Projecting Ising Model Parameters for Fast Mixing

Authored by:

Justin Domke
Xianghang Liu

Abstract

Inference in general Ising models is difficult, due to high treewidth making tree-based algorithms intractable. Moreover, when interactions are strong, Gibbs sampling may take exponential time to converge to the stationary distribution. We present an algorithm to project Ising model parameters onto a parameter set that is guaranteed to be fast mixing, under several divergences. We find that Gibbs sampling using the projected parameters is more accurate than with the original parameters when interaction strengths are strong and when limited time is available for sampling.

1   Paper Body

High-treewidth graphical models typically yield distributions where exact inference is intractable. To cope with this, one often makes an approximation based on a tractable model. For example, given some intractable distribution q, mean-field inference [14] attempts to minimize KL(p——q) over p ? TRACT, where TRACT is the set of fully-factorized distributions. Similarly, structured meanfield minimizes the KL-divergence, but allows TRACT to be the set of distributions that obey some tree [16] or a non-overlapping clustered [20] structure. In different ways, loopy belief propagation [21] and tree-reweighted belief propagation [19] also make use of tree-based approximations, while Globerson and Jaakkola [6] provide an approximate inference method based on exact inference in planar graphs with zero field. In this paper, we explore an alternative notion of a ?tractable? model. These are ?fast mixing? models, or distributions that, while they may be high-treewidth, have parameter-space conditions guaranteeing that Gibbs sampling will quickly converge to the stationary distribution. While the precise form of the parameter space conditions is slightly technical (Sections 2-3), informally, it is simply that interaction strengths between neighboring variables are not too strong. In the context of the Ising model, we attempt to use these models in the most basic way possible? by taking an

arbitrary (slow-mixing) set of parameters, projecting onto the fast-mixing set, using four different divergences. First, we show how to project in the Euclidean norm, by iteratively thresholding a singular value decomposition (Theorem 7). Secondly, we experiment with projecting using the ?zero-avoiding? divergence KL(q——p). Since this requires taking (intractable) expectations with respect to q, it is of only theoretical interest. Third, we suggest a novel ?piecewise? approximation of the KL divergence, where one drops edges from both q and p until a low-treewidth graph remains where the exact KL divergence can be calculated. Experimentally, this does not perform as well as the true KL-divergence, but is easy to evaluate. Fourth, we consider the ?zero forcing? divergence KL(q——p). Since this requires expectations with respect to p, which is constrained to be fast-mixing, it can be approximated by Gibbs sampling, and the divergence can be minimized through stochastic approximation. This can be seen as a generalization of mean-field where the set of approximating distributions is expanded from fully-factorized to fast-mixing. 1

2 Background The literature on mixing times in Markov chains is extensive, including a recent textbook [10]. The presentation in the rest of this section is based on that of Dyer et al. [4]. Given a distribution p(x), one will often wish to draw samples from it. While in certain cases (e.g. the Normal distribution) one can obtain exact samples, for Markov random fields (MRFs), one must generally resort to iterative Markov chain Monte Carlo (MCMC) methods that obtain a sample asymptotically. In this paper, we consider the classic Gibbs sampling method [5], where one starts with some configuration x, and repeatedly picks a node i, and samples x_i from $p(x_i —x_{?i})$. Under mild conditions, this can be shown to sample from a distribution that converges to p as t ? ?. It is common to use more sophisticated methods such as block Gibbs sampling, the Swendsen-Wang algorithm [18], or tree sampling [7]. In principle, each algorithm could have unique parameter-space conditions under which it is fast mixing. Here, we focus on the univariate case for simplicity and because fast mixing of univariate Gibbs is sufficient for fast mixing of some other methods [13]. Definition 1. Given two finite distributions p and q, the total variation distance —— ? ——T V is 1! ——p(X) ? q(X)——T V = —p(X = x) ? q(X = x)—. 2 x We need a property of a distribution that can guarantee fast mixing. The dependency R_{ij} of x_i on x_j is defined by considering two configurations x and x” , and measuring how much the conditional distribution of x_i can vary when $x_k = x”_k$ for all k $= j. Definition 2. Given a distribution p, the dependency matrix R is defined by $R_{ij} =$

max
x,x! :x?j =x!?j
——p(Xi —x?i) ? p(Xi —x”?i)——T V .

