Annealed Flow Transport Monte Carlo

Michael Arbel [*,1,†] Alexander G. D. G. Matthews [*, 2] Arnaud Doucet [2]

[*]Equal Contribution

[1]Gatsby Computational Neuroscience Unit, UCL, UK,

[2]DeepMind

[†]Currently at Inria, Grenoble Rhône-Alpes, France

Part I: Presentation of the method

Sampling from un-normalized densities

Target $\pi(x) = Z^{-1} e^{-V(x)}$

- Goal 1: Sampling from a target density π known up to a normalizing constant Z.
- Goal 2: Estimating the normalizing constant Z.

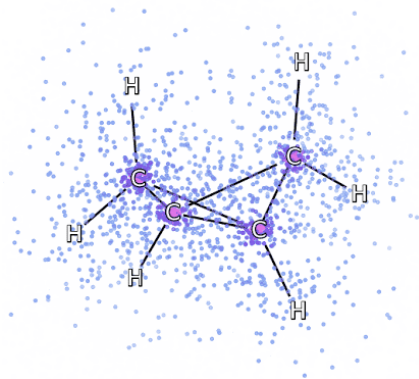
Applications in Statistical physics

Statistical physics, Molecular Dynamics.

- Configuration of physical systems described by a probability over microscopic state x given some macroscopic state y:

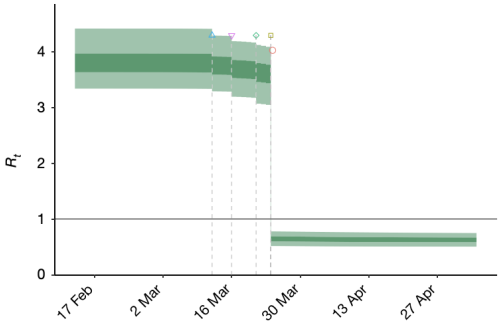
$$\pi(x) = Z(y)^{-1} e^{-V(x,y)} dx$$

- Normalizing constant $Z(y)$ describes the probability of being in a given macroscopic state y.
- Predicting the likelihood of chemical reactions: transition from a state y to state y' if $Z(y') > Z(y)$.



FermiNet project.
See Pfau, Spencer, Matthews and Foulkes.
Physical Review Research 2020.

Applications in Bayesian statistics



Estimating the effects of non-pharmaceutical interventions on COVID-19 in Europe.
See Flaxman, Mishra, Gandy et al. Nature 2020.

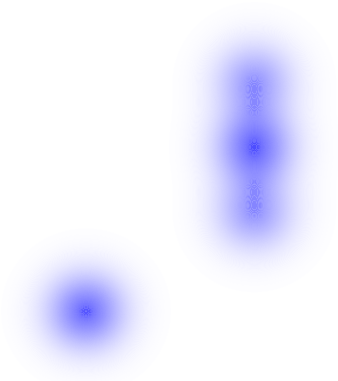
- Bayesian evidence obtained by integrating over latent parameters y:

$$\pi(x) = Z^{-1} \int e^{-V(x,y)} p(y) dy.$$

- Useful for model comparaison: model V better than V' if $\pi(x) > \pi'(x)$ over observation x.
- Inferring latent y from data x: sampling from $\pi(y|x) \propto e^{-V(x,y)} p(y)$.
- Useful for estimating unobserved effects from data.

Sampling from un-normalized densities: Challenges

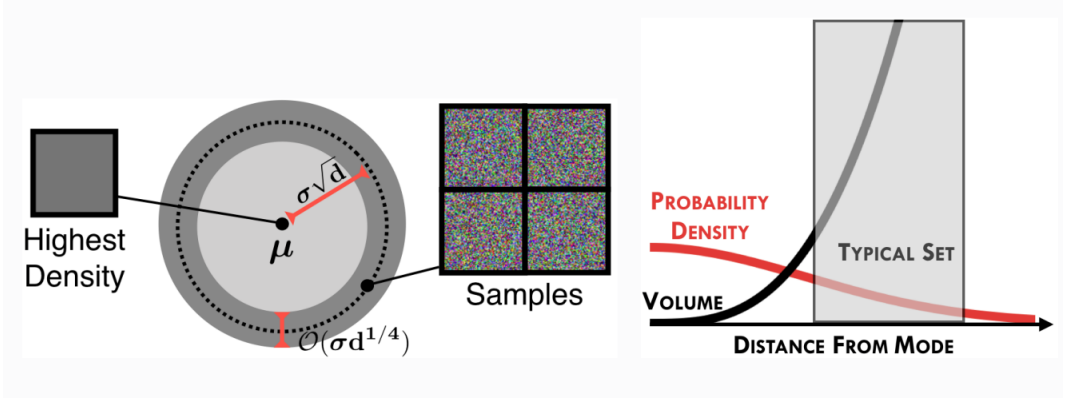
Target $\pi(x) = Z^{-1} e^{-V(x)}$



Challenges:

- Multimodality: Need to explore all the space to cover the different modes.

Sampling from un-normalized densities: Challenges



Borrowed from Tom Rainforth's Lecture on Advanced Bayesian Inference Methods: Lecture 5.

Challenges:

- Curse of dimension: Mass concentration in a typical set far from the mode.

Popular sampling methods: Variational Inference (VI)

- Key idea: introduce a parametric family of densities q_θ that is easy to sample from.
- Find the closest q_θ to π by minimizing the KL:

$$\theta^\star = \arg\min_\theta KL(q_\theta||\pi).$$

- Use samples from q_θ to approximate π.

