Error Bounds for Sliding Window Classifiers

Yaroslav Bulatov

January 18, 2010

Abstract

Below I provide examples of some results I developed that let me bound error rate of Bayes optimal sliding window classifiers on graphical models

Closed form for binary Forward classifier

Suppose you have a binary chain CRF of length n with edge potentials $\psi_1, \psi_2, \ldots, \psi_{n-1}$

$$\psi_i = \begin{pmatrix} e^{a_{11}^{(i)}} & e^{a_{12}^{(i)}} \\ e^{a_{21}^{(i)}} & e^{a_{22}^{(i)}} \end{pmatrix}$$

We are interested in the log-odds of the last state of the chain being 1, denoted by o_n It is determined recursively as

$$o_1 = 0 \tag{1}$$

$$o_{n+1} = \operatorname{arctanh}(\tanh q \tanh o_n + b_2) + b_1 \tag{2}$$

Where

$$b_1 = \frac{a_{11} + a_{12} - a_{21} - a_{22}}{4} \tag{3}$$

$$b_2 = \frac{a_{11} - a_{12} + a_{21} - a_{22}}{4} \tag{4}$$

$$q = \frac{a_{11} - a_{12} - a_{21} + a_{22}}{4} \tag{5}$$

Example

Suppose I have a chain of length 4, where each transition matrix has the following form.

Figure 1: Binary Chain

$$\psi_i = \begin{pmatrix} e^1 & e^{-1} \\ e^{-1} & e^1 \end{pmatrix} \tag{6}$$

The potential matrix is symmetric, so bias terms (b_1,b_2) are 0, and the log odds of the last state being 0 takes the following form

$$f(f(f(0))) \tag{7}$$

Where

$$f(x) = \operatorname{arctanh}(\tanh 1 \tanh x) \tag{8}$$

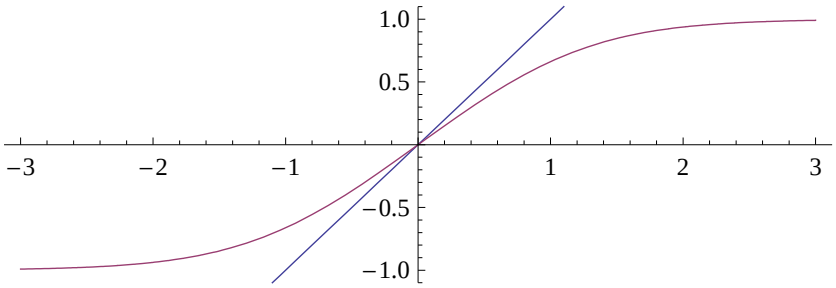

Figure 2: x and $f(x)$

Now, suppose we know the first state is 1, then log odds of the last state being 1 is

$$f(f(f(\infty))) \tag{9}$$

Using the definition of $f(x)$ from 8 we get $\operatorname{arctanh}(\tanh^3 1) \approx 0.474$

Suppose we now have the same situation, but with 5 states. The log-odds of last state being 1 is now $f(f(f(f(\infty)))) \approx 0.350$

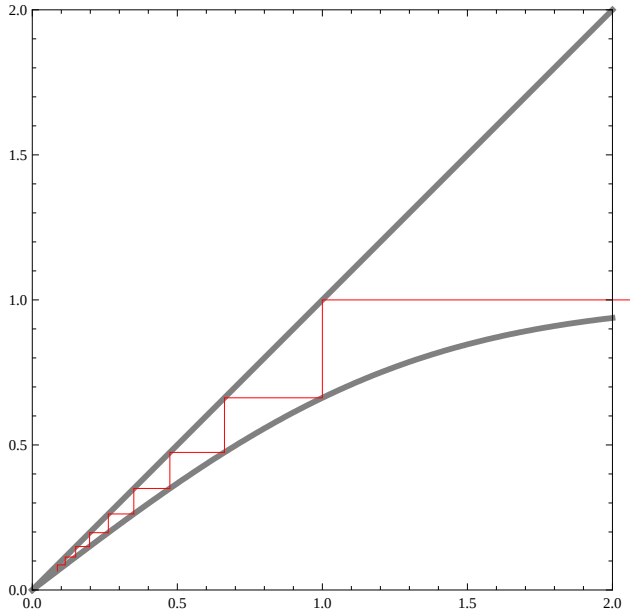

Figure 3: Fixed point iterations of $f(x)$

Repeating this operation, makes the estimated log-odds converge to 0 as is evident from the cobweb plot above.

The rate at which fixed point iterations converge to 0 corresponds to the rate of "forgetting" of far away observations. We can use that rate to bound the error incurred by discarding evidence outside of a certain window.

Example, for the function in Figure 2, the derivative is at most 0.7615. From this we can conclude that in order to make a correct prediction, the forward classifier only needs to consider k most recent observations, where

$$k = \begin{cases} \frac{\log \text{margin}}{\log 0.7615} & \text{if margin} < 1 \\ 1 & \text{otherwise} \end{cases} \tag{10}$$

Margin is the absolute value of the log-odds estimate produced by the Bayes-optimal classifier, it reflects how strongly the data favors a particular state for this model.

Classifiers for Binary-valued Trees

Sum-product classifier for tree-structued graphical models has closed form similar to 2. The simplest version is obtained when edge ij has the following potential matrix

$$\psi_{ij} = \begin{pmatrix} \exp J_{ij} & \exp -J_{ij} \\ \exp -J_{ij} & \exp J_{ij} \end{pmatrix} \tag{11}$$

