Appendix A. Proof of Convexity

Proposition 1. *The negative log pseudolikelihood in* (9) *is jointly convex in all the parameters* $\{\beta_{ss}, \beta_{st}, \alpha_s, \phi_{rj}, \rho_{sj}\}$ *over the region* $\beta_{ss} > 0$.

Proof To verify the convexity of $\tilde{\ell}(\Theta|x, y)$, it suffices to check that each term is convex. $-\log p(y_r|y_{\setminus r,}, x; \Theta)$ is jointly convex in ρ and ϕ since it is a multiclass logistic regression. We now check that $-\log p(x_s|x_{\setminus s}, y; \Theta)$ is convex. $-\frac{1}{2}\log \beta_{ss}$ is a convex function. To establish that $\frac{\beta_{ss}}{2}\left(\frac{\alpha_s}{\beta_{ss}} + \sum_j \frac{\rho_{sj}(y_j)}{\beta_{ss}} - \sum_{t\neq s}\frac{\beta_{st}}{\beta_{ss}}x_t - x_s\right)^2$ is convex, we use the fact that $f(u, v) = \frac{v}{2}(\frac{u}{v} - c)^2$ is convex. Let $v = \beta_{ss}$, $u = \alpha_s + \sum_j \rho_{sj}(y_j) - \sum_{t\neq s}\beta_{st}x_t$, and $c = x_s$. Notice that x_s, α_s, y_j, and x_t are fixed quantities and u is affinely related to β_{st} and ρ_{sj}. A convex function composed with an affine map is still convex, thus $\frac{\beta_{ss}}{2}\left(\frac{\alpha_s}{\beta_{ss}} + \sum_j \frac{\rho_{sj}(y_j)}{\beta_{ss}} - \sum_{t\neq s}\frac{\beta_{st}}{\beta_{ss}}x_t - x_s\right)^2$ is convex.

To finish the proof, we verify that $f(u, v) = \frac{v}{2}(\frac{u}{v} - c)^2 = \frac{1}{2}\frac{(u-cv)^2}{v}$ is convex over $v > 0$. The epigraph of a convex function is a convex set iff the function is convex. Thus we establish that the set $C = \{(u, v, t)|\frac{1}{2}\frac{(u-cv)^2}{v} \leq t, v > 0\}$ is convex. Let $A = \begin{bmatrix} v & u - cv \\ u - cv & t \end{bmatrix}$. The Schur complement criterion of positive definiteness says $A \succ 0$ iff $v > 0$ and $t > \frac{(u-cv)^2}{v}$. The condition $A \succ 0$ is a linear matrix inequality and thus convex in the entries of A. The entries of A are linearly related to u and v, so $A \succ 0$ is also convex in u and v. Therefore $v > 0$ and $t > \frac{(u-cv)^2}{v}$ is a convex set. ■

Appendix B. Sampling From The Joint Distribution

In this section we discuss how to draw samples $(x, y) \sim p(x, y)$. Using the property that $p(x, y) = p(y)p(x|y)$, we see that if $y \sim p(y)$ and $x \sim p(x|y)$ then $(x, y) \sim p(x, y)$. We have that

$$p(y) \propto \exp\left(\sum_{r,j}\phi_{rj}(y_r, y_j) + \frac{1}{2}\rho(y)^T B^{-1}\rho(y)\right) \tag{27}$$

$$(\rho(y))_s = \sum_j \rho_{sj}(y_j) \tag{28}$$

$$p(x|y) = No(B^{-1}(\alpha + \rho(y)), B^{-1}) \tag{29}$$

The difficult part is to sample $y \sim p(y)$ since this involves the partition function of the discrete MRF. This can be done with MCMC for larger models and junction tree algorithm or exact sampling for small models.

