Bayesian Convolutional Networks-based Generalized Linear Model

Yeseul Jeon, Won Chang, Seonghyun Jeong, Jaewoo Park

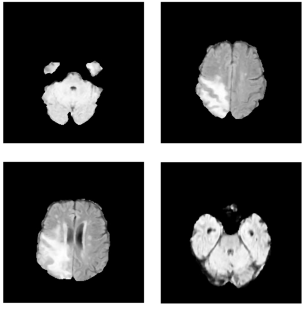
September 24, 2024

Overview

1. Research problems
2. Model
3. Applications
4. Summary

How to model the different types of data?

- High-dimensional correlated structured data
  - fMRI correlation matrix, MRI, and spatial basis function matrix
- Standard vector-type variables.
  - Demographic information (weight, age, gender, surgical history, etc), texture-based features, and environmental variables



Variable	Category	Frequency (f)	Percentage (%)
Race	African	396	29.1
	Coloured	183	13.4
	Indian	125	9.2
	White	658	48.3
Gender	Female	407	29.9
	Male	955	70.1
Age (in years)	0–19	0	0.0
	20–29	106	7.8
	30–39	406	29.8
	40–49	563	41.3
	50–59	276	20.3
	60–79	11	0.8
Occupational group	Manager	65	4.8
	Information technology	89	6.5
	Technicians	605	44.4
	Sales	238	17.5
	Supervisory	222	16.3
	Clerical or admin	143	10.5

n = 1362.

Statistical models

- Generalized linear models (GLMs), which estimate coefficients of covariates
- Spatial-temporal models or random-effect models, which consider data dependency
- **Limitations:**
  - Hard to directly model the correlated structured dataset (tensor)
  - Dimension issue
  - Adequate covariance structure (high computation cost)

Deep learning models

- **Convolutional neural networks (CNNs)**
  - Convolution layer: trains important neighborhood features from the input by shifting the kernels over all pixel locations with a certain step size (stride)
- **Limitations:**
  1. Uncertainty quantification
  2. Interpretation of covariates
  3. Stochastic gradient descent (SGD) algorithm is based on prediction accuracy (not on convergence in parameter estimation)

Research goal

- Study different types of variables simultaneously in various applications
- Estimate the coefficient of covariates
- Quantify the uncertainty in estimation and prediction
  - Posterior distribution of coefficient
  - Predictive distribution

Notations

- Dataset: $\mathbf{D} = \{(\mathbf{x}_n, \mathbf{y}_n), n = 1, \cdots, N\}$
- Layers: L layers, where the lth layer has k_l nodes for $l = 1, \cdots, L$
- A set of parameters $\boldsymbol{\theta}$: $(\mathbf{W}_l, \mathbf{b}_l)$
  - Weight matrix: $\mathbf{W}_l \in \mathbb{R}^{k_l \times k_{l-1}}$
  - Bias vector: $\mathbf{b}_l \in \mathbb{R}^{k_l}$

Neural network with dropout $\mathbf{d}_l$

$$\mathbf{o}_n = \sigma_L\Big(\mathbf{W}_L\sigma_{L-1}\Big(\cdots\sigma_3\Big(\mathbf{W}_3\sigma_2\Big(\mathbf{W}_2\sigma_1\Big(\mathbf{W}_1\mathbf{x}_n+\mathbf{b}_1\Big)\circ\mathbf{d}_2+\mathbf{b}_2\Big)\circ\mathbf{d}_3+\mathbf{b}_3\Big)\cdots\Big)\circ\mathbf{d}_L+\mathbf{b}_L\Big), \quad (1)$$