Given some threshold !, the mixing time is the number of iterations needed to guarantee that the total variation distance of the Gibbs chain to the stationary distribution is less than !. Definition 3. Suppose that {X t } denotes the sequence of random variables corresponding to running Gibbs sampling on some distribution p. The mixing time ? (!) is the minimum time t such that the total variation distance between X t and the stationary distribution is at most !. That

2

is, $\tau(\epsilon) = \min\{t : d(t) < \epsilon\}$, $d(t) = \max$ ——$P(X_t|X_0 = x) ? p(X)$——$_{TV}$
. x

Unfortunately, the mixing time can be extremely long, which makes the use of Gibbs sampling delicate in practice. For example, for the two-dimensional Ising model with zero field and uniform interactions, it is known that mixing time is polynomial (in the size of the grid) when the interaction strengths are below a threshold $?_c$, and exponential for stronger interactions [11]. For more general distributions, such tight bounds are not generally known, but one can still derive sufficient conditions for fast mixing. The main result we will use is the following [8]. Theorem 4. Consider the dependency matrix R corresponding to some distribution $p(X_1 , ..., X_n)$. For Gibbs sampling with random updates, if ——R——$_2 < 1$, the mixing time is bounded by $\tau(\epsilon) ?$

"n# n . ln $\frac{1}{?}$ ——R——$_2$!

Roughly speaking, if the spectral norm (maximum singular value) of R is less than one, rapid mixing will occur. A similar result holds in the case of systematic scan updates [4, 8]. Some of the classic ways of establishing fast mixing can be seen as special cases of this. For example, the Dobrushin $ criterion is that ——R——$_1 < 1$, which can be easier to verify in many cases, since ——R——$_1 = \max_j \sum_i$ —R_{ij} — does not require the computation of singular values. However, for symmetric matrices, it can be shown that ——R——$_2 ?$ ——R——$_1$, meaning the above result is tighter. 2

3 Mixing Time Bounds For variables $x_i ? \{?1, +1\}$, an Ising model is of the form $? ? \# \# ?_i x_i ? A(?, ?)? , p(x) = \exp ? ?_{ij} x_i x_j + i$

i,j

where $?_{ij}$ is the interaction strength between variables i and j, $?_i$ is the ?field? for variable i, and A ensures normalization. This can be seen as a member of the exponential family $p(x) = \exp (? ? f(x) ? A(?))$, where $f(x) = \{x_i x_j$ $?(i, j)\} ? \{x_i ?_i\}$ and ? contains both ? and ?. Lemma 5. For an Ising model, the dependency matrix is bounded by $R_{ij} ? \tanh$ —$?_{ij}$ — ? —$?_{ij}$ — Hayes [8] proves this for the case of constant ? and zero-field, but simple modifications to the proof can give this result. Thus, to summarize, an Ising model can be guaranteed to be fast mixing if the spectral norm of the absolute value of interactions terms is less than one.

4 Projection In this section, we imagine that we have some set of parameters ?, not necessarily fast mixing, and would like to obtain another set of parameters ? which are as close as possible to ?, but guaranteed to be fast mixing. This section derives a projection in the Euclidean norm, while Section 5 will build on this to consider other divergence measures. We will use the following standard result that states that given a matrix A, the closest matrix with a maximum spectral norm can be obtained by thresholding the singular values. Theorem 6. If A has a singular value decomposition $A = U S V^T$, and ——?——$_F$ denotes the Frobenius ! norm, then $B = \arg\min$ ——$A ? B$——$_F$ can be obtained as $B = U \hat{S} V^T$, where $S_{ii} = \min(S_{ii} , c_2)$. B:——B——$_2 ?_c$

We denote this projection by $B = ?_c [A]$. This is close to providing an algorithm for obtaining the closest set of Ising model parameters that obey a given spectral norm constraint. However, there are two issues. First, in general,

even if A is sparse, the projected matrix B will be dense, meaning that projecting will destroy a sparse graph structure. Second, this result constrains the spectral norm of B itself, rather than R = —B—, which is what needs to be controlled. The theorem below provides a dual method that fixed these issues. Here, we take some matrix Z that corresponds to the graph structure, by setting Zij = 0 if (i, j) is an edge, and Zij = 1 otherwise. Then, enforcing that B obeys the graph structure is equivalent to enforcing that Zij Bij = 0 for all (i, j). Thus, finding the closest set of parameters B is equivalent to solving min ——A ? B——F subject to ——D——2 ? c, Zij Dij = 0, D = —B—. (1) B,D

