# Streaming Bayes GFlowNets

Tiago da Silva, Daniel Augusto, Diego Mesquita

September 2nd, 2025

NeurIPS, 2024.

# Approximate Bayesian Inference

Let $\mathcal{X} = \{x_1, ..., x_N\}$ be a finite, compositional space.

Also, let $\mathcal{D}$ be the data space, $f(\cdot \mid x) : \mathcal{D} \to \mathbb{R}$ be a likelihood function, and $\pi : \mathcal{X} \to \mathbb{R}$ be a prior on $\mathcal{X}$.

# Approximate Bayesian inference

Our objective is to approximate the a sequence of posterior distributions,

$$\pi(x \mid D_1, ..., D_i) \propto f(D_1, ..., D_i \mid x)\pi(x) \text{ for } i \geq 1.$$

This is known as the **streaming Bayes problem.**

That is, Bayesian inference over **unbounded data.**

# Approximate Bayesian inference

For conciseness, we will let

$$R_i(x) = f(D_1, ..., D_i \mid x)\pi(x)$$

be the **unnormalized posterior distribution.**

**Important.**

We assume that $D_i$ and $D_j$ are independent.

BUT each $D_i$ may have its own likelihood function.

# Approximate Bayesian inference

We approximate

$$\left(R_i\right)_{i \geq 1}$$

with a tractable family $\left(p^{\theta_i}\right)_{i \geq 1}$ parametrized by $\theta$.

Obviously we are only interested in the latest $\theta_i$ for each $i$.

No need to store unboundedly many models!

# Approximate Bayesian inference

Traditional techniques (e.g., MFVI) rely on $R_i$'s gradient.

However, $\nabla_x R_i(x)$ does not exist in discrete spaces.

GFlowNets (Bengio, 2021) address this problem.

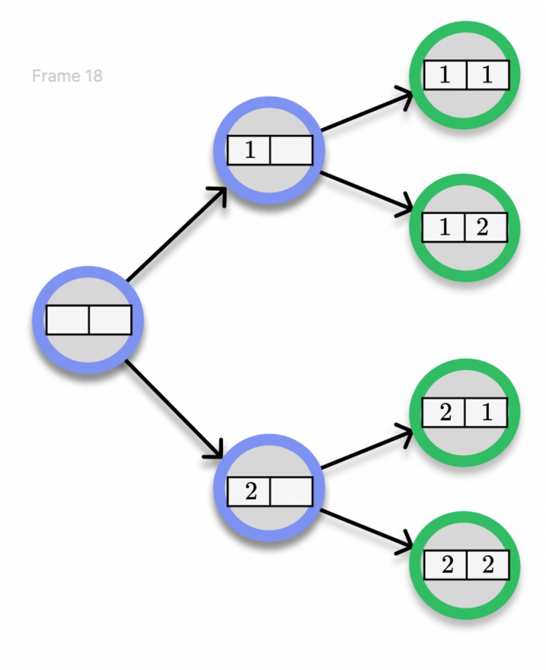
They can also be used for Streaming Bayes, as I will show.

# Generative Flow Networks (GFlowNets)

Compositional objects are uniquely characterized by:

1. A **state space,** $\mathcal{S}$, with both incomplete and complete objects. $\mathcal{S}$ also contains the **simplest of states** $s_o$.
2. A **terminal state space,** $\mathcal{X}$, which is our original parameter space. Clearly, $\mathcal{X} \subseteq \mathcal{S}$.
3. A **state graph** dictating how to navigate between states.

# Generative Flow Networks (GFlowNets)




State graph (SG).

A GFlowNet learns a **policy** $p_F(\cdot \mid s)$.

$p_F(\cdot \mid s)$ is a distribution over $s$'s children on the SG.

$p_F(\cdot \mid s)$ is a **neural network**

$$h_\theta : \mathcal{S} \to \Delta_{A-1}$$

with parameter $\theta$. $\Delta_{A-1}$ is a simplex, and $A$ is the SG's maximum outdegree.