- $\sigma_l(\cdot)$: an activation function (ReLu,Sofrplus..)
- $\mathbf{d}_l \in \mathbb{R}^{k_l} \sim$ Bernoulli(ψ_l): Dropout (Srivastava et al., 2014)
- $\mathbf{f}_{n,0} = \mathbf{x}_n \in \mathbb{R}^{k_0}, \mathbf{f}_{n,1} = \mathbf{W}_1\mathbf{x}_n + \mathbf{b}_1 \in \mathbb{R}^{k_1}$, and $\mathbf{f}_{n,l} = \mathbf{W}_l\phi_{n,l-1} + \mathbf{b}_l \in \mathbb{R}^{k_l}$, $l \geq 2$
- Nonlinear output feature from the lth layer: $\phi_{n,l} = \sigma_l(\mathbf{f}_{n,l}) \in \mathbb{R}^{k_l}$

Neural networks as a deep Gaussian process

- Deep Gaussian process (Deep GP) (Damianou and Lawrence, 2013)
- $\mathbf{F}_l = \{\mathbf{f}_{n,l}\}_{n=1}^N \in \mathbb{R}^{N \times k_l}$ and $\mathbf{F}_l^{(k)}$ $(k = 1, \cdots, k_l)$ is the kth column of $\mathbf{F}_l$

$$\begin{aligned} \mathbf{F}_l^{(k)} | \mathbf{F}_{l-1} &\sim N(0, \widehat{\mathbf{\Sigma}}_l), \quad l = 2, \ldots, L \\ \mathbf{y}_n | \mathbf{f}_{n,L-1} &\sim p(\mathbf{y}_n | \mathbf{f}_{n,L-1}), \end{aligned} \tag{2}$$

- Empirical covariance matrix $\widehat{\mathbf{\Sigma}}_l \in \mathbb{R}^{N \times N}$ is

$$\widehat{\mathbf{\Sigma}}_l = \frac{1}{k_l} \sigma_l(\mathbf{\Phi}_{l-1} \mathbf{W}_l^\top + \mathbf{b}_l) \sigma_l(\mathbf{\Phi}_{l-1} \mathbf{W}_l^\top + \mathbf{b}_l)^\top, \tag{3}$$

- $\mathbf{\Phi}_l = \{\boldsymbol{\phi}_{n,l}\}_{n=1}^N \in \mathbb{R}^{N \times K_l}$

Variational Bayes (VB) for deep Gaussian process

Normal mixture distribution as a variational distribution $q(\boldsymbol{\theta})$ to approximate the posterior distribution $\pi(\boldsymbol{\theta}|\{\mathbf{x}_n, \mathbf{y}_n\}_{n=1}^N)$ of deep GP. Specifically, the variational distributions are defined as

$$
\begin{aligned}
q(\mathbf{W}_l) &= \prod_{\forall i,j} q(w_{l,ij}), \quad q(\mathbf{b}_l) = \prod_{\forall i} q(b_{l,i}) \\
q(w_{l,ij}) &= p_l N(\mu_{l,ij}^w, \sigma^2) + (1 - p_l) N(0, \sigma^2) \\
q(b_{l,i}) &= p_l N(\mu_{l,i}^b, \sigma^2) + (1 - p_l) N(0, \sigma^2),
\end{aligned}
\tag{4}
$$

where $w_{l,ij}$ is the (i,j)th element of the weight matrix $\mathbf{W}_l \in \mathbb{R}^{k_l \times k_{l-1}}$ and $b_{l,i}$ is the ith element of the bias vector $\mathbf{b}_l \in \mathbb{R}^{k_l}$.

Evidence lower bound (ELBO)

Evidence lower bound (ELBO). With the independent variational distribution $q(\boldsymbol{\theta}) := \prod_{l=1}^{L} q(\mathbf{W}_l) q(\mathbf{b}_l)$, the log ELBO of the deep GP is

$$\begin{aligned}\mathcal{L}_{\text{GP-VI}} := \sum_{n=1}^{N} \int \cdots \int \prod_{l=1}^{L} q(\mathbf{W}_l) q(\mathbf{b}_l) \log p(\mathbf{y}_n | \mathbf{x}_n, \{\mathbf{W}_l, \mathbf{b}_l\}_{l=1}^{L}) d\mathbf{W}_1 d\mathbf{b}_1 \cdots d\mathbf{W}_L d\mathbf{b}_L \\ - \mathrm{KL}\Big(\prod_{l=1}^{L} q(\mathbf{W}_l) q(\mathbf{b}_l) \Big\| p(\{\mathbf{W}_l, \mathbf{b}_l\}_{l=1}^{L}) \Big).\end{aligned} \tag{5}$$