We find it convenient to solve this minimization by performing some manipulations, and deriving a dual. The proof of this theorem is provided in the appendix. To accomplish the maximization of g over M and ?, we use LBFGS-B [1], with bound constraints used to enforce that M ? 0. & The following theorem uses the ?triple dot product? notation of A ? B ? C = ij Aij Bij Cij . Theorem 7. Define R = —A—. The minimization in Eq. 1 is equivalent to the problem of maxM?0,? g(?, M), where the objective and gradient of g are, for D(?, M) = ?c [R+M ??'Z], g(?, M) =

1 ——D(?, M) ? R——2F + ? ? Z ? D(?, M) 2

dg = Z ' D(?, M) d? dg = D(?, M). dM 3

(2) (3) (4)

5 Divergences Again, we would like to find a parameter vector ? that is close to a given vector ?, but is guaranteed to be fast mixing, but with several notions of ?closeness? that vary in terms of accuracy and computational convenience. Formally, if ? is the set of parameters that we can guarantee to be fast mixing, and D(?, ?) is a divergence between ? and ?, then we would like to solve arg min D(?, ?).

(5)

???

As we will see, in selecting D there appears to be something of a trade-off between the quality of the approximation, and the ease of computing the projection in Eq. 5. In this section, we work with the generic exponential family representation p(x; ?) = exp(? ? f (x) ? A(?)).

We use ? to denote the mean value of f . By a standard result, this is equal to the gradient of A, i.e. ?(?) =

! x

p(x; ?)f (x) = ?A(?).

5.1 Euclidean Distance The simplest divergence is simply the l2 distance between the parameter vectors, D(?, ?) = ——? ? ?——2 . For the Ising model, Theorem 7 provides a method to compute the projection arg min??? ——?? ?——2 . While simple, this has no obvious probabilistic interpretation, and other divergences perform better in the experiments below. However, it also forms the basis of our projected gradient descent strategy for computing the projection in Eq. 5 under more general divergences D. Specifically, we will do this by iterating d 1. ? ” ? ? ? ? ? d? D(?, ?)

2. ? ? arg min??? ——? ” ? ?——2 for some step-size ?. In some cases, dD/d? can be calculated exactly, and this is simply projected gradient descent.

4

In other cases, one needs to estimate dD/d? by sampling from ?. As discussed below, we do this by maintaining a ?pool? of samples. In each iteration, a few Markov chain steps are applied with the current parameters, and then the gradient is estimated using them. Since the gradients estimated at each time-step are dependent, this can be seen as an instance of Ergodic Mirror Descent [3]. This guarantees convergence if the number of Markov chain steps, and the step-size ? are both functions of the total number of optimization iterations.

5.2 KL-Divergence Perhaps the most natural divergence to use would be the ?inclusive? KL-divergence D(?, ?) = KL(?——?) =

!

p(x; ?) log

x

p(x; ?) . p(x; ?)

(6)

This has the ?zero-avoiding? property [12] that ? will tend to assign some probability to all configurations that ? assigns nonzero probability to. It is easy to show that the derivative is dD(?, ?) = ?(?) ? ?(?), d?

(7)

where ?? = E? [f (X)]. Unfortunately, this requires inference with respect to both the parameter vectors ? and ?. Since ? will be enforced to be fast-mixing during optimization, one could approximate ?(?) by sampling. However, ? is presumed to be slow-mixing, making ?(?) difficult to compute. Thus, this divergence is only practical on low-treewidth ?toy? graphs. 4

5.3 Piecewise KL-Divergences Inspired by the piecewise likelihood [17] and likelihood approximations based on mixtures of trees [15], we seek tractable approximations of the KL-divergence based on tractable subgraphs. Our motivation is the the following: if ? and ? define the same distribution, then if a certain set of edges are removed from both, they should continue to define the same distribution1. Thus, given some graph T , we define the ?projection? ?(T) onto the tree such by setting all edge parameters to zero if not part of T . Then, given a set of graphs T , the piecewise KL-divergence is D(?, ?) = max KL(?(T)——?(T)). T