# GFlowNets

Our objective is to find a $\theta_i$ such that

$$\sum_{\tau \rightsquigarrow x} \underbrace{\prod_{1 \leq t \leq |\tau|} p_F^{\theta_i}(\tau_{t+1} \mid \tau_t)}_{p_F(\tau \mid s_o)} \propto R_i(x)$$

($\tau \rightsquigarrow x$ is a state-sequence from $s_o$ to $x \in \mathcal{X}$).

For this, we introduce a **backward policy** $p_B(\cdot \mid s)$.

$p_B(\cdot \mid s)$ is an **uniform distribution** over $s$'s parents.

We solve this problem by minimizing

$$\theta_i^\star, Z_i^\star = \min_{\theta_i, Z_i} \mathbb{E}_\tau \left[ \left( \log \frac{R_i(\tau_{-1}) p_B(\tau \mid \tau_{-1})}{Z_i \cdot p_F^{\theta_i}(\tau \mid \tau_o)} \right)^2 \right]$$

$\tau_{-1}$ is the last element of the trajectory $\tau$ (which is on $\mathcal{X}$).
$\tau_o = s_o$ is the first.

$Z_i$ is an auxiliary parameter.

We can demonstrate that the unconstrained global optima solves our sampling problem (Bengio, 2021).

## Streaming Bayes GFlowNets

Our prior approach results in a policy $p_F^{\theta_i}$.

This policy **generates approximate samples** from $R_{i(x)}$.

**Research question.**

How to update $p_F^{\theta_i}$ when novel data $D_{i+1}$ are observed?

Or: how to sample from $R_{i+1}$ given a model for $R_i$?

(Recall that $R_i(x) = f(D_1, ..., D_i \mid x)\pi(x)$).

# Streaming Bayes GFlowNets

We use the $i$th model as a **prior** for the $(i+1)$ model.

To understand this, notice that

$$R_{i+1}(x) = f(D_1, ..., D_{i+1} \mid x)\pi(x)$$

$$= \underbrace{f(D_1, ..., D_i \mid x)\pi(x)}_{R_i(x)} f(D_{i+1} \mid x)$$

$$= R_i(x) f(D_{i+1} \mid x).$$

# Streaming Bayes GFlowNets

$$R_{i+1}(x) = R_i(x) f(D_{i+1} \mid x).$$

Let $\tau_x$ be any trajectory from $s_o$ to $x$. Since

$$R_i(x) = \frac{Z_i^\star \cdot p_F^{\theta_i^\star}(\tau_x \mid s_o)}{p_B(\tau_x \mid x)},$$

then

$$R_{i+1}(x) = \frac{Z_i^\star \cdot p_F^{\theta_i^\star}(\tau_x \mid s_o)}{p_B(\tau_x \mid x)} \cdot f(D_{i+1} \mid x)$$

# Streaming Bayes GFlowNets

Our learning objective then becomes

$$\theta^\star_{i+1}, Z^\star_{i+1} = \min_{\theta_{i+1}, Z_{i+1}} \mathbb{E}_\tau \left[ \left( \log \frac{R_{i+1}(\tau_{-1}) p_B(\tau \mid \tau_{-1})}{Z_{i+1} \cdot p_F^{\theta_{i+1}}(\tau \mid \tau_o)} \right)^2 \right]$$

$$= \min_{\theta_{i+1}, Z_{i+1}} \mathbb{E}_\tau \left[ \left( \log \frac{p_B(\tau \mid \tau_{-1})}{Z_{i+1} \cdot p_F^{\theta_{i+1}}(\tau \mid \tau_o)} + \log \frac{Z^\star_i \cdot p_F^{\theta^\star_i}(\tau \mid \tau_o)}{p_B(\tau \mid \tau_{-1})} \cdot f(D_{i+1} \mid \tau_{-1}) \right)^2 \right]$$

This is the **streaming Bayes** objective ($L_{\mathrm{SB}}$).