Monte Carlo approximation

Since the direct maximization of (5) is challenging due to the intractable integration, Gal and Ghahramani (2016a) replaced it with MC approximation as

$$\begin{aligned}\mathcal{L}_{\text{GP-MC}} := \frac{1}{M}\sum_{m=1}^{M}\sum_{n=1}^{N}\log p(\mathbf{y}_n|\mathbf{x}_n, \{\mathbf{W}_l^{(m)}, \mathbf{b}_l^{(m)}\}_{l=1}^{L}) \\ - \text{KL}\Big(\prod_{l=1}^{L} q(\mathbf{W}_l)q(\mathbf{b}_l)\Big\|\prod_{l=1}^{L} p(\mathbf{W}_l)p(\mathbf{b}_l)\Big),\end{aligned} \tag{6}$$

where $\{\{\mathbf{W}_l^{(m)}, \mathbf{b}_l^{(m)}\}_{l=1}^{L}\}_{m=1}^{M}$ is MC samples from the variational distribution in (4).

MC dropout

- **MC dropout**: Variational Bayes (VB) for deep GP based on Monte Carlo (MC) approximation
  - Gal and Ghahramani (2016a) show applying dropout $\mathbf{d}_l$ after every hidden layer l can approximate the objective function of VB for deep GP

Idea: $\mathbf{\Phi}$

- Image (correlated structure) features $\mathbf{\Phi} \in \mathbb{R}^{N \times k_{L-1}}$: last layer nodes
- Summarizes high-dimensional input $\mathbf{X}$ (matrix or tensor) to a lower dimensional space (vector)
  - $\mathbf{\Phi}$ as a basis design matrix that encapsulates information of $\mathbf{X}$
  - summary statistic useful for predicting response variables

BayesCGLM

- Covariates $\mathbf{Z} \in \mathbb{R}^{N \times p}$, features $\mathbf{\Phi} \in \mathbb{R}^{N \times k_{L-1}}$, and response $\mathbf{Y}$
- **BayesCGLM**

$$g(E[\mathbf{Y}|\mathbf{Z}, \mathbf{\Phi}]) = \mathbf{Z}\boldsymbol{\gamma} + \mathbf{\Phi}\boldsymbol{\delta} = \mathbf{A}\boldsymbol{\beta} \tag{7}$$

- $\boldsymbol{\beta} = (\boldsymbol{\gamma}^\top, \boldsymbol{\delta}^\top)^\top \in \mathbb{R}^{p+k_{L-1}}$: corresponding regression coefficients
- $g(\cdot)$: a one-to-one continuously differential link function

Posterior distribution of coefficient

$$\pi(\beta|\mathbf{D}) = \int \pi(\beta|\mathbf{D}, \{\mathbf{W}_l, \mathbf{b}_l\}_{l=1}^{L-1}) \times \prod_{l=1}^{L-1} \pi(\mathbf{W}_l|\mathbf{D})\pi(\mathbf{b}_l|\mathbf{D}) d\mathbf{W}_1 d\mathbf{b}_1 \cdots d\mathbf{W}_{L-1} d\mathbf{b}_{L-1}, \tag{8}$$

where, $\pi(\beta|\mathbf{D}, \{\mathbf{W}_l, \mathbf{b}_l\}_{l=1}^{L-1})$ is the conditional posterior, and $\pi(\mathbf{W}_l|\mathbf{D})$, $\pi(\mathbf{b}_l|\mathbf{D})$ are marginal posteriors for weight and bias, respectively. Since it is challenging to compute (8) directly, we approximate it through MC dropout as

$$\int \pi(\beta|\mathbf{D}, \{\mathbf{W}_l, \mathbf{b}_l\}_{l=1}^{L-1}) \prod_{l=1}^{L-1} q(\mathbf{W}_l) q(\mathbf{b}_l) d\mathbf{W}_1 d\mathbf{b}_1 \cdots d\mathbf{W}_{L-1} d\mathbf{b}_{L-1}, \tag{9}$$

where $q(\mathbf{W}_l)$ and $q(\mathbf{b}_l)$ are variational distributions in (4)

