

Characterizing cancer subtypes as attractors of Hopfield networks

Stefan R. Maetschke¹ and Mark A. Ragan^{1,2,*}¹The University of Queensland, Institute for Molecular Bioscience, Brisbane, QLD 4072, Australia and ²Australian Research Council Centre of Excellence in Bioinformatics, Australia

Associate Editor: Martin Bishop

ABSTRACT

Motivation: Cancer is a heterogeneous progressive disease caused by perturbations of the underlying gene regulatory network that can be described by dynamic models. These dynamics are commonly modeled as Boolean networks or as ordinary differential equations. Their inference from data is computationally challenging, and at least partial knowledge of the regulatory network and its kinetic parameters is usually required to construct predictive models.

Results: Here, we construct Hopfield networks from static gene-expression data and demonstrate that cancer subtypes can be characterized by different attractors of the Hopfield network. We evaluate the clustering performance of the network and find that it is comparable with traditional methods but offers additional advantages including a dynamic model of the energy landscape and a unification of clustering, feature selection and network inference. We visualize the Hopfield attractor landscape and propose a pruning method to generate sparse networks for feature selection and improved understanding of feature relationships.

Availability: Software and datasets are available at <http://acb.qfab.org/acb/hclust/>

Contact: m.ragan@uq.edu.au

Supplementary information: Supplementary data are available at *Bioinformatics* online.

Received on September 20, 2013; revised on December 4, 2013; accepted on December 30, 2013

1 INTRODUCTION

Cellular function and development are controlled by networks that regulate the expression of genes. The importance of precise regulation of gene expression through development and cellular differentiation is well known and perturbations of gene regulation are associated with many complex diseases including cancer (del Sol *et al.*, 2010; Pe'er and Hachohen, 2011).

At any point in time, the current state of a cell can be described by its expression profile (transcripts and their concentrations), and the set of all possible states defines the state space of the cell (Huang *et al.*, 2005). Most states are unstable, and the system will converge to a stable state of low energy—an attractor state—when perturbed (Huang, 2009). Waddington's epigenetic landscape of cell development (Waddington, 1940, 1957) and Kauffman's attractor networks (Kauffman, 1969a,b) are examples of state space models of gene regulatory networks in

which attractors represent stable low-energy states that emerge from the dynamics of the underlying network. In such models, attractor states correspond to normal developmental stages of the cell (Huang *et al.*, 2005) or disease states such as cancer (Huang *et al.*, 2009b); trajectories in state space describe developmental processes or cancer progression (Huang, 2009; Huang and Ingber, 2006), and the energy (elevation of the landscape) is related to the transition rates or transition probabilities between states (Bhattacharya *et al.*, 2011; Wang *et al.*, 2011).

The existence of attractor states and of state-space trajectories of cell differentiation has been experimentally verified (Chang *et al.*, 2008; Huang *et al.*, 2005; Huang *et al.*, 2009a) and different methods (Layek *et al.*, 2011b) have been used to model such attractor systems, with Boolean networks (BNs) and ordinary differential equations (ODEs) being the most common ones (Cheng *et al.*, 2012). Current methods are unable to describe the full complexity of a complete gene regulatory network, and applications are, therefore, focused on specific aspects of subsystems. For instance, recent cancer-related applications of BNs include cancer networks described by logic circuit stuck-at fault models (Layek *et al.*, 2011a; Lin and Khatri, 2012), models of signaling and DNA repair pathways (Esfahani *et al.*, 2011; Fumiã and Martins, 2013; Saez-Rodriguez *et al.*, 2011; Rodriguez *et al.*, 2012), models of cell adhesion for tumorigenic cells (Guebel *et al.*, 2012) and cancer networks modeling the G1/S transition (Yang *et al.*, 2013). ODEs have been used to model cancer cell populations (Gyori *et al.*, 1988; Kansal *et al.*, 2000), immune response (de Pillis *et al.*, 2005; Kolev, 2003; Lucia and Maino, 2002) and therapy (d'Onofrio *et al.*, 2009; Isaeva and Osipov, 2009). Cheng *et al.* (2012) provide a comprehensive overview.

BN- and ODE-based models are typically small (a few genes), manually crafted and require intricate explicit knowledge of the regulatory subsystem and its molecular kinetics. There have been attempts to construct such networks from data, but this is difficult due to the complexity of the networks and the lack of suitable time-series data (Esfahani *et al.*, 2011). Here, we propose Hopfield networks as a novel approach to efficiently construct large attractor networks from static gene-expression data—neither pathway data nor kinetic information is required. It has been argued that cancer is an intrinsic attractor state of the molecular-cellular network (Ao *et al.*, 2008; Huang, 2011), and here we demonstrate that attractors of Hopfield networks can correspond to cancer subtypes. Specifically, we construct Hopfield networks for numerous cancer datasets, cluster the samples according to their attractor association and measure

*To whom correspondence should be addressed.

how accurately the clusters reflect the true subtypes of the dataset. In addition, we use Hopfield networks to perform feature selection, network pruning and network inference and present a novel method to visualize the attractor landscape of the network.

It is important to appreciate that two levels of description are in play: the real physical network of transcription factors and other biomolecules that, by regulating gene expression, generates and maintains the cell state; and the Hopfield network, the structure of which we infer from observed data (here, sets of gene-expression patterns) indicative of this cell state. Each level uses concepts of states, trajectories and attractors. Here, we explore mappings between these concepts, e.g. between cancer subtypes as attractors in a Waddington landscape and attractor states in a Hopfield model.

2 METHODS AND RESULTS

In the following sections, we first describe the data used, then introduce Hopfield networks and describe their application to the clustering of cancer subtype data. We then present novel approaches to prune networks and to visualize their high-dimensional energy functions. Finally, we perform an extensive comparison of Hopfield networks with other common methods for the clustering of cancer data.