To find log-odds of node 1, rearrange the tree so that 1 is the root and arrows point from root to leaves. Then letting ∂j indicate the list of children of node j, log odds of node 1 being in state 1 are o_1 where

$$o_i = \sum_{j \in \partial i} f_{ij}(o_j) \tag{12}$$

Where

$$f_{ij}(x) = \operatorname{arctanh}(\tanh J_{ij} \tanh x) \tag{13}$$

See figure 4 for plot of f for various values of J

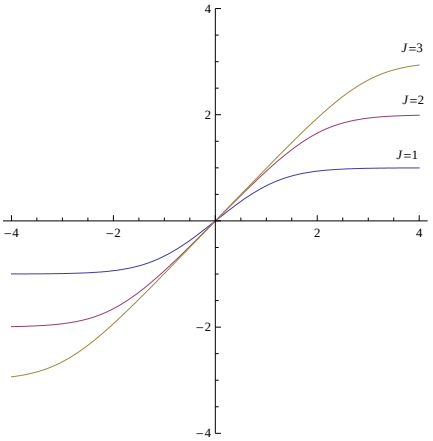

Figure 4: Plot of $f(x) = \operatorname{arctanh}(\tanh J \tanh x)$

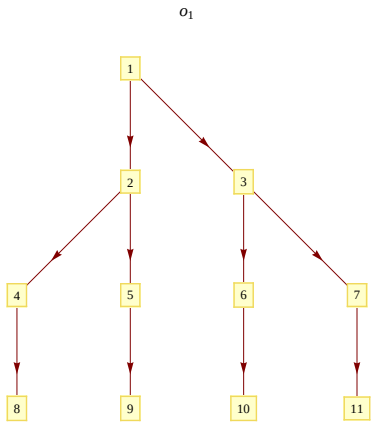

Figure 5: Tree rooted at 1

Example

Suppose we have have the graphical model structure as in the figure 5, with potentials of the form in (11), and let $J_{ij} = 1$. Fix all the nodes at level 2 (ie, nodes

4,5,6,7) to have state 1. This corresponds to setting o_4,o_5,o_6,o_7 to ∞. Log-odds at the root now becomes

$$o_1 = f(f(\infty) + f(\infty)) + f(f(\infty) + f(\infty)) \approx 1.875 \tag{14}$$

If we do the same, but now set those nodes to have state 0, we get log odds of about -1.875. From this we can show that if the margin for the Bayes-optimal classifier is greater than 1.875 we can discard any evidence more than 2 levels deep in the tree and still make a correct prediction.

A simpler way to bound the maximum error in log odds incurred by discarding any evidence beyond k levels is to instead consider the first order Taylor expansion of $f(x)$ around 0, see figure 6

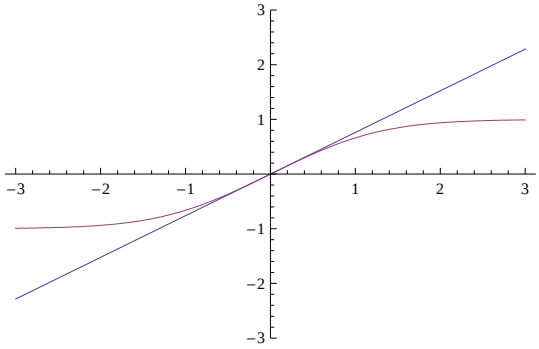

Figure 6: Plot of $f(x)$ and it's first order Taylor expansion $\hat{f}(x)$

The advantage of using $\hat{f}(x)$ instead of $f(x)$ is that if we use it to replace f in our sum-product classifier (14), the resulting classifier will be linear. Additionally, error bounds based on this simplified classifier are still valid for the Bayes-optimal classifier.

Consider how we can rewrite estimate of the root log-odds using this linearized classifier, for the same problem as in 14 –

$$\begin{aligned} \hat{o}_1 &= \hat{f}(\hat{f}(\infty) + \hat{f}(\infty)) + \hat{f}(\hat{f}(\infty) + \hat{f}(\infty)) && (15) \\ &= \hat{f}(\hat{f}(\infty)) + \hat{f}(\hat{f}(\infty)) + \hat{f}(\hat{f}(\infty)) + \hat{f}(\hat{f}(\infty)) && (16) \end{aligned}$$

You can see that we get 4 terms. Note that in our tree there are 4 paths of length 2. It is not a coincidence. More generally, for this "linearized sum-product"

classifier we will get a term for each path, where the level of nesting of f's in each term corresponds to length of the path. Another example, suppose we have a tree all potentials the same and two paths starting from root node, of length 1 and length 2. Then, estimate of root node log-odds will be $\hat{f}(x) + \hat{f}(\hat{f}(x))$

Using potentials of the form 11 and the linearization trick I just described, we get the following bound for the maximum influence on root log-marginal by nodes outside of radius (path-distance) k from root node

$$\sum_{p} J_{p_k,p_{k+1}} \tanh J_{p_1,p_2} \tanh J_{p_2,p_3} \ldots \tanh J_{p_{k-1},p_k} \tag{17}$$

Where the sum is taken over all paths of length $k+1$ starting at node 1.

If node 1 is at least k away from the closest leaf, all potentials are the same, and the degree of each node is d, this simplifies to

$$J \tanh J^k d^k \tag{18}$$

For our simple example, this approach gives a bound of $1 \cdot \tanh^2 1 \cdot 2^2 \approx 2.3201$. Stated another way, our estimate of log-odds will be off by at most 2.3201 if we discard observations outside of radius 2 in any 2-regular tree with binary valued states where neighbouring nodes have mutual information at most 1.48 bits. This may seem quite loose, but that's because in regular trees, the number of evidence nodes at distance d grows exponentially with d so without restrictions on tree size, there can be potentially quite a bit of evidence outside of any fixed window radius.

The approach used in equation 14 is optimal in a sense that without restrictions on the strength of local potentials, the bound is tight. However, we shall see that for general graphical models, optimal bound is intractable to compute, while linearization trick gives us tractable bounds.

Binary Valued General Graphical Models

Weitz (2006) has recently introduced a way to take a marginal in an arbitrary graph, and find a tree such that the marginal of a root node of this tree is the same as the marginal in the arbitrary graph. This construction is known as the self-avoiding walk tree. Figure 8 gives a self-avoiding walk representation of a marginal of node 3 in 7. "3-" and "3+" indicate that we fix that node to have value 0 or 1 respectively. More specifically the tree is constructed by considering all self-avoiding walks on a graph starting at the target node, then creating a tree by

merging walks with common prefixes. When loop is encountered while constructing the walk, the self-avoiding walk terminates. The node that closes the loop is included in the tree, fixed to either 0 or 1 value, depending on the direction in which the loop was traversed. Node 3 in figure 8 and Node 3 in figure 7 have the same marginal

