Statistical models in neuroscience

Matthias H Hennig

18th April 2016
Contents

Multi-neuron activity

Binary representations of neural activity

The maximum entropy principle

The pairwise maximum entropy model

The Hopfield model

The Boltzmann machine

Restricted Boltzmann machines
Big data

[from Kording, 2011]
Neural activity

[data from cultured neurons, Maccione/Berdondini]
Neural activity

[data from Retina, from Hilgen/Sernargor]
Neural activity

100k neurons, 0.8Hz

[https://www.youtube.com/watch?v=Nxa19uWC_oA]
Understanding population activity

▶ top-down: analyse firing patterns and determine features (e.g. decoding, can identify rates, timings etc. as relevant quantities)

▶ bottom-up: low-dimensional parameterisation of firing patterns that makes it accessible to analysis

[after Nirenberg and Victor 2007]
Binary representations

The state of each of $N$ neurons $i$ at time $t$ ($0 \ldots T$) is modelled as a spin variable:

$$\sigma_i(t) = \begin{cases} 1 & \text{spike in } [t : t + \Delta t] \\ -1 & \text{no spike in } [t : t + \Delta t] \end{cases}$$

This yields patterns/words

$$\underline{\sigma}(t) = \{ \sigma_1(t), \sigma_2(t), \ldots, \sigma_N(t) \}$$

There are $2^N$ possible patterns $\underline{\sigma}_k$ with probabilities

$$p(\underline{\sigma}_k)$$
Binary patterns

How to obtain $P(\{\sigma_k\})$?

What does $P(\{\sigma_k\})$ tell us?
Two simple models

Full independence:

- firing does not depend on other neurons:
  \[ p(\sigma_i(t) = x | \sigma_{j\neq i}(t)) = p(\sigma_i(t)) \]
- firing does not depend on firing history:
  \[ p(\sigma_i(t) = x | \sigma_i(t - 1)) = p(\sigma_i(t)) \]

- Bernoulli distribution
- wrong, but useful as null-model

The complete set:

\[ \{ P(\sigma_k) \} \]

- requires specification of \(2^N\) probabilities
- hard to estimate from real data
- we do not gain
Interactions matter

- Pairwise correlations in network activity tend to be weak (correlations $<$ 0.1) but significant.
- Temporal history can also be important.

[graphs from Schneidman et al. (2006)]
Maximum entropy

Find the least structured model of the population activity:

- maximise entropy given a set of constraints \( \{ C \} \)

\[
P(\sigma_k, \theta_{ME}| C) \leftarrow \max_{\{ C \}} S(P(\sigma_k, \theta| C))
\]

- constraints should be things we are sure about
- does not impose structure

For spike data:

- rates and correlations can be estimated from finite data, but other constraints are harder
- we can’t access the detailed neural parameters
- some aspects of the activity depend on interactions between neurons
Lagrange multipliers

Optimisation problem:
find the maximum of $S(p)$ subject to a set of constraints $\{f_j(p, \sigma)\}$

$$L(p, \lambda) = - \sum_{i=1}^{M} p_i \log p_i + \sum_{j=1}^{C} \lambda_j f_j(p, \sigma)$$

$$\frac{\partial L(p, \lambda)}{\partial p_i} = 0$$
$$\frac{\partial L(p, \lambda)}{\partial \lambda_j} = 0$$

requires differentiable $f_j(p, \sigma) \forall j \in C$
Range constraint

\( M = 2^N \) possible states, with unknown probabilities \( p_i \).
We know nothing except that the system is finite.
What is our best guess for the \( p_i \)'s?

\[
L = - \sum_{i=1}^{M} p_i \log p_i + \lambda \left( \sum_{i=1}^{M} p_i - 1 \right)
\]

\[
\implies p_i = \frac{1}{M}
\]
Mean constraint

\[ M = 2^N \] possible states, with unknown probabilities \( p_i \).

