Decoding From Spike Trains

Zoran Nenadic
Division of Engineering and Applied Science
California Institute of Technology
Pasadena, CA 91125

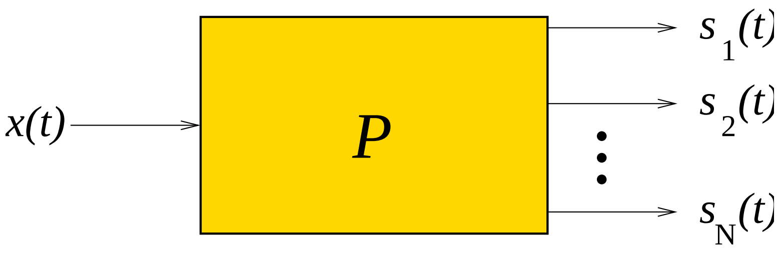

Figure 1: Population of neurons

1 Introduction

Suppose that we have a population $\mathcal{P}$ of an arbitrary number of neurons N. Suppose that the population responds to a time dependent scalar variable $x(t)$ by producing a number of spike trains as shown by Fig. 1, i.e.

$$S(t) = [s_1(t),\, s_2(t),\, \cdots,\, s_N(t)],$$

where

$$s_j(t) = \sum_{k=1}^{n_j(t)} \delta(t - t_k^j) \qquad \forall j \in \{1,\, 2 \cdots,\, N\},$$

and $n_j(t)$ is the total number of spikes produced by the $j-$th neuron in the interval $[0,\, t]$. We allow the possibility of $n_j = 0$, in which case $s_j(t) \equiv 0$.

Let us suppose that the population acts with a certain degree of randomness, i.e. the repetition of the same $x(t)$ does not produce identical response. Assuming that the collection of spike trains carries "enough" information about $x(t)$ (i.e. $x(t)$ is **encoded** in the spike trains), we can ask the following question: How would one decode an arbitrary function of x (and in particular x itself) from the spike train observations?

2 Detection Problem

Let $O(t)$ be the observed response of the population $\mathcal{P}$ to a time varying scalar function $x(t)$. In general $O(t)$ can be any set of features of the response $S(t)$. Using Bayes' rule one can easily write

$$P(x(t) \,|\, O(t)) = \frac{P(O(t) \,|\, x(t))\, P(x(t))}{P(O(t))} \tag{1}$$

If $x(t)$ takes values from a discrete set of functions $\{x_i(t)\}_1^M$, we can formulate the following **detection problem**

$$x(t) = \{x_m(t) \,|\, x_m(t) = \arg\max_i P(x_i(t) \,|\, O(t))\} \tag{2}$$

The idea given by (2) is not a new one and has been extensively used for various applications. For example, if

$$O(t) = [n_1(t),\, n_2(t),\, \cdots,\, n_N(t)]$$

where $n_j(t)$ is the number of spikes fired by $j-$ th neuron in time interval $[0,\, t]$, one has so-called **rate decoding**. Since neurons fire with a considerable degree of variability, one might argue that the precise spike timings are not important in the encoding process, and that the idea of rate decoding is justified. However, this encoding/decoding scheme is not very efficient, especially if the firing rates of neurons within the population are not sufficiently high. Assuming the signals are encoded in a sequence of spike times $\{t_k^j\}_1^{n_j}$ increases the information capacity of the population $\mathcal{P}$ tremendously. Here, we propose the decoding method that is based on full statistical description of the population response, namely we assume that $O(t) = S(t)$.

3 Decoding Algorithm

The goal of this section is to present a decoding method that utilizes full statistical description of the response $S(t)$. In this context the equation (1) becomes

$$\begin{aligned} P(x_i(t)\,|\,S(t)) &= \frac{P(S(t)\,|\,x_i(t))\,P(x_i(t))}{P(S(t))} \Rightarrow \\ P(x_i(t)\,|\,[s_1(t),\, s_2(t),\, \cdots,\, s_N(t)]) &= \\ &= \frac{P([s_1(t),\, s_2(t),\, \cdots,\, s_N(t)]\,|\,x_i(t))\,P(x_i(t))}{P([s_1(t),\, s_2(t),\, \cdots,\, s_N(t)])} \\ &= \frac{P([s_1(t),\, s_2(t),\, \cdots,\, s_N(t)]\,|\,x_i(t))\,P(x_i(t))}{\sum_{i=1}^{M} P([s_1(t),\, s_2(t),\, \cdots,\, s_N(t)]\,|\,x_i(t))\,P(x_i(t))} \end{aligned} \tag{3}$$

The proposed scheme is fairly general and often difficult to implement. To make the problem tractable and easy to implement we will make several important assumptions.