Popular sampling methods: VI using Normalizing Flows (NFs)

- A normalizing flow is parametric family of diffeomorphisms $x \mapsto T_\theta(x)$, with easy to compute Jacobian determinant.
- Can use normalizing flows and a proposal density p to define $q_\theta = (T_\theta)_\# p$ so that:

$$\log q_\theta(x) = \log p(T_\theta(x)) + \log |\nabla_x T_\theta(x)|.$$

- Under-estimates the tails of π [Domke and Sheldon 2018]

Popular sampling methods: Importance Sampling (IS)

- Key idea: Expectations $\pi[f]$ under π of a function f given by IS w.r.t. a proposal p:

$$\pi[f] = \frac{\int w(x) f(x) p(x) dx}{\int w(x) p(x) dx}$$

- Uses samples from a proposal $p(x)$ and re-weight them according to density ratio $w(x) = e^{-V(x)}/p(x)$.
- High variance estimates of Z.
- Sensitive to choice of the proposal.

Popular sampling methods: Markov Chain Monte Carlo

Target $\pi(x) = Z^{-1} e^{-V(x)}$

- Key idea: Use local moves to explore the typical set of π.
- Construct a Markov chain $(X_k)_{k \geq 0}$ using Markov kernel K invariant w.r.t. π: $X_k \sim K(X_{k-1}, .)$
- Metropolis Adjusted Langevin Algorithm

$$Y_k = X_{k-1} - \gamma \nabla \log \pi(X_{k-1}) + \sqrt{2\gamma} W_k$$
$$X_k \sim \delta_{Y_k} \alpha(X_{k-1}, Y_k) + \delta X_{k-1}(1 - \alpha(X_{k-1}, Y_k))$$

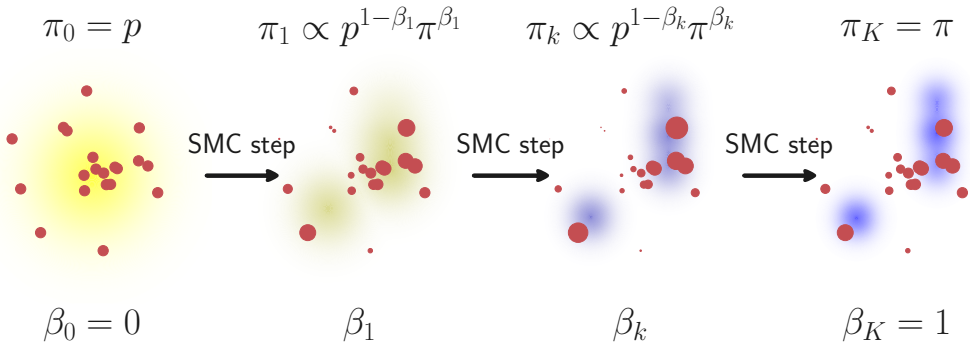
- Cannot estimate Z,
- Unable to explore multiple modes in a reasonable time.

Popular sampling methods: AIS/SMC

Target $\pi(x) = Z^{-1}e^{-V(x)}$

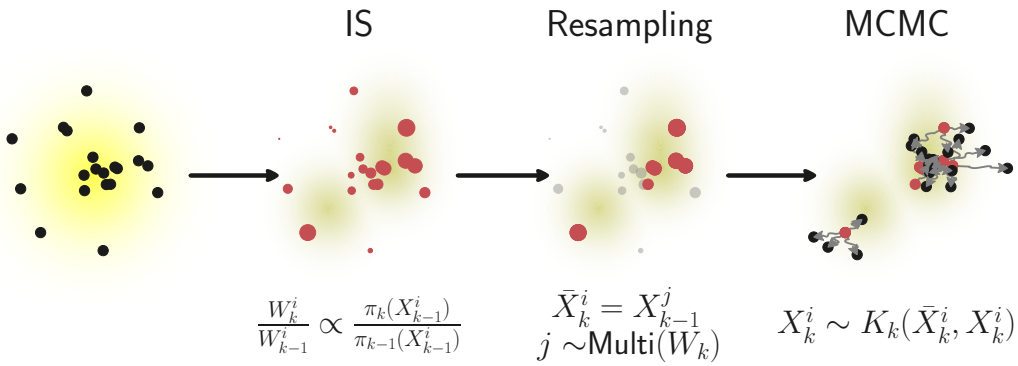
- Key idea: Combines MCMC with Importance sampling: SOTA samplers.
- Accurate estimates require careful design of the algorithms like AIS [Neal, 2001], SMC [Del Moral et al., 2006]

Sequential Monte Carlo (SMC)



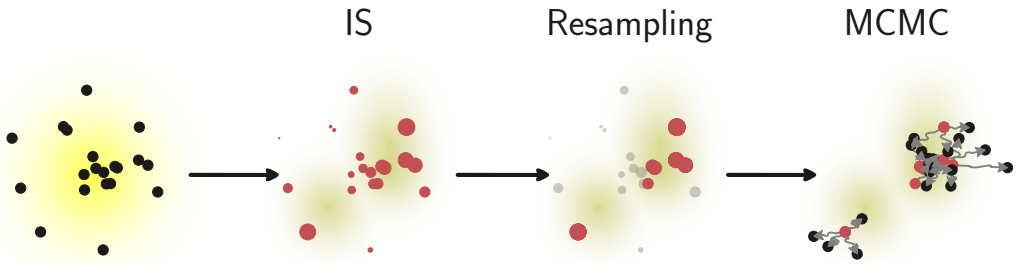
- Annealing: Introduce a sequence of densities π_k interpolating between a proposal p and the target π. β_k controls how π_k and π_{k-1} are close.
- Sequential sampling: Use approximate samples from π_{k-1} to compute approximate samples from π_k.
- Main Advantage: Easy to modify samples from π_{k-1} to get samples from π_k, when π_k is close to π_{k-1}.