Mean firing rates are known: 
\[ < \sigma_i >_t = \frac{1}{T} \sum_{t=1}^{T} \sigma_i(t) = \nu_i \]

\[
L = - \sum_{i=1}^{M} p_i \log p_i + \lambda \left( \sum_{i=1}^{M} p_i - 1 \right) + \sum_{j=1}^{N} h_j \left( \sum_{i=1}^{M} p_i \sigma_j - \nu_j \right)
\]

\[ \Rightarrow \]

\[
p_i = \frac{1}{\sum_{i=1}^{M} \exp(\sum_{j=1}^{N} h_j \sigma_j)} \exp(\sum_{j=1}^{N} h_j \sigma_j)
\]
The Maxwell-Boltzmann distribution

\[ \frac{N_i}{N} = \frac{1}{\sum_{j=1}^{M} \exp(-\beta E_j)} \exp(-\beta E_i) \]

where \( \beta = \frac{1}{k_B T} \) and \( E_i \) is the energy associated with state \( i \).
Mean and correlations constraint

\[ M = 2^N \] possible states, with unknown probabilities \( p_i \).

Correlations are known:

\[ < \sigma_i \sigma_j >_t = \frac{1}{T} \sum_{t=1}^{T} \sigma_i(t) \sigma_j(t) = c_{ij} \]

\[
L = - \sum_{i=1}^{N} p_i \log p_i + \lambda \left( \sum_{i=1}^{N} p_i - 1 \right) + \sum_{j=1}^{N} h_j \left( \sum_{i=1}^{M} p_i \sigma_j - \nu_j \right) + \sum_{j=1}^{N} \sum_{k=1}^{N} J_{jk} \left( \sum_{i=1}^{M} p_i \sigma_j \sigma_k - c_{jk} \right)
\]

\[ \implies p_i = \frac{\exp \left( \sum_{j=1}^{N} h_j \sigma_j + \frac{1}{2} \sum_{j \neq k}^{N} J_{jk} \sigma_j \sigma_k \right)}{\sum_{i=1}^{M} \exp \left( \sum_{j=1}^{N} h_j \sigma_j + \frac{1}{2} \sum_{j \neq k}^{N} J_{jk} \sigma_j \sigma_k \right)} \]
Spin models

This model is equivalent to the spin glass model with Hamiltonian (Energy):

\[ H = - \sum_{i=1}^{N} h_i \sigma_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} J_{ij} \sigma_i \sigma_j \]

\( \sigma_i \) are the magnetic moments or spins

\( <\sigma_i> \) is the magnetisation

\( h_i \) is the local field

\( J_{ij} \) is the symmetric coupling strength between spins

Maximum entropy distribution is the Boltzmann distribution:

\[ P \propto e^{-\frac{H}{k_B T}} \]

In our MaxEnt models, the temperature \( T \) is absorbed into the fields and couplings (and \( k_B = 1 \)).
Obtaining the parameters

Independent model:

\[
p(\{\sigma\}) = \frac{e^{\sum_{i=1}^{N} h_i \sigma_i}}{\prod_{i=1}^{N} 2 \cosh(h_i)}
\]

\[
h_i = \tanh^{-1}(\langle \sigma_i \rangle_t)
\]

Pairwise model:

\[
\delta h_i = \eta (\langle \sigma_i \rangle_{data} - \langle \sigma_i \rangle_{model})
\]

\[
\delta J_{ij} = \eta (\langle \sigma_i \sigma_j \rangle_{data} - \langle \sigma_i \sigma_j \rangle_{model})
\]

- Gradient ascent: Boltzmann learning with learning rate \( \eta \)
- Costly: \( Z \) has to be evaluated for every optimisation step
- Approximation: evaluate \( Z \) by Monte Carlo sampling, but the size of a Monte Carlo run should be similar to the data size, or use mean field approximations (see e.g. ?).
Independent versus pairwise model

Data from 10 neurons simultaneously recorded in the salamander retina. For example, 1011001010 occurs 1/min, but is predicted 1/3 yrs by independent model. from Schneidman et al. (2006)
Evaluating the model