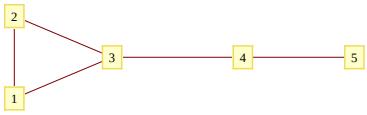

Figure 7: Example graphical model

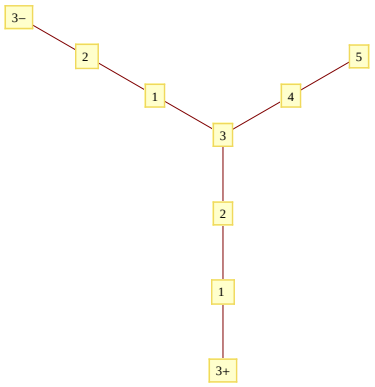

Figure 8: SAW-tree representation of node 3 marginal

Using self-avoiding walk representation, and the same potential as 6 we can have a similar closed form representation for log-odds of node 3

$$o_3 = f(f(0)) + f(f(f(\infty))) + f(f(f(-\infty))) \tag{19}$$

Where f is the same as in 8. Using linearization trick as in the previous section, we can obtain the following bound on the maximum influence on the log-marginal by nodes outside of some radius k

$$\sum_p J(\tanh^{|p|} J) \tag{20}$$

Here the sum is taken over all self-avoiding walks of length $k+1$. Note the similarity to equation 17

We can rewrite it as

$$J \sum_p c^{|p|} \tag{21}$$

Where $c = \tanh J$. The sum in the equation 21 is known as the generating function for self-avoiding walks. Those have been heavily studied in physics literature, and we can use those results to obtain bounds on sliding window classifier error for lattices. For instance, consider the square lattice structure in figure 9

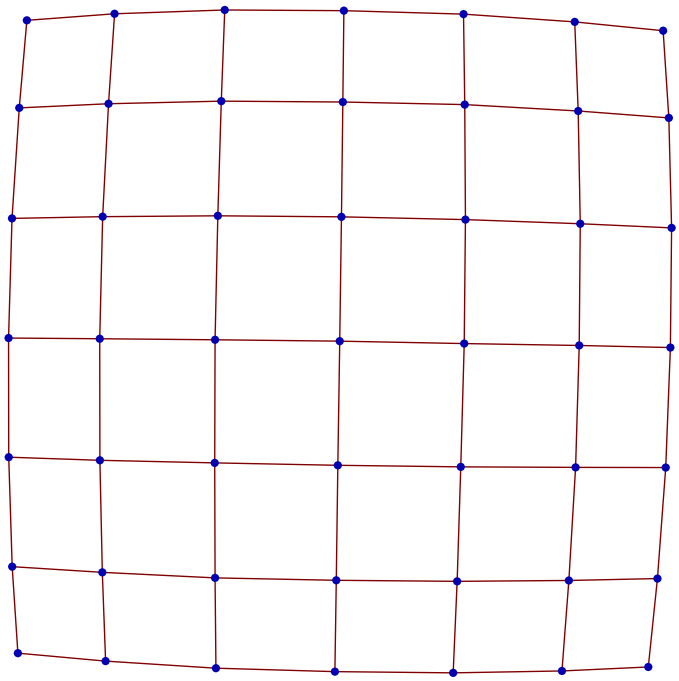

Figure 9: Piece of square lattice

If we again use the simple potential as in eq. 11, linearization trick and the results from self-avoiding walks on lattices (ie, from (Madras & Slade, 1996)), we get the following estimate for the amount of influence exerted by nodes more than k steps away from target node.

$$\sum_{n=k+1}^{\infty} n^{\frac{11}{32}} (\mu \tanh J)^n \tag{22}$$

Where μ is the unique positive root of polynomial $13x^4 - 7x^2 - 581 \approx 2.63816$. You can see that in order for this sum to converge, J has to be at most 0.398952.

This is related to the phenomenon of phase-transitions. The graphical model in the figure above corresponds to square-lattice Ising model. For this Ising model, it is known that above a certain value of J, there's no longer decay of correlations, in other words, the importance of far away states doesn't diminish with distance. In a true Ising model (allowing infinite square grid), there's perfect correlation between all states when $J > \text{arcsinh}(1)/2 \approx 0.441$. Therefore, any kind of bound on influence of far away observations must become trivial for $J > 0.441$. We introduce looseness by linearizing f, so our bound becomes trivial slightly earlier, for $J > 0.399$.

In order to get optimal sliding window size given margin (which as I mentioned depends on the strength and pattern of observations), we could solve equation 22 for k numerically. Alternatively, we could approximate the influence by dropping the polynomial term, then we can solve for k explicitly, we get

$$\frac{\log\left(\text{margin}(1-\mu\tanh J)\right)}{\log\left(\mu\tanh J\right)} \tag{23}$$

Where μ is the same as in equation 22. Suppose our margin is 0.5, then you can see the size of the sliding window of optimal sliding window classifier for large square lattice as a function of J in figure 10 (it defines the potential matrix as in equation 11)

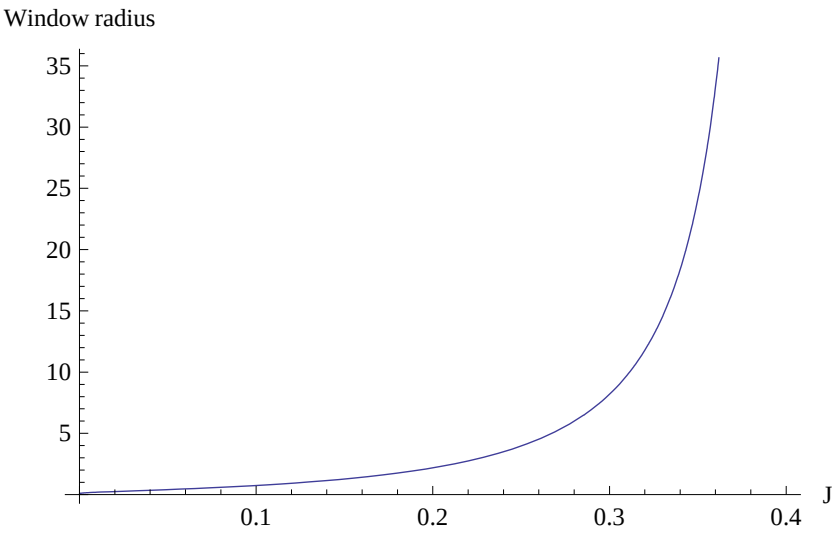

Figure 10: Bound on size of optimal sliding window classifier for square lattice (fig 9) where edge potentials have form (24)

$$\begin{pmatrix} \exp J & \exp -J \\ \exp -J & \exp J \end{pmatrix} \tag{24}$$

Linear Sliding Window for Finite Graphs

Obtaining error bounds and obtaining linear classifiers on graphs are related tasks – we can take a first-order Taylor expansion of log-marginal as a function of evidence, then consider the regret incurred by discarding far-away evidence for this linear classifier. This "linearized" regret bounds the true regret of optimal sliding window. Results from mathematical physics literature mentioned in previous section (ie, (Guttmann & Conway, 2001)) apply to large graphs with a high degree of symmetry. For finite graphs, we can use a more direct approach.