SMC steps



- Importance Sampling: re-weights particles from $k-1$ proportionally to $\frac{\pi_k}{\pi_{k-1}}$.
- Resampling: duplicate particles with large weights and discard those with small weights. (Recovers AIS (Neal, 2001) if no resampling).
- MCMC step: Move particles according to a Markov Kernel K_k with invariant distribution π_k: (HM, Gibbs-samplers, etc).

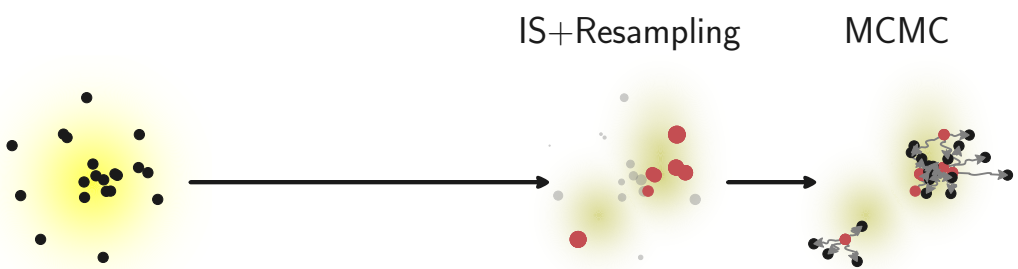
SMC steps



- Estimating normalizing constant Z_k sequentially:

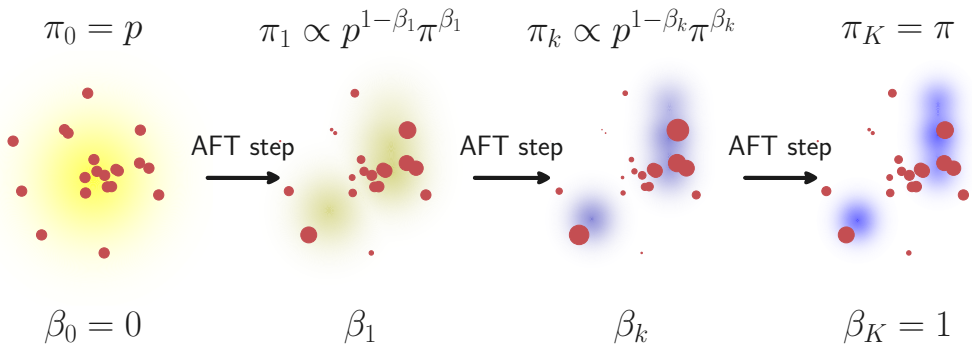
$$Z_k^N := Z_{k-1}^N \left(\sum_{i=1}^{N} W_{k-1}^i \frac{\pi_k\left(X_{k-1}^i\right)}{\pi_{k-1}\left(X_{k-1}^i\right)} \right)$$

SMC steps


IS+Resampling
MCMC
$\frac{W_k^i}{W_{k-1}^i} \propto \frac{\pi_k(X_{k-1}^i)}{\pi_{k-1}(X_{k-1}^i)}$
$\bar{X}_k^i = X_{k-1}^j,\ j \sim \mathsf{Multi}(W_k)$
$X_k^i \sim K_k(\bar{X}_k^i, .)$

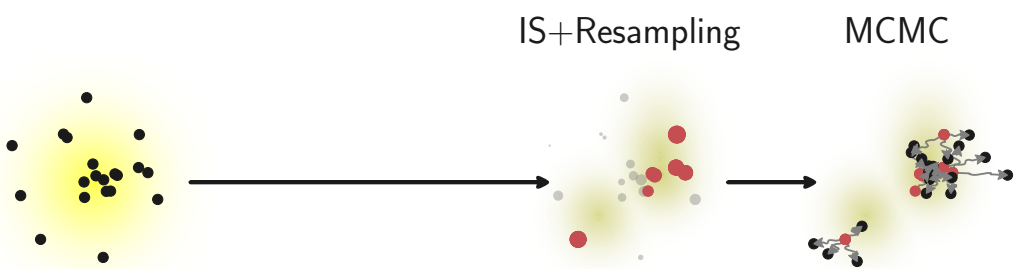
Annealed Flow Transport (AFT)

We combine SMC methods with NFs to gain the best from both approaches.

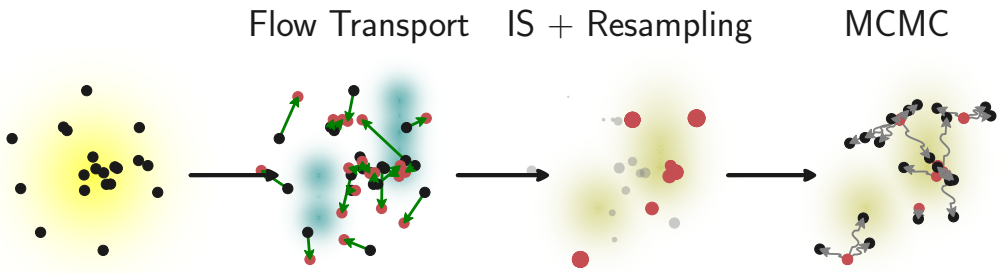


- Similarly to SMC: Introduce a sequence of densities π_k interpolating between a proposal p and the target π.
- Sequential sampling: Use samples from π_{k-1} to compute samples from π_k.
- AFT step: combines a Flow transport step followed by standard SMC steps.