Appendix C. Maximum Likelihood

The difficulty in MLE is that in each gradient step we have to compute $\hat{T}(x, y) - E_{p(\Theta)}[T(x, y)]$, the difference between the empirical sufficient statistic $\hat{T}(x, y)$ and the expected sufficient statistic. In both continuous and discrete graphical models the computationally expensive step is evaluating $E_{p(\Theta)}\left[T(x,y)\right]$. In discrete problems, this involves a sum over the discrete state space and in continuous problem, this requires matrix inversion. For both discrete and continuous models, there has been much work on addressing these difficulties. For discrete models, the junction tree algorithm is an exact method for evaluating marginals and is suitable for models with low tree width. Variational methods such as belief propagation and tree reweighted belief propagation work by optimizing a surrogate likelihood function by approximating the partition function $Z(\Theta)$ by a tractable surrogate $\widetilde{Z}(\Theta)$ (Wainwright and Jordan, 2008). In the case of a large discrete state space, these methods can be used to approximate $p(y)$ and do approximate maximum likelihood estimation for the discrete model. Approximate maximum likelihood estimation can also be done via Monte Carlo estimates of the gradients $\hat{T}(x,y) - E_{p(\Theta)}(T(x,y))$. For continuous Gaussian graphical models, efficient algorithms based on block coordinate descent (Friedman et al., 2008b; Banerjee et al., 2008) have been developed, that do not require matrix inversion.

The joint distribution and loglikelihood are:

$$
\begin{aligned}
p(x,y;\Theta) &= \exp\,(-\frac{1}{2}x^T Bx + (\alpha + \rho(y))^T x + \sum_{(r,j)} \phi_{rj}(y_r, y_j))/Z(\Theta) \\
\ell(\Theta) &= \left(\frac{1}{2}x^T Bx - (\alpha + \rho(y))^T x - \sum_{(r,j)} \phi_{rj}(y_r, y_j)\right) \\
&\quad + \log(\sum_{y'} \int dx \exp\,(-\frac{1}{2}x^T Bx + (\alpha + \rho(y'))^T x) \exp(\sum_{(r,j)} \phi_{rj}(y'_r, y'_j)))
\end{aligned}
$$

The derivative is

$$
\begin{aligned}
\frac{\partial \ell}{\partial B} &= \frac{1}{2}xx^T + \frac{\int dx(\sum_{y'} -\frac{1}{2}xx^T \exp(-\frac{1}{2}x^T Bx + (\alpha + \rho(y))^T x + \sum_{(r,j)} \phi_{rj}(y'_r, y'_j)))}{Z(\Theta)} \\
&= \frac{1}{2}xx^T + \int \sum_{y'} (-\frac{1}{2}xx^T p(x,y';\Theta)) \\
&= \frac{1}{2}xx^T + \sum_{y'} \int -\frac{1}{2}xx^T p(x|y';\Theta)p(y') \\
&= \frac{1}{2}xx^T + \sum_{y'} \int -\frac{1}{2}\left(B^{-1} + B^{-1}(\alpha + \rho(y'))(\alpha + \rho(y')^T)B^{-1}\right) p(y')
\end{aligned}
$$

The primary cost is to compute B^{-1} and the sum over the discrete states y.
The computation for the derivatives of $\ell(\Theta)$ with respect to ρ_{sj} and ϕ_{rj} are similar.

$$
\begin{aligned}
\frac{\partial \ell}{\phi_{rj}(a,b)} &= -1(y_r = a, y_j = b) + \sum_{y'} \int dx 1(y'_r = a, y'_j = b) p(x,y';\Theta) \\
&= -1(y_r = a, y_j = b) + \sum_{y'} 1(y'_r = a, y'_j = b) p(y')
\end{aligned}
$$

The gradient requires summing over all discrete states.
Similarly for $\rho_{sj}(a)$:

$$\begin{aligned}\frac{\partial \ell}{\rho_{sj}(a)} &= -1(y_j = a)x_s + \sum_{y'} \int dx (1(y'_j = a)x_s) p(x', y'; \Theta) \\ &= -1(y_j = a)x_s + \int dx \sum_{y'_{\backslash j}} x_s p(x|y'_{\backslash j}, y'_j = a) p(y'_{\backslash j}, y'_j = a)\end{aligned}$$

MLE estimation requires summing over the discrete states to compute the expected sufficient statistics. This may be approximated using using samples $(x, y) \sim p(x, y; \Theta)$. The method in the previous section shows that sampling is efficient if $y \sim p(y)$ is efficient. This allows us to use MCMC methods developed for discrete MRF's such as Gibbs sampling.