2.1 Data

We used two data suites to study Hopfield network and their performance. First, we examine a suite containing 12 cancer microarray datasets with clinically defined subtypes created by Wang *et al.* (2012) with the purpose of comparing different feature selection methods.

The second is a suite containing 35 cancer gene-expression datasets generated by de Souto *et al.* (2008) for the evaluation of clustering methods. The datasets in this suite have been preprocessed: 10% of the largest and smallest values are discarded, and only genes showing a sufficiently high fold-change have been selected [see de Souto *et al.* (2008) for details]. Missing data were detected in dataset *alizadeh-2000-v3*, and after consultation with the authors the two rows concerned were replaced by the corresponding rows from dataset *alizadeh-2000-v2*.

For all datasets, we performed sample-wise Z-score normalization, and in the case of single-channel arrays a gene-wise \log_2 transformation was applied before normalization.

The example data used in the Sections 2.2–2.5 are a reduced version of the *yeoh-2002-v1* dataset from de Souto's suite. Originally created by Yeoh *et al.* (2002), it contains gene-expression data of acute lymphoblastic leukemia (ALL), with 43 samples labeled as subtype T-ALL and 205 samples labeled as subtype B-ALL. We took the 43 samples of class T-ALL and the first 43 samples from class B-ALL in the order they appeared in the original dataset, and then extracted the 50 most-variable genes (largest variance) to construct a smaller balanced dataset with 50 genes and 86 samples. The comparison of clustering performance in Section 2.6 was performed on the Wang and the de Souto suites described earlier in the text.

2.2 Hopfield network

A Hopfield network is a graph composed of nodes $i \in 1, \dots, n$ and weighted but undirected edges w_{ij} between nodes i and j . In the original Hopfield network, nodes have binary states $s_i \in \{0, +1\}$. Here, we use a slight modification with ternary states $s_i \in \{-1, 0, +1\}$, with the only purpose to simplify the pruning of the network described later. The dynamic of the network is defined by the following recursive formula

$$s_i^{(t+1)} = \text{sgn}\left(\sum_j w_{ij}s_j^{(t)}\right), \quad (1)$$

where $s_i^{(t)}$ is the state of the i -th node at time step t and $\text{sgn}(\cdot)$ is the signum function

$$\text{sgn}(x) = \begin{cases} +1 & \text{for } x > 0 \\ 0 & \text{for } x = 0 \\ -1 & \text{for } x < 0. \end{cases} \quad (2)$$

Given any initial state $\mathbf{s}^0 = (s_1^0, \dots, s_n^0)^T$ in state space \mathfrak{S} , for $t \rightarrow \infty$ the network states $\mathbf{s}^{(t)}$ will converge to an attractor state with monotonically decreasing energy

$$E(\mathbf{s}) = -\frac{1}{2} \sum_{i,j} s_i w_{ij} s_j = -\frac{1}{2} \mathbf{s}^T \mathbf{W} \mathbf{s}. \quad (3)$$

Attractors are local minima of the energy function $E(\cdot)$, and the system can be interpreted as relaxing from a higher energy state to a minimum energy state. States or patterns $\mathbf{p}_k \in \mathfrak{S}$, $k \in \{1, \dots, m\}$ can be established (stored) as attractors of the network using Hebbian learning

$$\mathbf{W} = \frac{1}{m} \sum_k \mathbf{p}_k \mathbf{p}_k^T - \mathbf{I}, \quad (4)$$

where network weights are computed as the sum over the outer products of the pattern vectors. The resulting weight matrix \mathbf{W} is a normalized covariance matrix of the patterns \mathbf{p}_k but with a zero diagonal. Initializing the network state \mathbf{s}^0 to some input pattern and relaxing the network using Equation (1) allows the recall of stored patterns even when the input patterns are distorted or noisy.

2.3 Clustering

Here, we use Hopfield networks to characterize different cancer subtypes as network attractors. Let $\mathbf{P} = (\mathbf{p}_1, \dots, \mathbf{p}_m)$ be a matrix of patterns, constructed by discretizing the data \mathbf{D} of a gene-expression matrix $\mathbf{P} = \text{sgn}(\mathbf{D})$, where columns \mathbf{p}_k contain samples from different cancer subtypes and rows are genes.

We firstly construct a network \mathbf{W} from \mathbf{P} using Hebbian learning [see (4)], and then recall each sample until it reaches its attractor, using Equation (1). In matrix notation a recall step (relaxation) for all patterns in \mathbf{P} is simply the product of the pattern matrix and the weight matrix \mathbf{W} followed by discretization

$$\mathbf{S}_{(t+1)} = \text{sgn}(\mathbf{S}_{(t)} \mathbf{W}), \quad (5)$$

with $\mathbf{S}_{(0)} = \mathbf{P}$. Each column of the state matrix $\mathbf{S}_{(t)}$ describes one of the \mathbf{p}_k patterns after the t -th recall step.

We do not label patterns by cancer subtype class but instead leave it to the learning algorithm to construct attractors that capture the similarities between patterns. Specifically, we hope that the algorithm creates attractors that group samples according to subtype, i.e. essentially functions as a clustering method. During recall, input patterns would then converge to one of the attractors, where the state vector of the attractor represents the label and molecular signature of a specific subtype.

Figure 1(a) displays the expression matrix \mathbf{D} for ALL data and state matrices $\mathbf{S}_{(0)} \dots \mathbf{S}_{(7)}$ over eight iterations of recall (matrices are transposed to preserve space). \mathbf{D}^T has 50 columns (genes) and 86 samples (rows), with the top half containing samples of subtype B-ALL and the lower half of subtype T-ALL. In the last iteration, the state matrix is stable, indicating that all samples have converged to their respective attractors. Inspection of $\mathbf{S}_{(7)}$ reveals only two different states (attractors) and only one sample that has converged to the wrong attractor (marked by a triangle).