# Streaming Bayes GFlowNets (catch-up)

1. A sequence of independent data sets, $(D_i)_{i \geq 1}$.
2. A likelihood function, $f(\cdot \mid x)$, indexed by $x \in \mathcal{X}$.

Algorithm:

1. We first learn a policy $p_F^{\theta_1^\star}$ that approximates $\pi(\cdot \mid D_1)$.
2. We iteratively update $p_F^{\theta_i^\star}$ by minimizing $L_{\mathrm{SB}}$.

$$\theta_{i+1}^\star, Z_{i+1}^\star = \min_{\theta_{i+1}, Z_{i+1}} L_{\mathrm{SB}}(D_{i+1}, \theta_i^\star, Z_i^\star, f, \pi)$$

# Streaming Bayes GFlowNets

**Important.**

Streaming Bayes GFlowNets do not require

1. **likelihood evaluation** on the **entire data history**;
2. unbounded storage for **model snapshots**.

# Empirical illustration

**Example:** Linear preference learning with integer-valued features.

**Data.** $(d_{i1}, d_{i2}, p_i) \in \{1, 0\}^k \times \{1, 0\}^k \times \{1, 0\}$.

$p_i = 1$ means that $d_{i2} \succeq d_{i1}$.

$p_i = 0$ means that $d_{i1} \succeq d_{i2}$.

$d_{i1}$ and $d_{i2}$ are binary feature vectors.

**Model.** We use a simple logistic model,

$$p(d_{i1} \succeq d_{i2} \mid x) = \sigma\left(x^T(y_{i1} - y_{i2})\right).$$

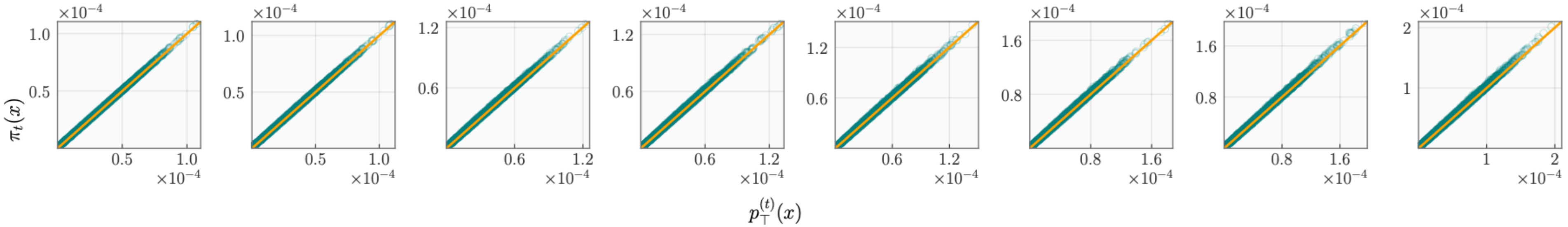
$x \in [[0, u]]^{\{k\}}$ is an integer vector.

We choose a **factorized truncated Poisson prior** for $x$.

We observe a (simulated) stream $(D_i)_{i \geq 1}$ of tuples $(d_1, d_2, p)$.

# Empirical illustration

We compare the true posterior (y axis) with our approximation (x axis) for $1 \leq i \leq 8$.




As we can see, Streaming Bayes GFlowNet accurately approximates the streaming posterior.

# Take home message

GFlowNets are state-of-the-art models for approximate inference over discrete and compositional distributions.

Traditional approaches to streaming Bayesian inference fail on discrete parameter spaces.

Streaming Bayes GFlowNets are a scalable solution for Bayesian inference in the face of continual data streams.

SB-GFlowNets pave the road for reduced environmental footprint for GFlowNet training via model reuse.