AFT steps with no flow = SMC steps



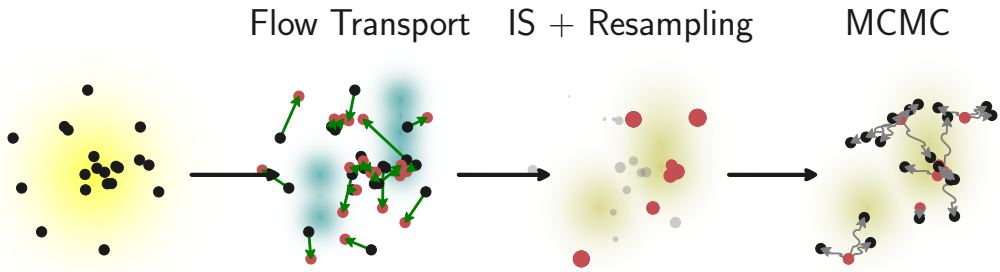
AFT steps with a general flow



- Flow Transport T_k moves X_{k-1}^i to new particles $\tilde{X}_k^i$ close to π_k.
- Closed-form expression for the IS weights to correct for inexact flow:

$$G_k(X, Y) = \frac{\pi_k(Y)}{\pi_{k-1}(X)} |\nabla T_k(X)|$$

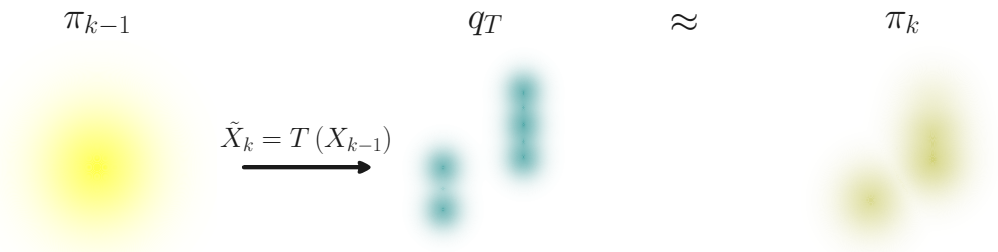
Annealed Flow Transport steps (with a flow)



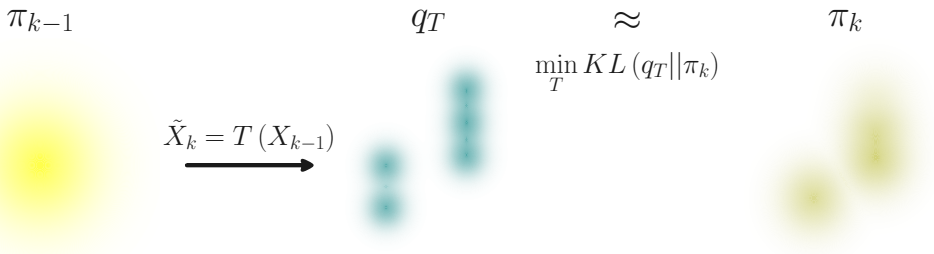
- Estimating normalizing constant Z_t sequentially:

$$Z_k^N := Z_{k-1}^N \left(\sum_{i=1}^N W_{k-1}^i G_k\left(X_{k-1}^i, X_k^i\right) \right)$$

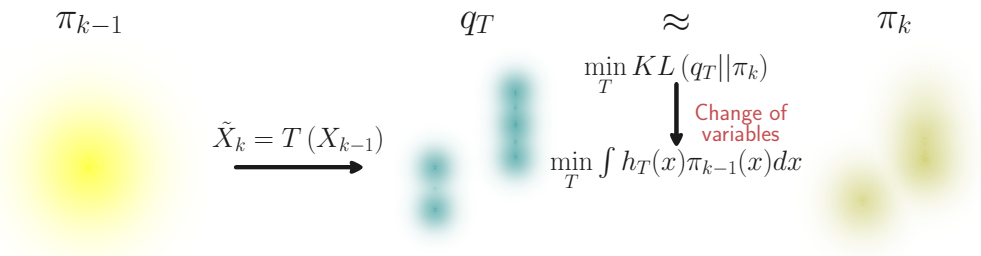
Learning the Normalizing Flows sequentially



Learning the Normalizing Flows sequentially



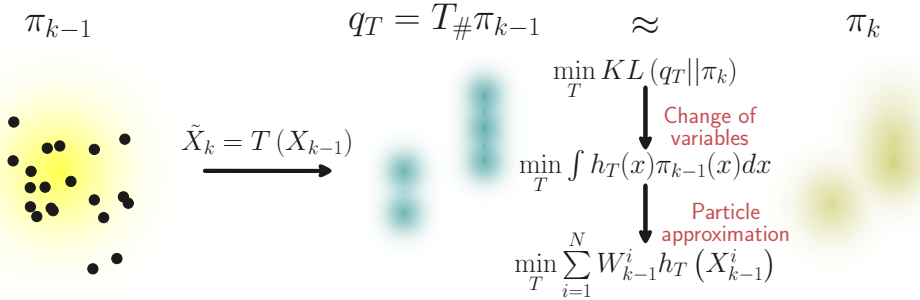
Learning the Normalizing Flows sequentially



