L1 regularization for reconstruction of a non-equilibrium Ising model

Hong Li Zeng1, John Hertz2,3 and Yasser Roudi2,4

1 Department of Applied Physics, Aalto University, FIN-00076 Aalto, Finland
2 Nordita, KTH Royal Institute of Technology and Stockholm University, Roslagstullsbacken 23, SE-106 91, Stockholm, Sweden
3 The Niels Bohr Institute, DK-2100 Copenhagen, Denmark
4 Kavli Institute for Systems Neuroscience, NTNU, NO-7030 Trondheim, Norway

E-mail: hong.zeng@aalto.fi, hertz@nbi.dk and yasserroudi@gmail.com

Received 20 November 2013, revised 6 June 2014
Accepted for publication 18 July 2014
Published 28 August 2014

Abstract
The couplings in a sparse asymmetric, asynchronous Ising network are reconstructed using an exact learning algorithm. L1 regularization is used to remove the spurious weak connections that would otherwise be found by simply maximizing the log likelihood of a finite data set. In order to see how L1 regularization works in detail, we perform the calculation in several ways including (1) by iterative minimization of a cost function equal to minus the log likelihood of the data plus an L1 penalty term, and (2) an approximate scheme based on a quadratic expansion of the cost function around its minimum. In these schemes, we track how connections are pruned as the strength of the L1 penalty is increased from zero to large values. The performance of the methods for various coupling strengths is quantified using receiver operating characteristic curves, showing that increasing the coupling strength improves reconstruction quality.

Keywords: sparse networks, nonequilibrium ising model, network reconstruction

(Some figures may appear in colour only in the online journal)

1. Introduction
A crucial step in understanding how a complex network operates is inferring its connectivity from observables in a systematic and controlled way. This learning of the connections from data is an inverse problem and recently, with the ongoing growth of available data, especially in biological systems, such inverse problems have attracted a lot of attention in the statistical physics community. Examples of applications include the reconstruction of a gene regulation network from gene expression levels [1] and identification of the protein-protein interactions from the correlations between amino acids [2]. One proxy for such a problem is the inverse Ising model, where the parameters of the model (fields and interactions) are inferred from observed spin history.

There has been a long history of inferring Gibbs equilibrium models, such as the equilibrium Ising model, where the fields and couplings are inferred from the measured means and correlations [3–9]. The methods developed for learning the connections in these models as originally formulated do not assume any prior belief about the network architecture and they only use the data to infer this architecture. Recently though, it has been shown that the connections can be inferred much more efficiently when a sparse prior, specifically the L1 regularizer, is taken into account [10–13].

In many practical applications, equilibrium and regression models that are typically studied in the context of L1 regularization are not the ideal choice for network identification and statistical modeling of the data. A mismatch between the statistical properties of the configurations of the system and an equilibrium fit, as well as a mismatch between the real couplings and the inferred ones, can be seen for large systems [14, 15]. This mainly stems from the out-of-equilibrium nature and the presence of asymmetric couplings between the elements of these systems. Several recent studies have thus moved to kinetic models, prescribing exact and approximate learnings for inferring the connections in non-equilibrium models [4, 15–18]. However this body of work has not yet exploited the potential power of L1 regularization in inferring the connections, and numerical and analytical
studies of $L_1$ regularization have not explored non-equilibrium stochastic dynamical systems. In this paper we make a first step toward filling this gap, treating an asynchronously updated, asymmetrically coupled Sherrington-Kirkpatrick (SK) model.

We study the performance and regularization path for a number of approximations for maximizing the $L_1$ regularized likelihood and compare these to the numerically exact procedure. We show that an approximation similar to those developed in [19, 20] performs well on the SK model, although it does not achieve the performance level of the exact algorithm. A significant body of work is devoted to studying the consistency of $L_1$ estimator in logistic and linear regression problems [13, 21, 22]. Asymptotic analysis shows that increasing correlations between input variables in a regression problem has a negative effect on the performance of $L_1$ for these problems [21, 22]. In the SK model, correlations between the spins are controlled by the magnitude of the interactions increases with increasing coupling strength.

The paper is organized as follows: The dynamics and the underlying network are described in section 2, an $L_1$-regularized learning rule for an asynchronously updated kinetic Ising model is described in section 3, approximate learning algorithms, based on an expansion of the cost function of section 3, are derived in section 4, and the performance of the learning rules is studied in section 5. The effects of different coupling strengths are explored in section 6. A discussion is given in section 7.

