Discovering governing reactions from concentration data

January 22, 2018

1 Introduction

When presented with a time series of possibly noisy non-equilibrium 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 a generating reaction network. In this paper we present an application of the shallow learning method SINDy [1]. By sparse regression, it is able to identify generating nonlinear dynamics in data that stems from dynamical systems. The parsimonious nature of the results avoids overfitting and provides interpretability. In our application we, as 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 is a law of mass action type dynamical system. To this end, let S be the number of species, then the observed concentration 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, is represented by the dynamical system

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

where ξ_r are the to-be estimated macroscopic rates.

When presented with a time series consisting of 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

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

where $\mathbf{X}_i$ denotes the i-th row in X. Applying the concentration trajectory to the library yields $\Theta(\mathbf{X}) \in \mathbb{R}^{T\times S\times R}$. Following the approach of SINDy, the goal is 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 potentially sparse regression tools such as elastic net regularization [8] applicable. To this end, one can consider the minimization problem to find $\hat{\Xi}$ such that

$$\hat{\Xi} = \arg\min_{\Xi}\left(\frac{1}{2T}\left\|\dot{\mathbf{X}} - \Theta(\mathbf{X})\Xi\right\|_F^2 + \alpha\lambda\|\Xi\|_1 + \alpha(1-\lambda)\|\Xi\|_2^2\right) \quad \text{subject to } \Xi \geq 0, \tag{7}$$

where $\|\cdot\|_F$ denotes the Frobenius norm, $\lambda \in [0,1]$ a hyperparameter that interpolates linearly between LASSO [7, 3] and Ridge [4] methods, and $\alpha \geq 0$ is a hyperparameter that, depending on λ, can induce sparsity and give preference to smaller solutions in the L_1 or L_2 sense.

For $\alpha = 0$ the minimization problem reduces to constrained least-squares. In order to solve (7) we apply the sequential least-squares minimizer SLSQP, originally described in [6], by the software package SciPy [5]. As initial guess for the regularized problem we use the solution of (7) with $\alpha = 0$.

3 Examples

(this has changed, rewrite!)

For generating time series data of concentrations we use the Gillespie method [2]. For given 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 data's temporal derivative that is needed in (7). Applying finite differences has the effect that, with decreasing time step, the derivative mostly constant zero and 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 **??** and thus obtain a piecewise constant derivative. (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. Let there be two species A and B which can unimolecularly convert into each other with macroscopic rates of $k_1 = 4.0$ and $k_2 = 0.5$, respectively. The initial amount of particles is set to 70 of type A and 0 of type B. The law of mass action solution and Gillespie realization corresponding to this system are depicted in Figure 1.

For each Δt, eight different realizations with identical starting conditions are averaged into a single trajectory, of which then the rates for reactions $A \to B$ and $B \to A$ are estimated. This procedure was repeated 30 times. In Figure 2, the average estimates with their standard deviations are shown. One can see that, with decreasing Δt, the estimates become better and the rates can be recovered. Further it can be observed that in this case a time step of $\Delta t \approx 10^{-2}$ suffices.

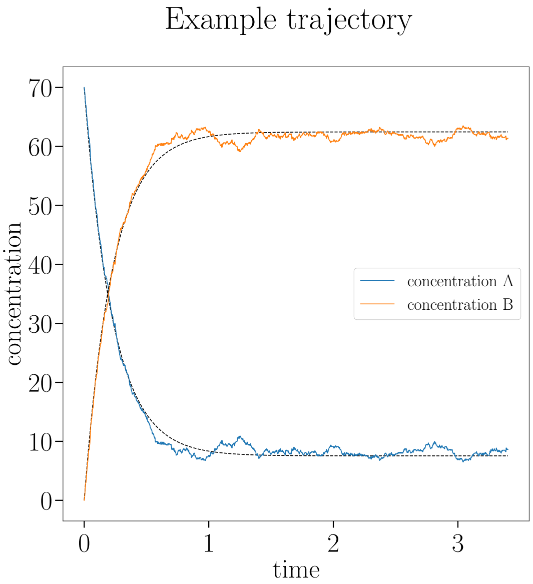


Figure 1: The law of mass action solution and a Gillespie realization of the example given in Section 3.1.

