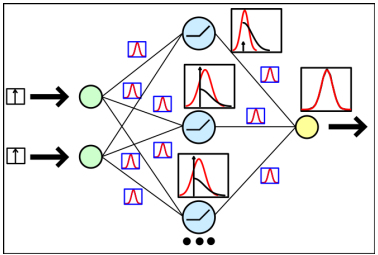



Variational Implicit Processes

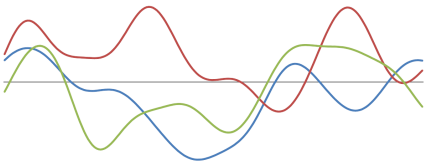
Chao Ma[1], Yingzhen Li[2], José Miguel Hernández-Lobato[1,2]

[1]University of Cambridge, [2]Microsoft Research Cambridge

A function space view of Bayesian Neural Networks



(a) weight space view



(b) function space view

- sampling a configuration of NN weights $\Leftrightarrow$ sampling a function
- more straight-forward to think about priors over functions
- "symmetric" modes in weight posterior $\Rightarrow$ one mode in function posterior

Implicit Stochastic Processes

Definition: An implicit stochastic process (IP) is a collection of random variables $f(\cdot)$, such that any finite collection $\mathbf{f} = (f(\mathbf{x}_1), ..., f(\mathbf{x}_N))^\top$ has joint distribution implicitly defined by the following generative process:

$$\mathbf{z} \sim p(\mathbf{z}), \quad f(\mathbf{x}_n) = g_\theta(\mathbf{x}_n, \mathbf{z}), \quad \forall\ \mathbf{x}_n \in \mathbf{X}. \tag{1}$$

A function distributed according to the above IP is denoted as $f(\cdot) \sim \mathcal{IP}(g_\theta(\cdot, \cdot), p_\mathbf{z})$.

Implicit Stochastic Processes

Note that $\mathbf{z}$ can be finite or infinite dimensional!

- Finite dimensional $\mathbf{z}$:
  prove via Kolmogorov extension theorem (proposition 1).

Implicit Stochastic Processes

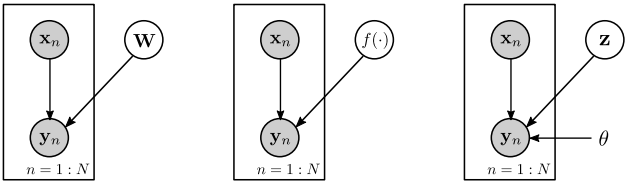
Note that $\mathbf{z}$ can be finite or infinite dimensional!

- Finite dimensional $\mathbf{z}$:
  prove via Kolmogorov extension theorem (proposition 1).
- Infinite dimensional case (here $\mathbf{z} = z(\cdot)$ is a random function):
  sufficient conditions:
  - $z(\cdot) \sim \mathcal{SP}(0, C(\cdot,\cdot))$ is a centered stochastic process on $\mathcal{L}^2(\mathbb{R}^d)$
  - $g(\mathbf{x}, z) = h(\int_{\mathbf{x}} \sum_{l=0}^{M} K_l(\mathbf{x}, \mathbf{x}') z(\mathbf{x}') d\mathbf{x}'),\ K_l \in \mathcal{L}^2(\mathbb{R}^d \times \mathbb{R}^d),\ |h(x)| \leq A|x|$

  Then $f(\cdot)$ is also a stochastic process (proposition 2).
  Proof: apply Karhunen-Loeve expansion to $z(\cdot)$ then prove $g(\mathbf{x}, z)$ converges in $\mathcal{L}^2(\mathbb{R}^d)$.

Implicit Stochastic Processes

Examples:



Also include many simulators in physics, ecology, climate science...

