Meta Particle Flow for Sequential Bayesian Inference

Le Song
Associate Professor, CSE
Associate Director, Machine Learning Center
Georgia Institute of Technology

Joint work with Xinshi Chen and Hanjun Dai
Bayesian Inference

Infer the posterior distribution of unknown parameter $x$ given

- Prior distribution $\pi(x)$
- Likelihood function $p(o|x)$
- Observations $o_1, o_2, ..., o_m$

$$p(x|o_{1:m}) = \frac{1}{z} \pi(x) \prod_{i=1}^{m} p(o_i|x)$$

$$z = \int \pi(x) \prod_{i=1}^{m} p(o_i|x) \, dx$$

Challenging computational problem for high dimensional $x$
Gaussian Mixture Model

- prior $x_1, x_2 \sim \pi(x) = \mathcal{N}(0, I)$
- observations $o | x_1, x_2 \sim p(o | x_1, x_2) = \frac{1}{2} \mathcal{N}(x_1, 1) + \frac{1}{2} \mathcal{N}(x_1 + x_2, 1)$
- With $(x_1, x_2) = (1, -2)$, the resulting posterior will have two modes: $(1, -2)$ and $(-1, 2)$

To fit only one posterior $p(x | o_{1:m})$ is already not easy.

[Results reported by Dai et al. (2016)]
Fundamental Principle for Machine Learning

Lots of applications in machine learning

- **Hidden Markov model**

- **Topic modeling**

- **Uncertainty quantification**

  - \( o_{t+1} = P o_{t-\tau} \exp\left(-\frac{y_{t-\tau}}{y_0}\right) e_t + o_t \exp(-\delta \epsilon_t) \),

  - \( e_t \sim \Gamma(\sigma_p^{-2}, \sigma_p^2), \epsilon_t \sim \Gamma(\sigma_d^{-2}, \sigma_d^2) \)

  - parameters \( x = (P, y_0, \sigma_p^2, \sigma_d^2, \tau, \delta) \)
Online Bayesian Inference

• Observations $o_1, o_2, ..., o_m$ arrive sequentially

An ideal algorithm should:

• Efficiently update $p(x|o_{1:m})$ to $p(x|o_{1:m+1})$ when $o_{m+1}$ is observed
• Without storing all historical observations $o_1, o_2, ..., o_m$

\[
p(x|o_{1:m}) \propto p(x|o_{1:m-1}) p(o_m|x)
\]
- updated posterior
- current posterior
- likelihood

Prior $\pi(x)$

\[
p(x|o_1) \rightarrow p(x|o_1) \rightarrow p(x|o_{1:2}) \rightarrow ... \rightarrow p(x|o_{1:m})
\]
Related Work

- **MCMC**
  - requires a complete scan of the data

- **Variational Inference (VI)**
  - requires re-optimization for every new observation

- **Stochastic approximate inference**
  - are prescribed algorithms to optimize the final posterior $p(x|o_{1:M})$
  - cannot exploit the structure of the sequential inference problem

![Diagram](https://via.placeholder.com/150)
Related Work

- **Sequential Monte Carlo** (Doucet et al., 2001; Balakrishnan & Madigan, 2006)
  - the state of art for online Bayesian Inference
  - but suffers from path degeneracy problem in high dimensions
  - rejuvenation steps can help but will violate online constraints (Canini et al., 2009)

Can we learn to perform efficient and effective sequential Bayesian update?
Operator View

Kernel Bayes’ Rule (Fukumizu et al., 2012)

- the posterior is represented as an embedding $\mu_m = \mathbb{E}_{p(x|o_{1:m})}[\phi(x)]$

- $\mu_{m+1} = \mathcal{K}(\mu_m, o_{m+1})$

- views the Bayes update as an operator in reproducing kernel Hilbert space (RKHS)

- conceptually nice but is limited in practice
Our Approach: Bayesian Inference as Particle Flow