Figure 2(a) shows the corresponding weight matrix of the network. Large weights (dark blue or red color) tend to occur in the upper left part of the weight matrix. This is because the genes in the ALL dataset were selected and ordered according to their variance (see Section 2.1).

To summarize, we constructed a Hopfield network from static gene-expression data and let samples converge to the network attractors.

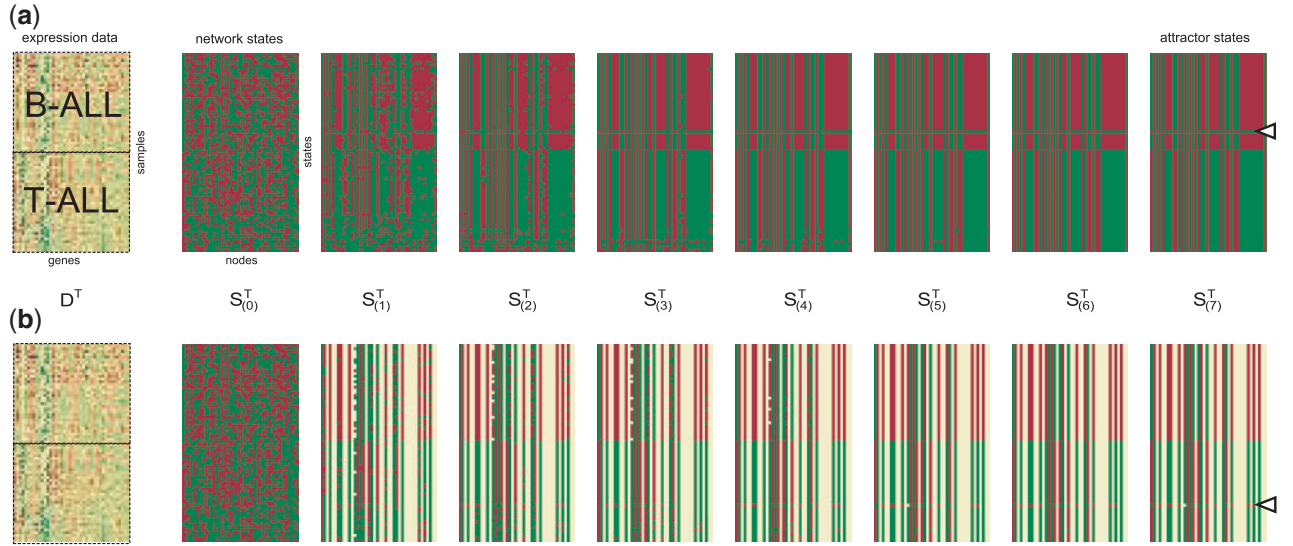


Fig. 1. Relaxation of all samples over eight iterations to their attractor states for the (a) unpruned and (b) the pruned weight matrix. Triangles in the state matrices of the last iteration mark samples that converged to the wrong attractor (misclassified)

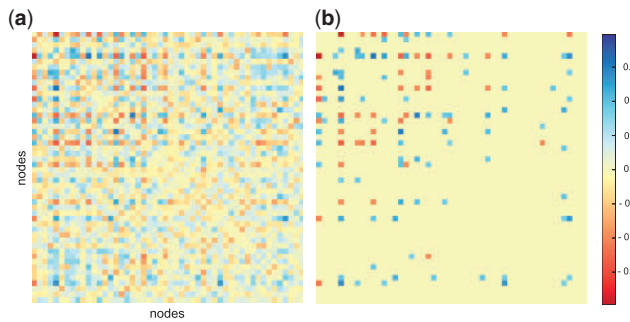


Fig. 2. Weight matrix of the Hopfield network for the ALL data (a) before and (b) after pruning. Blue color indicates positive weights (coexpression of genes), red color indicates negative weights (anti-coexpression) and yellow color indicates zero weights (no coexpression)

We use the network as a clustering algorithm in which clusters are identified by attractor states that represent molecular signatures and class labels of subtypes. We do not use labeled samples or specify the number of clusters during the learning phase.

2.4 Pruning

The weight matrices created by Hebbian learning are typically dense [see Fig. 2(a)], in contrast to gene regulatory networks that tend to be sparse (Huang *et al.*, 2005). Although the learning algorithm does not aim to provide an accurate reconstruction of the regulatory network underlying the expression data, a sparse network is nevertheless preferable, as memory consumption is lower, recall is more efficient, and most importantly, relationships between genes are more easily understood.

In the following, we introduce a method to prune the network without destroying its capability to classify samples. To this purpose, we set all weights w_{ij} with an absolute value smaller than a threshold τ to zero

$$\mathbf{W}(\tau) = \begin{cases} w_{ij} & \text{for } |w_{ij}| \geq \tau \\ 0 & \text{for } |w_{ij}| < \tau, \end{cases} \quad (6)$$

and determine the largest threshold τ_{max} that does not reduce the estimated clustering accuracy $ERI(\tau)$ of the network by more than a given factor α

$$\tau_{max} = \max_{s.t. ERI(\tau) > 1-\alpha} \tau. \quad (7)$$

Let C_{true} be the true sample labels (e.g. subtypes), C_0 the sample labels predicted by the network for the unpruned matrix $\mathbf{W}(0)$ and C_τ the labels predicted for the pruned matrix $\mathbf{W}(\tau)$. The true clustering accuracy $TRI(\tau)$ of the network for a specific threshold τ can be determined by comparing the true labels C_{true} with the predicted labels C_τ . Following de Souto *et al.* (2008), we also use the Adjusted Rand Index $ARI(\cdot, \cdot)$ by Hubert and Arabie (1985) (see Supplementary Material) to measure the clustering accuracy.