Implicit Process Regression

Implicit process regression model:

$$f(\cdot) \sim \mathcal{IP}(g_\theta(\cdot,\cdot), p_{\mathbf{z}}),\ y = f(\mathbf{x}) + \epsilon,\ \epsilon \sim \mathcal{N}(0, \sigma^2).$$

- Similar to GP regression, given dataset $\mathcal{D} = \{\mathbf{X}, \mathbf{y}\}$, we hope to compute

$$p(\mathbf{f}|\mathbf{X}, \mathbf{y}) \propto p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\mathbf{X})$$

- Then for predictive inference, compute

$$p(y^*|\mathbf{x}^*, \mathcal{D}) = \int p(y^*|f^*)p(f^*|\mathbf{X}, \mathbf{y})df^*$$

intractable due to the unknown distribution $p(\mathbf{f})$ (cannot use variational inference directly)

Variational Implicit Processes

Generalised wake-sleep applied to implicit processes

- **Sleep phase:** approximate $p_\theta(\mathbf{y}, \mathbf{f}|\mathbf{X}) \approx q(\mathbf{y}, \mathbf{f}|\mathbf{X})$
- **Wake phase:** approximate $\log p_\theta(\mathbf{y}|\mathbf{X}) \approx \log q(\mathbf{y}|\mathbf{X})$ then maximise w.r.t θ
- large-scale learning: spectral approximations lead to a Bayesian linear regression problem

Variational Implicit Processes

Sleep phase:

- Define $q_{\mathcal{GP}}(\mathbf{y}, \mathbf{f}|\mathbf{X}) = q(\mathbf{y}|\mathbf{f}) q_{\mathcal{GP}}(\mathbf{f}|\mathbf{X})$, $\underbrace{q(\mathbf{y}|\mathbf{f}) = p(\mathbf{y}|\mathbf{f})}_{\text{same likelihood term}}$
- for *any* $\mathbf{X}$, use $(\mathbf{y}, \mathbf{f}) \sim p(\mathbf{y}, \mathbf{f}|\mathbf{X})$ as targets to train q:

$$\min_q \ \mathrm{D}_{\mathrm{KL}}[p(\mathbf{y}, \mathbf{f}|\mathbf{X}) || q_{\mathcal{GP}}(\mathbf{y}, \mathbf{f}|\mathbf{X})]$$

- Reduce to matching mean & covariance functions:

$$m^\star_{\mathrm{MLE}}(\mathbf{x}) = \frac{1}{S} \sum_s f_s(\mathbf{x}), \quad \mathcal{K}^\star_{\mathrm{MLE}}(\mathbf{x}_1, \mathbf{x}_2) = \frac{1}{S} \sum_s \Delta_s(\mathbf{x}_1) \Delta_s(\mathbf{x}_2),$$

$$\Delta_s(\mathbf{x}) = f_s(\mathbf{x}) - m^\star_{\mathrm{MLE}}(\mathbf{x}), \quad f_s(\cdot) \sim \mathcal{IP}(g_\theta(\cdot, \cdot), p_{\mathbf{z}}).$$

$q^\star_{\mathcal{GP}}(\mathbf{f}|\mathbf{X}, m^\star_{\mathrm{MLE}}, \mathcal{K}^\star_{\mathrm{MLE}}, \theta)$ depends on θ

Wake phase:

- We want to maximise $\log p_\theta(\mathbf{y}|\mathbf{X})$ w.r.t. θ (intractable)
- Note that in sleep step we are minimising joint KL and

$$\mathrm{D}_{\mathrm{KL}}[p(\mathbf{y}, \mathbf{f}|\mathbf{X})||q_{\mathcal{GP}}(\mathbf{y}, \mathbf{f}|\mathbf{X})] \geq \mathrm{D}_{\mathrm{KL}}[p(\mathbf{y}|\mathbf{X})||q_{\mathcal{GP}}(\mathbf{y}|\mathbf{X})]$$

- Then we use $\log q^\star_{\mathcal{GP}}(\mathbf{y}|\mathbf{X}, \theta) \approx \log p_\theta(\mathbf{y}|\mathbf{X})$
- Note that $q^\star_{\mathcal{GP}}(\mathbf{y}|\mathbf{X}, \theta)$ depends on $\theta \Rightarrow$ just differentiate through

Wake phase:

For large dataset GP inference is very expensive (cubic complexity)!

