Discovering governing reactions from concentration data

November 2, 2017

1 Introduction

When presented with a non-equilibrium time series of possibly noisy concentration fluctuations of some species as output of, e.g., measurements from experiments or simulations that were parameterized by microscopic rates (cite ReaDDy?), one can ask for the corresponding macroscopic rates and generating reaction network. In this paper we present an application of the shallow learning method SINDy [1]. It is able to identify the generating parsimonious nonlinear dynamics in data that stems from dynamical systems, thus providing an interpretable result. In our application we, opposed to the original method, do not only look for macroscopic rates of net species change but investigate the specific reactions that might have lead to the observations. We demonstrate the algorithm on two toy problems - one problem showing that when there is no ambiguity in the system, one converges to the correct rates with increasing resolution of concentration fluctuations and one in which we compute a sparse reaction network for given data.

2 The method

The underlying model that we want to fit the data to is a law of mass action type dynamical system. To this end, let S be the number of species, then the concentration data at a time t can be represented by a vector

$$\mathbf{x}(t) = \begin{pmatrix} x_1(t) \\ \vdots \\ x_S(t) \end{pmatrix} \in \mathbb{R}^S. \tag{1}$$

Further, one can choose R possible ansatz reactions with their respective reaction function

$$\mathbf{y}_r(\mathbf{x}(t)) = \begin{pmatrix} y_{r,1}(\mathbf{x}(t)) \\ \vdots \\ y_{r,S}(\mathbf{x}(t)) \end{pmatrix} \tag{2}$$

so that the change of concentration for species i at time t_i, is represented by the dynamical system

$$\dot{\mathbf{x}}_i(t) = \sum_{r=1}^{R} y_{r,i}(\mathbf{x}(t))\xi_r, \quad i = 1, \ldots, S, \tag{3}$$

where ξ_r are the to-be estimated macroscopic rates.

When presented with a time series with T observations, the data can be represented as a matrix

$$\mathbf{X} = \begin{pmatrix} x_1(t_1) & x_2(t_1) & \cdots & x_S(t_1) \\ x_1(t_2) & x_2(t_2) & \cdots & x_S(t_2) \\ \vdots & \vdots & \ddots & \vdots \\ x_1(t_T) & x_2(t_T) & \cdots & x_S(t_T) \end{pmatrix} \in \mathbb{R}^{T \times S}. \tag{4}$$

Given this matrix, one can propose a library $\Theta(\mathbf{X}) = \begin{pmatrix} \theta_1(\mathbf{X}) & \theta_2(\mathbf{X}) & \cdots & \theta_R(\mathbf{X}) \end{pmatrix}$ of R ansatz reactions with corresponding reaction functions

$$\theta_r(\mathbf{X}) = \begin{pmatrix} \mathbf{y}_r(\mathbf{X}_1)^T \\ \vdots \\ \mathbf{y}_r(\mathbf{X}_T)^T \end{pmatrix} \in \mathbb{R}^{T \times S}, \quad r = 1, \ldots, R, \tag{5}$$

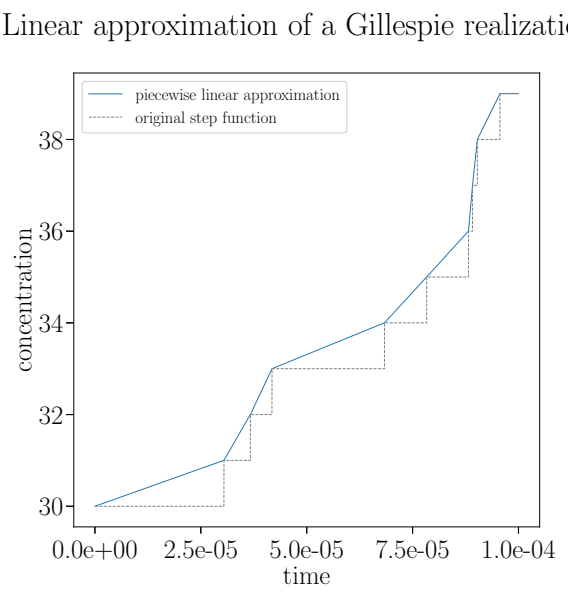


Figure 1: This figure depicts a possible output yielded by an application of the Gillespie method and the performed piecewise linear approximation in order to obtain a better behaving derivative.