$$TRI(\tau) = ARI(C_{true}, C_\tau) \quad (8)$$

The ARI is 1 for a perfect prediction and negative close to zero for predictions that are not better than chance. The computation of this True Rand Index $TRI(\tau)$ requires, however, the true labels C_{true} , which are unknown in practice. Instead we compute an Estimated Rand Index

$$ERI(\tau) = ARI(C_0, C_\tau) \quad (9)$$

by taking the accuracy of the unpruned network at $\tau = 0$ as a baseline and assuming C_0 to be the true labeling. As a consequence $ERI(0)$ is always 1 and the ERI will follow the TRI closely, provided $TRI(0)$ is close to 1.

Figure 3 plots the TRI , the Estimated Rand Index ERI and the density of the weight matrix for different pruning thresholds τ . The plots for TRI and ERI are almost identical, and for an allowed 20% decrease ($\alpha = 0.2$) of the ERI the maximum pruning threshold τ_{max} becomes 0.34. For this threshold, the density of the weight matrix is reduced from 92% to 5%, where density is the percentage of non-zero entries in the weight matrix. Figure 2 displays the weight matrix (a) before and (b) after pruning.

This pruning method ensures that the overall clustering accuracy of the network is largely maintained. However, what effect does pruning have on the attractors of the network and on the labeling of samples? Figure 1 shows recall and the attractors for the (a) unpruned and (b) pruned network. The expression matrix \mathbf{D} and the state matrix $\mathbf{S}_{(0)}$, which is simply $\mathbf{S}_{(0)} = \text{sgn}(\mathbf{D})$, are identical in both cases. Recall and attractors, however, are different. Because the pruned network contains nodes (genes) that

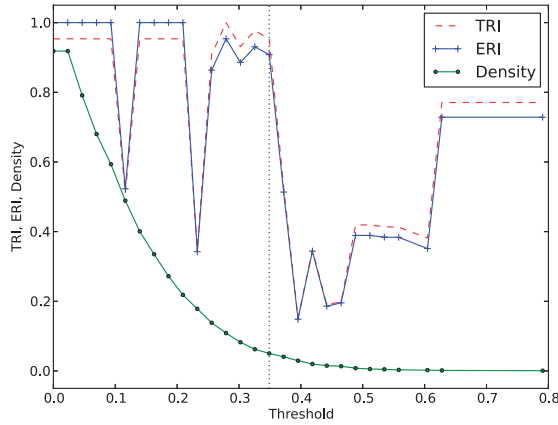


Fig. 3. True Rand Index (*TRI*), Estimated Rand Index (*ERI*) and density of the weight matrix for different pruning thresholds τ . Best threshold τ_{max} marked by dashed line

have lost all edges (interactions), their states become zero (yellow) and remain zero [see Equation (1)]. Consequently, the attractor states in $S_{(7)}$ differ only for genes that interact (via their gene products) with other genes, and the weight matrix [Fig. 2(b)] describes the strengths and signs of these interaction. Only one sample is misclassified (Fig. 1, marked by a triangle) for the pruned and the unpruned networks.

To summarize, we prune the network by removing small weights as long as the estimated classification accuracy decreases by no more than a given percentage (e.g. $\alpha = 0.2$). Pruning not only results in a sparse network but also functions as a feature selection method. Genes without interactions correspond to nodes fixed to zero state that do not affect attractors, and the molecular signatures that attractor states represent can be compressed by removing those genes. The pruned network furthermore provides insight into the relationships of the genes that make up the molecular signatures. Thus, Hopfield networks provide a framework to integrate clustering, feature selection and network inference.

2.5 Visualization

The energy function [see Equation (3)] of an attractor network is typically high-dimensional and cannot be visualized directly. Current pictures of energy landscapes are, therefore, either artistic renderings not based on actual data or are limited to systems with only two genes. Here, we propose a simple method to visualize the energy landscape of networks with arbitrary numbers of genes.

Let $D_{n \times m}$ be a matrix with expression data, containing n samples in rows and m genes in columns. First, we reduce the dimensionality of the expression matrix to two dimensions using principal component analysis

$$D_{n \times 2} = D_{n \times m} T_{m \times 2}, \quad (10)$$

where the transformation matrix T is composed of the eigenvectors of $D_{n \times m}^T D_{n \times m}$, and $T_{m \times 2}$ contains the two eigenvectors of T with the largest eigenvalues. After transformation, $D_{n \times 2}$ contains the first and second principal components of $D_{n \times m}$.

Next, we construct a regular 2D grid $G_{k \times 2}$ with k 2D points in the same range as the points in $D_{n \times 2}$ and perform an inverse principal component analysis to map the grid points to the high-dimensional space

$$G_{k \times m} = G_{k \times 2} T^{-1}, \quad (11)$$

where T^{-1} is the inversion of the transformation matrix T computed in the first step.

