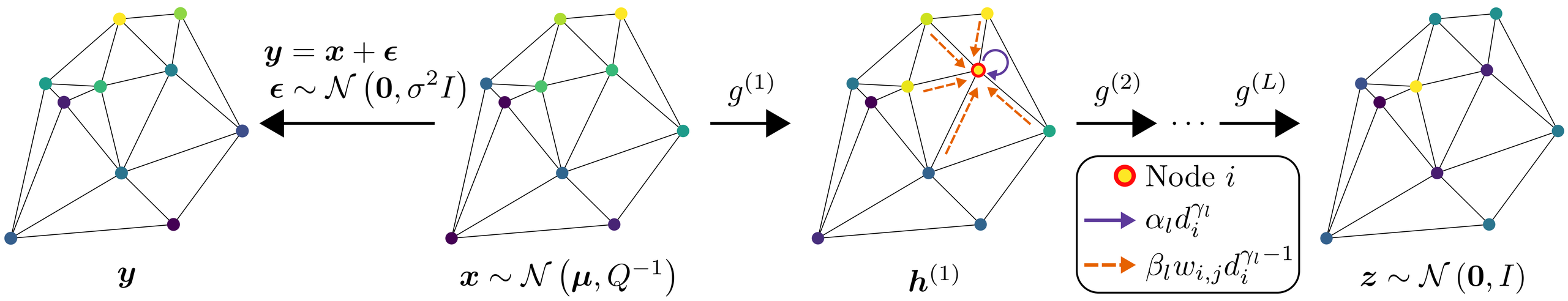
Scalable Deep Gaussian Markov Random Fields for General Graphs

Joel Oskarsson[1] Per Sidén[1 2] Fredrik Lindsten[1]

[1]Division of Statistics and Machine Learning, Department of Computer and Information Science, Linköping University, Linköping, Sweden
[2]Arriver Software AB



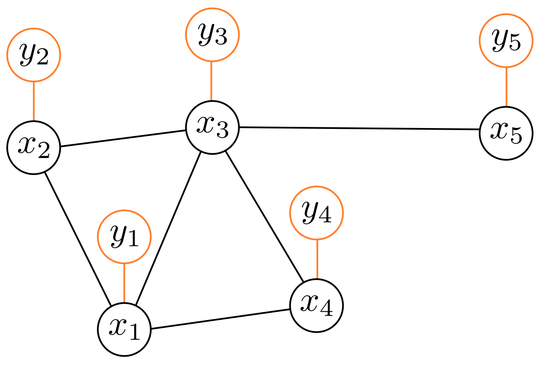
A Layered Gaussian Model on Graphs

We propose a flexible Gaussian Markov Random Field (GMRF) model for general graphs built on the multi-layer structure of Deep GMRFs. By designing a new type of layer we enable the model to scale to large graphs. The layer allows for efficient training using variational inference and existing software frameworks for Graph Neural Networks. For a Gaussian likelihood, close to exact Bayesian inference is available for the latent field. In experiments on synthetic and real world datasets our model compares favorably to other both Bayesian and deep learning methods.

Background: Gaussian Markov Random Fields

Gaussian Markov Random Fields (GMRFs)

- Graphical models where the nodes $\boldsymbol{x}$ are jointly Gaussian [2]
- Conjugate prior to Gaussian likelihood $p(\boldsymbol{y}|\boldsymbol{x}) = \mathcal{N}(\boldsymbol{y}|\boldsymbol{x}, \sigma^2 I)$
  - Posterior $p(\boldsymbol{x}|\boldsymbol{y})$ analytically tractable, also a GMRF!