Posterior distribution of coefficient

Then the MC approximation to (9) is

$$\frac{1}{M}\sum_{m=1}^{M}\pi(\boldsymbol{\beta}_m|\mathbf{D},\{\mathbf{W}_l^{(m)},\mathbf{b}_l^{(m)}\}_{l=1}^{L-1}). \tag{10}$$

Here $\{\{\mathbf{W}_l^{(m)},\mathbf{b}_l^{(m)}\}_{l=1}^{L-1}\}_{m=1}^{M}$ are sampled from (4).

Laplace approximation

1. Compute $\mathbf{\Phi}^{(m)}$ from the given $\{\mathbf{W}_l^{(m)}, \mathbf{b}_l^{(m)}\}_{l=1}^{L-1}$ through forward propagation
2. Obtain the maximum likelihood estimate (MLE) $\widehat{\boldsymbol{\beta}}_m$ using GLM by regressing $\mathbf{Y}$ on $\mathbf{\Phi}^{(m)}$ and $\mathbf{Z}$
3. Approximate the posterior of $\boldsymbol{\beta}_m$ as $\mathcal{N}(\widehat{\boldsymbol{\beta}}_m, \widehat{\mathbf{B}}_m^{-1})$
4. $\widehat{\mathbf{B}}_m \in \mathbb{R}^{(p+k_{L-1})\times(p+k_{L-1})}$: observed Fisher information matrix from the mth MC samples

$$\frac{1}{M}\sum_{m=1}^{M} \varphi(\boldsymbol{\beta}; \widehat{\boldsymbol{\beta}}_m, \widehat{\mathbf{B}}_m^{-1}), \tag{11}$$

where $\varphi(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ is a multivariate normal density with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$.

Predictive distribution of the linear predictor

- $\mathbf{A}_m^* = [\mathbf{Z}^*, \mathbf{\Phi}^{*(m)}] \in \mathbb{R}^{N_{\text{test}} \times (p + k_{L-1})}$, and $\widehat{\boldsymbol{\beta}}_m$ from (11) for $m = 1, \cdots, M$
- $\mathbf{\Phi}^{*(m)} \in \mathbb{R}^{n_{\text{test}} \times k_{L-1}}$ given $\mathbf{X}^*$ and $\mathbf{Z}^*$
- The predictive distribution of the linear predictor is

$$\frac{1}{M} \sum_{m=1}^{M} \varphi(\mathbf{A}^* \boldsymbol{\beta}; \mathbf{A}_m^* \widehat{\boldsymbol{\beta}}_m, \mathbf{A}_m^* \widehat{\mathbf{B}}_m^{-1} \mathbf{A}_m^{*\top}). \tag{12}$$

- Gaussian response: $\mathbf{Y}^* \sim \mathcal{N}(\frac{1}{M} \sum_{m=1}^{M} \mathbf{A}^* \widehat{\boldsymbol{\beta}}_{\boldsymbol{m}}, \widehat{\sigma}^2)$, $\widehat{\sigma}^2 = \sum_{m=1}^{M} (\mathbf{A}^{(m)} \widehat{\boldsymbol{\beta}}_m - \mathbf{Y})^\top (\mathbf{A}^{(m)} \widehat{\boldsymbol{\beta}}_m - \mathbf{Y}) / NM$
- Count response: $\mathbf{Y}^* \sim \text{Poisson}(\frac{1}{M} \sum_{m=1}^{M} \mathbf{A}^* \widehat{\boldsymbol{\beta}}_{\boldsymbol{m}})$

Why use a two-stage approach?