2. Glauber dynamics and network

We consider a kinetic Ising model endowed with Glauber dynamics [23]. Glauber dynamics describes the evolution of the joint probability of the spin states $p(s_1, s_2, ..., s_N; t)$ in time $t$, following the master equation

$$
\frac{dp(s; t)}{dt} = \sum_i \omega_i (s) p(s, ..., -s_i, ..., s_N; t) - \sum_i \omega_i (-s) p(s, ..., s_i, ..., s_N; t)
$$

(1)

where

$$\omega_i (s) = \frac{\gamma_0}{1 + \exp \{2s_i H_i(t)\}] = \frac{\gamma_0}{2 \left(1 - s_i \tanh H_i(t)\right)}$$

(2)

is the probability for spin $i$ to change its state from $s_i(t)$ to $-s_i(t)$ during the time interval $dt$. Here, we choose time units so that $\sqrt{\gamma_0} = 1$. The quantity $H_i(t) = h_i + \sum_j J_{ij} s_j(t)$ is the instantaneous field acting on spin $i$. The external field $h_i$ can be dependent on time, but for the sake of simplicity we focus on the stationary case, i.e., time-independent $h_i$, here.

One way to implement the Glauber dynamics is as follows: Make a discretization of the evolution process with very small time steps $\delta t \ll 1/N$. At each step, every spin is selected for updating with probability $\delta t$. For $\delta t \ll 1/N$, almost certainly at most one spin at a time will be updated. The next value of the spin selected for updating is chosen according to

$$p\left(s_i(t) + \delta t\right) \left| \left( s_i(t) \right) \right) = \exp \left[ s_i(t + \delta t) H_i(t) \right] 2 \cosh H_i(t)
$$

(3)

Note that the updated spin might not change its value; an update is not necessarily a flip. In this paper, we will take the dynamics to be defined in this doubly stochastic way and assume that the data accessible to us include both the times at which every spin is selected for updating (determined by independent Poisson processes for each spin) and the result of those updates (whose outcomes are given by (2), i.e., the spin history). The problem may also be treated by other algorithms that only assume knowledge of the spin history (not of all the update times); these are discussed in other work [26], but we do not consider them here. In our computations, in order not to waste lots of time not updating any spins, we have, at each time step, chosen one spin at random for updating. For finite $N$ this is not exactly the dynamics described above, but we do not see any difference when we compare the results of our computations with those done following the correct dynamics exactly.

We study a diluted binary asymmetric SK model with these dynamics. For the original SK model, the pairwise interactions $J_{ij}$ between spins $i$ and $j$ were independent and identically distributed (i.i.d.) Gaussian variables (except $J_{ii} = J_{ij}$) with variance $g^2/N$ and mean 0. In the model we study here, the network is diluted, $J_{ij}$ is independent of $J_{ji}$, and the interactions vary only in sign, not in magnitude: Each coupling has the distribution

$$p(J) = \frac{c}{N} \delta \left( J - \frac{g}{\sqrt{c}} \right) + \frac{c}{N} \delta \left( J + \frac{g}{\sqrt{c}} \right) + \left(1 - \frac{c}{N}\right) \delta(J),
$$

(4)

where $c$ is the average in-degree (and out-degree). We are interested in sparse networks, i.e., $c \ll N$. In our computations, we use $N = 40$ and $c = 5$. Furthermore, as mentioned above, we model asymmetrically coupled spins, taking each $J_{ij}$ independent of $J_{ji}$. This model can have a stationary distribution (and does for the parameters we use here), but it is not of Gibbs-Boltzmann form, and no simple expression for it is known.
3. Exact learning

As described above, we suppose we know the full history of the system—both the data matrix $\{s_i(t)\}$, with $1 \leq i \leq N$ and $1 \leq t \leq L$, where $L = T/\delta t$ is the data length, and the update times $\{\tau_i\}$. Note that $t$ here means something different from the one that appears in the master equation (1). The $i$ in (1) is a continuous time, while here it represents the index of Monte Carlo time step, which is an integer. We can reconstruct the couplings $J_{ij}$ and external fields $h_i$ by performing the gradient ascent on the log-likelihood of this history. Given as date of both spin history and update history, the likelihood is $P(s, \tau) = P(s|\tau)p(\tau)$. For each spin $i$, the $\{\tau_i\}$ are Poisson process: every $t$ has probability $\delta t$ of being a member of the set $\tau$. Thus, the probability of the update history $p(\tau)$ is independent of the model parameter, and the objective function could be taken as $\mathcal{L}_0 = \log P(s|\tau)$:

$$\mathcal{L}_0 = \sum_i \sum_{\tau_i} \left[ s_i(\tau_i + \delta t)H_i(\tau_i) - \log 2 \cosh H_i(\tau_i) \right]. \quad (4)$$

We can maximize the log-likelihood by simple gradient ascent with a learning rate $\eta$:

$$\delta J_{ij} = \eta \frac{\partial \mathcal{L}_0}{\partial J_{ij}} = \eta \sum_{\tau_i} \left[ s_i(\tau_i + \delta t) - \tanh H_i(\tau_i) \right] s_j(\tau_i). \quad (5)$$

This equation includes the learning rule for the external field $h_i$ under the convention $J_{ij} = h_i$, $s_i(t) = 1$. It has the same form as that for a synchronous model, except that changes for spin $i$ are made only at times $\tau_i$.