Deep GMRFs (DGMRFs) [3]

- GMRF $\boldsymbol{x}$ defined implicitly by affine map g

$$\boldsymbol{z} = g(\boldsymbol{x}) = G\boldsymbol{x} + \boldsymbol{b}, \quad \boldsymbol{z} \sim \mathcal{N}(\mathbf{0}, I) \tag{1}$$

- g defined as a combination of L simpler layers $g = g^{(L)} \circ g^{(L-1)} \circ \cdots \circ g^{(1)}$
- Originally restricted to lattice graphs (image-structured data)

Method: Graph DGMRF

- Consider graph $\mathcal{G}$ with adjacency matrix A and degree matrix $D = \mathrm{diag}([d_1, d_2, \ldots, d_N]^\intercal)$
- Define a layer construct

$$\boldsymbol{h}^{(l)} = g^{(l)}(\boldsymbol{h}^{(l-1)}) = G^{(l)}\boldsymbol{h}^{(l-1)} + b_l \mathbf{1} \tag{2}$$
$$G^{(l)} = \alpha_l D^{\gamma_l} + \beta_l D^{\gamma_l - 1} A \tag{3}$$

  - b_l, α_l, β_l and γ_l are trainable parameters

- Closely corresponds to Graph Neural Network (GNN) formulation. Existing frameworks can be utilized for automatic differentiation and GPU acceleration.
- Variational training, maximizing the Evidence Lower Bound (ELBO)
- Training requires efficient evaluation of $\log\left|\det\left(G^{(l)}\right)\right|$. We develop two solutions:
  - Exact method based on pre-computing the eigenvalues of $D^{-1}A$
  - Scalable approximate method based on reformulating log-determinant as a power series [1]

Experiments

- Experiments on prediction for a subset of unobserved nodes. We train on a fraction of the nodes and evaluate on remaining ones.

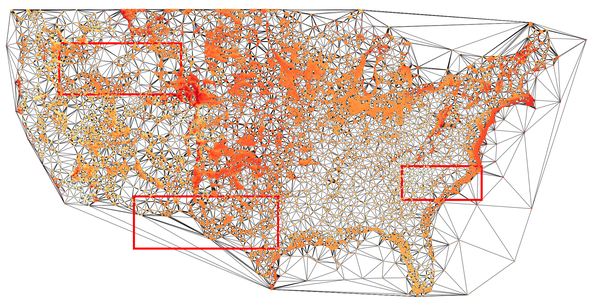
Wikipedia Graph

- 11 631 Wikipedia pages as nodes
- Edges are links between pages
- Target $\boldsymbol{y}$ is log of average monthly traffic
- CRPS, probabilistic metric taking uncertainty into account

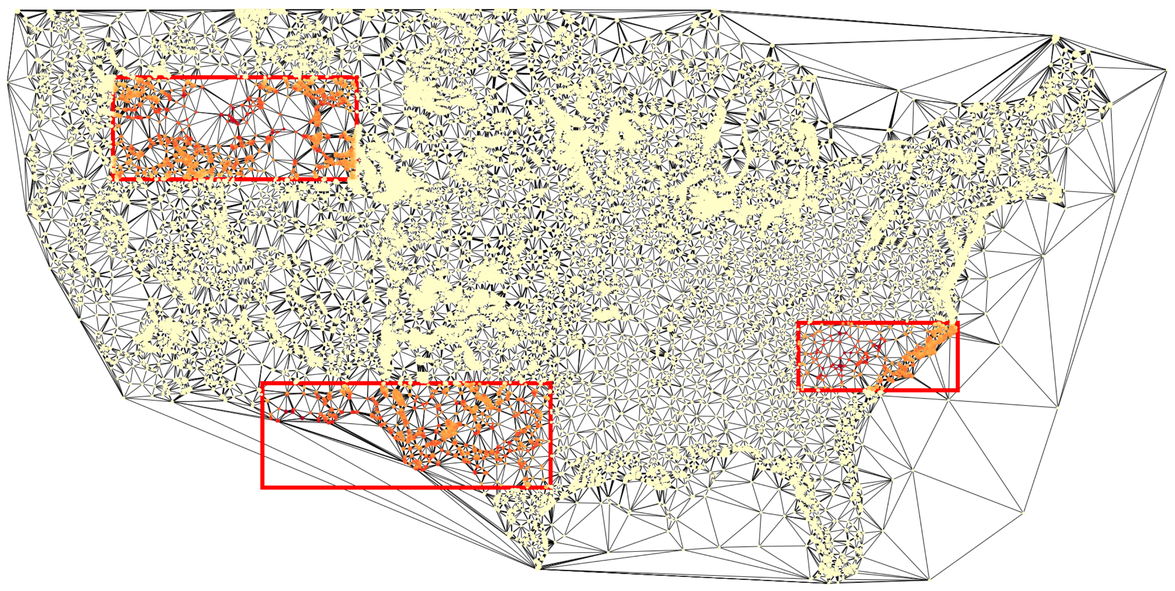
	RMSE	CRPS
Baselines		
Graph GP	2.169	1.251
DGP (GNN)	1.308	0.786
I-GMRF	1.526	0.939
DGMRF		
1 Layer	1.311	0.704
3 Layer	1.228	0.652
5 Layer	**1.169**	**0.614**