- One-stage approach: BayesCNN (Gal and Ghahramani (2016a))
- Limitation of BayesCNN: poor convergence in parameter estimation, especially with high-dimensional data (Goodfellow et al., 2014; Dauphin et al., 2014)
- Our approach: BayesCGLM
- Make the complex optimization into simple nonparametric regression problems with a fixed basis function of $\mathbf{\Phi}$
- $\mathbf{\Phi}$ is still informative when predicting responses because it is obtained by minimizing the loss function

Why use a two-stage approach?

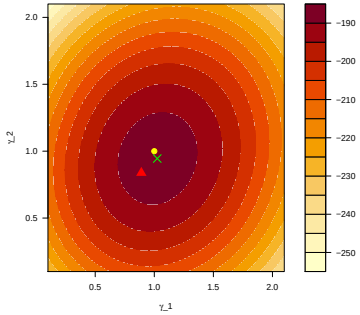


Figure: BayesCNN

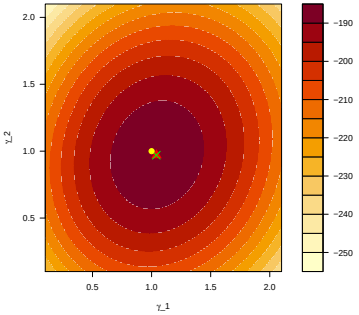


Figure: BayesCGLM

Figure: The profile log-likelihood for γ given other parameters. The yellow circles: true coefficient $\gamma = (1, 1)$, the green x: the profile likelihood estimates, and the red triangles: the Bayes estimates obtained by BayesCNN and BayesCGLM, respectively.

Real data application: malaria incidence

- **Malaria in the African Great Lakes Region**
  - $\mathbf{Y}$: 4,741 cases of malaria
  - $\mathbf{Z}$: average annual rainfall ($\mathbf{Z}_1$), vegetation index of the region ($\mathbf{Z}_2$), and proximity to water ($\mathbf{Z}_3$)
  - $\mathbf{X}$: spatial basis function matrix with 239 knots
- $N_{\text{train}} = 3,500$ and $N_{\text{test}} = 1,241$
- Compare with BayesCNN and a spatial basis regression model

Result

Table: Inference results for the malaria dataset from different methods. For all methods, the posterior mean of γ, 95% HPD interval, RMSPE, prediction coverage, and computing time (min) are reported in the table.

		BayesCGLM $M = 500$	**BayesCNN** $M = 500$	**Spatial model**
γ_1 (vegetation index)	Mean	0.099	0.103	0.115
	95% Interval	$(0.092, 0.107)$	-	$(0.111, 0.118)$
γ_2 (proximity to water)	Mean	0.074	0.058	-0.269
	95% Interval	$(0.068, 0.080)$	-	$(-0.272, -0.266)$
γ_3 (rainfall)	Mean	0.036	0.027	-0.122
	95% Interval	$(0.027, 0.045)$	-	$(-0.126, -0.117)$
Prediction	RMSPE	27.438	28.462	42.393
	Coverage	0.950	0.947	0.545
Time (min)		57.518	30.580	41.285