where $\mathbf{X}_i$ denotes the i-th row in X. Applying the concentration trajectory to the library yields a data tensor $\Theta(\mathbf{X}) \in \mathbb{R}^{T\times S\times R}$. Following the approach of SINDy, the goal is now to find coefficients $\Xi = \begin{pmatrix}\xi_1 & \xi_2 & \cdots & \xi_R\end{pmatrix}^T$, so that

$$\dot{\mathbf{X}} = \Theta(\mathbf{X})\Xi = \sum_{r=1}^{R} \theta_r(\mathbf{X})\xi_r. \tag{6}$$

In particular, the system is linear in the coefficients Ξ, which makes sparse regression tools such as LASSO [6, 3] applicable. To this end, one can consider the minimization problem to find $\hat{\Xi}$ such that

$$\hat{\Xi} = \arg\min_{\Xi} \left(\frac{1}{2T} \left\| \dot{\mathbf{X}} - \Theta(\mathbf{X})\Xi \right\|_F^2 + \alpha \|\Xi\|_1 \right) \quad \text{subject to } \Xi \geq 0, \tag{7}$$

where $\alpha \geq 0$ is a sparsity inducing hyperparameter and $\|\cdot\|_F$ denotes the Frobenius norm. For $\alpha = 0$ this problem reduces to constrained least-squares. For solving (7) we apply the sequential least squares minimizer SLSQP, originally described in [5], contained in, e.g., the software package SciPy [4]. As initial guess for the regulated problem we use the solution of (7) with $\alpha = 0$.

3 Examples

For generating time series data of concentrations we use the Gillespie method [2]. For a given set of initial conditions, we produce several realizations which then are converted to a trajectory with fixed time step and averaged. Since these trajectories are piecewise constant, one has to take special care when approximating the temporal derivative. Simply taking finite differences has the effect that, with decreasing time step, the derivative is most of the time constant zero and

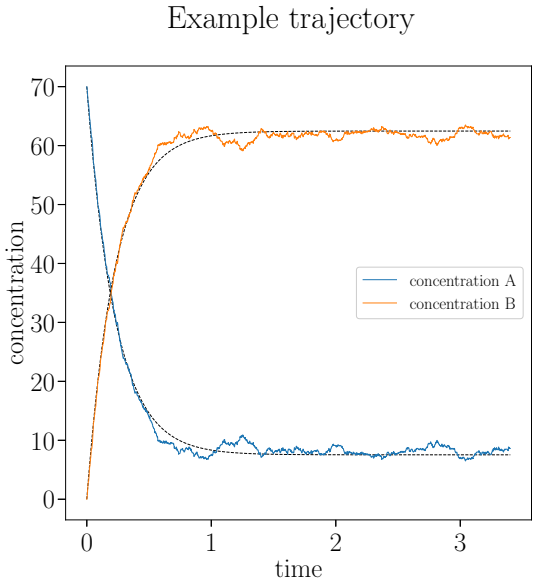


Figure 2: A realization of the example given in Section 3.1.

approaches infinity at the jump discontinuities. To counter this effect, we beforehand perform a linear approximation between each two adjacent discontinuities, as depicted in Figure 1. (Use regularized derivative?)

3.1 Regression without regularization

Here we demonstrate that, with decreasing time step Δt and $\alpha = 0$, the estimated rates converge to the true rates if there is no ambiguity in the system. The systems contains two species A and B which can convert into each other with macroscopic rates of $r_1 = 4.0$ and $r_2 = 0.5$, respectively. The initial amount of particles is set to 70 of type A and 0 of type B. A possibly resulting change of concentrations over time is depicted in Figure 2.

For each Δt, we averaged eight different realizations with identical starting conditions into one single trajectory and estimated the rates for reactions $A \to B$ and $B \to A$. This procedure, we repeated 30 times. In Figure 3, the average estimates alongside with their standard deviations are shown. One can see that, with decreasing Δt, also the estimate for the rates becomes better and they can be recovered.

3.2 Regression with regularization

In this example there are four species A, B, C, and D with reactions $A \xrightarrow{k_1} D$, $D \xrightarrow{k_2} A$, $D \xrightarrow{k_3} B$, $A + B \xrightarrow{k_4} C$, and $C \xrightarrow{k_5} D + B$, where $k_1 = k_2 = 2 \cdot 10^{-2}$, $k_3 = 1 \cdot 10^{-2}$, and $k_4 = k_5 = 1 \cdot 10^{-4}$. These reactions are inserted into the ansatz library $\Theta(\cdot)$. Additionally the reactions $A \leftrightarrow B$, $A \leftrightarrow C$, and $A + C \leftrightarrow D$ are inserted into the library.