Shlens et al. (2006)
Likelihood tests

Shlens et al. (2006)
Kullback-Leibler divergence between data $d$ and model $m$ (= negative of average log likelihood, hence invariant to data size):

$$D_{KL}(P_d || P_m) = \sum_{i \in \{\sigma\}} p_d(\sigma_i) \log \frac{p_d(\sigma_i)}{p_m(\sigma_i)}$$

[problematic for high dimensional and/or sparse data]
Pairwise model as Null model

Higher order interactions seem relevant at the scale of cortical microcolumns, but not beyond. Data from macaque V1, ?
Multi-information in an interacting system

Idea: hierarchy of entropies

\[ S_1 \geq S_2 \geq S_3 \geq \ldots \geq S_N = S(\{\sigma\}) \]

Joint entropy:

\[ S(\{\sigma\}) = -\sum_{\{\sigma_i\}} p(\sigma_i) \log p(\sigma_i) \leq \sum_{j=1}^{N} S(\sigma_j) \]

Multi-information: Entropy difference between independent model and full system, or \( D_{KL} \) between joint distribution and independent model produced from marginals \( \prod_j p(\sigma_i) \):

\[ I(\sigma) = \sum_j S(\sigma_j) - S(\{\sigma\}) = \sum_{\sigma_i} p(\sigma_i) \log \frac{p(\sigma_i)}{\prod_j p(\sigma_j)} \]
Decomposition of multi-information

$$I(\{\sigma_i\}) = \sum_{k=2}^{N} I^{(k)}(\{\sigma_i\})$$

With connected information:

$$I^{(k)}_C(\{\sigma_i\}) = S \left[ \tilde{p}^{(k-1)}(\{\sigma_i\}) \right] - S \left[ \tilde{p}^{(k)}(\{\sigma_i\}) \right] \geq 0$$

Where $\tilde{p}^{(k)}(\{\sigma_i\})$ is the ME distribution consistent with $k^{th}$-order marginals.

This allows to measure the contributions of all orders to the total entropy difference between independent model and data.

See Schneidman et al. (2003) for examples.
Multi-information in retinal recordings

from Schneidman et al. (2006)

But note: relative measures such as $I_2/I_N$ improve with very small time bins, low firing rates or small groups (?). It is therefore not always possible to extrapolate from small models.
Inferring functional connectedness

Schneidman et al. (2006)
Error correction

- Spiking can be unreliable.
- Suppose 1 neuron, 2 stimuli $S = \{1, 2\}$, $p(S = 1) = 0.5$
- Correct response: $p(\sigma_1 = 1|S = 1) = p_{S_1}$
- Incorrect: $p(\sigma = -1|S = 1) = 1 - p_{S_1}$
- Error probability for 3 identical neurons:

\[
(1 - p_{S_1})^3 + 3(1 - p_{S_1})^2 p_{S_1} < 1 - p_{S_1}
\]

- Simple majority rule yields correct answers for large $N$.
- But potentially inefficient.
Quantifies the entropy reduction through correlations: $I_N = S_1 - S_N$

When $I_N \to S_1$, correlations dominate the activity.

Schneidman et al. (2006) find: $I_N \to S_1$ for $N \approx 200$
Error correction

Information $N$ cells provide about the $(N+1)$th cell.
But: extrapolated from $N = 15$ + natural (correlated) stimuli
Schneidman et al. (2006)
Model extensions: Markov model

One time step constraint:
\[ < \sigma_i(t) \sigma_j(t + 1) >_t = \frac{1}{T} \sum_{t=1}^{T} \sigma_i(t) \sigma_j(t + 1) = c_{ij}^1 \]

\[ P(\{\sigma\}^t, \{\sigma\}^{t+1}) = \frac{\exp \left( \sum_{j=1}^{N} h_j \sigma_j^t + \frac{1}{2} \sum_{j \neq k}^{N} J_{jk}^t \sigma_j^t \sigma_k^t + \frac{1}{2} \sum_{j \neq k}^{N} J_{jk}' \sigma_j^t \sigma_k^{t+1} \right)}{\sum_{i=1}^{M} \exp \left( \sum_{j=1}^{N} h_j \sigma_j^t + \frac{1}{2} \sum_{j \neq k}^{N} J_{jk} \sigma_j^t \sigma_k^t + \frac{1}{2} \sum_{j \neq k}^{N} J_{jk}' \sigma_j^t \sigma_k^{t+1} \right)} \]

Higher orders in time are possible, then we have a n-step Markov model.