Computing the derivative of this divergence is not hard? one simply computes the KL-divergence for each graph, and uses the gradient as in Eq. 7 for the maximizing graph. There is some flexibility of selecting the graphs T . In the simplest case, one could simply select a set of trees (assuring that each edge is covered by one tree), which makes it easy to compute the KLdivergence on each tree using the sum-product algorithm. We will also experiment with selecting low-treewidth graphs, where exact inference can take place using the junction tree algorithm. 5.4 Reversed KL-Divergence We also consider the ?zero-forcing? KL-divergence D(?, ?) = KL(?——?) =
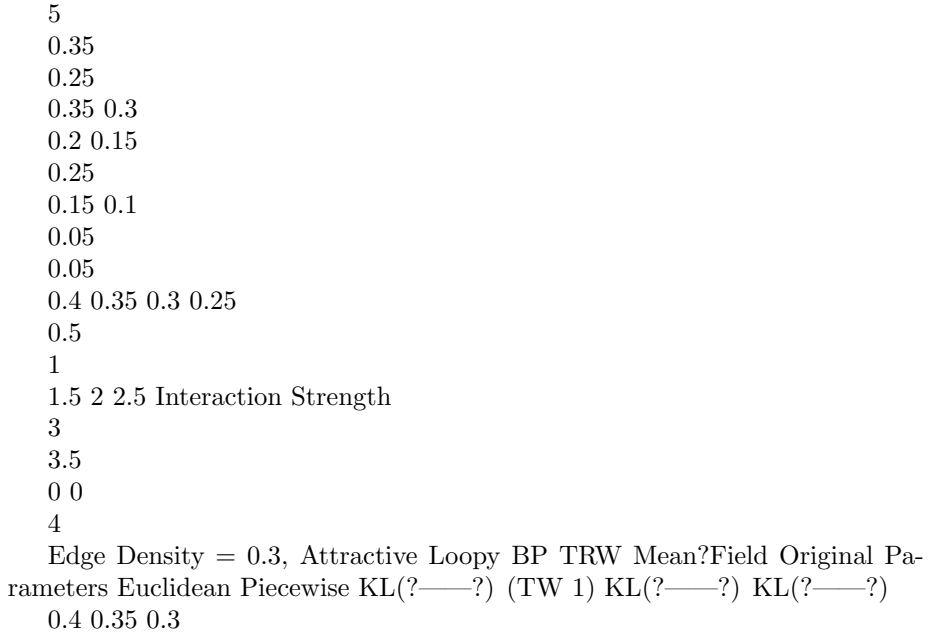
! x

p(x; ?) log

p(x; ?) . p(x; ?)

Theorem 8. The divergence D(?, ?) = KL(?——?) has the gradient ! d D(?, ?) = p(x; ?)(? ? ?) ? f (x) (f (x) ? ?(?)) . d? x Arguably, using this divergence

is inferior to the ?zero-avoiding? KL-divergence. For example, since the parameters ? may fail to put significant probability at configurations where ? does, using importance sampling to reweight samples from ? to estimate expectations with respect to ? could have high variance Further, it can be non-convex with respect to ?. Nevertheless, it often work well in practice. Minimizing this divergence under the constraint that the dependency matrix R corresponding to ? have a limited spectral norm is closely related to naive mean-field, which can be seen as a degenerate case where one constrains R to have zero norm. This is easier to work with than the ?zero-avoiding? KL-divergence in Eq. 6 since it involves taking expectations with respect to ?, rather than ?: since ? is enforced to be fast-mixing, these expectations can be approximated by sampling. Specifically, suppose that one has generated a set of samples x1 , ..., xK using the current parameters ?. Then, one can first approximate the marginals "K 1 k by ? ?=K k=1 f (x), and then approximate the gradient by g? =

K $# $ 1 !# (? ? ?) ? f (xk) f (xk) ? ? ? . K

(8)

k=1

It is a standard result that if two estimators are unbiased and independent, the product of the two estimators will also be unbiased. Thus, if one used separate sets of perfect samples to estimate ? ? and g?, then g? would be an unbiased estimator of dD/d?. In practice, of course, we generate the samples by Gibbs sampling, so they are not quite perfect. We find in practice that using the same set of samples twice makes makes little difference, and do so in the experiments. 1 Technically, here, we assume that the exponential family is minimal. However, in the case of an overcomplete exponential family, enforcing this will simply ensure that ? and ? use the same reparameterization.

