs Domain knowledge is needed to determine a distribution of ICs.

Estimating likelihoods

Sample $n \gg 1$ from a distribution and tally results.

ODEs Sampling hypercubes allows the reporting of volumes.

DDEs Domain knowledge is needed to determine a distribution of ICs.

For the earlier (arbitrary) distribution, with $n = 50\,000$:

Fixed point 31%

Small LC 41%

Large LC 28%

References

Ahmed N, Fisher P (1970) Study of algorithmic properties of Chebyshev coefficients. International Journal of Computer Mathematics 2:307–317

Olver S, Slevinsky R, Townsend A (2020) Fast algorithms using orthogonal polynomials. Acta Numerica 29:573–699

Rousseeuw P (1987) Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. Journal of Computational and Applied Mathematics 20:53–65

Scarabel F, Polner M, Wylde D, et al (2025) Bistability and complex bifurcation diagrams generated by waning and boosting of immunity. Journal of Mathematical Biology 91:30

Stender M, Hoffmann N (2022) bSTAB: an open-source software for computing the basin stability of multi-stable dynamical systems. Nonlinear Dynamics 107:1451–1468

Sugiyama M (2007) Dimensionality Reduction of Multimodal Labeled Data by Local Fisher Discriminant Analysis. Journal of Machine Learning Research 8:1027–1061

Zelnik-Manor L, Perona P (2005) Self-tuning spectral clustering. In: Saul L, Weiss Y, Bottou L (eds) Advances in Neural Information Processing Systems 17. pp 1601–1608

Contributions

Devised a scheme to sample initial conditions for non-linear DDEs, which can be used to:

- detect attractors; and
- estimate likelihoods of outcomes.

A dimensionality reduction method to detect deciding features of initial conditions.

We applied these to a SIR model with waning and boosting of immunity, finding a new attractor and a projection of the initial conditions separating the basins.