This is a special case of a GLM.

Marre et al. (2009)
Model extensions: Markov model

Marre et al. (2009)
Model extensions: K-pairwise model

Tkačik et al. (2014)
Model extensions: K-pairwise model

Add constraint on population firing rate:

$P_N(K)$: probability that $K$ of $N$ neurons fire simultaneously

$$
P_N(K) = \left\langle \delta \left( \sum_{i=1}^{N} \sigma_i, 2K - N \right) \right\rangle
$$

$$
\Rightarrow
$$

$$
P(\{\sigma\}) = \frac{1}{Z} \exp \left( E_{pair} + \sum_{K=1}^{N} \lambda_K \left( \sum_{j=1}^{N} \sigma_j \right)^K \right)
$$

Tkačík et al. (2014)
Model extensions: K-pairwise model

Good match of higher order structure

Tkačik et al. (2014)
Energy landscapes

Identified metastable states.
Tkačik et al. (2014)
Metastable states

Identified metastable states.
Tkačik et al. (2014)
Reproducible sequences of metastable states

Tkačík et al. (2014)
A small detour: Associative memory

- Retrieval of computer memory is address-based
  - localised: one address
  - error-prone: gone if one bit flipped in address
  - reliability through check-sums etc.

- In the brain memory retrieval appears content-addressable
  - associative: partial cues sufficient for recall
  - distributed: neurons may participate in multiple memories
  - error correcting: 'An American politician who was very intelligent and whose politician father did not like broccoli.' (MacKay, 2003)
  - robust: tolerates loss of neurons
The Hopfield network

Network of N neurons $s_i$ connected by weights $w_{ij}$:

$$s_i(t + 1) = \Theta \left( \sum_{j=1}^{N} w_{ij} s_j(t) - \theta_i \right)$$

$$\Theta(a) = \begin{cases} 
1 & a \geq 0 \\
-1 & a < 0 
\end{cases}$$

- updates can be synchronous or asynchronous
- without (much) loss of generality, we set $\theta_i = 0$
- Aim: activities $s_i$ should reflect a stored pattern when presented with similar inputs
Storing one pattern

Neurons $S = \{s_i\}$
Pattern $P = \{p_i\}$
Weights $W = \{w_{ij}\}$

\[
p_i = \Theta \left( \sum_{j=1}^{N} w_{ij} s_j \right)
\]

\[
w_{ij} \propto p_i p_j
\]
Hebbian learning

Store $M$ patterns $P = \{p^m\}$, each as a stable state of the activity rule.

\[
\Delta w_{ij} = \eta \sum_{m=1}^{M} p_i^m s_j^m
\]

- typically $\eta = 1/N$
- For $M \leq N$, this converges a network with separable attractors.
- Silence ($s_i = s_j = -1$) leads to positive weight change, contrary to Hebb proposal.
- Dale’s principle is also violated.
Stability

N Neurons $S = \{s_i\}$

M Patterns $P = \{p^m\}$

Weights $W = \{w_{ij}\}$

Stability condition:

$$\Theta(h_i^m) = p_i^m$$

with inputs

$$h_i^m = \sum_j w_{ij} p_j^m = \frac{1}{N} \sum_j \sum_{m'} p_i^{m'} p_j^{m'} p_j^m$$

$$= p_i^m + \frac{1}{N} \sum_j \sum_{m' \neq m} p_i^{m'} p_j^{m'} p_j^m$$

If the second term (crosstalk term) $< 1$, the pattern is stable. This depends on $M$, generally $M \leq N$. 