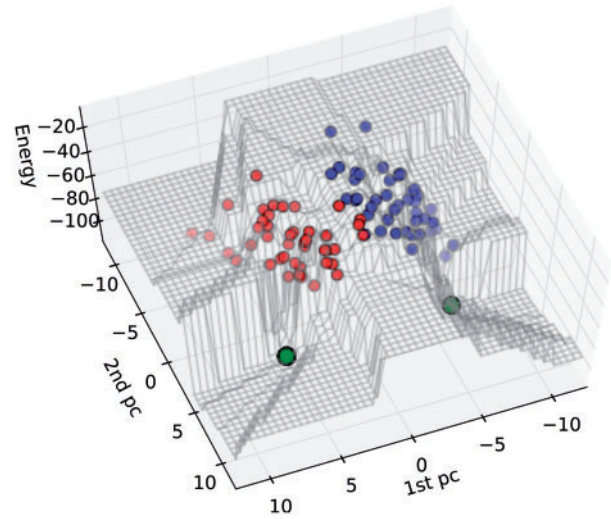


Fig. 4. Energy landscape of the leukemia network. The x- and y-axes show the first and second principal components of dimensionality reduced data. Red points indicate samples of the T-ALL subtype, blue points indicate B-ALL samples and attractor states are in green

In the third step, we compute the energies $E_g = E(G_{k \times m})$ for the grid data and $E_d = E(D_{n \times m})$ for the sample data in the high-dimensional space, with

$$E(S) = -\frac{1}{2} S^T W S. \quad (12)$$

By interpolating between the energy values E_g over the grid data $G_{k \times 2}$, we can render the surface of the energy landscape in three dimensions. Similarly, the expression data $D_{n \times 2}$ and their energies E_d can be plotted. Figure 4 depicts the energy surface (grid) of the Hopfield network, the samples of the leukemia data (red and blue dots) and the two attractor states (green) of the network.

Several properties of the landscape are of interest. First, the attractors are minima of the energy function, and their basins of attraction are clearly visible. Second, in contrast to clustering methods such as k-means, attractors are not the centers of the point clouds defined by the two subtypes. Instead, samples are divided by a ridge separating the subtypes. Third, the landscape is shaped like a staircase due to the discrete state function $sgn(\cdot)$ of the network nodes.

The Supplementary Material contains additional renderings of the energy surface including a plot displaying the trajectories of samples converging to their attractors. The visualization method is not limited to Hopfield networks but can be applied whenever there is an energy function and a dataset with points of interest.

2.6 Clustering comparison

In Section 2, we applied Hopfield networks to clustering, feature selection and network inference on a small example dataset. Here, we focus on the clustering aspect and study the performance of Hopfield networks in comparison with a selection of other clustering algorithms on a larger suite of datasets.

Hopfield networks classify samples without specifying the number of clusters or any other parameter. Therefore, we limit our comparison with clustering algorithms that also detect the number of clusters automatically such as DBSCAN (Ester *et al.*, 1996), affinity propagation (Frey and Dueck, 2007), mean shift (Barash and Comaniciu, 2004), k-means (MacQueen, 1967) and Ward's hierarchical clustering (Ward, 1963). For the latter two, the silhouette coefficient (Rousseeuw, 1987) is

computed to determine the number of clusters. All methods were used with their default settings as implemented in the *scikit-learn* Python library, with the exception of DBSCAN, for which the default setting $\epsilon = 0.0$ resulted in a poor performance, and we chose $\epsilon = 1.0$ instead.

To avoid bias due to a specific feature selection method, we firstly measure the clustering accuracy (ARI) of the various algorithms for increasing numbers of genes (features), where genes are sorted according to their variance (most-variable first). Figure 5 plots the ARI averaged over all datasets in Wang's suite for 10–2000 genes. Most clustering algorithms reach their peak accuracy between 200 and 1000 features and stabilize thereafter. Adding further features does not improve or decrease the ARI substantially, with the notable exception of the HOPFIELD method, which shows a strong loss in accuracy with increasing numbers of genes (black curve). Apparently, the Hopfield learning algorithm is sensitive to the growing noise and diminishing sample differences for sample vectors of increasing dimensionality.

The poor performance of the Hopfield clustering for larger number of features can be avoided by using a similarity matrix instead of the sample matrix \mathbf{S} as input. Let $\mathbf{D} = [d(s_i, s_j)]_{ij}$ be a matrix containing the Euclidean distances $d(\cdot, \cdot)$ between sample vectors s_i and s_j , and $\mathbf{M} = \mathbf{I} - \mathbf{D}/\max(\mathbf{D})$ the corresponding similarity matrix. Describing samples not by their feature vectors (gene-expression values) but by the similarities between feature vectors is a common transformation of input data. For instance, affinity propagation, mean shift and DBSCAN all require the similarity matrix as input. We mark all methods using the similarity matrix with a star (*).

Figure 5 shows that HOPFIELD* performs consistently better than HOPFIELD and does not suffer from the decrease in accuracy for growing numbers of features. The accuracy of HOPFIELD* is comparable with the best-performing methods KMEANS+SIL, WARD+SIL and DBSCAN*, whereas AFFPROP* and MEANSHIF* achieve considerably lower accuracies over the entire range of feature numbers.

In Section 2.4, we introduced pruning for Hopfield networks; here we are interested in its impact on the clustering accuracy. In contrast to the previous evaluation of accuracy for different numbers of features, in the following we select the number of features that explains most of the variance in the data. Specifically, we rank genes in Wang's datasets according to variance, plot variance versus the number of genes and find the elbow of the plot (point closest to origin). This common method to select features does not require labeled samples (in contrast to fold-change, for instance). Feature selection for de Souto's datasets was performed by de Souto *et al.* (2008) (see Section 2.1), and we use the original data.

Table 1 lists the prediction accuracies (ARI) averaged over Wang's and de Souto's data suites for the different clustering algorithms including Hopfield clustering with (HOPFIELD+P*, HOPFIELD+P) and without pruning (HOPFIELD*, HOPFIELD). As the data show, pruning usually improves prediction accuracy of Hopfield networks slightly and leads to sparser networks (95 \rightarrow 35% density for $\alpha = 0.2$) but has the disadvantage of increasing the computation time.