Suppose $f_i(x_1, x_2, \ldots, x_n)$ represents log-odds of node i as a function of evidence on nodes $1 \ldots n$. Then we have the following result for binary valued graphical models

$$\frac{\partial f_i}{\partial x_j} \leq \frac{\partial y_j}{\partial x_j} \sum_p s(p_1, p_2) s(p_2, p_3) \ldots s(p_{n-1}, p_n) \tag{25}$$

The sum is taken over all self-avoiding walks between node i and j. The function $s(n_1, n_2)$ represents the degree of informativeness of the edge between nodes n_1 and n_2. For simple potentials of the form (24) it is $\tanh J$, for general potentials on binary models () it is $\tanh q$ where q is defined as in (5).

The bound will become tight if we introduce loop correction factors, more on this below. As it stands, it is tight in the limit of weak loop interactions. In other words, if we want our bound to be loose by at most a factor of e, we can find pick loop potentials where this tightness is satisfied, for any e

If we consider models with more than two states, f_i is now a vector valued function. In this case, the following holds

$$\| \frac{\partial f_i}{\partial x_j} \| \leq C \| \frac{\partial y_j}{\partial x_j} \| \sum_p s(p_1, p_2) s(p_2, p_3) \ldots s(p_{n-1}, p_n) \tag{26}$$

C is a constant that depends on the norm $\|.\|$. Function $s(.,.)$ is now the Birkhoff contraction coefficient (Hartfiel, 2002), of the potential matrix for the corresponding edge. When the norm used is a certain polyhedral norm related to Hilbert's projective metric (Kohlberg & Pratt, 1982), $C = 1$ and the bound is tight, subject to conditions on loop potentials as before.

Maximum error incurred by dropping evidence from node j is

$$C\,\Delta(\phi_i)\sum_{p} s(p_1,p_2)s(p_2,p_3)\ldots s(p_{n-1},p_n) \tag{27}$$

Where Δ is the range of the image of the emission potential on node i. It is closely related to Birkhoff contraction coefficient. For a potential matrix $\{a\}_{ij}$, Δ and a certain polyhedral norm $\|.\|_h$ (explained below), (also brought up in (Harpe, 1991)) Δ is defined

$$\Delta = \max_{ijkl} \frac{a_{ik}a_{jl}}{a_{il}a_{jk}} \tag{28}$$

This formula is derived in greater detail in (Bapat & Raghavan, 1997).

C is a constant that depends on the metric used to measure error, same C as in (26). For instance, if you measure error in terms of L_2 distance between estimated log-odds and true log-odds, and we have 3 states, C=$\frac{\sqrt{3}}{2}$. If we use distance measure introduced by Darwiche/Hei Chan (2002), C=1. In fact, this measure has been rediscovered several times, and is also known as Dynamic Range (Ihler, 2007), "Hilbert's projective metric" ((Hartfiel, 2002)), also it is the same as the quotient metric for the l_∞ norm introduced in (Mooij & Kappen, 2007). A few other places where it comes up are (Waser, 1986),(Liebetrau, 1983),(Altham,),(Atar & Zeitouni, 1996)

This metric seems to be the most natural to use for measuring information loss because it's been shown that it is the only metric (up to monotonic bijection) under which two belief vectors are brought closer together when convolved with any positive edge potential, (Kohlberg & Pratt, 1982). Diagram below marks contours of common metrics centered at 0, our metric is marked as $\|.\|_h$.

For models with 3 states, level sets of Hilbert's projective metric are regular hexagons. For 4 states, they are rhombic dodecahedra as in Figure 12

For higher dimensions, the level sets of this metric are known as strombiated simplices. We can obtain bound on error in more familiar metrics like l_2 distance in log odds by comparing ratio of circumscribed (hyper)sphere to inscribed sphere for one such simplex. For instance, the constant C=$\frac{\sqrt{3}}{2}$ for error in l_2 metric in Eq (27) is precisely the ratio of circumscribed to inscribed circles for a regular hexagon.