Hopfield energy function

$N$ Neurons $S = \{s_i\}$

$M$ Patterns $P = \{p^m\}$

Weights $W = \{w_{ij}\}$

$$H = -\sum_m \left( \sum_i s_i p_i^m \right)^2$$

$$= -\sum_{i,j} \left( \frac{1}{N} \sum_m p_i^m p_j^m \right) s_i s_j$$

$$= -\sum_{i,j} w_{ij} s_i s_j$$

This looks much like the Ising energy function.
Moving through energy landscapes

[from Wikipedia]
Moving through state space

[from Wikipedia]
Recall

(a) D J C M

(b) D D

(c) J J

(d) J J

(e) Q C

(f) K M

(g) I D

(h) X M

Figure 42.3. Binary Hopfield network storing four memories. (a) The four memories, and the weight matrix. (b–h) Initial states that differ by one, two, three, four, or even five bits from a desired memory are restored to that memory in one or two iterations. (i–m) Some initial conditions that are far from the memories lead to stable states other than the four memories; in (i), the stable state looks like a mixture of two memories, 'D' and 'J'; stable state (j) is like a mixture of 'J' and 'C'; in (k), we find a corrupted version of the 'M' memory (two bits distant); in (l) a corrupted version of 'J' (four bits distant) and in (m), a state which looks spurious until we recognize that it is the inverse of the stable state (l).

[from MacKay (2003)]
Hopfield models for optimisation/constraint satisfaction

Figure 42.10. Hopfield network for solving a travelling salesman problem with \( K = 4 \) cities. (a1,2) Two solution states of the 16-neuron network, with activities represented by black = 1, white = 0; and the tours corresponding to these network states. (b) The negative weights between node B2 and other nodes; these weights enforce validity of a tour. (c) The negative weights that embody the distance objective function.

[from MacKay (2003)]
Theories of computation

All of this will lead to theories [of computation] which are much less rigidly of an all-or-none nature than past and present formal logic. They will be of a much less combinatorial, and much more analytical, character. In fact, there are numerous indications to make us believe that this new system of formal logic will move closer to another discipline which has been little linked in the past with logic. This is thermodynamics, primarily in the form it was received from Boltzmann, and is that part of theoretical physics which comes nearest in some of its aspects to manipulating and measuring information.

(John Von Neumann, *Collected Works* Vol. 5, p. 304)

[from Ackley et al. (1985)]
The Boltzmann machine

A stochastic Hopfield model:

\[ H = - \sum_{i=1}^{N} \theta_i x_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} w_{ij} x_i x_j \]

\[ P(\{x_i\}) = \frac{1}{Z} e^{-H} \]

\[ Z = \sum_{m=1}^{M} \exp \left( \sum_{i} \theta_i x_i + \frac{1}{2} \sum_{i \neq j} w_{ij} x_i x_j \right) \]

with

\[ x_i = \{0, 1\} / \{-1, 1\} \]

\[ w_{ij} = w_{ji} \]

\[ w_{ii} = 0 \]

This is equivalent to the Ising model.
Activity rule

\[
P(x_i = 1|h_i) = \frac{1}{1 + e^{-h_i}}
\]
\[
P(x_i = -1|h_i) = 1 - P(x_i = 1|h_i)
\]

\[
h_i = \sum_{n=1}^{N} w_{ij}x_i
\]

This can be used to minimise a cost function, and implements Gibbs sampling for \( P(\{x_i\}) \).
Fully visible BM

Each unit \( x_i \) is to represent a part of the data:

\[
\frac{\partial \log P(\{x_i\})}{\partial \theta_i} = \eta (\langle x_i \rangle_{data} - \langle x_i \rangle_{model})
\]

\[
\frac{\partial \log P(\{x_i\})}{\partial w_{ij}} = \eta (\langle x_i x_j \rangle_{data} - \langle x_i x_j \rangle_{model})
\]

Interpretation:

- ’waking’ (real world) - weight increase
- ’sleeping’ (model dreams generatively) - weight decrease
- when both match, weights do not change
Adding hidden units

The FVBM can only exactly account for pairwise correlations. This is often insufficient.