- Change of variables: KL as an expectation under π_{k-1} of a function $h_T(x)$

$$h_T(x) = \log \pi_{k-1}(x) - \log \pi_k(T(x)) - \log |\nabla T(x)| + C$$

Learning the Normalizing Flows sequentially

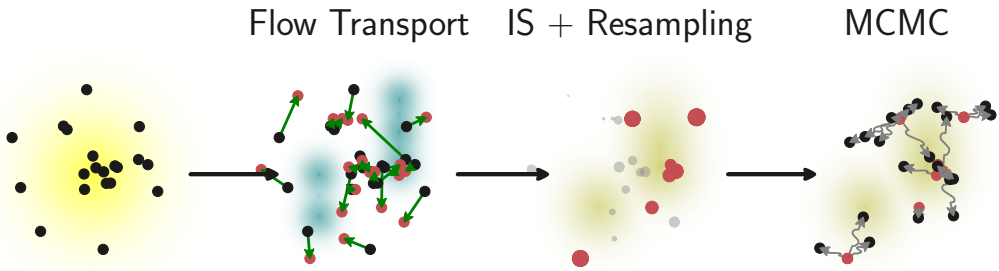


- Change of variables: KL as an expectation under π_{k-1} of a function $h_T(x)$

$$h_T(x) = \log \pi_{k-1}(x) - \log \pi_k(T(x)) - \log |\nabla T(x)| + C$$

- Particle approximation: Use particles X_{k-1}^i and weights W_{k-1}^i to estimate expectation of h_T under π_{k-1}.

Annealed Flow Transport steps (with a flow)



- Given particles $X_{k-1}^i \sim \pi_{k-1}$, learn a flow T_k.
- Compute transported particles $\tilde{X}_k^i$ using flow T_k.
- Apply IS+Resampling and MCMC to get new particles $X_k^i \sim \pi_k$

Part II: Algorithm and Experiments

Algorithm 1 Annealed Flow Transport

Input: number of particles N, unnormalized annealed targets $\{\gamma_k\}_{k=0}^K$ such that $\gamma_0 = \pi_0$ and $\gamma_K = \gamma$, resampling threshold $A \in [1/N, 1)$.
Ouput: Approximations π_K^N and Z_K^N of π and Z.
Sample $X_0^i \sim \pi_0$ and set $W_0^i = \frac{1}{N}$ and $Z_0^N = 1$.
for $k = 1, \ldots, K$ **do**
  Compute $\mathcal{L}_k^N(T)$ using (8).
  Solve $T_k \leftarrow \text{argmin}_{T \in \mathcal{T}} \mathcal{L}_k^N(T)$ using e.g. SGD.
  Transport particles: $\widetilde{X}_k^i = T_k(X_{k-1}^i)$.
  Estimate normalizing constant Z_k:
  $Z_k^N \leftarrow Z_{k-1}^N \left(\sum_{i=1}^N W_{k-1}^i G_{k,T_k}(X_{k-1}^i) \right)$.
  Compute IS weights:
  $w_k^i \leftarrow W_{k-1}^i G_{k,T_k}(X_{k-1}^i)$ // unnormalized
  $W_k^i \leftarrow \frac{w_k^i}{\sum_{j=1}^N w_k^j}$ // normalized
  Compute effective sample size ESS_k^N using (10).
  if $\text{ESS}_k^N / N \leq A$ **then**
    Resample N particles denoted abusively also $\widetilde{X}_k^i$ according to the weights W_k^i, then set $W_k^i = \frac{1}{N}$.
  end if
  Sample $X_k^i \sim K_k(\widetilde{X}_k^i, \cdot)$. // MCMC
end for

- It is possible to overfit to the loss because we use a finite number of particles.
- We would like to have unbiased estimates of normalizing constant as in SMC.

Algorithm 2 Annealed Flow Transport: Detailed Version

Input: Number of training, test and validation particles N_{train}, N_{test}, N_{val}, unnormalized annealed targets $\{\gamma_k\}_{k=0}^{K}$ such that $\gamma_0 = \pi_0$ and $\gamma_K = \gamma$, resampling thresholds $A_a \in [1/N_a, 1)$ for $a \in \{\text{train}, \text{test}, \text{val}\}$, number of training iterations J.

Ouput: Approximations $\pi_K^{N_{\text{test}}}$ and $Z_K^{N_{\text{test}},\text{test}}$ of π and Z.

for $a \in \{\text{train}, \text{test}, \text{val}\}$ **do**

  Sample $X_0^{i,a} \sim \pi_0$ and set $W_0^{i,a} \leftarrow \frac{1}{N_a}$ and $Z_0^{N,a} \leftarrow 1$.

end for

for $k = 1, \ldots, K$ **do**

  Learn the flow $T_k \leftarrow \texttt{LearnFlow}\left(J, \left\{X_{k-1}^{i,\text{train}}, W_{k-1}^{i,\text{train}}\right\}_{i=1}^{N_{\text{train}}}, \left\{X_{k-1}^{i,\text{val}}, W_{k-1}^{i,\text{val}}\right\}_{i=1}^{N_{\text{val}}}\right)$

  for $a \in \{\text{train}, \text{test}, \text{val}\}$ **do**

    Transport particles: $\widetilde{X}_k^{i,a} \leftarrow T_k(X_{k-1}^{i,a})$.