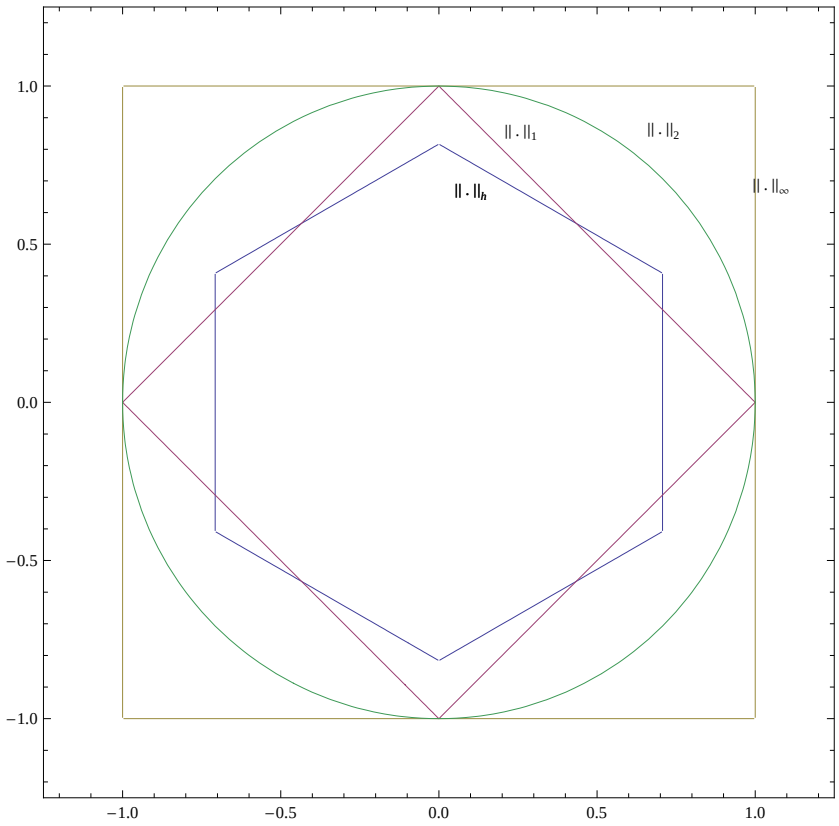

Figure 11: Level-1 sets of different metrics centered at 0

Computing Correlations for Linear Sliding Window

In order to compute a linearized sliding window classifier we need to compute all correlations from the equation (25). The difficulty with that formula is that it requires enumerating over all self-avoiding walks, which can potentially take exponential time. We can relax the requirement and compute an upper bound instead by enumerating over all walks, not necessarily self-avoiding.

Consider the following matrix

$$A = \begin{pmatrix} s(1,1) & s(1,2) & \dots & s(1,n) \\ s(2,1) & s(2,2) & \dots & s(2,n) \\ \dots & \dots & \dots & \dots \\ s(n,1) & s(n,2) & \dots & s(n,n) \end{pmatrix} \tag{29}$$

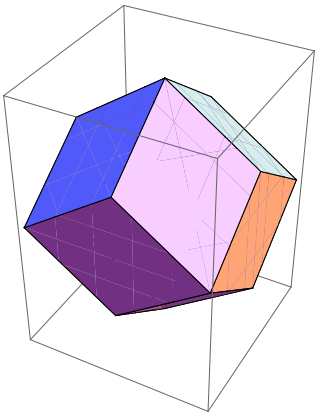

Figure 12: Level set of dynamic range norm for 4-state model

Note that $(A^n)_{ij}$ is the same as

$$\sum_p s(p_1,p_2)s(p_2,p_3)\ldots s(p_{n-1},p_n) \tag{30}$$

Where the sum is taken over walks of length n.

In order to get an upper bound on (25), we need to consider walks of all lengths. The sum in the equation (25) will be the ijth entry in the matrix defined by the following

$$I + A^1 + A^2 + A^3 + \ldots = (I - A)^{-1} \tag{31}$$

This is known as the Neumann series, (Meyer & Meyer, 2001) page 126 and the equality above holds when spectral radius of A is < 1. When the spectral radius is $>= 1$, the above equality doesn't hold, the series diverges, and this represents the fact that the bound becomes trivial.

Example

Consider binary valued graphical model in figure 13. Suppose all potentials (edge and observation potentials) have the form

$$\psi_i = \begin{pmatrix} e^{0.1} & e^{-0.1} \\ e^{-0.1} & e^{0.1} \end{pmatrix} \tag{32}$$

We are interested in linear classifier that predicts state 1 as a function of observations at nodes 2,3,4. Use equation 25 to compute approximate linear expansion

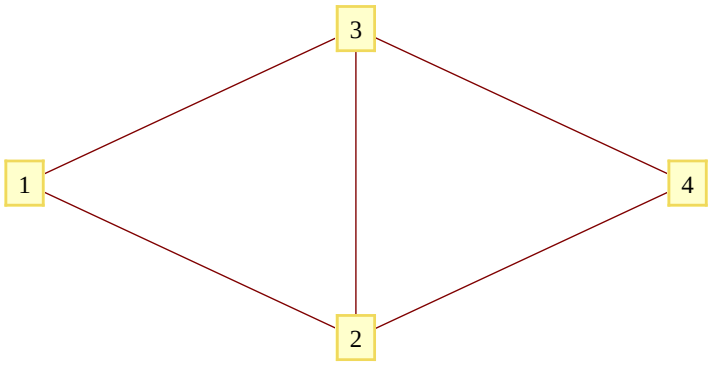

Figure 13: Simple graph

of the log-marginal around 0 evidence point. For instance, consider the derivative $\frac{\partial f_1}{\partial y_4}$, the corresponding sum will have 4 terms, visualized in figure 14

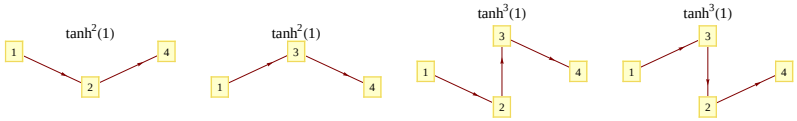

Figure 14: Self avoiding walks between nodes 1 and 4

Summing up the corresponding weights we get

$$\frac{\partial f_1}{\partial y_4} = \tanh^2 .1 + \tanh^2 .1 + \tanh^3 .1 + \tanh^3 .1 \approx 0.0218476 \tag{33}$$

Carrying out the procedure for all the variables we get the following linear expansion of the log-marginal at node 1 in terms of local potentials x_i for the graph 13

$$y_1 = x_1 + 0.110592x_2 + 0.110592x_3 + 0.0218476x_4 \tag{34}$$

x variables are local potentials, more specifically, x_i gives log-odds of state 1 over state 0 for node i conditioned on local evidence given all other potentials 0, for the graph 13

Note that the equation above is a linear expansion of log-odds around 0 evidence point (x's are 0) in the limit of weak loop interactions. I chose low edge-potential strength here to make this expansion method look better.