The feature selection method described earlier in the text tends to result in rather large feature sets (> 2000 genes) for Wang's datasets, and the accuracy of the HOPFIELD method is therefore poor (see Fig. 5). Using the similarity matrix as input greatly improves Hopfield clustering (compare HOPFIELD with HOPFIELD*) but is not beneficial to k-means and Ward's hierarchical clustering in most cases. The best-performing methods are WARD+SIL and KMEANS+SIL. DBSCAN* is the fastest method and highly accurate on Wang's data but shows low accuracy on de Souto's data. HOPFIELD+P* is the fourth and fifth ranking method on Wang's and de Souto's datasets, respectively. Although Hopfield clustering is not the best-performing method, it is the simplest, with average-to-good accuracy and speed.

Overall prediction accuracies are fairly low, but it has to be taken into account that most datasets are composed of many subtypes and that

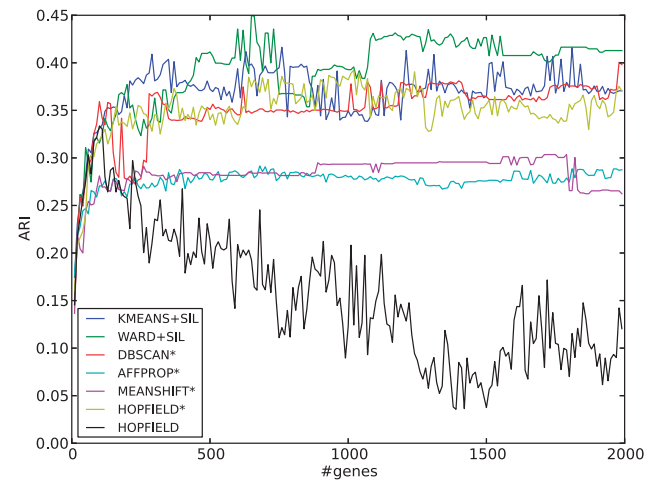


Fig. 5. Adjusted Rand Index (ARI) of different clustering algorithms averaged over Wang's datasets for increasing numbers of genes (features). Genes were ranked and selected by variance. Methods marked with asterisk operate on the similarity matrix instead of the sample matrix

Table 1. Comparison of clustering methods

Method	Wang ARI \pm std	Time	de Souto ARI \pm std	Time
KMEANS+SIL	0.373 \pm 0.29	6.65	0.305 \pm 0.29	20.40
WARD+SIL	0.417 \pm 0.31	3.52	0.254 \pm 0.24	5.96
KMEANS+SIL*	0.340 \pm 0.25	5.50	0.300 \pm 0.29	11.12
WARD+SIL*	0.336 \pm 0.24	2.31	0.271 \pm 0.29	4.14
DBSCAN*	0.374 \pm 0.32	0.33	0.142 \pm 0.20	0.39
AFFPROP*	0.283 \pm 0.19	0.51	0.165 \pm 0.14	1.66
MEANSHIF*	0.260 \pm 0.28	1.49	0.075 \pm 0.21	6.00
HOPFIELD*	0.356 \pm 0.24	1.41	0.242 \pm 0.22	3.36
HOPFIELD+P*	0.366 \pm 0.25	5.65	0.249 \pm 0.22	13.75
HOPFIELD	0.069 \pm 0.19	7.65	0.129 \pm 0.18	17.96
HOPFIELD+P	0.073 \pm 0.19	7.79	0.128 \pm 0.17	18.54

Note: Adjusted Rand Index (ARI) with standard deviation (std) and computation time (wall time) in seconds for different clustering algorithms averaged over datasets in Ward's and de Souto's suite. Methods marked with star (*) use the similarity matrix instead of the sample matrix as input.

accuracy varies greatly between datasets and clustering algorithms (see Supplementary Material). Which algorithm performs well depends strongly on the dataset. We recommend, therefore, that a variety of methods should be evaluated for any given dataset.

3 DISCUSSION

3.1 Hopfield networks as cancer attractor model

Following the hypotheses that cancers are attractors within gene regulatory networks (Ao *et al.*, 2008; Huang, 2011), we constructed Hopfield networks from cancer expression data and demonstrate that network attractors correspond to cancer subtypes. Hopfield networks are simple models, and because they are inferred from static data, they cannot be expected to model

the topology or the dynamics of the real regulatory network with great accuracy. Inference of networks from data is ill-posed in general, and different networks can generate the same dynamics (Hickman and Hodgman, 2009). Despite these limitations, Hopfield networks can capture important properties of real physical networks. As we have shown, they can address cancer subtype characterization in the framework of state-space attractors. As coexpression networks themselves, they can provide valuable information about relationships among genes and enable the selection of important genes.

For other attractor network models, the energy function is related to the transcription rates or probabilities between states (Bhattacharya *et al.*, 2011; Wang *et al.*, 2011). The Hopfield model is derived from the Ising model (Ising, 1925) in which energy is correlated with the probability of a state. However, Ising models are not constructed by Hebbian learning, nor are standard Hopfield networks probabilistic. Therefore, it is not straightforward to link Hopfield energies with cell-state probabilities.

Finally, the inner working of Hopfield models and of real regulatory models is fundamentally different, and thus their overall dynamics should be expected to differ. Exploration of this potential mapping is not made easier by the paucity of data about network states underlying real cell-state attractors. Nevertheless, as shown here, Hopfield networks can be used to derive molecular signatures for cancer subtypes from their Hopfield attractors.

3.2 Alternative learning algorithms

We used Hebbian learning to construct the weight matrix of the Hopfield network. For networks with sufficiently many edges, and for a broad range of topologies, Hebbian learning creates stable attractors (Bar-Yam and Epstein, 2004). For randomly chosen patterns, the maximum asymptotic value $m_c(n)$ to recover all m patterns exactly is $m_c(n) = n/(4 \log n)$ (McEliece *et al.*, 1987), although for correlated patterns the storage capacity of the network is much lower (Storkey and Valabregue, 1999). Although for datasets with several hundred genes the capacity of the network is typically sufficient to store the molecular signatures of a few subtypes, other learning algorithms potentially are more-accurate classifiers.