**Particle Flow**

- **Start with** $N$ **particles**
  
  $\mathcal{X}_0 = \{x_0^1, ..., x_0^N\}$, sampled i.i.d. from prior $\pi(x)$

- **Transport particles to next posterior via solution of an initial value problem (IVP)**
  
  $$
  \frac{dx}{dt} = f(\mathcal{X}_0, o_1, x(t)), \forall t \in [0, T] \quad \text{and} \quad x(0) = x_0^n 
  $$

  $$
  \Rightarrow \text{solution } x_1^n = x(T)
  $$
• **Continuity Equation** expresses the law of *local conservation of mass*:
  
  – Mass can neither be created nor destroyed
  – nor can it ‘teleport’ from one place to another

\[
\frac{\partial q(x, t)}{\partial t} = -\nabla_x \cdot (q f)
\]

• **Theorem.** If \( \frac{dx}{dt} = f \), then the change in log-density follows the differential equation

\[
\frac{d \log q(x, t)}{dt} = -\nabla_x \cdot f
\]

• **Notation**

  – \( \frac{dq}{dt} \) is material derivative that defines the rate of change of \( q \) in a given particle as it moves along its trajectory \( x = x(t) \)

  – \( \frac{\partial q}{\partial t} \) is partial derivative that defines the rate of change of \( q \) at a particular point \( x \)
Particle Flow for Sequential Bayesian Inference

\[ x_{m+1}^n = x_m^n + \int_0^T f(X_m, o_{m+1}, x(t)) dt \]

\[ -\log p_{m+1}^n = -\log p_m^n + \int_0^T \nabla_x \cdot f(X_m, o_{m+1}, x(t)) dt \]

- Other ODE approaches (eg. Neural ODE of Chen et al 18), are not for sequential case.
Does a shared flow velocity $f$ exist for different Bayesian inference tasks involving different priors and different observations?

**A simple Gaussian Example**

- Prior $\pi(x) = \mathcal{N}(0, \sigma_0)$, likelihood $p(o|x) = \mathcal{N}(x, \sigma)$, observation $o = 0$
- $\Rightarrow$ posterior $p(x|o = 0) = \mathcal{N}(0, \frac{\sigma \cdot \sigma_0}{\sigma + \sigma_0})$
- Whether a shared $f$ exists for priors with different $\sigma_0$? What is the form for it?
  - E.g. $f$ in the form of $f(o, x(t))$ won’t be able to handle different $\sigma_0$. 

\begin{align*}
x(0) &\sim \pi(x) \\
x(t) &\sim p(x|o_1) \\
x(T) &= x(0) + \int_0^T f(\text{inputs}) dt
\end{align*}
Existence: Connection to Stochastic Flow

- **Langevin dynamics** is a *stochastic* process
  
  \[ dx(t) = \nabla_x \log \pi(x)p(o|x) \, dt + \sqrt{2} \, dw(t), \]
  
  where \( dw(t) \) is a standard Brownian motion.

- **Property.** If the potential function \( \Psi(x) := -\log \pi(x)p(o|x) \) is smooth and \( e^{-\Psi} \in L^1(\mathbb{R}^d) \), the Fokker-Planck equation has a unique stationary solution in the form of Gibbs distribution,

  \[ q(x, \infty) = \frac{e^{-\Psi}}{Z} = \frac{\pi(x)p(o|x)}{Z} = p(x|o) \]
Existence: Connection to Stochastic Flow

• The probability density $q(x, t)$ of $x(t)$ follows a deterministic evolution according to the **Fokker-Planck equation**

$$
\frac{\partial q}{\partial t} = -\nabla_x \cdot (q \nabla_x \log \pi(x)p(o|x)) + \Delta_x q(x, t)
\Rightarrow
= -\nabla_x \cdot (q(\nabla_x \log \pi(x)p(o|x) - \nabla_x \log q(x, t))),
$$

which is in the form of **Continuity Equation**.