For the exact learning rule, we take the initial couplings input $J_{ij}^{(0)} = 0$ and iterate equation (5), obtaining the corrections $\delta J_{ij}^{(n+1)}$ using the $n$th estimate $J_{ij}^{(n)}$, on the right-hand side. Inserting each $J_{ij}^{(n)}$ into equation (4) gives $\mathcal{L}_0^{(n)}$ at each iteration step $n$. If we find an increase in likelihood $\mathcal{L}_0^{(n)} - \mathcal{L}_0^{(n-1)} < 10^{-5}$, we consider the iteration to be convergent and stop.

For finite data length $L$, this procedure will in general produce a dense connection matrix. To sparsify it, we add a simple regularization term that penalizes dense connectivity in a controllable fashion. We then minimize a cost function

$$E = -\mathcal{L}_0 + \lambda \sum_{ij} |J_{ij}|, \quad (6)$$

where the first term is the negative log-likelihood and the second term is the $L_1$ norm. There are several efficient methods that have been used to minimize the cost function (6), e.g., the interior-point method [24, 25]. However, in order to see how $L_1$ regularization works in detail, we study a simple gradient descent algorithm here. Gradient descent on this cost function leads to an additional term in the learning rule for couplings:

$$\delta J_{ij} = \eta \left\{ \sum_{\tau_i} \left[ s_i(\tau_i + \delta t) - \tanh H_i(\tau_i) \right] s_j(\tau_i) - \lambda \text{sgn} (J_{ij}) \right\}. \quad (7)$$

The log-likelihood function $\mathcal{L}_0$ is smooth and convex as a function of the $J_{ij}$ and $h_i$, so the cost function is concave except on the hyperplanes where any $J_{ij} = 0$. This leads to complications in the minimization whenever a minimum of $E$ is at $J_{ij} = 0$. We deal with this problem by setting $J_{ij} = 0$ whenever the change (7) would cause $J_{ij}$ to change sign. Then, if the minimum of $E$ truly lies at this $J_{ij} = 0$, the estimated $J_{ij}$ will oscillate between zero and a small nonzero value (using sgn (0) = 0). However, the size of these oscillations is proportional to the learning rate $\eta$, so a sufficiently small $\eta$ ensures that these couplings can be pruned by a simple rounding procedure, with negligible chance of removing couplings that are not truly zero at the minimum. In the case that $J_{ij}$ is not zero at the minimum, its estimated value will continue to change and it will move toward its optimal value after the step where it was set to zero.

For this learning algorithm with $L_1$ regularization, we take as initial couplings the $J_{ij}$ obtained as described above without regularization. Then, for each value of $\lambda$, we iterate equation (7) to obtain successive parameter estimates. At each step $n$, we compute the cost function $E^{(n)}$ using the current parameter estimates and stop the iteration process if $E^{(n-1)} - E^{(n)} < 10^{-5}$. The resulting $J_{ij}$ are then taken as the initial couplings for the next value of $\lambda$. This procedure is carried out for all the values of $\lambda$ for which we want to evaluate the cost function.

Another way to deal with the non-differentiability of the cost function (6) is to use $\lambda \mu \sum_{ij} \text{log cosh} (J_{ij}/\mu)$ as the penalty term and take the limit $\mu \to 0$. This term leads to the replacement of the $\lambda \text{sgn} (J_{ij})$ by $\lambda \tanh (J_{ij}/\mu)$. For any nonzero $\mu$, this modified cost function is totally convex. We checked some of our computations by doing the regularization this way. No difference between these results and those done as described above was found.

4. An approximate learning scheme

We can get some insight into the dynamics of the learning with regularization by expanding the cost function (6) to second order around its minimum $J^0$ when $\lambda = 0$. Up to a constant, we have

$$\frac{E}{T} = \frac{1}{2} \sum_{ij} C_{ij}^{(0)} v_{ij} v_{ij} + \lambda \sum_{ij} |J_{ij}^0 + v_{ij}|. \quad (8)$$

where $v_{ij} = J_{ij} - J_{ij}^0$, $T = LN$ is the number of updates per
spin, $\lambda = \Lambda/T$, and
\[ C^{(i)}_{jk} = \frac{1}{T} \sum_{\tau_i} \left( 1 - \tanh^2 H^{(i)}_{\tau_i} \right) s_j(\tau_i) s_k(\tau_i), \tag{9} \]
where $H^{(i)}_{\tau_i}$ is $H_{\tau_i}$ evaluated with $J_{ij} = J_{ij}^0$.