Wind Speed Dataset

- Spatial graph with 126 652 nodes
- Target $\boldsymbol{y}$ is average wind speed
- Nodes inside red rectangles are unobserved



Posterior Mean



Posterior marginal std.-dev.

More Information

Code, Link to Paper:
`github.com/joeloskarsson/graph-dgmrf`

Correspondence to:
Joel Oskarsson, `joel.oskarsson@liu.se`

References

[1] J. Behrmann, W. Grathwohl, et al. Invertible residual networks. In *Proceedings of the 36th International Conference on Machine Learning*, 2019.

[2] H. Rue and L. Held. *Gaussian Markov random fields: theory and applications*. Number 104 in Monographs on statistics and applied probability. Chapman & Hall/CRC, 2005.

[3] P. Sidén and F. Lindsten. Deep gaussian markov random fields. In *Proceedings of the 37th International Conference on Machine Learning*, 2020.

Scalable Deep Gaussian Markov Random Fields for General Graphs

Joel Oskarsson[1] Per Sidén[1 2] Fredrik Lindsten[1]

[1]Division of Statistics and Machine Learning, Department of Computer and Information Science, Linköping University, Linköping, Sweden
[2]Arriver Software AB

Variational Training

- Directly maximizing the log marginal likelihood $\log p(\boldsymbol{y}_m|\theta)$ is not computationally feasible for large graphs
- Maximizing the ELBO allows for scalable and efficient learning of the model parameters
- Let $\boldsymbol{m} \in \{0,1\}^N$ be an observation mask with 1 at observed nodes

$$M = \sum_{i=1}^{N} m_i \qquad \boldsymbol{y}_m = \boldsymbol{y} \odot \boldsymbol{m} \qquad I_m = \operatorname{diag}(\boldsymbol{m})$$

$$\begin{aligned}\operatorname{ELBO}(\theta,\phi) = &-\frac{1}{2}\mathbb{E}_{q(\boldsymbol{x}|\boldsymbol{\phi})}\left[g(\boldsymbol{x})^\intercal g(\boldsymbol{x}) + \frac{1}{\sigma^2}(\boldsymbol{y}_m - \boldsymbol{x})^\intercal I_m (\boldsymbol{y}_m - \boldsymbol{x})\right] \\ &+ \sum_{l=1}^{L} \log\left|\det\left(G^{(l)}\right)\right| + \mathrm{H}[q(\boldsymbol{x}|\phi)] - M\log\sigma + \text{const.}\end{aligned} \tag{4}$$

Variational Distribution

- Gaussian variational distribution $q(\boldsymbol{x}|\phi) = \mathcal{N}(\boldsymbol{x}|\boldsymbol{\nu}, SS^\intercal)$
- Defined in opposite direction of DGMRF

$$\boldsymbol{x} = S\boldsymbol{r} + \boldsymbol{\nu}, \quad \boldsymbol{r} \sim \mathcal{N}(\boldsymbol{0}, I) \tag{5}$$