Appendix D. Choosing the Weights

We first show how to compute w_{sj}. The gradient of the pseudo-likelihood with respect to a parameter $\rho_{sj}(a)$ is given below

$$\begin{aligned}\frac{\partial \tilde{\ell}}{\partial \rho_{sj}(a)} &= \sum_{i=1}^{n} -2 \times \mathbb{1}\left[y_j^i = a\right] x_s^i + E_{p_F}(\mathbb{1}[y_j = a] x_s | y^i_{\backslash j}, x^i) + E_{p_F}(\mathbb{1}[y_j = a] x_s | x^i_{\backslash s}, y^i) \\ &= \sum_{i=1}^{n} -2 \times \mathbb{1}\left[y_j^i = a\right] x_s^i + x_s^i p(y_j = a) + \mathbb{1}\left[y_j^i = a\right] \mu_s \\ &= \sum_{i=1}^{n} \mathbb{1}\left[y_j^i = a\right] \left(\hat{\mu}_s - x_s^i\right) + x_s^i \left(\hat{p}(y_j = a) - \mathbb{1}\left[y_j^i = a\right]\right) \\ &= \sum_{i=1}^{n} \left(\mathbb{1}\left[y_j^i = a\right] - \hat{p}(y_j = a)\right) \left(\hat{\mu}_s - x_s^i\right) + \left(x_s^i - \hat{\mu}_s\right) \left(\hat{p}(y_j = a) - \mathbb{1}\left[y_j^i = a\right]\right) \end{aligned} \tag{30}$$

$$= \sum_{i=1}^{n} 2 \left(\mathbb{1}\left[y_j^i = a\right] - \hat{p}(y_j = a)\right) \left(\hat{\mu}_s - x_s^i\right) \tag{31}$$

Since the subgradient condition includes a variable if $\left\| \frac{\partial \tilde{\ell}}{\partial \rho_{sj}} \right\| > \lambda$, we compute $E \left\| \frac{\partial \tilde{\ell}}{\partial \rho_{sj}} \right\|^2$. By independence,

$$E_{p_F}\left(\left\| \sum_{i=1}^{n} 2 \left(\mathbb{1}\left[y_j^i = a\right] - \hat{p}(y_j = a)\right) \left(\hat{\mu}_s - x_s^i\right) \right\|^2 \right) \tag{32}$$

$$= 4n E_{p_F}\left(\left\| \mathbb{1}\left[y_j^i = a\right] - \hat{p}(y_j = a) \right\|^2 \right) E_{p_F}\left(\left\| \hat{\mu}_s - x_s^i \right\|^2 \right) \tag{33}$$

$$= 4(n-1) p(y_j = a)(1 - p(y_j = a)) \sigma_s^2 \tag{34}$$

The last line is an equality if we replace the sample means $\hat{p}$ and $\hat{\mu}$ with the true values p and μ. Thus for the entire vector ρ_{sj} we have $E_{p_F} \left\| \frac{\partial \tilde{\ell}}{\partial \rho_{sj}} \right\|^2 = 4(n-1) \left(\sum_a p(y_j = a)(1 - p(y_j = a)) \right) \sigma_s^2$.

If we let the vector z be the indicator vector of the categorical variable y_j, and let the vector $p = p(y_j = a)$, then $E_{p_F} \left\| \frac{\partial \tilde{\ell}}{\partial \rho_{sj}} \right\|^2 = 4(n-1)\sum_a p_a(1-p_a)\sigma^2 = 4(n-1)\mathbf{tr}(\mathbf{cov}(z))\mathbf{var}(x)$ and $w_{sj} = \sqrt{\sum_a p_a(1-p_a)\sigma_s^2}$.

We repeat the computation for β_{st}.

$$
\begin{aligned}
\frac{\partial \ell}{\partial \beta_{st}} &= \sum_{i=1}^{n} -2x_s^i x_t + E_{p_F}(x_s^i x_t^i | x_{\backslash s}, y) + E_{p_F}(x_s^i x_t^i | x_{\backslash t}, y) \\
&= \sum_{i=1}^{n} -2x_s^i x_t^i + \hat{\mu}_s x_t^i + \hat{\mu}_t x_s^i \\
&= \sum_{i=1}^{n} x_t^i(\hat{\mu_s} - x_s^i) + x_s^i(\hat{\mu}_t - x_t^i) \\
&= \sum_{i=1}^{n} (x_t^i - \hat{\mu}_t)(\hat{\mu_s} - x_s^i) + (x_s^i - \hat{\mu_s})(\hat{\mu}_t - x_t^i) \\
&= \sum_{i=1}^{n} 2(x_t^i - \hat{\mu}_t)(\hat{\mu_s} - x_s^i)
\end{aligned}
$$