Since the quantities in the sum in (9) are insensitive to whether spin $i$ is updated, the average over updates may safely be replaced by an average over all times,
\[ C^{(i)}_{jk} = \left. \left( 1 - \tanh^2 H^{(i)}(t) \right) s_j(t) s_k(t) \right|_t, \tag{10} \]
the Fisher information matrix for spin $i$, which is a more robust quantity.

Minimizing (8), we get, to first order in $\lambda$,
\[ \sum_k C^{(i)}_{jk} v_{ik} = -\lambda \text{sgn} \left( J_{ij}^0 + v_{ij} \right) \approx -\lambda \text{sgn} \left( J_{ij}^0 \right). \tag{11} \]
Solving this equation for $v_{ij}$, we obtain:
\[ v_{ij} = -\lambda \sum_k \left[ C^{(i)} \right]^{-1}_{jk} \text{sgn} \left( J_{ik}^0 \right). \tag{12} \]
This equation shows how the regularization term shrinks the magnitudes of the couplings.

In the weak coupling limit (small $g$ or, equivalently, high temperature) and for a uniform external field, the right-hand side of (10) is proportional to $\delta_{jk}$ so the $J_{ij}$ are just shrunk in magnitude proportional to $\lambda$ until they reach zero and are pruned. This is a trivial kind of regularization: We know that the couplings that survive the pruning procedure the longest are simply the ones with the biggest initial absolute values. In this case, there is no need to go through the elaborate learning-with-regularization procedure of (7). However, at larger coupling this is not the case. Some $J_{ij}$ will be shrunk more rapidly than others, depending on the size and signs of the terms in the sum in (12).

Based on the quadratic expansion (8), we can carry out the pruning in an approximate alternative fashion, as follows: Starting from $J_{ij}^0$ and a small value of $\lambda$, we calculate the shifts $v_{ij}$ by (12) and remove any $J_{ij}$ that would go though zero. Starting from the resulting new $J_{ij}$ (some of them now equal to zero), increase $\lambda$, recompute the Fisher information matrix and calculate new shifts in the parameter values. Again remove any couplings that change sign, and continue until the desired degree of pruning has been achieved. This amounts to numerical integration of the differential equation, describing a kind of dynamics of regularization under increasing $\lambda$:
\[ \frac{dJ_{ij}(\lambda)}{d\lambda} = -\sum_k \left[ C^{(i)}(\lambda) \right]^{-1}_{jk} \text{sgn} \left( J_{ik}(\lambda) \right). \tag{13} \]

Note that the matrix $C^{(i)}(\lambda)$ depends on $\lambda$, since it is computed in the absence of the bonds that were removed at preceding steps. An algorithm like this [19] and elaborations thereon [20] have been studied in the machine learning literature for simpler systems and has also been recently used for the reconstruction of sparse Hopfield networks [12].

This algorithm offers potential speed advantages over the full $L_1$ calculation, at least if the optimal value of $\lambda$ is not known a priori. In the full calculation, one then has to perform the iterative learning procedure (7) for each value of $\lambda$.

In the present approach one must still generate data (with the pruned couplings set to zero), but no learning loops are required: it is only necessary to recompute and invert the Fisher information matrix (for each $\lambda$). If one knows the optimal $\lambda$, the advantage is not so clear, because then the time to run the learning procedure once has to be compared with the time to simulate the partially pruned models at all the values of $\lambda$ less than the optimal one. Furthermore, the approximate algorithm may have a cost in accuracy. Below, we compare its performance with that of the full procedure.

5. Performance of learning algorithms

We consider the problem of identifying the positive and negative couplings in the network, i.e., correctly classifying every potential bond as $+$, $-$ or 0. Consider first the couplings $J_{ij}$ found with no regularization, i.e., $\lambda = 0$. For given $g$, $c$ and $N$, for very large $T$ the inferred $J_{ij}$ will be very close to the true ones. A histogram of their values will have three narrow peaks around 0 and $\pm g/\sqrt{c}$, and it will be simple to identify the true nonzero couplings and their signs (figures 1(a)–(b)). In the opposite limit (small $T$), the data are not sufficient to estimate the couplings well. The histogram will be unimodal, and it will be more or less hopeless to solve the problem, even with the help of $L_1$ regularization (figures 1(c)–(d)). The interesting case is that of intermediate data length, for which the partial histograms from the zero and nonzero-$J$ classes overlap, but the separations between their means are not much smaller than their widths (figures 1(e)–(f)). We would also like to avoid the trivial weak-coupling case mentioned above, so in the following results we take $g = 1/\sqrt{2}$. For this case, a $T$ of 200 realizes the interesting intermediate-data-length case.