5

0.35

0.25

0.35 0.3

0.2 0.15

0.25

0.15 0.1

0.05

0.05

0.4 0.35 0.3 0.25

0.5

1

1.5 2 2.5 Interaction Strength

3

3.5

0 0

4

Edge Density = 0.3, Attractive Loopy BP TRW Mean?Field Original Parameters Euclidean Piecewise KL(?——?) (TW 1) KL(?——?) KL(?——?)

0.4 0.35 0.3

0.2 0.15

0.25

0.05

0.05

1

1.5 2 2.5 Interaction Strength

3

3.5

0 0

4

1

1.5 2 2.5 Interaction Strength

3

3.5

4

3

3.5

4

Edge Density = 0.3, Mixed Loopy BP TRW Mean?Field Original Parameters Euclidean Piecewise KL(?——?) (TW 1) KL(?——?) KL(?——?)

0.15 0.1

0.5

0.5

0.2

0.1

0 0

Grid, Attractive LBP TRW Mean?Field Original Parameters Euclidean Piecewise KL(?——?) (TW 1) Piecewise KL(?——?) (TW 2) KL(?——?) KL(?——?)

0.2

0.1

0 0

Marginal Error

0.4

Marginal Error

Marginal Error

0.3

Grid, Mixed LBP TRW Mean?Field Original Parameters Euclidean Piecewise KL(?——?) (TW 1) Piecewise KL(?——?) (TW 2) KL(?——?) KL(?——?)

Marginal Error

0.4

0.5

1

1.5 2 2.5 Interaction Strength

Figure 1: The mean error of estimated univariate marginals on 8x8 grids (top row) and low-density random graphs (bottom row), comparing 30k iterations of Gibbs sampling after projection to variational methods. To approximate the

computational effort of projection (Table 1), sampling on the original parameters with 250k iterations is also included as a lower curve. (Full results in appendix.)

6 Experiments Our experimental evaluation follows that of Hazan and Shashua [9] in evaluating the accuracy of the methods using the Ising model in various configurations. In the experiments, we approximate randomly generated Ising models with rapid-mixing distributions using the projection algorithms described previously. Then, the marginals of rapid-mixing approximate distribution are compared against those of the target distributions by running a Gibbs chain on each. We calculate the mean absolute distance of the marginals as the accuracy measure, with the marginals computed via the exact junction-tree algorithm. We evaluate projecting under the Euclidean distance (Section 5.1), the piecewise divergence (Section 5.3), and the zero-forcing KL-divergence KL(?——?) (Section 5.4). On small graphs, it is possible to minimize the zero-avoiding KL-divergence KL(?——?) by computing marginals using the junctiontree algorithm. However, as minimizing this KL-divergence leads to exact marginal estimates, it doesn?t provide a useful measure of marginal accuracy. Our methods are compared with four other inference algorithms, namely loopy belief-propagation (LBP), Tree-reweighted belief-propagation (TRW), Naive mean-field (MF), and Gibbs sampling on the original parameters. LBP, MF and TRW are among the most widely applied variational methods for approximate inference. The MF algorithm uses a fully factorized distribution as the tractable family, and can be viewed as an extreme case of minimizing the zero forcing KL-divergence KL(?——?) under the constraint of zero spectral norm. The tractable family that it uses guarantees ?instant? mixing but is much more restrictive. Theoretically, Gibbs sampling on the original parameters will produce highly accurate marginals if run long enough. However, this can take exponentially long and convergence is generally hard to diagnose [2]. In contrast, Gibbs sampling on the rapid-mixing approximation is guaranteed to converge rapidly but will result in less accurate marginals asymptotically. Thus, we also include time-accuracy comparisons between these two strategies in the experiments. 6