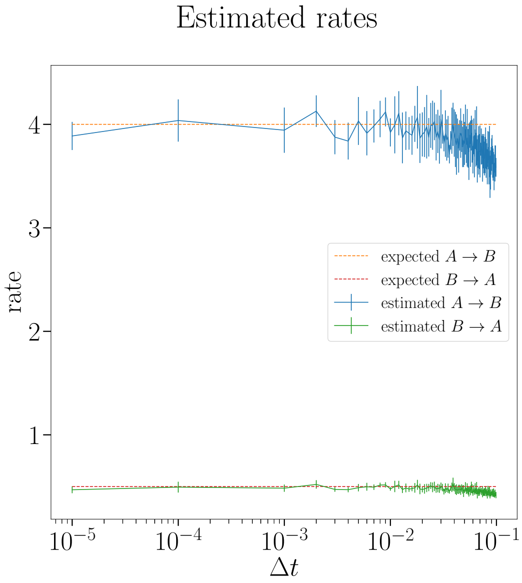
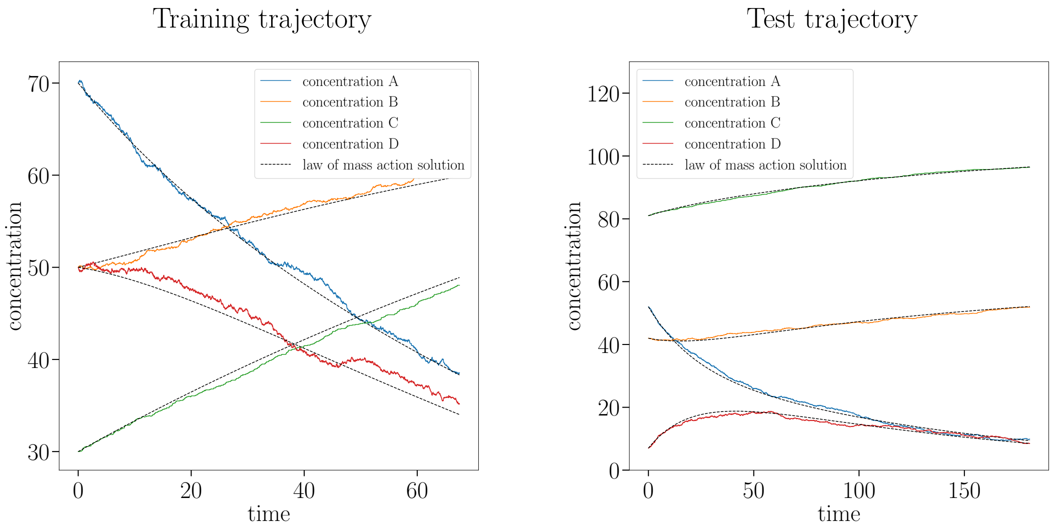


Figure 2: Convergence of estimated rates with decreasing Δt.



(a) Train data set.

(b) Test data set.

Figure 3: Train and test data sets for the example given in Section 3.2.

3.2 Regression with regularization

In this example there are four species A, B, C, and D. The system was generated 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 the rates are given in Table 1. 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 yielded. To this end, cross-validation is applied. The trajectories that are used as training and test data set are depicted in Figure 3. Perhaps use several test trajectories with different initial conditions to get something like a confidence interval? As an indicator to the estimation error one can use the scoring function

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

where the best possible score is $f_\alpha = 0$ and larger values indicate worse models. For a selection of hyperparameters $\alpha \in [0, 100]$, the model parameters Ξ are estimated and the scoring function (8) evaluated, yielding a dependency of the score on α as depicted in Figure 4.

One can see that the minimum score is achieved at

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

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 as in Table 1.