It is worth remarking that Approximation 1, based on the expansion of the log likelihood to second order around its maximum, is in principle exact in the limit of large data. However, in that limit, the empirical histograms of the inferred couplings would look rather like the top panels in figure 1, and the problem could be solved rather trivially. It is not a priori obvious how good the approximation is for interesting intermediate data lengths. We therefore investigate this question here.

Based on $J$s inferred with $\lambda_0 = 0$ as shown in figures 1(c) and (d), four pruning methods were employed. Figure 1 shows how the $J$s inferred by each method vary as the regularization coefficient $\lambda$ is increased. Here, we only show positive $J$s; graphs of the negative ones would look like the ones shown, reflected about the horizontal axis. Bonds actually present in the model (a realization of (3)) are plotted in black and bonds which are absent in red.

Figure 2(a) shows the $J$s inferred using exact learning with $L_1$ regularization (7). It is apparent that the pruning process for the case shown here is not trivial in the way it would be in the weak-coupling limit: Some true (black) bonds, for which rather small values were inferred at $\lambda = 0$ because of insufficient data, are ‘rescued’ (they fall off more
Figure 1. Distribution of the inferred couplings without L1 regularization, $\gamma = 1/\sqrt{\langle S \rangle}$ for various data lengths. Top: $T = 2000$ updates/spin. Middle: $T = 200$. Bottom: $T = 50$. In each row, the left panel shows a histogram of the $J_{ij}$ obtained, and the right panel shows these sorted according to whether the bond was present (black) or absent (red) in the network that generated the data.

Figure 2. Inferred couplings as functions of regularization coefficient $\lambda$ for four methods: (a) full L1 regularization using (7), (b) integration of (13), (c) integration of (13) with diagonal approximation of the inverse Fisher matrix, (d) linear extrapolation in $\lambda$ of the curves in (c). Black lines represent bonds actually present, while red lines represent ones equal to zero in the network used to generate the data. We show an equal number of red and black ones.
slowly with $\lambda$ than red ones with nearly the same initial inferred $J$s, and some spurious (red) bonds with high inferred values at $\lambda = 0$ are driven to zero faster than black ones with the same initial inferred $J$s. Thus, as $\lambda$ increases the red and black lines tend to get separated, and one can do the pruning almost correctly just by turning $\lambda$ up until the desired number of bonds have been removed.

Figure 2(b) shows the inferred $J$s using the quadratic expansion (8) in the fashion described at the end of section 4. We call this ‘approximation 1’. The qualitative features of figure 2(a) are apparently reproduced in this approximation.

Figure 2(c) shows the result when off-diagonal elements of $\left[ C^{(i)}(\lambda) \right]_{ij}$ are ignored in (13). We refer to this procedure as ‘approximation 2’. The separation of red and black curves is not as good in this case. We also tried making a diagonal approximation of the Fisher matrix itself, rather than its inverse: $C^{(i)}_{\lambda}$ by $C^{(i)}_{\lambda} \delta_{\lambda}$. However, this gave much worse results (not shown) than making the diagonal approximation on the inverse Fisher matrix.

In figure 2(c), it is evident that the slopes of the $J_0(\lambda)$ curves vary rather slowly with $\lambda$. Therefore, we also tried a linear extrapolation based on the slopes of the curves in figure 2(c) at $\lambda = 0$. We denote this method as ‘Approximation 3’. To the extent that this simple procedure works, one can identify the nonzero bonds with very little computation: One needs only to do the learning at $\lambda = 0$ (to get the $J_0(\lambda)$) and calculate the Fisher matrices (to get the $dJ_0/d\lambda$). Figure 2(d) shows the result of this minimal algorithm.

For Approximation 3, the inferred $J$s that have been shrunk to zero are removed permanently. For the other three approaches, the inferred $J$s have a chance to be ‘resurrected’ with increasing $\lambda$, though in the results presented here we haven’t observed this.

One could also try similar linear extrapolation based on the initial slopes of the upper panels of figure 2. However, these curves show significant curvature for $\lambda < 30$ or so, so the initial slopes are not good guides to the ultimate fate of the bonds at large $\lambda$, and we do not present any results for these methods.

In what follows, we quantify the performances of these four pruning algorithms. For the three classes of bonds in the actual network, $-\lambda$ and $\lambda = 0$, we can compute the empirical classification errors. These errors can be either false positives (FP) (identifying a bond which is really absent as present), or false negatives (FN) (identifying a bond which is actually present as absent). In addition, a $+$ bond could be misclassified as $-$ or vice versa, but this does not happen for the data length we are studying here.