Assumption The responses of individual neurons are statistically independent. The consequence of this assumption is that

$$P([s_1(t),\, s_2(t),\, \cdots,\, s_N(t)]\,|\,x_i(t)) = \prod_{j=1}^{N} P(s_j(t)\,|\,x_i(t)) \tag{4}$$

Assumption The prior probabilities of individual inputs are equal i.e.

$$P(x_i(t)) = \frac{1}{M} \qquad \forall i = 1,\, 2,\, \cdots,\, M \tag{5}$$

Applying (4) and (5) to the decoding scheme (3) we have

$$P(x_i(t)\,|\,[s_1(t),\, s_2(t),\, \cdots,\, s_N(t)]) = \frac{\displaystyle\prod_{j=1}^{N} P(s_j(t)\,|\,x_i(t))}{\displaystyle\sum_{i=1}^{M}\prod_{j=1}^{N} P(s_j(t)\,|\,x_i(t))} \tag{6}$$

The chief difficulty of decoding in the present context is determining the conditional probability of $j-$th response given $i-$th input.

Let $s_j(t) = \sum_{k=1}^{n_j(t)} \delta(t - t_k^j)$ be the response of the $j-$th neuron in the population to the input $x_i(t)$. Our goal is to evaluate the conditional probability $P(s_j(t)\,|\,x_i(t))$. Since we consider only one input-response pair at a time, the indices i and j will be dropped for simplicity. The signal $s(t)$ is fully characterized by the sequence of times $\{t_1,\, t_2,\, \cdots,\, t_n\}$ so we have

$$P(s(t)\,|\,x(t)) = P(\theta_1 = t_1,\, \theta_2 = t_2,\, \cdots,\, \theta_n = t_n,\, \text{no spikes in}\, [t_n,\, t]\,|\,x(t)),$$

where θ_k is a random variable that corresponds to the arrival time of the $k-$th spike in the spike train. Clearly, θ_k is a continuous random variable, so the probability of the event above is equal to 0, and we are better off with its likelihood (probability density function), defined by

$$\begin{aligned} f(s(t)\,|\,x(t)) &\triangleq \frac{\partial^n P(s(t)\,|\,x(t))}{\partial t_1 \partial t_2 \cdots \partial t_n} = \\ &= \lim_{dt_1 \to 0} \cdots \lim_{dt_n \to 0} \frac{P(\theta_1 \in [t_1,\, t_1 + dt_1], \cdots, \theta_n \in [t_n,\, t_n + dt_n],\, N_{[t_n + dt_n,\, t]} = 0)}{dt_1 \cdots dt_n}, \end{aligned}$$

where conditioning on $x(t)$ has been dropped for simplicity and $N_{[t_n + dt_n,\, t]} = 0$ means that we have no spikes on the interval $[t_n + dt_n,\, t]$.

Assumption The arrivals (non-arrivals) at instant t are only dependent on the previous arrival. This Markov-type assumption means that θ_n depends on θ_{n-1} only.

Under this assumption the conditional probability calculation further simplifies to

$$\begin{aligned} P(s(t)\,|\,x(t)) &= P(N_{[t_n + dt_n,\, t]} = 0\,|\,\theta_n, \cdots, \theta_1)\, P(\theta_n, \cdots, \theta_1) = \\ &= P(N_{[t_n + dt_n,\, t]} = 0\,|\,\theta_n)\, P(\theta_n, \cdots, \theta_1) \\ &= P(N_{[t_n + dt_n,\, t]} = 0\,|\,\theta_n)\, P(\theta_n\,|\,\theta_{n-1}, \cdots, \theta_1)\, P(\theta_{n-1}, \cdots, \theta_1) \\ &\ \ \vdots \\ &= P(N_{[t_n + dt_n,\, t]} = 0\,|\,\theta_n)\, P(\theta_n\,|\,\theta_{n-1}) \cdots P(\theta_2\,|\,\theta_1)\, P(\theta_1), \end{aligned}$$