    Estimate normalizing constant Z_k:
    $Z_k^{N_a,a} \leftarrow Z_{k-1}^{N_a,a}\left(\sum_{i=1}^{N_a} W_{k-1}^{i,a} G_{k,T_k}(X_{k-1}^{i,a})\right)$.

    Compute IS weights:
    $w_k^{i,a} \leftarrow W_{k-1}^{i,a} G_{k,T_k}(X_{k-1}^{i,a})$ // unnormalized
    $W_k^{i,a} \leftarrow \frac{w_k^{i,a}}{\sum_{j=1}^{N_a} w_k^{j,a}}$ // normalized

    Compute effective sample size $\text{ESS}_k^{N_a}$
    $\text{ESS}_k^{N_a} \leftarrow \sum_{i=1}^{N_a}\left(W_k^{i,a}\right)^2$.

    if $\text{ESS}_k^{N_a}/N_a \leq A_a$ **then**

      Resample N_a particles from split a denoted abusively also $\widetilde{X}_k^{i,a}$ according to the weights $W_k^{i,a}$,

      Set $W_k^{i,a} \leftarrow \frac{1}{N_a}$.

    end if

    Sample $X_k^{i,a} \sim K_k(\widetilde{X}_k^{i,a}, \cdot)$. // MCMC

  end for

end for

Introduce three sets of particles.
-Train, Validation and Test.

Training set is used to estimate the gradients of the loss.

Validation set is used for early stopping of the loss.

Test set is not used to estimate the flow. Gives unbiased estimates of normalizing constant and robust samples.

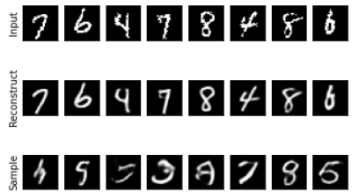
Initialize the flow to the identity which corresponds to SMC.

Evaluation Setup

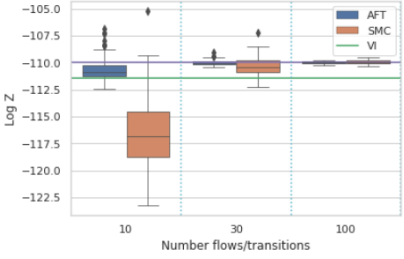
- We use the trained algorithm
- We use number of transitions/flows as a proxy for compute time.
- We use a simple element-wise affine flow. This has a linear memory/time in the dimension.
- Not very expressive though, but worked well in our experiments.

Algorithm: II

Variational Autoencoder Latent Space



Digits that are harder for variational inference.

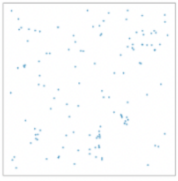


Variational inference works reasonably but is exceeded by SMC and AFT eventually.

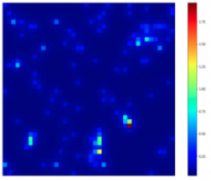
AFT has lower variance than SMC particularly for smaller number of temperatures.

Algorithm: II

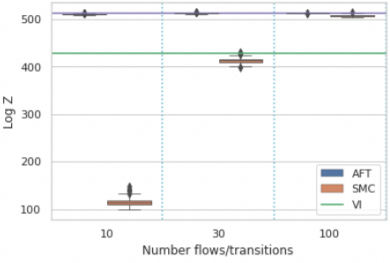
Log Gaussian Cox Process Posterior



Observed positions



Posterior rate



Density: $\gamma(x) = \mathcal{N}(x; \mu, K) \prod_{i \in [1:M]^2} \exp(x_i y_i - a \exp(x_i)).$

Becomes harder as lattice resolution M increases.

We use a 40 x 40 lattice giving 1600 dimensions.

AFT **significantly outperforms baselines**.

All methods could be further tailored.

Part III: Theory

Theory: Consistency and Asymptotic Normality

AFT produces estimates (π_K^N, Z_K^N) of (π, Z) using N particles X_K^i and weights W_K^i.

- Consistency:

$$\pi_K^N[f] \xrightarrow{N} \pi[f],$$
$$Z_K^N \xrightarrow{N} Z.$$

- Central Limit theorem:

$$\sqrt{N}\left(\pi_K^N[f] - \pi[f]\right) \xrightarrow{N} \mathcal{N}(0, V^\pi[f])$$
$$\sqrt{N}\left(Z_K^N - Z\right) \xrightarrow{N} \mathcal{N}(0, V^Z)$$

- Variance is optimal if the flows T_k exactly map π_{k-1} to π_k.
- Extends results of SMC algorithms, but standard proofs do not apply because NFs are stochastic.
- Need uniform CLT $\rightarrow$ Empirical process theory.
- Means controlling the richness of the set of flows: finite entropy numbers.

Theory: Consistency and Asymptotic Normality

AFT produces estimates (π_K^N, Z_K^N) of (π, Z) using N particles X_K^i and weights W_K^i.

- Consistency:

$$\pi_K^N[f] \xrightarrow[p]{N} \pi[f],$$
$$Z_K^N \xrightarrow[p]{N} Z.$$

- Central Limit theorem:

$$\sqrt{N}\left(\pi_K^N[f] - \pi[f]\right) \xrightarrow[D]{N} \mathcal{N}(0, V^\pi[f])$$
$$\sqrt{N}\left(Z_K^N - Z\right) \xrightarrow[D]{N} \mathcal{N}(0, V^Z)$$

- Key challenges:
  - Chaining method/bracketing entropy does not apply because particles are biased and not independent.
  - Uniform entropy method does not require independence but requires strong boundedness conditions otherwise Uniform entropy is infinite.
- Approach: To make it work we introduced a localization technic to the Uniform entropy method: Locally Uniform entropy remains finite.