At $\lambda = 0$, where in general all bonds will be estimated to have nonzero values, there will be no FNs and $N(N - e)$ FPs. In the other limit $\lambda \to \infty$, all bonds will be removed, so there will be $eN$ FNs and no FPs. The empirical numbers of FPs and FNs versus $\lambda$ are plotted in the left panel of figure 3. The total misclassification error, i.e., the sum of the FPs and FNs (shown in the right panel of figure 3) has a minimum at a certain $\lambda$ value, which could be considered as the optimal $\lambda$ for regularization $\lambda_{opt}$. We find that $\lambda_{opt}$ is around $33.5$ for full $L_1$ regularization. For Approximation 1 it is about $31.5$, while for Approximation 2 it is $30$, and for Approximation 3 it is $24$.

To compare algorithms in a general way, we calculate receiver operating characteristic (ROC) curves for them. For a given $\lambda$, the false positive rate (FPR) is defined as the number of FPs divided by the actual number of zero bonds, and the false negative rate (FNR) is defined as the number of FNs divided by the actual number of non-zero bonds. A true positive (TP) is the identification of a bond which is actually present as present, and the true positive rate (TPR) is the number of TPs normalized by the actual number of bonds present. It is equal to $1 - \text{FNR}$. The ROC curve is a plot of TPR versus FPR. Each value of $\lambda$ gives one point on the curve. In figure 4, we plot the ROC curves for all of our methods. We also measure the performance of the different methods quantitatively by defining an error measure, $e$:

$$ e = 1 - \text{area under ROC curve}. \quad (14) $$

The values of $e$ for full L1 and Approximations 1, 2 and 3 are $0.03, 0.06, 0.08, 0.09$ respectively. Thus, full L1 algorithm performs best, followed by Approximation 1. Approximation 2 works worse than them and it is only a little better than Approximation 3 for most values of $\lambda$, as can be seen in figure 4.

To establish a baseline for the performance of our methods, we also performed a simple pruning procedure that does not require any L1 regularization calculation. For a given cut value $\hat{J}$, we identify the bonds whose $J$s lie in the range $[-\hat{J}, \hat{J}]$ as absent and those outside that interval as present. The black $J$s in figure 1(d) which lie within the interval are FNs and the red ones outside the interval are FPs. Varying $\hat{J}$, we obtain an ROC curve. We refer to this procedure as ‘30-cut’. If there were a panel showing it in figure 2, the lines (of both colours) would all be parallel. The curve with light blue squares in figure 4 is calculated using this method. It gives the same value of $e(0.09)$ as Approximation 3, and the ROC curves nearly coincide.

Since Approximations 2 and 3 seem not to offer significant advantages over this trivial pruning procedure, we do not explore them further here.

**6. Effects of coupling strength $g$ on L1 regularization**

The results reported so far were all done for $g = 1/\sqrt{2}$. How does changing $g$ change the performance of $L_1$? We address this question in this section.

A well-known result in the L1 literature is that the presence of correlations between covariates in an L1 regularized regression model will have a negative effect on the consistency of the reconstruction [21, 22]. Given the likelihood (4), one can think about inferring the connections in the kinetic Ising model as a regression problem, where the spin configurations at time $t$ are the predictors of the values at time $t + 1$. One can therefore naively expect that increasing the strength of the connections, $g$, and thus the correlations between the covariates, will have a negative effect on the L1-
regularized inference. This is, however, not true, as shown
and discussed in this section.
To study the effect of the couplings and correlations, we
thus performed the calculations described in sections 3 and 4
for full L1 regularization and Approximation 1, respectively,
for two other values of $g = 1/2, 1$. Figure 6 shows how the ROC curves change as we
change $g$ for a fixed data length. The first observation is that
for the J0-cut, shown by dashed lines, increasing $g$ helps recover the correct connections. We see the same trend for the exact L1 regularization as well as Approximation 1, shown by solid curves. This can be understood intuitively in the following way. Increasing $g$ has two negative effects. First, it increases the equal time correlations, that is, thinking about the problem as a regression problem, one would be dealing with more correlated covariates. Second, increasing $g$ increases the correlation time and therefore different data

Figure 3. Dependence of classification errors on $\lambda$. Left column: number of misclassified $-$, $+$, and 0 (absent) bonds. Numbers of false negatives for $-$s are shown in green, for $+$s in red, and false positives for zero-bonds in blue. Right column: the sum of false negatives and false positives versus $\lambda$. Because the Js are symmetrically distributed, red and green curves almost coincide, with mostly only the green ones visible here. From top to bottom: full L1 regularization and Approximations 1, 2, 3, respectively.

Figure 4. ROC curves for full L1 regularization, Approximations 1, 2, 3, and the J0-cut method are shown in red, green, blue, pink and light blue, respectively.
points will be more correlated and less informative about the presence or absence of a connection. On the other hand, with larger $g$, the parameters to be inferred are bigger and can be identified more easily. It is the relative strength of these three factors that determine the net effect of increasing $g$ on the inference, and, as we see, the last of these wins out over the other two for the coupling strengths we have studied.