![Diagram](image)

**Figure 43.1.** The ‘shifter’ ensembles. (a) Four samples from the plain shifter ensemble. (b) Four corresponding samples from the labelled shifter ensemble.

[from MacKay (2003)]
Adding hidden units

The FVBM can only exactly account for pairwise correlations. This is often insufficient.

Proposed solutions:

- explicitly introduce higher orders (complicated, see above)
- introduce latent variables as hidden units $h_i$

$$H = - \sum_{i=1}^{N} \theta_i y_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} w_{ij} y_i y_j$$

$$y = (\{x_i\}, \{h_i\})$$
Learning rule

Compare equilibrium distributions when visible units are 'clamped' (wake) to data and when the network is 'free' (sleep).

The difference is minimised when:

\[
\Delta w_{ij} = \eta (p_{ij} - p'_{ij})
\]

\[
p_{ij} = p(y_i = 1, y_j = 1)_{stim}
\]

\[
p'_{ij} = p(y_i = 1, y_j = 1)_{model}
\]

For derivation, see Ackley et al. (1985).
Boltzmann trivia

BMs are universal approximators of discrete distributions.

Some possible extensions:

- Higher-order BMs, sampling gets harder.
- Conditional BMs: clamp hidden units during sleep can model conditional distributions.
- Mean field BMs: neurons have real-valued states, allow parallel updating but are limited in approximating distributions.
- Non-binary neurons, e.g. Potts model, follow the same principle as the energy is linear in parameters.

General problem: very slow convergence
The restricted Boltzmann machine

[figure by Dominik Spicher]
Energy and likelihood

\[ H = - \sum_{i=1}^{N_v} \theta_i^v v_i - \sum_{j=1}^{N_h} \theta_j^h h_j - \frac{1}{2} \sum_{i=1}^{N_v} \sum_{j=1}^{N_h} w_{ij} v_i h_j \]

Likelihood (sum runs over all \(2^{N_h}\) configurations of \(\{h\}_m\)):

\[ P(\{v_i\}) = \frac{1}{Z} \sum_m \exp(-H(\{v_i\}, \{h\}_m)) \]

\[ = \frac{1}{Z} \prod_i \exp(\theta_i^v x_i) \prod_j \left(1 + \exp(\theta_j^h + \sum_i v_i w_{ij})\right) \]

This leads to an interpretation as Products of Experts.

See also Hinton (2002).
Learning rule

- Boltzmann learning (still slow as sleep state has to be sampled)

\[
\Delta w_{ij} = \eta \left( \langle v_i h_j \rangle_{data} - \langle v_i h_j \rangle_{model} \right)
\]

- Contrastive Divergence:
  1. clamp \( v_i \) with a data vector, sample \( h_i \)
  2. \( \langle v_i h_j \rangle \) gives the positive phase
  3. repeat k times (fix \( h'_i \leftarrow h_i \), sample \( v'_i \))
Boltzmann machine learning rules

\[ \langle x_i x_j \rangle P(h | v^n) \quad - \quad \langle x_i x_j \rangle P(h, v) \]

[figure by David Reichert]
RBM RFs

[figure from Salakhutdinov (2009)]
Deep RBMs

[figure from David Reichert]
Deep Belief Network

(a) general BM
(b) restricted BM (RBM)
(c) deep BM (DBM)
(d) Deep Belief Net (DBN)

[figure from David Reichert, cf. Hinton and Salakhutdinov (2006)]
Auto-encoders

[figure from Salakhutdinov (2009)]
Auto-encoders

Reconstruction from auto-encoder (b) and PCA (c).

[figure from Salakhutdinov (2009)]
RBM as brain models?

- non-linear elements
- local learning
- unsupervised
- hierarchical structure
- distributed
- generative

But:

- binary neurons
- symmetric connections
- stationary distributions (ergodicity)
References