and the probability density function (pdf) becomes

$$\begin{aligned} & f(s(t)\,|\,x(t)) = \\ &= \lim_{dt_1 \to 0} \cdots \lim_{dt_1 \to 0} \frac{P(N_{[t_n + dt_n,\, t]} = 0\,|\,\theta_n)\, P(\theta_n\,|\,\theta_{n-1}) \cdots P(\theta_2\,|\,\theta_1)\, P(\theta_1)}{dt_1 \cdots dt_n} \\ &= P(N_{[t_n,\, t]} = 0\,|\,\theta_n = t_n)\, f_{\theta_n\,|\,\theta_{n-1}}(t_n\,|\,t_{n-1}) \cdots f_{\theta_2\,|\,\theta_1}(t_2\,|\,t_1)\, f_{\theta_1}(t_1), \end{aligned}$$

where $f_{\theta_n \,|\, \theta_{n-1}}$ represent **transition densities**. It is often more useful to use inter-spike intervals (ISI) defined by $T_n = \theta_n - \theta_{n-1}$ ($\theta_0 = 0$) instead of spike arrivals θ_n. The conditional density then becomes

$$f(s(t) \,|\, x(t)) = \\ = K\, P(N_{[t_n,t]} = 0 \,|\, T_n = \tau_n)\, f_{T_n \,|\, T_{n-1}}(\tau_n \,|\, \tau_{n-1}) \cdots f_{T_2 \,|\, T_1}(\tau_2 \,|\, \tau_1)\, f_{T_1}(\tau_1),$$

where $\tau_n = t_n - t_{n-1}$ ($t_0 = 0$) and K is a normalization constant that makes $f(s(t) \,|\, x(t)$ a valid pdf candidate. The transition densities $f_{T_n \,|\, T_{n-1}}$ are to be found using either parametric or non-parametric methods. Parametric methods rely on assuming the transition densities are parameterized by a number of unknown parameters which are found from experimental observations. Non-parametric methods rely on direct (point-wise) estimate of the transition densities. Both methods are based on experimental data. Using densities instead of probabilities the decoding algorithm (6) becomes

$$f(x_i(t) \,|\, [s_1(t),\, s_2(t),\, \cdots,\, s_N(t)]) = \frac{\displaystyle\prod_{j=1}^{N} f(s_j(t) \,|\, x_i(t))}{\displaystyle\sum_{i=1}^{M} \prod_{j=1}^{N} f(s_j(t) \,|\, x_i(t))} \tag{7}$$

To illustrate the application of the algorithm above, let us suppose that the underlying spike generating mechanism is a Poisson process with a constant rate λ. One can easily show that

$$f_{T_n \,|\, T_{n-1}}(\tau_n \,|\, \tau_{n-1}) = f_{T_n}(\tau_n) = \lambda\, e^{-\lambda\, \tau_n} \qquad \text{(renewal assumption)}$$

In particular one has

$$f(s(t) \,|\, x(t)) = K\, e^{-\lambda\,(t-t_n)} \lambda\, e^{-\lambda\, \tau_n} \cdots \lambda\, e^{-\lambda\, \tau_1} = K\, \lambda^{n(t)}\, e^{-\lambda\,(t - t_n + \tau_n + \cdots + \tau_1)} = \\ = K\, \lambda^{n(t)}\, e^{-\lambda\,(t - t_n + t_n - t_{n-1} + \cdots + t_2 - t_1 + t_1)} = K\, \lambda^{n(t)}\, e^{-\lambda\, t}.$$