Given the result on the negative effect of correlations on $L_1$ in regression problems, a more surprising result, shown in what follows, is that, not only the performance of $L_1$ regularization, but also its relative benefit over the simple $J_0$-pruning increases with $g$. One way to see this is by calculating the percentage of the area above the ROC curve, that is $\epsilon$, in figure 5 that will be eliminated by $L_1$ as a function of $g$. For $g = 1$, we found that 80% of the area above the ROC curve would be eliminated by exact $L_1$ regularization while these numbers are 55% and 30% for $g = 1/\sqrt{2}$ and $g = 1/2$. For Approximation 1 these numbers are 40%, 22% and 10%. These results are consistent with the observation made in section 4: in the weak-coupling limit $g \to 0$, the Fisher information matrices $C^l$ are all proportional to the identity matrix, thus in this limit all the regularization methods we have discussed here will give the same results as the trivial $J_0$-cut procedure and will not offer any advantages.

To further understand how the benefit of $L_1$ changes with $g$, we considered the same three values of $g$ but this time adjusted the length of data for each $g$ such that without $L_1$ the problem for different $g$ has the same intrinsic difficulty, as measured by the areas under their ROC curves computed by simple $J_0$-cut pruning. The results are shown in figure 6. The error measure $\epsilon$ is 0.033 for $g = 1/2$, 0.023 for $g = 1/\sqrt{2}$ and 0.012 for $g = 1$. Similarly, in figure 6(b), the solid lines are ROC curves found using Approximation 1, with $\epsilon$ = 0.047, 0.039 and 0.035 for the three cases, respectively.

It is apparent from these results that as couplings get stronger, $L_1$ regularization is able to provide increasingly more accurate network reconstructions and more benefit over an un-regularized reconstruction. This relative advantage is evident both in the percentage of the error that is eliminated by $L_1$ at fixed data length, and when adjusting the data length

Figure 5. ROC curves for full $L_1$ regularization (left, solid lines) and Approximation 1 (right, solid lines) with $g = 0.5, 1/\sqrt{2}, 1$ respectively. The green lines for $g = 0.5$, red for $g = 1/\sqrt{2}$ and black for $g = 1$. Corresponding dashed lines are for $J_0$-cut method of these $g$s. The length of the data is $L = 6730$.

Figure 6. ROC curves for full $L_1$ regularization (left, solid lines) and Approximation 1 (right, solid lines) with $g = 1/2, 1/\sqrt{2}, 1$ respectively. The green lines are for $g = 1/2$, red for $g = 1/\sqrt{2}$ and black for $g = 1$. Corresponding dashed lines are for $J_0$-cut method of these $g$s.
with the coupling so that all three cases studied have the same naive degree of difficulty (defined by equal error measure under J0-cut).

Comparison of the left and right panels in figures 5 and 6 make evident another point: While one sees the same qualitative effect for Approximation 1 as for full L1 regularization, it is weaker for the approximate algorithm. Not only are the errors $\epsilon$ larger than for the exact algorithm, but also the reduction in the error with increasing coupling strength is smaller. Our interpretation of this finding is that (1) higher-order corrections to the quadratic expansion of the log likelihood are not negligible for the data lengths used here, and (2) these corrections get larger with decreasing data length. In fact, we have found that, for larger data lengths than the one shown in figure 5, the relative improvement offered by Approximation 1 as a function of $g$ does not follow the monotonic increase we see for the full L1. This again emphasizes the point that Approximation 1 is not a perfect substitute for exact L1 minimization.

The conclusion that increasing the couplings helps L1 regularization appears to be rather general. In figure 7, ROC curves for very sparse networks ($c = 2$) are shown for network sizes $N = 40$ (upper plots) and $N = 80$ (lower plots). For each size, the left-hand graph is for $g = 0.3$ and the right-hand one for $g = 0.5$. The data lengths are 8500 for $N = 40$ and 24000 for $N = 80$. Comparing the left- and right-hand columns, we observe that, for this (fixed) value of $c$ and both the network sizes studied, better reconstruction is obtained for larger $g$.

7. Discussion

We have studied the reconstruction of sparse asynchronously updated kinetic Ising networks. With finite data length, simple maximization of the log likelihood of the system history will infer nonzero values for many bonds that are actually not present. For large data length, this is generally not a problem, since the inferred bond distribution will consist of well-separated peaks. The ones with the smallest absolute values can then safely be identified as spurious and removed ‘by hand’. However, for smaller data lengths, these peaks can overlap strongly, and nontrivial methods are required to perform optimal pruning of the inferred coupling set. Here we used L1 regularization to do this, minimizing a cost function that includes the L1-norm of the parameter vector as a penalty term. We performed this minimization in four ways, one exact and the other three involving various degrees of approximation.