	Grid, Strength 1.5		Grid, Strength 3		Random Graph, Strength 3	
	Gibbs Steps	SVDs	Gibbs Steps	SVDs	Gibbs Steps	SVDs
30,000 Gibbs steps	30k / 0.17s		30k / 0.17s		30k / 0.04s	
250,000 Gibbs steps	250k / 1.4s		250k / 1.4s		250k / 0.33s	
Euclidean Projection		22 / 0.04s		78 / 0.15s		17 / .0002s
Piecewise-1 Projection		322 / 0.61s		547 / 1.0s		408 / 0.047s
KL Projection	30k / 0.17s	265 / 0.55s	30k / 0.17s	471 / 0.94s	30k / 0.04s	300 / 0.037s

Table 1: Running Times on various attractive graphs, showing the number of Gibbs passes and Singular Value Decompositions, as well as the amount of computation time. The random graph is based on an edge density of 0.7. Mean-Field, Loopy BP, and TRW take less than 0.01s.

6.1 Configurations Two types of graph topologies are used: two-dimensional 8 ? 8 grids and random graphs with 10 nodes. Each edge is independently present with probability pe ? {0.3, 0.5, 0.7}. Node parameters ?i are uniformly drawn from unif(?dn , dn) and we fix the field strength to dn = 1.0. Edge parameters ?ij are uniformly drawn from unif(?de , de) or unif(0, de) to obtain mixed or attractive interactions respectively. We generate graphs with different

interaction strength de = 0, 0.5, . . . , 4. All results are averaged over 50 random trials. To calculate piecewise divergences, it remains to specify the set of subgraphs T . It can be any tractable subgraph of the original distribution. For the grids, one straightforward choice is to use the horizontal and the vertical chains as subgraphs. We also test with chains of treewidth 2. For random graphs, we use the sets of random spanning trees which can cover every edge of the original graphs as the set of subgraphs. A stochastic gradient descent algorithm is applied to minimize the zero forcing KL-divergence KL(?——?). In this algorithm, a ?pool? of samples is repeatedly used to estimate gradients as in Eq. 8. After each parameter update, each sample is updated by a single Gibbs step, consisting of one pass over all variables. The performance of this algorithm can be affected by several parameters, including the gradient search step size, the size of the sample pool, the number of Gibbs updates, and the number of total iterations. (This algorithm can be seen as an instance of Ergodic Mirror Descent [3].) Without intensive tuning of these parameters, we choose a constant step size of 0.1, sample pool size of 500 and 60 total iterations, which performed reasonably well in practice. For each original or approximate distribution, a single chain of Gibbs sampling is run on the final parameters, and marginals are estimated from the samples drawn. Each Gibbs iteration is one pass of systematical scan over the variables in fixed order. Note that this does not take into account the computational effort deployed during projection, which ranges from 30,000 total Gibbs iterations with repeated Euclidean projection (KL(?——?)) to none at all (Original parameters). It has been our experience that more aggressive parameters can lead to this procedure being more accurate than Gibbs in a comparison of total computational effort, but such a scheduling tends to also reduce the accuracy of the final parameters, making results more difficult to interpret. In Section 3.2, we show that for Ising models, a sufficient condition for rapid-mixing is the spectral norm of pairwise weight matrix is less than 1.0. However, we find in practice using a spectral norm bound of 2.5 instead of 1.0 can still preserve the rapid-mixing property and gives better approximation to the original distributions. (See Section 7 for a discussion.)

7 Discussion Inference in high-treewidth graphical models is intractable, which has motivated several classes of approximations based on tractable families. In this paper, we have proposed a new notion of ?tractability?, insisting not that a graph has a fast algorithm for exact inference, but only that it obeys parameter-space conditions ensuring that Gibbs sampling will converge rapidly to the stationary distribution. For the case of Ising models, we use a simple condition that can guarantee rapid mixing, namely that the spectral norm of the matrix of interaction strengths is less than one. 7

Grid, Interaction Strength 4.0, Mixed

0.4

0.3

0.4

0.3

0.2

0.2

0.1

0.1

0 0 10

1

10

2

3

4

10 10 10 Number of samples (log scale)

0 0 10

5

10

Edge Density = 0.3, Interaction Strength 3.0, Mixed

0.3

0.1

0.1

1

2

3

4

10 10 10 Number of samples (log scale)

4

5

10

LBP TRW Mean?Field Original Parameters Euclidean Piecewise KL(?——?) (TW 1) KL(?——?) KL(?——?)