One can see in Figure 5 that with this simpler set of reactions the observed concentration curves can be explained as well. Further, it can be argued that the second reaction of the original model $D \xrightarrow{k_2} A$ is 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 other unimolecular reactions' rates. The parameters k_3 and k_4 are approximately recovered by $\hat{k}_3$ and $\hat{k}_4$, respectively.

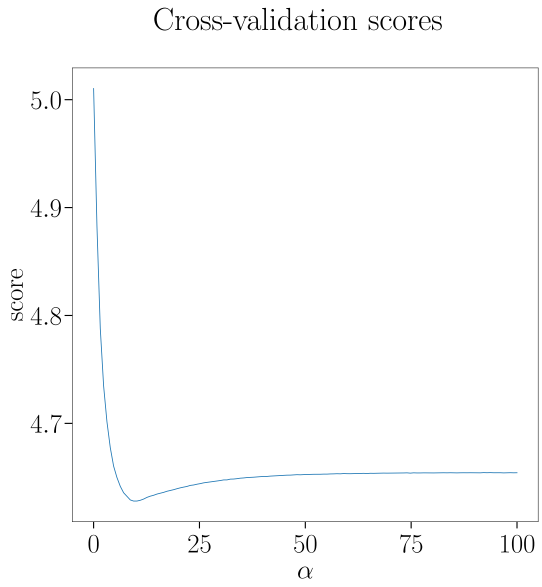


Figure 4: Score function (8) evaluated for a range of α values.

Reaction	original rate	estimated rate
$A \to D$	$k_1 = 2 \cdot 10^{-2}$	$\hat{k}_1 \approx 3.5 \cdot 10^{-3}$
$D \to A$	$k_2 = 2 \cdot 10^{-2}$	$\hat{k}_2 = 0$
$D \to B$	$k_3 = 1 \cdot 10^{-2}$	$\hat{k}_3 \approx 9.1 \cdot 10^{-3}$
$A + B \to C$	$k_4 = 1 \cdot 10^{-4}$	$\hat{k}_4 \approx 8.7 \cdot 10^{-5}$
$C \to D + B$	$k_5 = 1 \cdot 10^{-4}$	$\hat{k}_5 = 0$

Table 1: Original and estimated reaction rates for the example in Section 3.2.

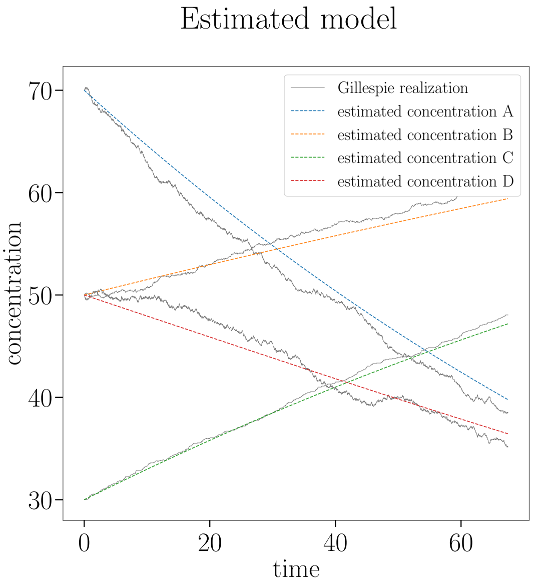


Figure 5: Integrated law of mass action equations (3) for the sparse estimated parameters and the corresponding Gillespie realization for the example in Section 3.2.

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 noisy data, but also yield, in this case, a sparse set of rates with respect to generating reactions (5).

In two examples it was demonstrated that despite noisy data and unavailable derivative measurements, a parsimonious generating reaction network that is qualitatively able to explain the observed data can be estimated. In particular it was shown in the first example that if there is no ambiguity in the underlying model and ansatz reaction library, the actual rates can be recovered with decreasing time step, i.e., increasing resolution of the jump process. In the second example we could obtain an even simpler model than what was used to generate data 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] A. E. Hoerl and R. W. Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12:55–67, 1970.

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

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

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

[8] H. Zou and T. Hastie. Regularization and variable selection via the elastic-net. *Journal of the Royal Statistical Society*, 67(2):301–320, 2005.