We can get exact linear expansion by considering exact formula for log-marginal of node 1 for the graph in Figure 13

$$\frac{1}{2}\text{Log}\left[\frac{e^{\frac{1}{10}+\text{x1}-\text{x2}-\text{x3}-\text{x4}}+e^{-\frac{1}{10}+\text{x1}+\text{x2}-\text{x3}-\text{x4}}+e^{-\frac{1}{10}+\text{x1}-\text{x2}+\text{x3}-\text{x4}}+e^{\frac{1}{10}+\text{x1}+\text{x2}+\text{x3}-\text{x4}}+e^{-\frac{3}{10}+\text{x1}-\text{x2}-\text{x3}+\text{x4}}+e^{-\frac{1}{10}+\text{x1}+\text{x2}-\text{x3}+\text{x4}}+e^{-\frac{1}{10}+\text{x1}-\text{x2}+\text{x3}+\text{x4}}+e^{\frac{1}{2}+\text{x1}+\text{x2}+\text{x3}+\text{x4}}}{e^{\frac{1}{2}-\text{x1}-\text{x2}-\text{x3}-\text{x4}}+e^{-\frac{1}{10}-\text{x1}+\text{x2}-\text{x3}-\text{x4}}+e^{-\frac{1}{10}-\text{x1}-\text{x2}+\text{x3}-\text{x4}}+e^{-\frac{3}{10}-\text{x1}+\text{x2}+\text{x3}-\text{x4}}+e^{\frac{1}{10}-\text{x1}-\text{x2}-\text{x3}+\text{x4}}+e^{-\frac{1}{10}-\text{x1}+\text{x2}-\text{x3}+\text{x4}}+e^{-\frac{1}{10}-\text{x1}-\text{x2}+\text{x3}+\text{x4}}+e^{\frac{1}{10}-\text{x1}+\text{x2}+\text{x3}+\text{x4}}}\right] \tag{35}$$

Another way of representing this quantity is using the self-avoiding walk representation as in previous section. The quantity in (35 is equivalent to)

$$\text{x1}+f\left[\text{x3}+f\left[\text{x4}+f\left[\frac{1}{5}+\text{x2}\right]\right]+f\left[\frac{1}{10}+\text{x2}+f\left[-\frac{1}{10}+\text{x4}\right]\right]\right]+f\left[\text{x2}+f[\text{x4}+f[\text{x3}]]+f\left[-\frac{1}{10}+\text{x3}+f\left[-\frac{1}{10}+\text{x4}\right]\right]\right] \tag{36}$$

Where $f(x) = \text{arctanh}(\tanh(0.1)\tanh(x))$. By explicitly taking first order Taylor expansion of either of the closed forms above around 0, we get

$$y_1 = x_1 + 0.110461x_2 + 0.110461x_3 + 0.0218022x_4 \tag{37}$$

Comparing this equation to 34 we see that approach based on self-avoiding walk generating function (Figure 14) gives almost the same result as one based on exact differentiation of the closed form. Also, coefficients in 34 are upper bounds on coeffients in 37

Obtaining linearized classifiers from equation for exact log-marginal as in (35) is intractable for large models. Obtaining approximate linearized classifiers from 25 is also intractable because self-avoiding walk enumeration is hard. However, self-avoiding walk representation suggests a natural sequence of approximations – instead of self-avoiding walks, use memory-k walks, which can be enumerated in polynomial time, and give valid upper bounds on both coefficients, and regret.

Using the approach of previous section, lets compute coefficients for our example using regular walks instead.

Correlation matrix (from 29) for this example is

$$\begin{pmatrix} 0 & \tanh 0.1 & \tanh 0.1 & 0 \\ \tanh 0.1 & 0 & \tanh 0.1 & \tanh 0.1 \\ \tanh 0.1 & \tanh 0.1 & 0 & \tanh 0.1 \\ 0 & \tanh 0.1 & \tanh 0.1 & 0 \end{pmatrix} \tag{38}$$

Spectral radius of this matrix is ≈ 0.255 so the bound based on regular walks will be non-trivial and Neumann series equality (**??**) holds. We get the following matrix of correlations

$$(I-A)^{-1} = \begin{pmatrix} 1.02309 & 0.115813 & 0.115813 & 0.0230856 \\ 0.115813 & 1.03567 & 0.126309 & 0.115813 \\ 0.115813 & 0.126309 & 1.03567 & 0.115813 \\ 0.0230856 & 0.115813 & 0.115813 & 1.02309 \end{pmatrix} \tag{39}$$

From this, we get the following linear expansion

$$y_1 = 1.023x_1 + 0.11581x_2 + 0.11581x_3 + 0.02308x_4 \tag{40}$$

Using formula for diameter of image after convolution (28) and formula above we can conclude that regret from dropping observations at node 4 is at most $0.02308 \cdot 2 \cdot \tanh 0.1 \approx 0.00460167$.

This approach can be extended to memory-k walks. We simply create a graph that corresponds to memory-k walks by creating nodes for nodes with different histories. For instance, for our example, memory-1 walk graph would be the one below. In Figure 15, we have a node for every directed edge, node $\{3,4\}$ represents a walker on state 4, which has visited state 3 in previous step. Note that this construction is the same as the directed line graph of 13 with backtracking edges removed.

Now we use the same approach as before to get

$$A = \begin{pmatrix} 0 & 0 & 0 & 0.099668 & 0.099668 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.099668 & 0.099668 & 0 & 0 \\ 0 & 0.099668 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.099668 & 0 & 0.099668 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.099668 \\ 0.099668 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.099668 & 0 & 0.099668 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.099668 & 0 \\ 0 & 0 & 0.099668 & 0.099668 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.099668 & 0.099668 & 0 & 0 & 0 \end{pmatrix} \tag{41}$$

Spectral radius of this matrix is 0.1516, lower than for matrix of regular walks and also below 1, so bound will be non-trivial. Repeating procedure as before, we get a 10x10 matrix. We can then add up correlations for equivalent states (ie, state 4 with history $\{3\}$ and state 4 with history $\{2\}$ will be collapsed) and we get the following linear expansion.

$$y_1 = 1.0011x_1 + 0.11102x_2 + 0.11102x_3 + 0.02213x_4 \tag{42}$$

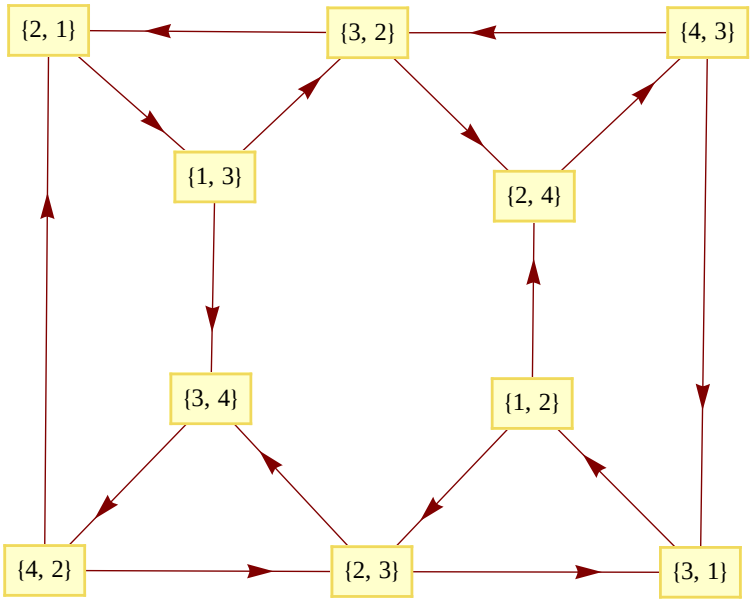

Figure 15: Graph for memory-1 walks on graph in Fig.13

Connections to belief propagation

Consider a binary graphical model with structure as in 13 with uniform edge potentials. Loopy belief propagation entails iterating the following set of equations until fixed point. We have a variable for each node and each directed edge