Uncertainty quantification

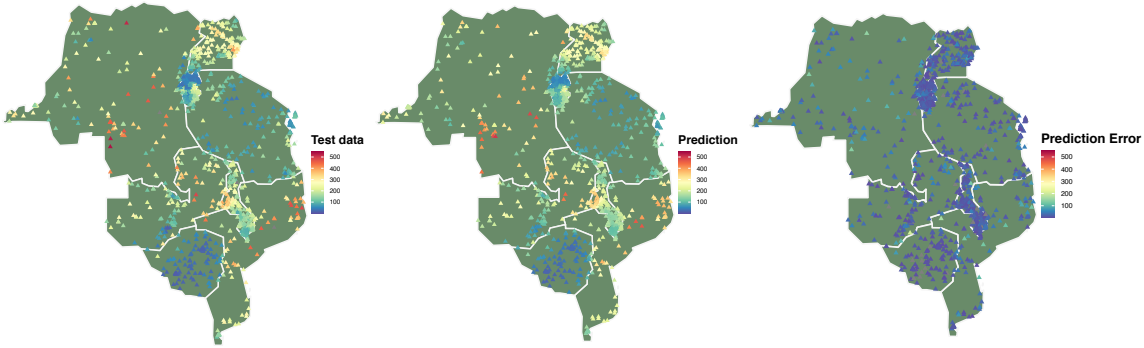
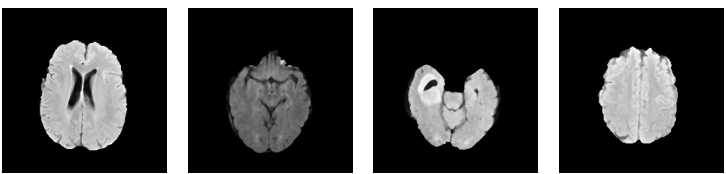


Figure: **Left:** Test data **Middle:** Prediction **Right:** Prediction error

Real data applications: brain tumor

- **Brain tumor MRI**
  - **Y**: whether 4,515 patients have a brain tumor or not
  - **Z**: first order feature of image ($\mathbf{Z}_1$) and second order feature of image ($\mathbf{Z}_2$)
  - **X**: 240×240 pixel gray images
- $N_{\text{train}} = 2,508$ and $N_{\text{test}} = 2,007$
- Compare with BayesCNN and a logistic regression model



Result

Table: Inference results for the brain tumor dataset from different methods. For all methods, the posterior mean of γ, 95% HPD interval, accuracy, recall, precision, and computing time (min) are reported in the table.

		BayesCGLM $M = 500$	**BayesCNN** $M = 500$	**GLM**
γ_1 (first order feature)	Mean	-5.332	0.248	-2.591
	95% Interval	$(-7.049, -3.704)$	-	$(-2.769, -2.412)$
γ_2 (second order feature)	Mean	4.894	0.160	2.950
	95% Interval	$(3.303, 6.564)$	-	$(2.755, 3.144)$
Prediction	Accuracy	0.924	0.867	0.784
	Recall	0.929	0.787	0.783
	Precision	0.901	0.907	0.715
Time (min)		293.533	103.924	0.004

Uncertainty quantification

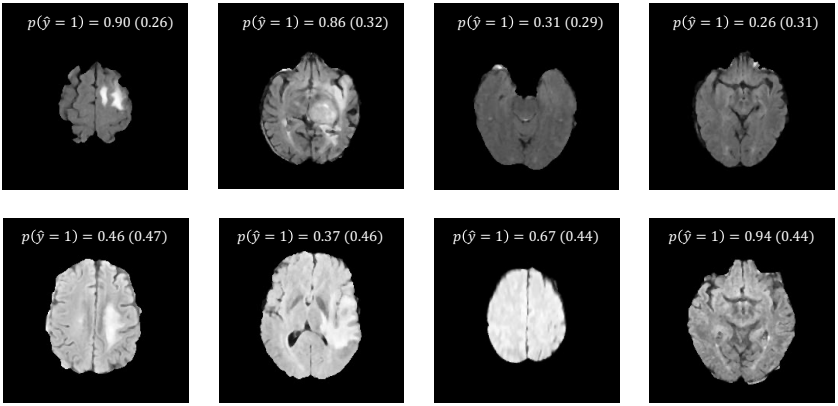


Figure: The top panel illustrates correctly specified images with small prediction errors. The bottom panel illustrates misclassified images with large prediction errors.

Takeaways

- Unified framework for analyzing both correlated high-dimensional variables (e.g., images) with standard vector-type variables.
  - Spatial basis function matrix, MRI images, fMRI correlation matrix
  - Improved prediction accuracy along with interpretation of covariates
- **Uncertainty quantification**
  - Inference of coefficient posterior distribution and predictive distribution
  - A credible interval means a lot!
- This work is published in January, 2024 in *Biometrics*
- Always welcome to discuss!