We evaluated high-capacity learning rules by Kanter and Sompolinsky (1987) and Storkey and Valabregue (1999) that allow storage of n linearly independent patterns. However, they have $O(n^4)$ complexity, resulting in prohibitively long learning times, whereas Hebbian learning is of quadratic complexity. Furthermore, the clustering accuracy was considerably lower than that of Hebbian learning (data not shown) because attractors for individual samples were generated instead of grouping similar patterns by assigning a single attractor.

We use Hopfield networks in a novel way by not storing a single pattern (e.g. a molecular signature) for each cancer subtype, but instead leave it to the learning algorithm to discover subtypes and establish network attractors from the complete set of unlabeled samples.

3.3 Network pruning

The networks generated by Hebbian learning are usually dense, with >95% of the weights non-zero. Gene regulatory networks,

on the other hand, tend to be sparse and scale-free (Huang *et al.*, 2005). Apart from the *ERI*, we explored other measures to determine good pruning thresholds such as the silhouette coefficient (Rousseeuw, 1987) or maintaining a fixed number of clusters but found the *ERI* to work best (results not shown).

We have shown that the accuracy of Hopfield clustering can be improved if the similarity matrix is used as input instead of the sample matrix. For the common case of gene-expression datasets with a large number of genes and a small number of samples, the similarity matrix is much smaller than the sample matrix, resulting in lower memory consumption and computation time. However, pruning of a network constructed from similarity matrices does not filter unimportant genes but samples.

4 CONCLUSION

We introduced Hopfield networks as a state-space model in which attractors characterize cell states, and used the model to identify cancer subtypes in gene-expression data. Compared with other clustering algorithms, the Hopfield network is not necessarily the most accurate. However, it has other important advantages, including the unification of clustering, feature selection and network inference, and as a modeling framework for epigenetic landscapes. Furthermore, Hopfield networks are simple to implement and do not require any meta-parameters when used for clustering.

We presented methods to visualize the high-dimensional energy landscape of the model and to prune the network to derive sparse networks that are more open to a biological interpretation than traditional clustering or feature-selection methods. Furthermore, Hopfield networks can be efficiently simulated on quantum computers; recent improvements (www.dwavesys.com) now enable the implementation of larger networks (512 qubit) with potentially dramatic speedups.

Many interesting questions remain, including the power of network pruning as a feature-selection method, the accuracy of inferred regulatory networks, whether and how known regulatory interactions can be incorporated, whether the energy function can reflect the probability or frequency of transitions among actual cell states, and whether developmental trajectories can be mapped to trajectories in this landscape (Waddington, 1940, 1957). The standard Hopfield network is likely too simple a model for many applications; directed stochastic Hopfield-like networks with more advanced learning algorithms will likely be required.

Attractor models of regulatory networks offer great potential. Genes are not treated independently but in combination, and trajectories of normal cell development or tumor progression can be predicted. If an accurate model can be constructed, time points, amounts and combinations of treatments can be identified that most effectively influence cell fates with minimum intervention.

ACKNOWLEDGEMENT

The authors thank Josha Inglis, Melissa Davis and Sriganesh Srihari for helpful discussions and suggestions.

Funding: Australian Research Council (DP110103384 and CE034822).

Conflict of Interest: none declared.