$$
\begin{aligned}
\text{m1} &\leftarrow \text{m12} + \text{m13} + \text{x1} \\
\text{m2} &\leftarrow \text{m21} + \text{m23} + \text{m24} + \text{x2} \\
\text{m3} &\leftarrow \text{m31} + \text{m32} + \text{m34} + \text{x3} \\
\text{m4} &\leftarrow \text{m42} + \text{m43} + \text{x4} \\
\text{m12} &\leftarrow f(\text{m23} + \text{m24} + \text{x2}) \\
\text{m13} &\leftarrow f(\text{m32} + \text{m34} + \text{x3}) \\
\text{m21} &\leftarrow f(\text{m13} + \text{x1}) \\
\text{m23} &\leftarrow f(\text{m31} + \text{m34} + \text{x3}) \\
\text{m24} &\leftarrow f(\text{m43} + \text{x4}) \\
\text{m31} &\leftarrow f(\text{m12} + \text{x1}) \\
\text{m32} &\leftarrow f(\text{m21} + \text{m24} + \text{x2}) \\
\text{m34} &\leftarrow f(\text{m42} + \text{x4}) \\
\text{m42} &\leftarrow f(\text{m21} + \text{m23} + \text{x2}) \\
\text{m43} &\leftarrow f(\text{m31} + \text{m32} + \text{x3})
\end{aligned}
\tag{43}
$$

Where $f(x)$ is the same contraction function used before, it depends on strength of potentials, for instance could be the same as in 36. The first four equations obtain the values of the log-marginals and have no effect on fixed point iteration. So the solution is obtained by iterating the last 10 equations to get fixed point value for edge variables, then substituting their values into equation for node variables.

You can view the fixed-point iteration process as consisting of two kinds of steps:

1. Addition: M43=m31+m32+x3
2. Convolution: m43=f(M43)

(44)

We can visualize the set of equations above by creating a graph with a node for each edge message, there will be an edge from message1 to message2 if message1 enters into the update equation for message2. The resulting graph is the same as the graph for non-backtracking (memory-1) walks in Figure (15). The equivalence of fixed point of Loopy belief propagation and root marginal of non-backtracking walk tree has been noted by Jordan/Tatikonda (2002) before.

The set of equations in 44 is non-linear because of convolution step which makes analysis harder. It is instructive to study a linearized version, which is close to exact when loop interactions and observation potentials are weak.

1. Addition: $M43 = m31 + m32 + x3$
2. Convolution: $m43 = M43 \cdot tanhJ$

(45)

The new set of equations is linear, hence we can represent one step linearized loopy-belief propagation update with a matrix multiplication. If $J = 0.1$, single step of this linearized belief propagation corresponds to the following equation

$$x \leftarrow Ax + o \tag{46}$$

Where A is the same matrix as the matrix of generating function for non-backtracking walks in (41). o is a vector of local potentials. Because this equation is linear, there's an easy condition on the convergence of this "linearized loopy belief propagation" – spectral radius of A has to be < 1.

Consider again the update matrix for linearized loopy belief propagation for the graph 13, letting J denote edge strength.

$$A = \begin{pmatrix} 0 & 0 & 0 & \tanh J & \tanh J & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \tanh J & \tanh J & 0 & 0 \\ 0 & \tanh J & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \tanh J & 0 & \tanh J & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \tanh J \\ \tanh J & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \tanh J & 0 & \tanh J & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \tanh J & 0 \\ 0 & 0 & \tanh J & \tanh J & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \tanh J & \tanh J & 0 & 0 & 0 \end{pmatrix} \tag{47}$$

Spectral radius of that matrix is $\tanh J \cdot r$ where r is the unique real root of equation $x^3 - x - 2 = 0$. This suggests that $J_c = \operatorname{arctanh}(1/r)$ is the critical point for this inference method. In fact, this is also the critical point for regular loopy belief propagation for this graph, see figures 16,17,18

Spectral radius condition in this example is a sufficient condition for the convergence of loopy belief propagation, although it's not a necessary condition since it ignores local potentials. This is the same as the condition given in (Mooij & Kappen, 2007).

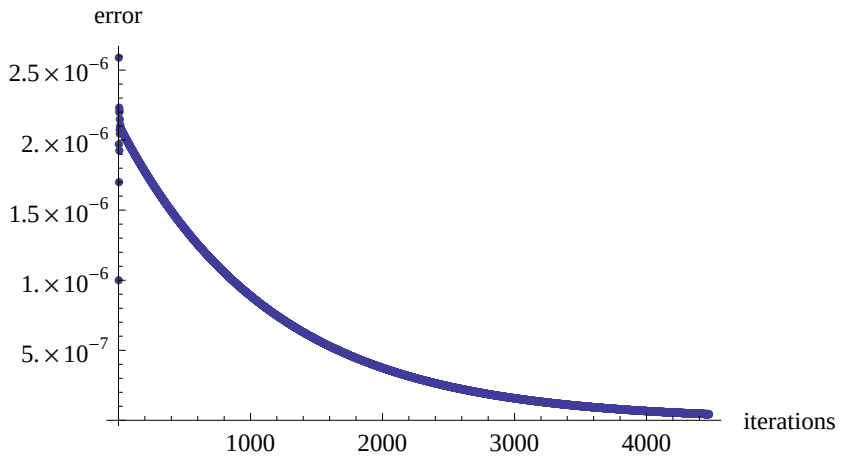

Figure 16: Behavior of error as a function of iterations for $J < J_c$

Summary

There are two technical challenges in obtaining tight bounds for graphical models – nonlinearity and inference complexity. For graphical models with strong observation potentials, dependence of marginal on observations is highly non-linear, for instance, in a chain, two strong observations can render the rest of the chain irrelevant. In a linear classifier, feature relevance doesn't depend on other features. Non-linearity also precludes a lot of simplification. Obtaining tight non-linear bounds for general graphical models can be done using approach similar to (Ihler, 2007). Basically the bounds are obtained in a way similar to belief propagation. This approach suggests an algorithm for bound calculation, rather than interpretable formulas. We can let graphical model take a particular form, ie a chain with uniform observation and interaction potentials, and come up with approximate formulas in terms of few variables, see first paper attached to this report.