Calculations on a model network at intermediate coupling strength revealed that the exact L1 regularization classified the bonds significantly better than a naive method based on retaining the strongest bonds. Approximation 1 was somewhat worse than the exact algorithm, but still
significantly better than the naive method. Our other two approximations, obtained by successive simplifications of Approximation 1, however, did not perform measurably better than the naive way, as measured by the areas under their ROC curves. These conclusions are general with respect to the coupling strengths we used. The regularization helps more with stronger coupling strengths.

This work is the first that we know of that takes a detailed look at how $L_1$ regularization works in a non-equilibrium model, by studying how bonds are removed successively as the regularization parameter $\Lambda$ is increased. Some insight into how this happens was made possible by studying the quadratic expansion of the cost function about its minimum, which also led to the relatively successful Approximation 1. The process would have been more transparent if we could have made further simplifying approximations, as we did for Approximation 2, where we ignored off-diagonal elements of the inverse Fisher matrices. The fact that this approximation performed rather poorly (while Approximation 1 did quite well) indicates that the off-diagonal terms in (13) are necessary, and we lack generic insight about them.

Perhaps surprisingly, we showed that in the kinetic SK model, increasing the couplings and thus the correlations helps the performance of $L_1$. This was true both for the exact $L_1$ solution and, for small data length, Approximation 1. Although at first glance this might seem inconsistent with the results of the regression studies with $L_1$ [21, 22], a closer look shows that this is not the case. In regression problems, correlations between the input covariates and the strength of the couplings between the inputs and the output are independent parameters. This is not the case for the model studied here or for many other kinetic models in which these two effects covary in a way that is controlled by the magnitude of the couplings and have opposing effects on network reconstruction.

**Acknowledgements**

We are grateful to E Aurell and M Alava for useful discussions about the work and to Nordita and Niels Bohr Institute for hospitality. The work of H-L Z was supported by the Academy of Finland as part of its Finland Distinguished Professor program, project 129024/Aurell.

**References**

[1] Bailly-Bechet A et al. 2010 BMC Bioinformatics 11 355
[2] Weigt M, White R A, Szurmant H, Hoch J A and Hwa T 2009 Proc. Natl. Acad. Sci. 106 67
[3] Ackley D H, Hinton G E and Sejnowski T J 1985 Cogn. Sci. 9 147
[4] Cocco S, Leibler S and Monasson R 2009 Proc. Natl. Acad. Sci. 106 14058
[5] Kappen H J and Rodriguez F B 1998 Neural. Comput. 10 1137
[6] Roudi Y, Tynrcha J and Hertz J 2009 Phys. Rev. E 79 051915
[7] Tanaka T 1998 Phys. Rev. E 58 2302
[8] Schneiderman E, Berry M J II, Segev R and Bialek W 2006 Nature 440 1007
[9] Vasquez J C, Palacios A, Marme O, Berry M J II and Cessac B 2012 J. Physiol. Paris 120 106
[10] Aurell E and Ekeberg M 2012 Phys. Rev. Lett. 108 090201
[11] Decelle A and Ricci-Tersenghi F 2014 Phys. Rev. Lett. 112 070603
[12] Huang P 2012 Eur. Phys. J. B. 86 1–7
[13] Ravikumar P, Wainwright M J and Lafferty J D 2010 Ann. Stat. 38 1287
[14] Roudi Y, Nirenberg S and Latham P 2009 PLoS Comp. Biology. 5 e1000380
[15] Roudi Y and Hertz J 2011 Phys. Rev. Lett. 106 48702
[16] Pillow J W, Shlens J, Paninski L, Sher A, Litke A M, Chichilnisky E J and Simoncelli E P Nature 454 99
[17] Zeng H-L, Aurell E, Alava M and Mahmoudi H 2011 Phys. Rev. E 83 041135
[18] Zhang P 2012 J. Stat. Phys. 148 502
[19] Rosset S 2005 Adv. Neur. Inf. Proc. Sys. 1153 1153–60
[20] Friedman J, Hastie T and Tibshirani R 2010 J. Stat. Software 33 1
[21] Knight K and Fu W 2000 Asymptotics for lasso-type estimators Ann. of Stat. 28 1356
[22] Zhao P and Yu B 2006 J. Mach. Lear. Res. 7 2541
[23] Glauber R J 1963 J. Math. Phys. 4 294
[24] Koh K, Kim S J and Boyd S 2007 J. Mach. Learn. Res. 8 1519
[25] Ye Y 1997 Wiley-Interscience Series in Discrete Math. and Optim., Interior Point Algorithms: Theory and Analysis (NJ, USA: John Wiley & Sons)
[26] Zeng H-L, Alava M, Aurell E, Hertz J and Roudi Y 2013 Phys. Rev. Lett. 110 210601