A Toric Variety from Machine Learning

Zhangsheng Lai   Orlando Marigliano

April 11, 2018

Given a directed graph G without multiple edges, we introduce a biology-inspired statistical model, the *McCulloch-Pitts process (MPP)*, for recurrent neural networks. We associate a toric variety to such a model and compute its Hilbert polynomial in a special case.

Motivation

I'm placing the connections to machine and RNNs here, seems weird to have it in simulation section.

The artificial neural networks that we have in deep learning today are inspired by biological neurons in mammalian brains. Biological neurons are electrical excitable, where a neuron spikes and discharges electrical signals through their synapses, which are the complex membrance junctions that transmit signals to other neurons. An example given in Figure 1 is the feedforward neural network, where in the case of computer vision, we feed the network with an image to do classification of the image into one of the many classes.

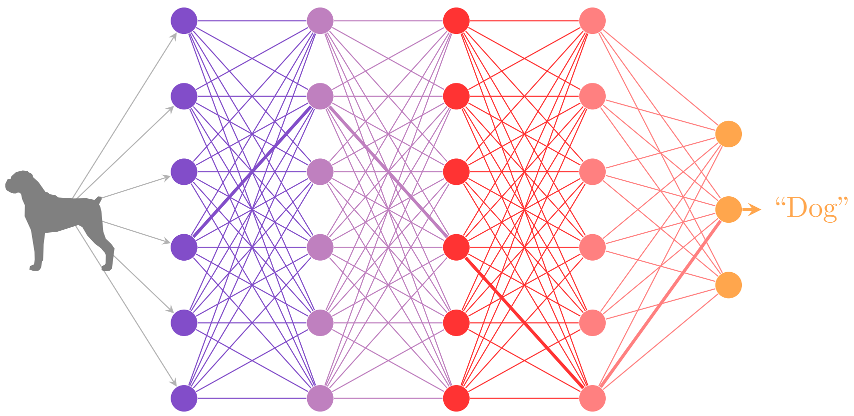

Figure 1: Feedforward Neural Network

However, in a feedforward network, the connection of the different layers of the neuron do not form a cycle. This is not true for biological neurons as there is no one direction of the connection of the neurons. Thus we look to recurrent neural networks (RNN), where the connections between the neurons to form a directed cycle is allowed. One of the popular RNNs is long short term memory (LSTM), which are able to connect previous information to the present task. The neurons in LSTM communicate with real values, which is different from how biological neurons communicate in mammalian brains. Therefore, we only allow the neurons in our network to be binary-valued, $\{0, 1\}$ where a transition $0 \to 1$ represents a spiking activity and a transition $1 \to 0$ represents the recovery of a neuron to an armed state.

The statistical model

Given a directed graph $G = (V, E)$ without multiple edges, with vertex weights $\beta_i > 0$ and edge weights $\alpha_{ij} > 0$, a *McCulloch-Pitts process (MPP)* is an activity-based process with binary states $x \in \{0, 1\}^{|V|}$ and transitions xy where state y is one-bit away from state x. If y and x differs in the i-th bit, we define the transition rate

$$F_{xy} = [\beta_i^{\sigma_i} \alpha_i^{x\sigma_i}]^{1/\tau}$$

where $\alpha_i^x = \alpha_{1i}^{x_1} \alpha_{2i}^{x_2} \dots \alpha_{di}^{x_d}$ with $|V| = d$ and $\sigma_i = 1 - 2x_i \in \{-1, +1\}$ denotes the change in the state of the i-th bit.

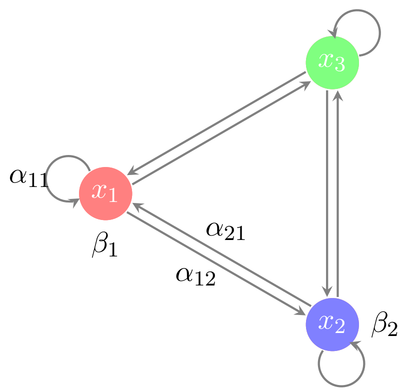

Figure 2: McCulloch-Pitts process with three neurons.

