Probabilistic Signaling Networks

Mechanistic Modeling in Markov Categories

Marius P. Furter
University of Zurich
marius.furter@math.uzh.ch

Biology needs principled modeling

Life is complex

Even single bacteria contain millions of interacting molecules.

E. Coli	
Proteins:	3 million 4k types
Fats:	20 million
DNA:	5 million basepairs 4k genes
mRNA:	1500

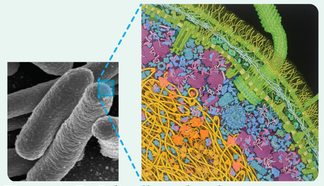


Figure 1: *E. Coli* cells under electron microscope (**left**) with illustration of internal structure (**right**). Adapted from [1].

Modern experiments require automated analysis

High-throughput techniques perform thousands of measurements at once. To evaluate such data we need automated methods for expressing and updating our incomplete state of knowledge.

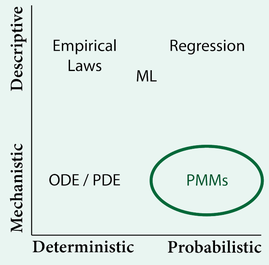
DNA sequencing	Protein mass spectrometry
10 billion basepairs / run	1500 protein concentrations / run

Probabilistic mechanistic models (PMMs) express physical understanding but allow uncertainty

Molecular Biology often uses deterministic models based on physical mechanisms. Probability is used to study noisy processes.

Using probability more broadly to express incomplete knowledge provides a principled way to incorporate experimental evidence.

PMMs are generative probability models based on physical mechanisms. Unlike black-box models, they are highly structured and have meaningful parameters with strong causal interpretations.



Modeling in Markov categories

Directly specifying a probability model is impractical when working with systems containing hundreds of components. Moreover, biologists often receive little training in probability and statistics.

Instead, we propose a modular hierarchical framework where each level can be automatically compiled to the one below it.

Domain specific modeling languages

On the top level, we provide a problem specific language that domain experts interact with. These can be based on ones already in use in a particular field and can hide many implementation details.

We will be interested in modeling languages defined by constructions on Markov categories. Objects of such languages compile to morphisms representing generalized probability models.

String diagrams as graphical models

Markov categories provide an abstract setting for probability theory [2] where morphism represent non-deterministic maps. Describing such morphisms with string diagrams [3] can be seen as a type of probabilistic graphical model [4].

Because of the mechanistic basis of our models, causal interpretations are warranted. For instance, we can use them to predict the effect of interventions [5] and to guide future experimentation.

Bayesian inference can be formulated abstractly in Markov categories [6]. Moreover, Markov categories enriched in divergence measures [7] could serve as the basis for variational inference.

The compositional nature of theses methods allows them to be performed according to several decompositions, some of which may be more computationally efficient. This should allow us to exploit the sparse structure of our models.

Computation as probabilistic programs

Models specified as string diagrams can be compiled to probabilistic programs on which inference techniques can be applied.

Probabilistic signaling networks (PSNs)

A PSN is a network consisting of interconnected transducers. At each time-step, a transducer updates its state according the inputs it receives. The updates can be probabilistic.

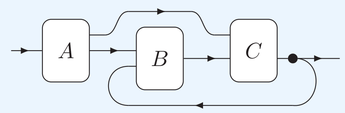


Figure 2: A protein signaling network with feedback.

Figure 2 shows proteins in a signaling network that interact along directed wires. We can model this as a PSN where each protein is a transducer activated according to

$$\frac{\Delta\alpha}{\Delta t} = r(H(\boldsymbol{i}) - \alpha)$$

with state α, input vector $\boldsymbol{i}$, rate r and hyper-Hill function H [8].

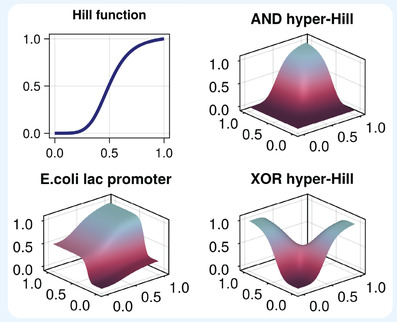


Figure 3: Hyper-Hill functions are multi-input generalizations of Hill functions (**top left**). They sigmoidally interpolate values specified on the corners of the hypercube. They generalized boolean functions and can capture biological activation landscapes (**bottom left**).

PSNs as lenses

PSNs are lenses in BorelStoch, the category of Markov kernels and standard Borel spaces. Each transducer has an update map

$$up_T \colon [0,1] \times [0,1]^{k_T} \to [0,1]$$

representing the probability $p(\alpha[t+1] \mid \alpha[t], i_1[t], \dots, i_{k_T}[t])$ where k_T is its number of inputs. Transducers are wired together by lens composition [9] according to the network structure. This yields the global update and output maps shown in Figure 4.

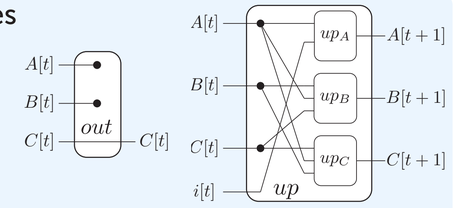


Figure 4: The global output (**left**) and update (**right**) maps can be derived from the clearer network representation.