Spectral approximation: **Bayesian linear regression (BLR) on top of function samples**

$$\log q^{\star}_{\mathcal{GP}}(\mathbf{y}|\mathbf{X},\theta) \approx \log \int \prod_{n} q^{\star}(y_n|\mathbf{x}_n,\mathbf{a},\theta)p(\mathbf{a})d\mathbf{a},$$

$$q^{\star}(y_n|\mathbf{x}_n,\mathbf{a},\theta) = \mathcal{N}\left(y_n;\mu(\mathbf{x}_n,\mathbf{a},\theta),\sigma^2\right), p(\mathbf{a}) = \mathcal{N}(\mathbf{a};0,\mathbf{I}),$$

$$\mu(\mathbf{x}_n,\mathbf{a},\theta) = m^{\star}(\mathbf{x}_n) + \frac{1}{\sqrt{S}}\sum_{s}\Delta_s(\mathbf{x}_n)a_s.$$

For scalability we do variational inference for BLR
(by constructing $q(\mathbf{a}|\boldsymbol{\mu},\boldsymbol{\Sigma}) \approx p(\mathbf{a}|\mathbf{X},\mathbf{y})$)

Variational Implicit Processes

Prediction

With $f_s(\cdot) \sim \mathcal{IP}(g_\theta(\cdot,\cdot), p_{\mathbf{z}})$:

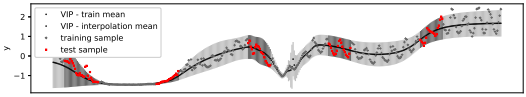
- GP predictive inference
- BLR with coefficients sampled from $q(\mathbf{a})$

Computational complexity (on BNNs):

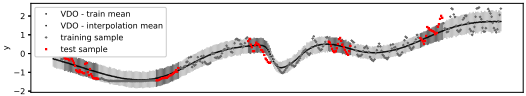
assume doing inference for M inputs with S random functions,
each function evaluation costs $\mathcal{O}(C)$:

- Training: $\mathcal{O}(CMS + MS^2 + S^3)$ vs $\mathcal{O}(CMS)$ for weight-space inference
- Test: $\mathcal{O}(CMS + S^3)$ vs $\mathcal{O}(CMS)$ for weight-space inference

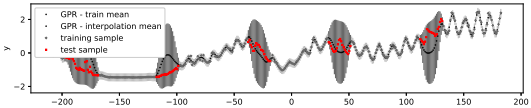
Some Experimental Results



(c) VIP-BNN



(d) Variational dropout (VDO-BNN)

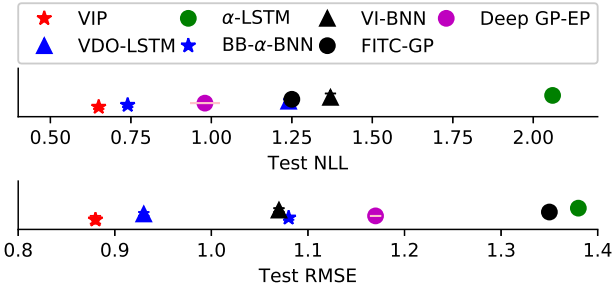


(e) GP regression (GPR)

Solar irradiance prediction:

- methods: VIP, VDO, GPR
- Capturing the predictive mean: VIP $>$ GPR;
- Uncertainty estimates: VIP $>$ VDO;

Some Experimental Results



VIP applied to Bayesian LSTM:

- CEP Data: $>$1 million datapoints, each $\mathbf{x}$ is a string represneting a molecule;
- Goal: predict power conversion efficiency
- Baselines: (deep) GP, BNN (hand-crafted features) & Bayesian LSTM (directly raw features), with different inference methods;
- VIP works significantly better for both NLL and RMSE.

Summary and Future Directions

Our contributions:

- We formally defined the implicit stochastic processes
- We introduced a generalised wake-sleep algorithm to train IP models
- We validated the VIP algorithm on a wide range of regression tasks

Future work:

- Non-GP posterior approximations
- Beyond regression tasks

Welcome to our poster #225 for discussions :)