0.3

0.2

10

3

0.4

0.2

0 0 10

2

10 10 10 Number of samples (log scale)

0.5

Marginal error

Marginal error

0.4

1

10

Edge Density = 0.3, Interaction Strength 3.0, Attractive

LBP TRW Mean?Field Original Parameters Euclidean Piecewise KL(?——?) (TW 1) KL(?——?) KL(?——?)

0.5

LBP TRW Mean?Field Original Parameters Euclidean Piecewise KL(?——?) (TW 1) Piecewise KL(?——?) (TW 2) KL(?——?) KL(?——?)

0.5

Marginal error

0.5

Marginal error

Grid, Interaction Strength 4.0, Attractive

LBP TRW Mean?Field Original Parameters Euclidean Piecewise KL(?——?) (TW 1) Piecewise KL(?——?) (TW 2) KL(?——?) KL(?——?)

0 0 10

5

10

1

10

2

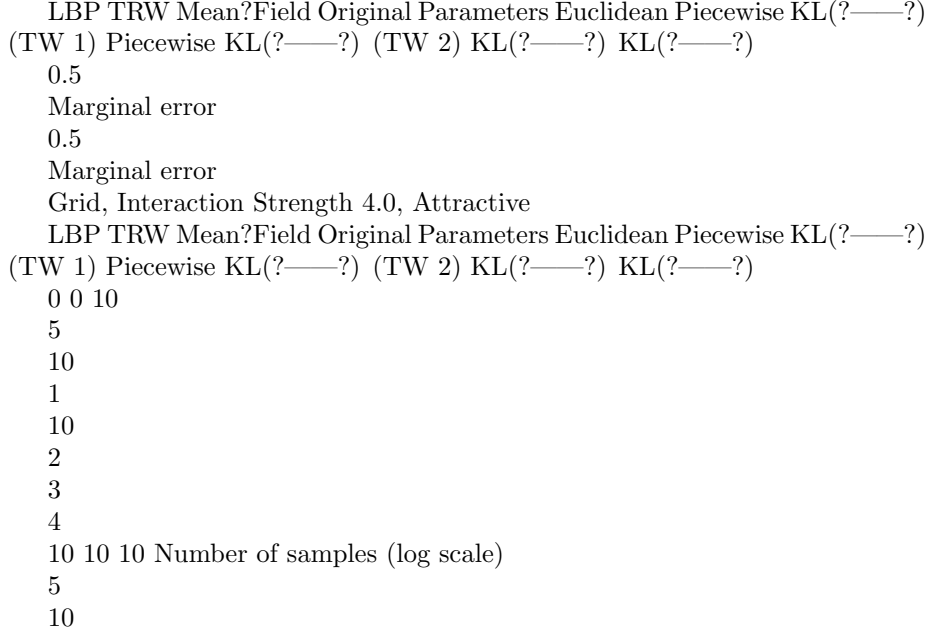3

4

10 10 10 Number of samples (log scale)

5

10

Figure 2: Example plots of the accuracy of obtained marginals vs. the number of samples. Top: Grid graphs. Bottom: Low-Density Random graphs. (Full results in appendix.) Given an intractable set of parameters, we consider using this approximate family by ?projecting? the intractable distribution onto it under several divergences. First, we consider the Euclidean distance of parameters, and derive a dual algorithm to solve the projection, based on an iterative thresholding of the singular value decomposition. Next, we extend this to more probabilistic divergences. Firstly, we consider a novel ?piecewise? divergence, based on computing the exact KL-divergnce on several low-treewidth subgraphs. Secondly, we consider projecting onto the KL-divergence. This requires a stochastic approximation approach where one repeatedly generates samples from the model, and projects in the Euclidean norm after taking a gradient step. We compare experimentally to Gibbs sampling on the original parameters, along with several standard variational methods. The proposed methods are more accurate than variational approximations. Given enough time, Gibbs sampling using the original parameters will always be more accurate, but with finite time, projecting onto the fast-mixing set to generally gives better results. Future work might extend this approach to general Markov random fields. This will require two technical challenges. First, one must find a bound on the dependency matrix for general MRFs, and secondly, an algorithm is needed to project onto the fast-mixing set defined by this bound. Fast-mixing distributions might also be used for learning. E.g., if one is doing maximum likelihood learning using MCMC to estimate the likelihood gradient, it would be natural to constrain the parameters to a fast mixing set. One weakness of the proposed approach is the apparent looseness of the spectral norm bound. For the two dimensional Ising model with no univariate terms, and a constant interaction strength ?, ?