To signify that the rate λ depends on both input and cell, we write

$$\lambda = \Lambda_j(x_i),$$

where $x_i(t) = x_i = const$ and $\Lambda_j(x)$ is so-called **tuning curve** of the $j-$th cell. One can easily show that in this case $K = 1/n(t)!$. Written more detailed, the conditional pdf is given by

$$f(s_j(t) \,|\, x_i(t)) = \frac{[\Lambda_j(x_i)]^{n_j(t)}}{n_j(t)!}\, e^{-\Lambda_j(x_i)\, t},$$

and finally the decoding scheme (7) simply becomes

$$f(x_i(t) \,|\, [s_1(t),\, s_2(t),\, \cdots,\, s_N(t)]) = \frac{\displaystyle\prod_{j=1}^{N} \tfrac{[\Lambda_j(x_i)]^{n_j(t)}}{n_j(t)!} e^{-\Lambda_j(x_i)\, t}}{\displaystyle\sum_{i=1}^{M} \prod_{j=1}^{N} \tfrac{[\Lambda_j(x_i)]^{n_j(t)}}{n_j(t)!} e^{-\Lambda_j(x_i)\, t}} \tag{8}$$

This result coincides with decoding scheme based on firing rates only. Namely, if $n_j(t)$ is the number of spikes fired by the $j-$th cell on the interval $[0,\, t]$, one can rewrite (7) as

$$P(x_i(t)\,|\,[n_1(t),\, n_2(t),\, \cdots,\, n_N(t)]) = \frac{\prod P(n_j(t)\,|\,x_i(t))}{\sum\prod P(n_j(t)\,|\,x_i(t))}, \tag{9}$$

and

$$P(n_j(t)\,|\,x_i(t)) = \frac{[\Lambda_j(x_i)\,t]^{n_j(t)}}{n_j(t)!}\,e^{-\Lambda_j(x_i)\,t}.$$

Finally (9) becomes

$$\begin{aligned} P(x_i(t)\,|\,[n_1(t),\, n_2(t),\, \cdots,\, n_N(t)]) &= \frac{\prod \frac{[\Lambda_j(x_i)\,t]^{n_j(t)}}{n_j(t)!}\,e^{-\Lambda_j(x_i)\,t}}{\sum\prod \frac{[\Lambda_j(x_i)\,t]^{n_j(t)}}{n_j(t)!}\,e^{-\Lambda_j(x_i)\,t}} = \\ = \frac{t^{n_1(t)+\cdots+n_N(t)}\prod \frac{[\Lambda_j(x_i)]^{n_j(t)}}{n_j(t)!}\,e^{-\Lambda_j(x_i)\,t}}{t^{n_1(t)+\cdots+n_N(t)}\sum\prod \frac{[\Lambda_j(x_i)]^{n_j(t)}}{n_j(t)!}\,e^{-\Lambda_j(x_i)\,t}} &= \frac{\prod \frac{[\Lambda_j(x_i)]^{n_j(t)}}{n_j(t)!}e^{-\Lambda_j(x_i)\,t}}{\sum\prod \frac{[\Lambda_j(x_i)]^{n_j(t)}}{n_j(t)!}e^{-\Lambda_j(x_i)\,t}} \end{aligned} \tag{10}$$

which is result identical to (8). This result is not surprising since Poisson process is completely determined by the rate λ and taking into account full statistical description of the spike trains does not yield any new information.

4 Simulation Results

Suppose that we have a population of $N = 40$ neurons, and suppose that the neurons are firing according to Poisson model with dead time $\triangle = 2$ ms (this can be taught of as absolute refractory period, i.e. the time interval following a spike in which neuron is unable to fire an action potential). Let us assume that the 40 cells within the population have their mean firing rates as shown by Fig. 2. These curves could be experimentally obtained by averaging the response of individual neurons to a sequence of constant stimuli of different magnitude. In particular, the tuning curves from Fig. 2 are obtained as raised cosine functions

$$\Lambda_j(x) = k_j\,\cos(x - x_j) + c_j \qquad j = 1, 2, \cdots, 40 \tag{11}$$

where $x \in [0,\, 2\pi]$, x_j is the **preferred direction** of $j-$th neuron and k_j and c_j are constants that determine the height and the width of the tuning curves. For simplicity we take $k_j = c_j$ and we draw x_j from a uniform distribution over $[0,\, 2\pi]$. Note that our maximum firing rate in the population does not exceed 24 spikes/s, which should make decoding process more challenging.