• **Theorem.** When the deterministic transformation of random variable $x(t)$ follows

$$
\frac{dx}{dt} = \nabla_x \log \pi(x)p(o|x) - \nabla_x \log q(x, t),
$$

its probability density $p(x, t)$ converges to the posterior $p(x|o)$ as $t \to \infty$. 
Close loop to Open loop

- Fokker-Planck equation leads to close loop flow, depend not just on $\pi(x)$ and $p(o|x)$, but also on flow state $q(x,t)$.

- Is there an equivalent form independent of $q(x,t)$ which can achieve the same flow? Optimization problem

\[
\min_w \quad d(q(x, \infty), p(x|o))
\]

\[
s.t. \quad \frac{dx}{dt} = \nabla_x \log \pi(x)p(o|x) - w,
\]

- **Positive answer:** there exists a fixed and deterministic flow velocity $f$ of the form

\[
\frac{dx}{dt} = \nabla_x \log \pi(x)p(o|x) - w^*(\pi(x), p(o|x), x, t)
\]
Parameterization

\[
\frac{dx}{dt} = \nabla_x \log \pi(x)p(o|x) - w^*(\pi(x), p(o|x), x, t)
\]

- \(\pi(x) \Rightarrow \mathcal{X}\)
  - use samples \(\mathcal{X}\) as surrogates, feature space embedding

\[
\mu_{\mathcal{X}}(p) := \int_{\mathcal{X}} \phi(x)p(x) \, dx \approx \frac{1}{N} \sum_{n=1}^{N} \phi(x^n), \quad x^n \sim \pi
\]

  - Ideally, if \(\mu_{\mathcal{X}}\) is an injective mapping from the space of probability measures

- \(p(o|x) \Rightarrow (o, x(t))\) for a fixed likelihood function

- With two neural networks \(\phi\) and \(h\), the overall parameterization

\[
f(\mathcal{X}, o, x(t), t) = h \left( \frac{1}{N} \sum_{n=1}^{N} \phi(x^n), o, x(t), t \right)
\]
Multi-task Framework

- Training set $\mathcal{D}_{train}$ for meta learning
  - containing multiple inference tasks, with diverse prior and observations
- Each task $\mathcal{T} \in \mathcal{D}_{train}$, $\mathcal{T} := (\pi(x), p(\cdot \mid x), \{o_1, o_2, \ldots, o_M\})$
  - prior likelihood $M$ observations

Loss Function

- Minimize negative evidence lower bound (ELBO) for each task $\mathcal{T}$

$$
\mathcal{L}(\mathcal{T}) = \sum_{m=1}^{M} KL(q_m(x) || p(x|o_{1:m})) = \sum_{m=1}^{M} \sum_{n=1}^{N} \log q_m^n - \log p(x_m^n, o_{1:m}) + \text{const.}
$$

$$
\begin{align*}
x_{m+1}^n &= x_m^n + \int_0^T f(x_m, o_{m+1}, x(t), t) dt \\
-\log q_{m+1}^n &= -\log q_m^n + \int_0^T \nabla_x \cdot f(x_m, o_{m+1}, x(t), t) dt
\end{align*}
$$
Experiments: Benefit for High Dimension

Multivariate Gaussian Model
- prior $x \sim \mathcal{N}(\mu_x, \Sigma_x)$
- observation conditioned on prior $o|x \sim \mathcal{N}(x, \Sigma_o)$

Experiment Setting
- Training set only contains sequences of 10 observations, but a diverse set of prior distributions.
- Testing set contains 25 different sequences of 100 observations.

Result
- As the dimension of the model increases, our MPF operator has more advantages.
Gaussian Mixture Model

- prior $x_1, x_2 \sim \mathcal{N}(0,1)$
- observations $o|x_1, x_2 \sim \frac{1}{2} \mathcal{N}(x_1, 1) + \frac{1}{2} \mathcal{N}(x_1 + x_2, 1)$
- With $(x_1, x_2) = (1, -2)$, the resulting posterior will have two modes: $(1, -2)$ and $(-1, 2)$

To fit only one posterior $p(x|o_{1:m})$ is already not easy.