- [3] use mean-field approximation (diagonal S)
- We can re-use our layer construct through $S = \operatorname{diag}(\boldsymbol{\xi})\tilde{G}\operatorname{diag}(\boldsymbol{\tau})$, where $\tilde{G}$ is one or more layers as defined in Eq. 3.
  - Introduces off-diagonal elements in the covariance matrix of q

Computing the Log-Determinant

Eigenvalue Method

$$\log\left|\det\left(G^{(l)}\right)\right| = \sum_{i=1}^{N} \gamma_l \log(d_i) + \log|\alpha_l + \beta_l \lambda_i'| \tag{6}$$

- $\{\lambda_i'\}_{i=1}^N$ are the eigenvalues of $D^{-1}A$
  - Depends only on graph structure $\Rightarrow$ Can be pre-computed!
- Exact, but does not scale to massive graphs

Power Series Method

- With $\tilde{A} = D^{-\frac{1}{2}}AD^{-\frac{1}{2}}$

$$\log\left|\det\left(G^{(l)}\right)\right| = N\log(\alpha_l) + \sum_{i=1}^{N}\gamma_l\log(d_i) + \log\left|\det\left(I + \frac{\beta_l}{\alpha_l}\tilde{A}\right)\right| \tag{7}$$

- Formulate last log-determinant as power series [1]

$$\log\left|\det\left(I + \frac{\beta_l}{\alpha_l}\tilde{A}\right)\right| = \sum_{k=1}^{\infty} -\frac{1}{k}\left(-\frac{\beta_l}{\alpha_l}\right)^k \operatorname{Tr}\left(\tilde{A}^k\right). \tag{8}$$

- Truncate at some large $k = K$ during training
  - $\operatorname{Tr}\left(\tilde{A}^k\right)$ can be pre-computed for $k \in \{1,\ldots,K\}$
- Approximate, but highly scalable

Acknowledgements

This research is financially supported by the Swedish Research Council via the project *Handling Uncertainty in Machine Learning Systems* (contract number: 2020-04122), the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation, and the Excellence Center at Linköping–Lund in Information Technology (ELLIIT).

Posterior Inference

- GMRF prior is conjugate to Gaussian likelihood $p(\boldsymbol{y}|\boldsymbol{x}) = \mathcal{N}\left(\boldsymbol{y}\middle|\boldsymbol{x}, \sigma^2 I\right)$

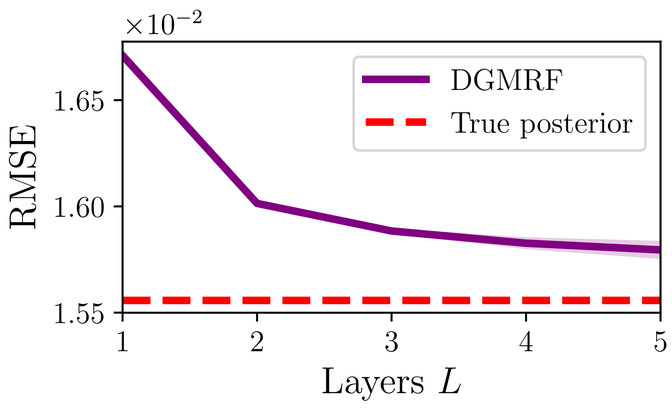
Prior	**Posterior**
$\boldsymbol{x} \sim \mathcal{N}\left(\boldsymbol{\mu}, Q^{-1}\right)$	$\boldsymbol{x}\|\boldsymbol{y}_m \sim \mathcal{N}\left(\tilde{\boldsymbol{\mu}}, \tilde{Q}^{-1}\right)$
$\boldsymbol{\mu} = -G^{-1}\boldsymbol{b}$	$\tilde{\boldsymbol{\mu}} = \tilde{Q}^{-1}\left(Q\boldsymbol{\mu} + \frac{1}{\sigma^2}\boldsymbol{y}_m\right)$
$Q = G^\intercal G$	$\tilde{Q} = Q + \frac{1}{\sigma^2}I_m$