In the spirit of detection problem defined by equation (2), let us assume that x takes its values from the set $\mathcal{X} = \{0,\, \pi/4,\, \pi/2, \cdots,\, 7\,\pi/4\}$ $(M = 8)$. For each $x \in \mathcal{X}$ and for each neuron from the population, the mean firing rate λ is calculated

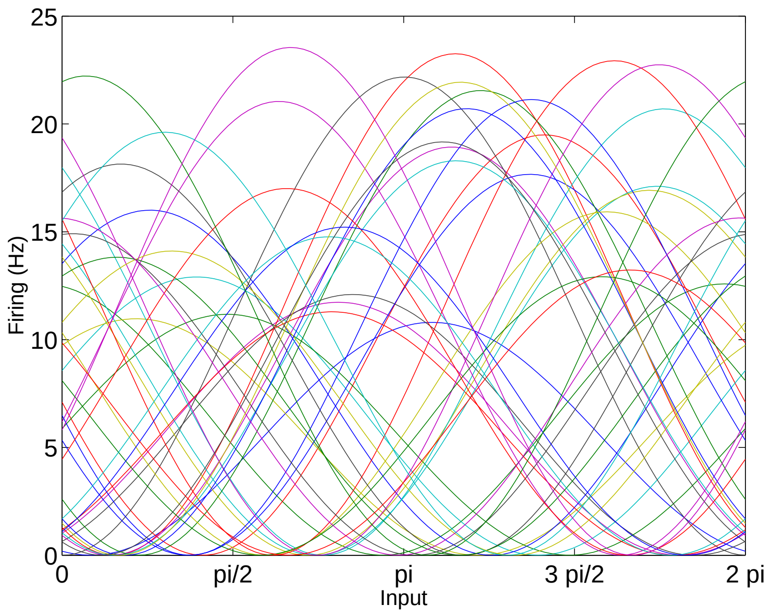

Figure 2: Tuning curves.

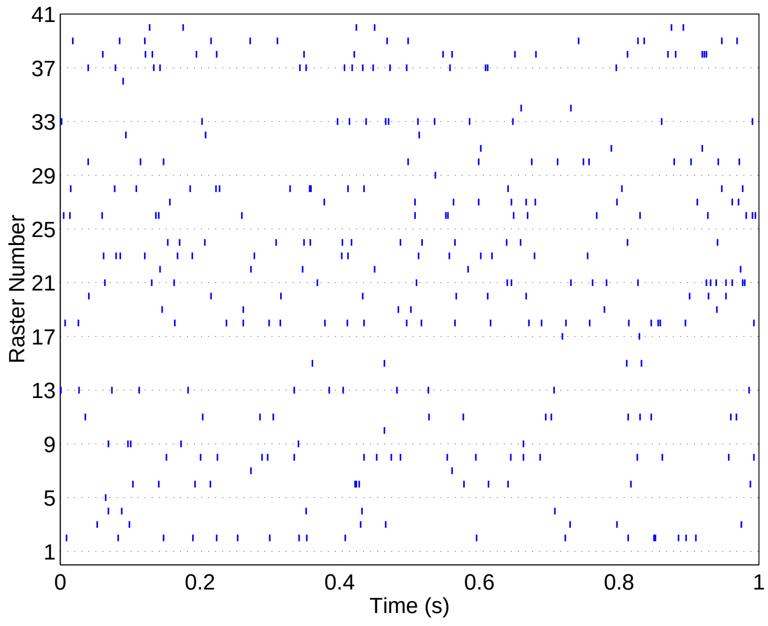

Figure 3: Raster plots.

according to (11), and sequences of spikes are generated using Poisson generator with dead time $\triangle = 2$ ms. The realization of one such collection of random processes is given by Fig. 3. The raster plots from Fig. 3 correspond to one of the eight inputs from $\mathcal{X}$. Given a conditional response, such as the one shown by Fig. 3, our goal is to estimate the most likely value of the input that elicited such response. Since we know what inputs are indeed behind every response, we can use this knowledge for cross-validation of our results. The decoding is performed according to (7).

The results of the decoding procedure across 8 inputs are shown in Fig. 4. The top plot in each subfigure shows the relative frequency of decoded directions. It is not surprising that the decoded input changes in time, despite the fact that the encoded input is constant in time. However, the decoded value stabilizes after some time, and does not change any more. The settling time is different for different inputs, e.g. 300 ms for input 4 ($x_4 = 3\,\pi/4$). The middle plot in each subfigure shows the traces of decoded inputs as a function of time. Input 7 is decoded with a 100% accuracy, i.e. this input emmerges as dominant (most likely) for all times. The bottom image in each subfigure shows the color coded likelihood of each input. The colorbar indicates that the likelihoods are normalized between 0 and 1.