Scaling limit: Infinitely many auxiliary densities

- Setting:
  - Population limit: Infinitely many particles $N \to +\infty$
  - Continuous-time limit: Infinitely many auxiliary densities $(\pi_k)_{k=1}^K \to (\pi_t)_{[0,1]}$.
  - Use the unadjusted Langevin kernel for K_k: gradient descent on $-\log \pi_t$ + gaussian noise.

Scaling limit: Infinitely many auxiliary densities

- Setting:
  - Population limit: Infinitely many particles $N \to +\infty$
  - Continuous-time limit: Infinitely many auxiliary densities $(\pi_k)_{k=1}^K \to (\pi_t)_{[0,1]}$.
  - Use the unadjusted Langevin kernel for K_k: gradient descent on $-\log \pi_t$ + gaussian noise.
- AFT recovers a weighted controlled diffusion:
  - Sample paths $X_{0,t}$ follows a controlled SDE with control α_t:

$$dX_t = (\alpha_t^\star(X_t) + \nabla \log \pi_t(X_t))dt + \sqrt{2}dB_t$$

Scaling limit: Infinitely many auxiliary densities

- Setting:
  - Population limit: Infinitely many particles $N \to +\infty$
  - Continuous-time limit: Infinitely many auxiliary densities $(\pi_k)_{k=1}^K \to (\pi_t)_{[0,1]}$.
  - Use the unadjusted Langevin kernel for K_k: gradient descent on $-\log \pi_t$ + gaussian noise.
- AFT recovers a weighted controlled diffusion:
  - Sample paths $X_{0,t}$ follows a controlled SDE with control α_t:

$$dX_t = (\alpha_t^\star(X_t) + \nabla \log \pi_t(X_t))dt + \sqrt{2}dB_t$$

  - Sample paths $X_{0,t}$ are re-weighted according to importance weights:

$$w_t^{\alpha^\star}(X_{[0,t]}) := \exp\left(\int_0^t g_s^{\alpha^\star}(X_s)ds\right), \qquad g_s^\alpha(X_s) := div_x(\alpha_t) + \alpha_t^\top \nabla_x \log \pi_t + \partial_t \log \pi_t$$

Scaling limit: Infinitely many auxiliary densities

- Setting:
  - Population limit: Infinitely many particles $N \to +\infty$
  - Continuous-time limit: Infinitely many auxiliary densities $(\pi_k)_{k=1}^K \to (\pi_t)_{[0,1]}$.
  - Use the unadjusted Langevin kernel for K_k: gradient descent on $-\log \pi_t$ + gaussian noise.
- AFT recovers a weighted controlled diffusion:
  - Sample paths $X_{0,t}$ follows a controlled SDE with control α_t:

$$dX_t = (\alpha_t^\star(X_t) + \nabla \log \pi_t(X_t))dt + \sqrt{2}dB_t$$

  - Sample paths $X_{0,t}$ are re-weighted according to importance weights:

$$w_t^{\alpha^\star}(X_{[0,t]}) := \exp\left(\int_0^t g_s^{\alpha^\star}(X_s)ds\right), \qquad g_s^\alpha(X_s) := div_x(\alpha_t) + \alpha_t^\top \nabla_x \log \pi_t + \partial_t \log \pi_t$$

  - Instantaneous work g_s^α measures how much the density of X_t differs from π_t.

Scaling limit: Infinitely many auxiliary densities

- Setting:
  - Population limit: Infinitely many particles $N \to +\infty$
  - Continuous-time limit: Infinitely many auxiliary densities $(\pi_k)_{k=1}^K \to (\pi_t)_{[0,1]}$.
  - Use the unadjusted Langevin kernel for K_k: gradient descent on $-\log \pi_t$ + gaussian noise.
- AFT recovers a weighted controlled diffusion:
  - Sample paths $X_{0,t}$ follows a controlled SDE with control α_t:

$$dX_t = (\alpha_t^\star(X_t) + \nabla \log \pi_t(X_t))dt + \sqrt{2}dB_t$$

  - Sample paths $X_{0,t}$ are re-weighted according to importance weights:

$$w_t^{\alpha^\star}(X_{[0,t]}) := \exp\left(\int_0^t g_s^{\alpha^\star}(X_s)ds\right), \qquad g_s^\alpha(X_s) := div_x(\alpha_t) + \alpha_t^\top \nabla_x \log \pi_t + \partial_t \log \pi_t$$

  - Instantaneous work g_s^α measures how much the density of X_t differs from π_t.
  - Weights ensure the marginals of weighted diffusion match π_t exactly.