Dynamics

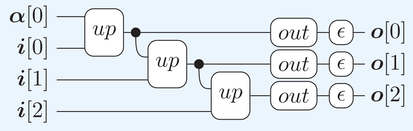


Figure 5: Dynamic model describing the probability of an output sequence given and input sequence and initial state. The pattern can be extended arbitrarily.

The update map of a PSN inductively defines a morphism in BorelStoch representing its discrete dynamics (Fig. 5). We add Gaussian noise to the output to model measurement error. The result forms a type of hidden Markov model.

Simulation

If we fix all parameters of our PSN, we can simulate its dynamics. Suppose that A and C are only activated if all their inputs are active. Assume further that C provides negative feedback to B so that B is only activated if A is active and C inactive. Driving the network with a constant input of 1 results in the dynamics shown in Figure 6. We emulate measurement of C by adding Gaussian noise.

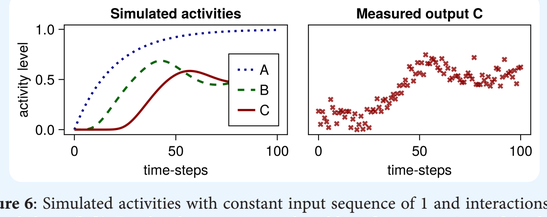


Figure 6: Simulated activities with constant input sequence of 1 and interactions described above (**left**). Emulated measurements by adding Gaussian noise to C (**right**).

Inference

We now recover the parameter values from our synthetic data using Markov chain Monte Carlo (MCMC) inference in Turing [10]. The necessary computer code can be automatically generated from a graphical specification of the PSN in software.

We suppose we have prior knowledge that C inhibits B and that A is activated by its input. We do not assume any knowledge about the other interaction parameters.

The results of the inference are shown in Figures 7 and 8. We faithfully reconstruct the hidden states as well as most interaction parameters. The two parameters that remain unidentified cannot in principle be inferred from our experiment.

Using MCMC is convenient but does not exploit the structure of the problem. Finding general methods for efficient inference on string diagrams is the focus of current work.

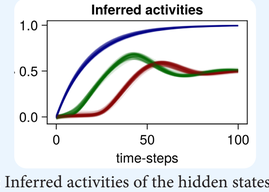


Figure 7: Inferred activities of the hidden states.

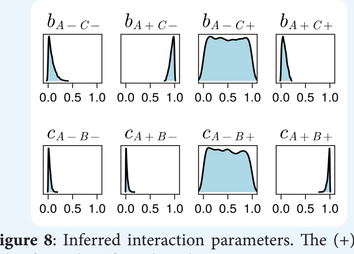


Figure 8: Inferred interaction parameters. The (+) and (−) refer to the effect when the input is active or inactive.

References

[1] David S. Goodsell. *Escherichia Coli Bacterium*. 2021. doi: 10.2210/rcsb_pdb/goodsell-gallery-028.

[2] Tobias Fritz. *"A synthetic approach to Markov kernels, conditional independence and theorems on sufficient statistics"*. Advances in Mathematics 370 (2020), p. 107239. doi: 10.1016/j.aim.2020.107239. eprint: arXiv:1908.07021.

[3] Peter Selinger. *"A Survey of Graphical Languages for Monoidal Categories"*. In: New Structures for Physics. Springer Berlin Heidelberg, 2010, pp. 289–355. doi: 10.1007/978-3-642-12821-9_4.

[4] Daphne Koller and Nir Friedman. *Probabilistic graphical models. Principles and techniques*. MIT Press, 2010. isbn: 9780262013192.

[5] Robin Lorenz and Sean Tull. *"Causal models in string diagrams"* (2023). doi: 10.48550/arXiv.2304.07638.

[6] Kenta Cho and Bart Jacobs. *"Disintegration and Bayesian inversion via string diagrams"*. Mathematical Structures in Computer Science 29.7 (2019), pp. 938–971. doi: 10.1017/s0960129518000488. eprint: arXiv: 1709.00322.

[7] Paolo Perrone. *"Markov Categories and Entropy"* (2022). doi: 10.48550/arXiv.2212.11719.

[8] Dominik M. Wittmann, Jan Krumsiek, Julio Saez-Rodriguez, Douglas A. Lauffenburger, Steffen Klamt, and Fabian J. Theis. *"Transforming Boolean models to continuous models: methodology and application to T-cell receptor signaling"*. BMC Systems Biology 3.1 (2009). doi: 10.1186/1752-0509-3-98.

[9] David Jaz Myers. *Categorical Systems Theory*. Work in progress. Jan. 2023. url: http://davidjaz.com/Papers/DynamicalBook.pdf.

[10] Hong Ge, Kai Xu, and Zoubin Ghahramani. *"Turing: A Language for Flexible Probabilistic Inference"*. In: Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics. Vol. 84. Proceedings of Machine Learning Research. PMLR, 2018, pp. 1682–1690. url: https://proceedings.mlr.press/v84/ge18b.html.

Acknowledgments

I thank the University of Zurich and the Digital Society Initiative for funding this work. I also acknowledge the partial support of SNF Grant No. 200020_192080.