Another approach is to look at bounds in the limit of weak interaction potentials. When potentials are weak enough, the log-odds of node marginals are almost linear functions of observations. Using self-avoiding walk tree representation of marginal, bounds then become values of the two-point correlation function for self-avoiding walks on the graph. Two point-correlation function for memory-k walks gives a looser upper bound which is easier to compute. Easiest to compute is a bound which is based on memory-1 walks. In fact this bound is tight for inference based on loopy-belief propagation. This suggests a connection between approximate inference and bounds – we can craft approximate inference methods

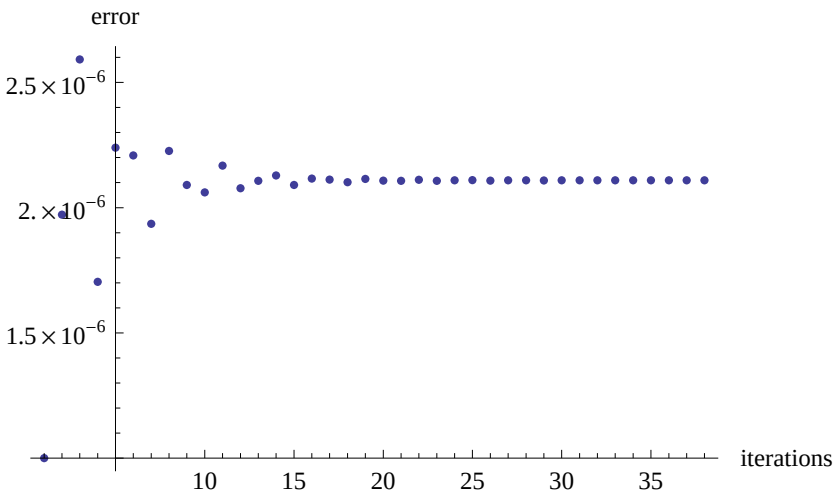

Figure 17: Behavior of error as a function of iterations for $J = J_c$

for which bound derived from memory-k walks is tight, and those methods will be more accurate than loopy belief propagation, see second attached paper.

References

Altham, P. M. E. The measurement of association of rows and columns for an r × s contingency table.

Atar, R., & Zeitouni, O. (1996). Exponential stability for nonlinear filtering.

Bapat, R. B., & Raghavan, T. E. S. (1997). *Nonnegative matrices and applications (encyclopedia of mathematics and its applications)*. Cambridge University Press.

Chan, H., & Darwiche, A. (2002). A distance measure for bounding probabilistic belief change. *Eighteenth national conference on Artificial intelligence* (pp. 539–545). Menlo Park, CA, USA: American Association for Artificial Intelligence.

Guttmann, A. J., & Conway, A. R. (2001). Square lattice self-avoiding walks and polygons. *Annals of Combinatorics*, *5*, 319–345.

Harpe, D. L. (1991). *On hilbert's metric for simplices*.

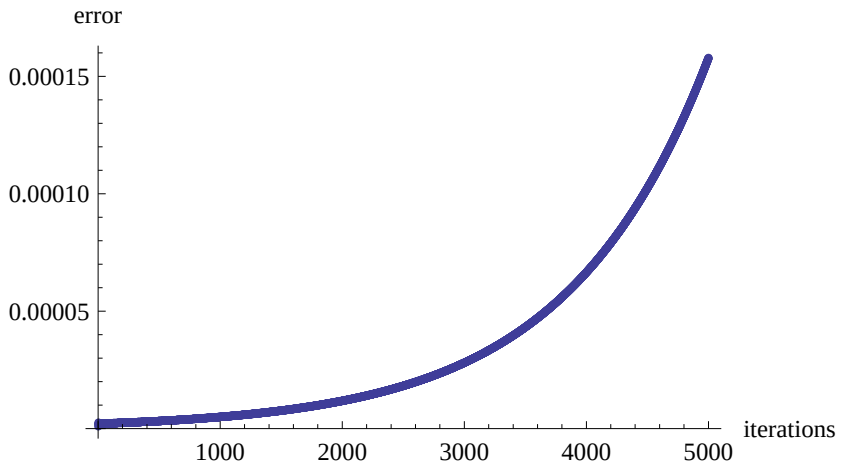

Figure 18: Behavior of error as a function of iterations for $J > J_c$

Hartfiel, D. J. (2002). *Nonhomogeneous matrix products*. World Scientific Publishing Company.

Ihler, A. (2007). Accuracy bounds for belief propagation. *Proceedings of UAI 2007*.

Kohlberg, E., & Pratt, J. W. (1982). The contraction mapping approach to the perron-frobenius theory: Why hilbert's metric? *Mathematics of Operations Research*, *7*, 198–210.

Liebetrau, A. M. (1983). *Measures of association (quantitative applications in the social sciences)*. Sage Publications, Inc.

Madras, N., & Slade, G. (1996). *The self-avoiding walk (probability and its applications)*. Birkhauser.

Meyer, C. D., & Meyer, C. (2001). *Matrix analysis and applied linear algebra*. Soc for Industrial & Applied Math.

Mooij, J. M., & Kappen, H. J. (2007). Sufficient conditions for convergence of the sum-product algorithm.

Tatikonda, S., & Jordan, M. (2002). Loopy belief propagation and gibbs measures.

Waser, N. M. (1986). Flower constancy: Definition, cause, and measurement. *The American Naturalist*, *127*, 593–603.

Weitz, D. (2006). Counting independent sets up to the tree threshold. *STOC '06: Proceedings of the thirty-eighth annual ACM symposium on Theory of computing* (pp. 140–149). New York, NY, USA: ACM.