The problem now is to find a suitable α such that a parsimonious solution is picked. To this end, cross-validation is applied. The trajectories that were used as training and test data set are depicted in Figure 4. Perhaps use several test trajectories with different initial conditions to get something like a confidence interval? As an indicator to the estimation error we use the

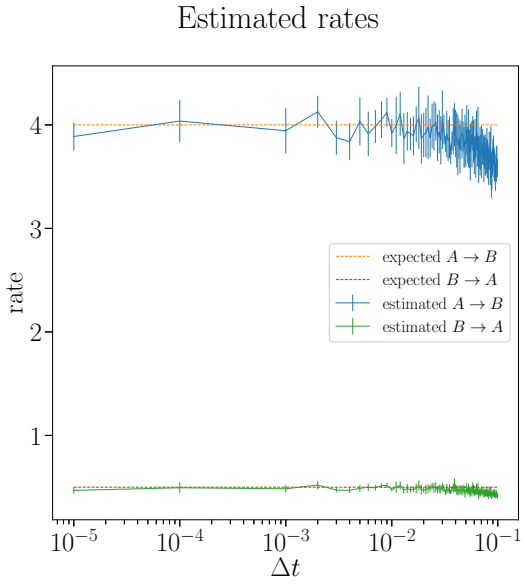
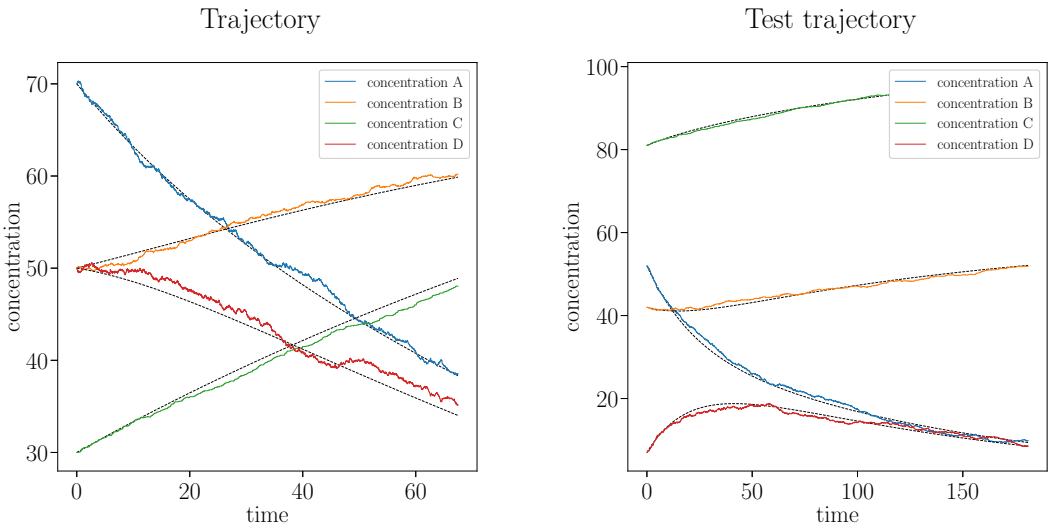


Figure 3: Convergence of estimated rates to the ground truth with decreasing Δt.



(a) Train data set.

(b) Test data set.

Figure 4: Train and test data sets for the example given in Section 3.2.

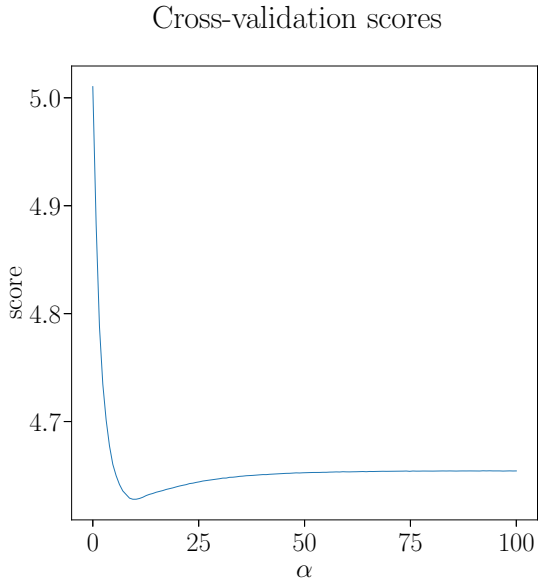


Figure 5: Score function (8) evaluated for a range of α values.

scoring function

$$f_\alpha = \frac{1}{T}\|\dot{\mathbf{X}}_{\text{test}} - \Theta(\mathbf{X}_{\text{test}})\Xi_{\text{train}}\|_F^2. \tag{8}$$

For a selection of values $\alpha \in [0, 100]$, the model parameters Ξ were estimated and the scoring function (8) evaluated, yielding the graph that is depicted in Figure 5.