Working with the example in Figure 2 and choosing $\tau = 1$, the transition rate matrix is given by

$$F = \begin{array}{cc} & \begin{matrix} 000 & 001 & 010 & 011 & 100 & 101 & 110 & 111 \end{matrix} \\ \begin{matrix} 000 \\ 001 \\ 010 \\ 011 \\ 100 \\ 101 \\ 110 \\ 111 \end{matrix} & \begin{bmatrix} * & \beta_3 & \beta_2 & 0 & \beta_1 & 0 & 0 & 0 \\ \beta_3^{-1}\alpha_{33}^{-1} & * & 0 & \beta_2\alpha_{32} & 0 & \beta_1\alpha_{31} & 0 & 0 \\ \beta_2^{-1}\alpha_{22}^{-1} & 0 & * & \beta_3\alpha_{23} & 0 & 0 & \beta_1\alpha_{21} & 0 \\ 0 & \beta_2^{-1}\alpha_{22}^{-1}\alpha_{32}^{-1} & \beta_3^{-1}\alpha_{23}^{-1}\alpha_{33}^{-1} & * & 0 & 0 & 0 & \beta_1\alpha_{21}\alpha_{31} \\ \beta_1^{-1}\alpha_{11}^{-1} & 0 & 0 & 0 & * & \beta_3\alpha_{13} & \beta_2\alpha_{12} & 0 \\ 0 & \beta_1^{-1}\alpha_{11}^{-1}\alpha_{31}^{-1} & 0 & 0 & \beta_3^{-1}\alpha_{13}^{-1}\alpha_{33}^{-1} & * & 0 & \beta_2\alpha_{12}\alpha_{32} \\ 0 & 0 & \beta_1^{-1}\alpha_{11}^{-1}\alpha_{21}^{-1} & 0 & \beta_2^{-1}\alpha_{12}^{-1}\alpha_{22}^{-1} & 0 & * & \beta_3\alpha_{13}\alpha_{23} \\ 0 & 0 & 0 & (\beta_1\alpha_1^{111})^{-1} & 0 & (\beta_2\alpha_2^{111})^{-1} & (\beta_3\alpha_3^{111})^{-1} & * \end{bmatrix} \end{array}$$

where $*$ denotes the negative of the sum of its corresponding row, and $\alpha_i^{111} = \alpha_{1i}\alpha_{2i}\alpha_{3i}$.

Simulation

The simulation of the McCulloch-Pitts process starts by drawing an inital state $x^{(0)}$ from a distribution $p^{(0)}$. Then for any state x, it holds for some time

$$\Delta t \sim \mathrm{Exp}(\lambda_x)$$

where $\lambda_x = \sum_{y \neq x} F_{xy}$ and it transits to state y which is one hop away with probability

$$F_{xy}/\lambda_x$$

Thus the temporal data obtained from the simulation are binary tuples of length $|V|$ and an associated holding time for each pair of consecutive state.

```
0000: state [1 1 0 1 1 1 1 0] | neuron 002 | holding time 0.0032901008027207005
0001: state [1 1 1 1 1 1 1 0] | neuron 003 | holding time 0.16460101982700073
0002: state [1 1 1 0 1 1 1 0] | neuron 007 | holding time 0.019009826806025597
0003: state [1 1 1 0 1 1 1 1] | neuron 003 | holding time 0.13344528911418524
0004: state [1 1 1 1 1 1 1 1] | neuron 006 | holding time 0.03776971154447096
0005: state [1 1 1 1 1 1 0 1] | neuron 007 | holding time 0.003934267162875663
0006: state [1 1 1 1 1 1 0 0] | neuron 006 | holding time 0.38924525541203236
0007: state [1 1 1 1 1 1 1 0] | neuron 006 | holding time 0.011087533081789665
```

Figure 3: Temporal data generated from simulation for a MPP with eight neurons

The Toric Variety

We consider the space of weights $W := \mathbb{C}^{d+|E|} = \{(\beta_i, \alpha_{jk}) \mid i \in V, (j,k) \in E\}$ and the space of transition rates $T := \mathbb{C}^{2^d d} = \{(F_{xy}) \mid x, y \text{ binary states differing at one bit}\}$. We have a map $f \colon W \to T$ defined by $f(\alpha, \beta) = (F_{xy}(\alpha, \beta))_{xy}$. We define the toric variety X as the Zariski closure of the image of f.

In the above example, we get the map $f\colon \mathbb{C}^{12} \to \mathbb{C}^{24}$ and the induced toric variety X. Using Polymake, we can compute the Hilbert series of (the closure in $\mathbb{P}^{24}$ of) X. We obtain The Hilbert series

$$\frac{P(x)}{(1-x)^{12}}$$

with

$$P(x) = x^6 + 12x^5 + 51x^4 + 88x^3 + 51x^2 + 12x + 1.$$

The dimension of X is thus 12, i. e. the degree of the denominator, and the degree of X is $P(1) = 216$.

A group action

Let π be a graph isomorphism of G, seen as a map $V \to V$. The map π acts on the space of weights by sending an element $(\beta_i, \alpha_{j,k})$ to $(\beta_{\pi i}, \alpha_{\pi j, \pi k})$, and similarly it acts on the space of transition rates. Since f is a Laurent map, for all weights (α, β) we have $\pi f(\alpha, \beta) = f \pi(\alpha, \beta)$. Hence we obtain a group action of $\mathrm{Aut}(G)$ on the variety X.

In the above example we see for instance that the permutation group S_6 acts on X.