Thus

$$
\begin{aligned}
&E\left(\left\| \sum_{i=1}^{n} 2(x_t^i - \hat{\mu}_t)(\hat{\mu_s} - x_s^i) \right\|^2 \right) \\
&\quad = 4nE_{p_F}\left\| x_t - \hat{\mu}_t \right\|^2 E_{p_F}\left\| x_s - \hat{\mu}_s \right\|^2 \\
&\quad = 4(n-1)\sigma_s^2\sigma_t^2
\end{aligned}
$$

Thus $E_{p_F} \left\| \frac{\partial \ell}{\partial \beta_{st}} \right\|^2 = 4(n-1)\sigma_s^2\sigma_t^2$ and taking square-roots gives us $w_{st} = \sigma_s\sigma_t$. We repeat the same computation for ϕ_{rj}. Let $p_a = Pr(y_r = a)$ and $q_b = Pr(y_j = b)$.

$$
\begin{aligned}
\frac{\partial \tilde{\ell}}{\partial \phi_{rj}(a,b)} &= \sum_{i=1}^{n} -\mathbb{1}\left[y_r^i = a\right]\mathbb{1}\left[y_j^i = b\right] + E\left(\mathbb{1}[y_r = a]\mathbb{1}[y_j = b] | y_{\backslash r}, x\right) \\
&\quad + E\left(\mathbb{1}[y_r = a]\mathbb{1}[y_j = b] | y_{\backslash j}, x\right) \\
&= \sum_{i=1}^{n} -\mathbb{1}\left[y_r^i = a\right]\mathbb{1}\left[y_j^i = b\right] + \hat{p}_a\mathbb{1}\left[y_j^i = b\right] + \hat{q}_b\mathbb{1}\left[y_r^i = a\right] \\
&= \sum_{i=1}^{n} \mathbb{1}\left[y_j^i = b\right](\hat{p}_a - \mathbb{1}\left[y_r^i = a\right]) + \mathbb{1}\left[y_r^i = a\right](\hat{q}_b - \mathbb{1}\left[y_j^i = b\right]) \\
&= \sum_{i=1}^{n} (\mathbb{1}\left[y_j^i = b\right] - \hat{q}_b)(\hat{p}_a - \mathbb{1}\left[y_r^i = a\right]) + (\mathbb{1}\left[y_r^i = a\right] - \hat{p}_a)(\hat{q}_b - \mathbb{1}\left[y_j^i = b\right]) \\
&= \sum_{i=1}^{n} 2(\mathbb{1}\left[y_j^i = b\right] - \hat{q}_b)(\hat{p}_a - \mathbb{1}\left[y_r^i = a\right])
\end{aligned}
$$

Thus we compute

$$\begin{aligned} E_{p_F} \left\| \frac{\partial \tilde{\ell}}{\partial \phi_{rj}(a,b)} \right\|^2 &= E\left(\left\| \sum_{i=1}^{n} 2(\mathbb{1}\left[y_j^i = b\right] - \hat{q}_b)(\hat{p}_a - \mathbb{1}\left[y_r^i = a\right]) \right\|^2 \right) \\ &= 4nE_{p_F} \left\| \hat{q}_b - \mathbb{1}[y_j = b] \right\|^2 E_{p_F} \left\| \hat{p}_a - \mathbb{1}[y_r = a] \right\|^2 \\ &= 4(n-1)q_b(1-q_b)p_a(1-p_a) \end{aligned}$$

From this, we see that $E_{p_F} \left\| \frac{\partial \tilde{\ell}}{\partial \phi_{rj}} \right\|^2 = \sum_{a=1}^{L_r} \sum_{b=1}^{L_j} 4(n-1)q_b(1-q_b)p_a(1-p_a)$ and $w_{rj} = \sqrt{\sum_{a=1}^{L_r} \sum_{b=1}^{L_j} q_b(1-q_b)p_a(1-p_a)}$.