REFERENCES

- Ao, P. *et al.* (2008) Cancer as robust intrinsic state of endogenous molecular-cellular network shaped by evolution. *Med. Hypotheses*, **70**, 678–684.
- Bar-Yam, Y. and Epstein, I. (2004) Response of complex networks to stimuli. *Proc. Natl Acad. Sci. USA*, **101**, 4341–4345.
- Barash, D. and Comaniciu, D. (2004) A common framework for nonlinear diffusion, adaptive smoothing, bilateral filtering and mean shift. *Image Vis. Comput.*, **22**, 73–81.
- Bhattacharya, S. *et al.* (2011) A deterministic map of Waddington's epigenetic landscape for cell fate specification. *BMC Syst. Biol.*, **5**, 85.
- Chang, H. *et al.* (2008) Transcriptome-wide noise controls lineage choice in mammalian progenitor cells. *Nature*, **453**, 544–547.
- Cheng, T. *et al.* (2012) Understanding cancer mechanisms through network dynamics. *Brief. Funct. Genomics*, **11**, 543–560.
- de Pillis, L. *et al.* (2005) A validated mathematical model of cell-mediated immune response to tumor growth. *Cancer Res.*, **65**, 7950–7958.
- de Souto, M. *et al.* (2008) Clustering cancer gene expression data: a comparative study. *BMC Bioinformatics*, **9**, 497.
- del Sol, A. *et al.* (2010) Diseases as network perturbations. *Curr. Opin. Biotechnol.*, **21**, 566–571.
- d'Onofrio, A. *et al.* (2009) On optimal delivery of combination therapy for tumors. *Math. Biosci.*, **222**, 13–26.
- Esfahani, M. *et al.* (2011) Probabilistic reconstruction of the tumor progression process in gene regulatory networks in the presence of uncertainty. *BMC Bioinformatics*, **12** (Suppl. 10), S9.
- Ester, M. I. (1996) A density-based algorithm for discovering clusters in large spatial databases with noise. In: *Second International Conference on Knowledge Discovery and Data Mining*. Vol. 96, AAAI Press, Menlo Park, California, pp. 226–231.
- Frey, B. and Dueck, D. (2007) Clustering by passing messages between data points. *Science*, **315**, 972–976.
- Fumiã, H. and Martins, M. (2013) Boolean network model for cancer pathways: predicting carcinogenesis and targeted therapy outcomes. *PLoS One*, **8**, e69008.
- Guebel, D. *et al.* (2012) Analysis of cell adhesion during early stages of colon cancer based on an extended multi-valued logic approach. *Mol. Biosyst.*, **8**, 1230–1242.
- Gyori, I. *et al.* (1988) Time-dependent subpopulation induction in heterogeneous tumors. *Bull. Math. Biol.*, **50**, 681–696.
- Hickman, G. and Hodgman, T. (2009) Inference of gene regulatory networks using Boolean-network inference methods. *J. Bioinform. Comput. Biol.*, **7**, 1013–1029.
- Huang, S. (2009) Reprogramming cell fates: reconciling rarity with robustness. *BioEssays*, **31**, 546–560.
- Huang, S. (2011) On the intrinsic inevitability of cancer: from foetal to fatal attraction. *Semin. Cancer Biol.*, **21**, 183–199.
- Huang, S. and Ingber, D. (2006) A non-genetic basis for cancer progression and metastasis: self-organizing attractors in cell regulatory networks. *Breast Dis.*, **26**, 27–54.
- Huang, S. *et al.* (2005) Cell fates as high-dimensional attractor states of a complex gene regulatory network. *Phys. Rev. Lett.*, **94**, 128701.
- Huang, A. *et al.* (2009a) Using cell fate attractors to uncover transcriptional regulation of HL60 neutrophil differentiation. *BMC Syst. Biol.*, **3**, 20.
- Huang, S. *et al.* (2009b) Cancer attractors: a systems view of tumors from a gene network dynamics and developmental perspective. *Semin. Cell Dev. Biol.*, **20**, 869–876.
- Hubert, L. and Arabie, P. (1985) Comparing partitions. *J. Classif.*, **2**, 193–218.
- Isaeva, O. and Osipov, V. (2009) Different strategies for cancer treatment: mathematical modelling. *Comput. Math. Methods Med.*, **10**, 253–272.
- Ising, E. (1925) Beitrag zur Theorie des Ferromagnetismus. *Zeitschrift für Physik*, **31**, 253–258.
- Kansal, A. *et al.* (2000) Emergence of a subpopulation in a computational model of tumor growth. *J. Theor. Biol.*, **207**, 431–441.
- Kanter, I. and Sompolinsky, H. (1987) Associative recall of memory without errors. *Phys. Rev. A*, **35**, 380.
- Kauffman, S. (1969a) Homeostasis and differentiation in random genetic control networks. *Nature*, **224**, 177–178.
- Kauffman, S. (1969b) Metabolic stability and epigenesis in randomly constructed genetic nets. *J. Theor. Biol.*, **22**, 437–467.
- Kolev, M. (2003) Mathematical modelling of the competition between tumors and immune system considering the role of the antibodies. *Math. Comput. Model.*, **37**, 1143–1152.
- Layek, R. *et al.* (2011a) Cancer therapy design based on pathway logic. *Bioinformatics*, **27**, 548–555.
- Layek, R. *et al.* (2011b) From biological pathways to regulatory networks. *Mol. Biosyst.*, **7**, 843–851.
- Lin, P.-C. K. and Khatri, S. (2012) Application of Max-SAT-based ATPG to optimal cancer therapy design. *BMC Genomics*, **13** (Suppl. 6), S5.
- Lucia, U. and Maino, G. (2002) Thermodynamical analysis of the dynamics of tumor interaction with the host immune system. *Physica A*, **313**, 569–577.
- MacQueen, J. (1967) Some methods for classification and analysis of multivariate observations. In: Cam, L. M. L. and Neyman, J. (eds) *Proceeding of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*. Vol. 1, University of California Press, Berkeley, CA, pp. 281–297.
- McEliece, R. *et al.* (1987) The capacity of the hopfield associative memory. *IEEE Trans. Inf. Theory*, **33**, 461–482.
- Pe'er, D. and Hachohen, N. (2011) Principles and strategies for developing network models in cancer. *Cell*, **144**, 864–873.
- Rodriguez, A. *et al.* (2012) A Boolean network model of the FA/BRCA pathway. *Bioinformatics*, **28**, 858–866.
- Rousseeuw, P. (1987) Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *J. Comput. Appl. Math.*, **20**, 53–65.
- Saez-Rodriguez, J. *et al.* (2011) Comparing signaling networks between normal and transformed hepatocytes using discrete logical models. *Cancer Res.*, **71**, 5400–11.
- Storkey, A. and Valabregue, R. (1999) The basins of attraction of a new Hopfield learning rule. *Neural Netw.*, **12**, 869–876.
- Waddington, C. (1940) *Organisers and genes*. Cambridge University Press, Cambridge, United Kingdom.
- Waddington, C. (1957) *The strategy of the genes*. George Allan & Unwin, London.
- Wang, C. *et al.* (2012) mCOPA: analysis of heterogeneous features in cancer expression data. *J. Clin. Bioinforma.*, **2**, 22.
- Wang, J. *et al.* (2011) Quantifying the Waddington landscape and biological paths for development and differentiation. *Proc. Natl Acad. Sci. USA*, **108**, 8257–8262.
- Ward, J. (1963) Hierarchical grouping to optimize an objective function. *J. Am. Stat. Assoc.*, **58**, 236–244.
- Yang, L. *et al.* (2013) Robustness and backbone motif of a cancer network regulated by mir-17-92 cluster during the G1/S transition. *PLoS One*, **8**, e57009.
- Yeoh, E. *et al.* (2002) Classification, subtype discovery, and prediction of outcome in pediatric acute lymphoblastic leukemia by gene expression profiling. *Cancer Cell*, **1**, 133–143.