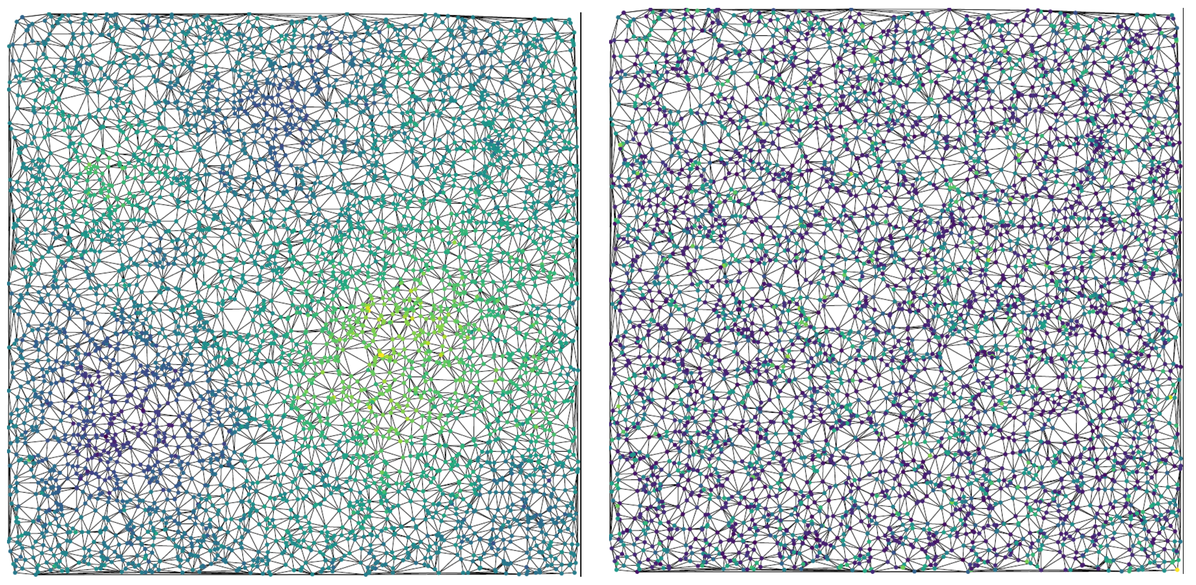
- Conjugate Gradient method used to avoid computing inverse in $\tilde{\boldsymbol{\mu}}$

Additional Experiments

Synthetic GMRF Dataset

- Sample from known GMRF created as random mix of multiple DGMRFs
- 1-5 layer DGMRFs trained and evaluated





Posterior mean | Posterior marginal std.-dev.

California Housing Dataset

- Data on 20 640 housing blocks with socio-economic features
- Target $\boldsymbol{y}$ is log of median house value in block
- Spatial graph with edges weighted by inverse distances
- We consider the dataset both with and without node features
  - In **No features** the methods use only the graph/spatial coordinates
  - In DGMRFs node features are handled using an auxiliary Bayesian linear model

	No features		**Features**	
Model	RMSE	CRPS	RMSE	CRPS
Bayes LR	13.319	7.521	8.872	4.834
MLP	11.086	7.915	7.094	4.525
GCN	8.760	5.683	6.837	4.273
GAT	9.166	6.049	6.788	4.348
SVGP	10.172	5.689	7.287	3.930
Graph GP	11.202	6.350	-	
IGMRF	6.989	3.841	-	
DGMRF, $L = 1$	6.909	3.665	5.894	3.078
DGMRF, $L = 2$	**6.853**	**3.651**	5.810	3.041
DGMRF, $L = 3$	**6.853**	3.656	**5.804**	**3.039**