One can see that the minimum is at about

$$\hat{\alpha} = \underset{\alpha\in[0,100]}{\arg\min} f_\alpha \approx 10. \tag{9}$$

When evaluating the minimization problem (7) with $\hat{\alpha}$ as hyperparameter for the regularizer and applying a cut-off of 10^{-10} to the yielded parameters, one obtains the simpler set of reactions $A \xrightarrow{\hat{k}_1} D$, $D \xrightarrow{\hat{k}_3} B$, and $A+B \xrightarrow{\hat{k}_4} C$ with rates $\hat{k}_1 \approx 3.5\cdot10^{-3}$, $\hat{k}_3 \approx 9.1\cdot10^{-3}$, and $\hat{k}_4 \approx 8.7\cdot10^{-5}$, respectively.

One can see in Figure 6 that with this simpler set of reactions the observed concentration curves can be explained as well. It can be argued that the second reaction of the generating model $D \xrightarrow{k_2} A$ can be compensated by a smaller rate $\hat{k}_1 < k_1$ and that the fifth reaction of the generating model can be neglected altogether in the observed timescale as its rate is two orders of magnitude smaller than the rates of the other unimolecular reactions. The rates k_3 and k_4 are approximately recovered by $\hat{k}_3$ and $\hat{k}_4$, respectively.

4 Conclusion

In this work we have successfully applied and extended the SINDy method to not only parsimoniously detect potentially nonlinear terms in a dynamical system from possibly noisy data, but also pick up on, in this case, a sparse set of generating reactions (5) and their respective rates.

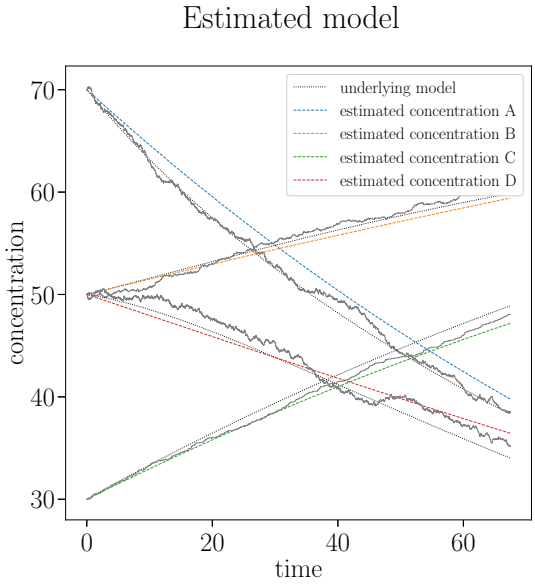


Figure 6: Estimated and generating concentration curves for the example in Section 3.2.

In two examples it was demonstrated that, despite noisy data and unavailability of derivative measurements, a parsimonious generating reaction network that is able to explain the observed data can be recovered. In particular it was shown in the example given in Section 3.1 that if there is no ambiguity in the underlying model and ansatz reaction library, the actual rates can be recovered. In the example of Section 3.2 an even simpler model could be obtained by making use of sparse regression and cross-validation.

References

[1] S. L. Brunton, J. L. Proctor, and J. N. Kutz. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *PNAS*, 1(609):1–26, 2015.

[2] D. T. Gillespie. A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. *Journal of Computational Physics*, 22(4):403–434, 1976.

[3] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer New York, New York, NY, 2009.

[4] E. Jones, T. Oliphant, P. Peterson, et al. SciPy: Open source scientific tools for Python, 2001–. [Online; accessed October 27, 2017].

[5] D. Kraft. A software package for sequential quadratic programming. *Technical Report DFVLR-FB 88-28, Institut für Dynamik der Flugsysteme, Oberpfaffenhofen*, 1988.

[6] R. Tibshirani. Regression Selection and Shrinkage via the Lasso, 1996.