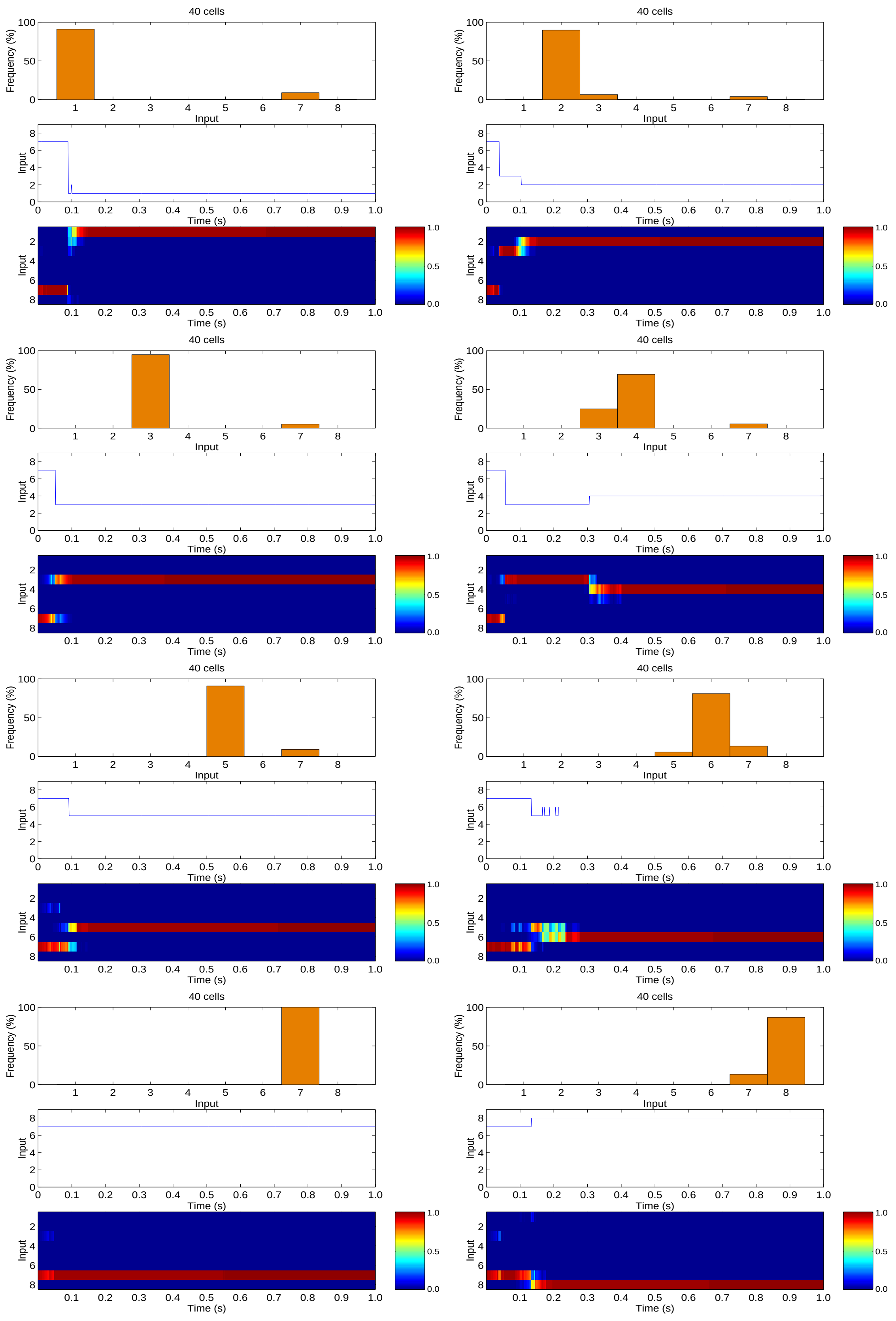

Figure 4: Detection results across 8 inputs.