there is a well-known threshold $?c = 12 \ln(1 + 2) ? .4407$, obtained using more advanced techniques than the spectral norm [11]. Roughly, for $? ¡ ?c$, mixing is known to occur quickly (polynomial in the grid size) while for $? ¿ ?c$, mixing is exponential. On the other hand, the spectral bound norm will be equal to one for $? = .25$, meaning the bound is too conservative in this case by a factor of $?c /.25 ? 1.76$. A tighter bound on when rapid mixing will occur would be more informative. 8

2   References

[1] Richard H. Byrd, Peihuang Lu, Jorge Nocedal, and Ciyou Zhu. A limited memory algorithm for bound constrained optimization. SIAM J. Sci. Comput., 16(5):1190?1208, 1995. [2] Mary Kathryn Cowles and Bradley P. Carlin. Markov chain monte carlo convergence diagnostics: A comparative review. Journal of the American Statistical Association, 91:883?904, 1996. [3] John C. Duchi, Alekh Agarwal, Mikael Johansson, and Michael I. Jordan. Ergodic mirror descent. SIAM Journal on Optimization, 22(4):1549?1578, 2012. [4] Martin E. Dyer, Leslie Ann Goldberg, and Mark Jerrum. Matrix norms and rapid mixing for spin systems. Ann. Appl. Probab., 19:71?107, 2009. [5] Stuart Geman and Donald Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. IEEE Trans. Pattern Anal. Mach. Intell., 6(6):721?741, 1984. [6] Amir Globerson and Tommi Jaakkola. Approximate inference using planar graph decomposition. In NIPS, pages 473?480, 2006. [7] Firas Hamze and Nando de Freitas. From fields to trees. In UAI, 2004. [8] Thomas P. Hayes. A simple condition implying rapid mixing of single-site dynamics on spin systems. In FOCS, pages 39?46, 2006. [9] Tamir Hazan and Amnon Shashua. Convergent message-passing algorithms for inference over general graphs with convex free energies. In UAI, pages 264?273, 2008. [10] David A. Levin, Yuval Peres, and Elizabeth L. Wilmer. Markov chains and mixing times. American Mathematical Society, 2006. [11] Eyal Lubetzky and Allan Sly. Critical Ising on the square lattice mixes in polynomial time. Commun. Math. Phys., 313(3):815?836, 2012. [12] Thomas Minka. Divergence measures and message passing. Technical report, 2005. [13] Yuval Peres and Peter Winkler. Can extra updates delay mixing? arXiv/1112.0603, 2011. [14] C. Peterson and J. R. Anderson. A mean field theory learning algorithm for neural networks. Complex Systems, 1:995?1019, 1987. [15] Patrick Pletscher, Cheng S. Ong, and Joachim M. Buhmann. Spanning Tree Approximations for Conditional Random Fields. In AISTATS, 2009. [16] Lawrence K. Saul and Michael I. Jordan. Exploiting tractable substructures in intractable networks. In NIPS, pages 486?492, 1995. [17] Charles Sutton and Andrew Mccallum. Piecewise training for structured prediction. Machine Learning, 77:165?194, 2009. [18] Robert H. Swendsen and Jian-Sheng Wang. Nonuniversal critical dynamics in monte carlo simulations. Phys. Rev. Lett., 58:86?88, Jan 1987. [19] Martin Wainwright, Tommi Jaakkola, and Alan Willsky. A new class of upper bounds on the log partition function. IEEE Transactions on Information Theory, 51(7):2313?2335, 2005.

[20] Eric P. Xing, Michael I. Jordan, and Stuart Russell. A generalized mean field algorithm for variational inference in exponential families. In UAI, 2003.

[21] Jonathan Yedidia, William Freeman, and Yair Weiss. Constructing free energy approximations and generalized belief propagation algorithms. IEEE Transactions on Information Theory, 51:2282?2312, 2005.

9