[Results reported by Dai et al. (2016)]
Gaussian Mixture Model

- prior $x_1, x_2 \sim \mathcal{N}(0,1)$
- observations $o|x_1, x_2 \sim \frac{1}{2} \mathcal{N}(x_1, 1) + \frac{1}{2} \mathcal{N}(x_1 + x_2, 1)$
- With $(x_1, x_2) = (1, -2)$, the resulting posterior will have two modes: $(1, -2)$ and $(-1, 2)$

Our more challenging experimental setting:

- The learned MPF operator will be tested on sequences that is not observed in the training set.
- It needs to fit all intermediate posteriors $p(x|o_1), p(x|o_{1:2}), ..., p(x|o_{1:m})$.

Visualization of the evolution of posterior density from left to right.
Hidden Markov Model – Linear Dynamical System

- \( x_m = A x_{m-1} + \epsilon_m, \quad \epsilon_m \sim \mathcal{N}(0, \Sigma_1) \)
- \( o_m = B x_m + \delta_m, \quad \delta_m \sim \mathcal{N}(0, \Sigma_2) \)

Marginal posteriors update: \( p(x_1|o_1) \rightarrow p(x_2|o_{1:2}) \rightarrow \cdots \rightarrow p(x_m|o_{1:m}) \)

**Transition sampling + MPF operator:**

1. \( \tilde{x}^n_m = A \tilde{x}^n_{m-1} + \epsilon_m \)

2. \( x^*_m = \mathcal{F}(\tilde{X}_m, \tilde{x}^n_{m}, o_{m+1}) \) – A Bayesian update from \( p(x_m|o_{1:m-1}) \) to \( p(x_m|o_{1:m}) \)
Experiments: Generalization across Multitasks

Bayesian Logistic Regression on MNIST dataset 8 vs 6

- About 1.6M training images and 1932 testing images
- Each data point \( o_m : = (\phi_m, c_m) \)
  
  - feature \( \phi \)
  
  - label \( c \)

- Logistic Regression \( y = \sigma(\theta^T \phi_m) \)
- Likelihood function \( p(o_m|\theta) = y^{c_m}(1 - y)^{1-c_m} \)

Multi-task Environment

- We reduce the dimension of the images to 50 by PCA
- We rotate the first two components by an angle \( \psi \sim [-15^\circ, 15^\circ] \)
  
  - The first two components account for more variability in the data
- Different rotation angle \( \psi \Rightarrow \) different decision boundary \( \Rightarrow \) different tasks

Multi-task Training

- MPF operator will be learned from multiple training tasks with different \( \psi \)
- Use the learned MPF as an online-learning algorithm during testing
Experiments: Generalization across Multitasks

Testing as Online learning:

1. All algorithms start with a set of particles sampled from the prior;
2. Each algorithm makes predictions to the encountered batch of 32 images;
3. All algorithms observe true labels of the encountered batch;
4. Each algorithm updates its particles and then make predictions to next batch;

\[
\text{predict} \rightarrow \text{observe true labels} \rightarrow \text{update particles} \rightarrow \text{predict} \rightarrow \text{observe true labels} \rightarrow \text{update} \ldots
\]

accuracy $r_1$  
accuracy $r_2$
Conclusion and Future Work

- An ODE-based Bayesian operator for sequential Bayesian inference
  - Existence
  - Parametrization
  - Meta-learning framework

\[ \begin{align*}
\mathcal{X}_0 &= \{x_0^1, \ldots, x_0^N\} \\
\mathcal{X}_0 &\sim \pi(\mathcal{x}) \\
\mathcal{X}_1 &= \{x_1^1, \ldots, x_1^N\} \\
\mathcal{X}_1 &\sim p(x|o_1)
\end{align*} \]

\[ x_1^n = x_0^n + \int_0^T f(\mathcal{X}_0, o_1, x(t), t)dt \]

- Future work
  - Architecture
  - Stable flow
  - Improve training