Scaling limit: Infinitely many auxiliary densities

- Setting:
  - Population limit: Infinitely many particles $N \to +\infty$
  - Continuous-time limit: Infinitely many auxiliary densities $(\pi_k)_{k=1}^K \to (\pi_t)_{[0,1]}$.
  - Use the unadjusted Langevin kernel for K_k: gradient descent on $-\log \pi_t$ + gaussian noise.
- AFT recovers a weighted controlled diffusion:
  - Sample paths $X_{0,t}$ follows a controlled SDE with control α_t:

$$dX_t = (\alpha_t^\star(X_t) + \nabla \log \pi_t(X_t))dt + \sqrt{2}dB_t$$

  - Sample paths $X_{0,t}$ are re-weighted according to importance weights:

$$w_t^{\alpha^\star}(X_{[0,t]}) := \exp\left(\int_0^t g_s^{\alpha^\star}(X_s)ds\right), \qquad g_s^\alpha(X_s) := div_x(\alpha_t) + \alpha_t^\top \nabla_x \log \pi_t + \partial_t \log \pi_t$$

  - Instantaneous work g_s^α measures how much the density of X_t differs from π_t.
  - Weights ensure the marginals of weighted diffusion match π_t exactly.
  - Optimal control $\alpha^\star$ obtained by minimizing the variance of Instantaneous work:

$$\alpha^\star := \frac{1}{2} \arg\min_\alpha \int_0^1 dt \left(\pi_t[(g_t^\alpha)^2] - \pi_t[g_t^\alpha]^2\right).$$

Conclusion

- AFT extends SMC to take advantage of Normalizing flows.
- Known asymptotic behavior
- Known scaling limit
- In practice, the choice of the NF is problem dependent. Are there more principled ways for such choice? Exploiting symmetries?
- The optimization problem for learning the NF is generally non-convex. Can one consider losses/models that result in convex problem? Kernel methods?

Thank you !

Asymptotic Normality: Assumptions I

Let $\mathcal{C}_2$ be the class of continuous functions with quadratic growth and $\mathcal{LC}_2 \subset \mathcal{C}_2$ satisfying:

$$\left\|f(x) - f(x')\right\| \leq C\left(1 + \|x\|^3 + \left\|x'\right\|^3\right)\|x - x'\|.$$

General assumptions

- The Markov kernel K_k preserves the classes $\mathcal{LC}_2$, $\mathcal{C}_2$
- π_k admit 8-th order moments.
- The potentials $V_k(x) = -\log(\pi_k(x))$ are L-smooth.
- The importance weights $G_{k,T}(x)$ are bounded uniformly over x and T.

Asymptotic Normality: Assumptions I

Let $\mathcal{C}_2$ be the class of continuous functions with quadratic growth and $\mathcal{LC}_2 \subset \mathcal{C}_2$ satisfying:

$$\left\|f(x) - f(x')\right\| \leq C\left(1 + \|x\|^3 + \left\|x'\right\|^3\right)\|x - x'\|.$$

General assumptions

- The Markov kernel K_k preserves the classes $\mathcal{LC}_2$, $\mathcal{C}_2$
- π_k admit 8-th order moments.
- The potentials $V_k(x) = -\log(\pi_k(x))$ are L-smooth.
- The importance weights $G_{k,T}(x)$ are bounded uniformly over x and T.

Assumptions on the NFs

- The NFs T are of the form $T(x) = \tau_\theta(x)$, with θ in a convex compact set.
- The NFs family is non-degenerate: singular values of jacobian are uniformly bounded away from 0.
- $(\theta, x) \mapsto \tau_\theta(x)$ is jointly Lipschitz in θ and x and admits higher order derivatives $\nabla_\theta \tau_\theta(x)$, $\partial_{\theta_i}\partial_{x_l}\tau_\theta(x)$ and $\partial_{\theta_i}\partial_{\theta_j}\partial_{x_l}\tau_\theta(x)$, $H_x\tau_\theta(x)$ with linear growth.

Asymptotic Normality: Assumptions II

Let $\mathcal{L}_k(\theta)$ bet the population loss:

$$\mathcal{L}_k(\theta) = KL\Big((\tau_\theta)_\# \pi_{k-1} || \pi_k\Big)$$

Let θ_k^N be obtained by the algorithm optimizing the particle loss $\mathcal{L}_k^N(\theta)$ (with N particles).

Assumptions on the NF optimizer

- Assume θ_k^N is an approximate local minimizer of $\theta \mapsto \mathcal{L}_k^N(\theta)$:

$$\nabla \mathcal{L}_k^N(\theta_k^N) = o_{\mathbb{P}}(1)$$
$$H\mathcal{L}_k^N(\theta_k^N) \geq o_{\mathbb{P}}(1).$$

- There exits a local minimizer $\theta_k^\star$ of $\mathcal{L}_k$ such that:

$$\mathbb{P}\left[\theta_k^\star \in \arg\min_{\theta \in \Theta_k^\star} \left\|\theta_k^N - \theta\right\|\right] \xrightarrow{N} 1.$$

Set $T_k^\star = \tau_{\theta_k^\star}$ and $T_k^N = \tau_{\theta_k^N}$.

Asymptotic Normality: Main result

Define the unnormalized particle approximation $\gamma_k^N = Z_k^N \pi_k^N$.

Theorem (Version with no adaptive resampling)

Under the previous assumptions, γ_k^N and π_k^N are consistent and for $0 \leq k \leq K$:

$$(CLT_k): \quad \begin{cases} \sqrt{N}\left(\gamma_k^N[f] - \gamma_k[f]\right) & \xrightarrow{\mathcal{D}} \mathcal{N}(0, \mathbb{V}_k^\gamma[f]), \\ \sqrt{N}\left(\pi_k^N[f] - \pi_k[f]\right) & \xrightarrow{\mathcal{D}} \mathcal{N}(0, \mathbb{V}_k^\pi[f]), \end{cases}$$

$\mathbb{V}_k^\gamma[f]$ and $\mathbb{V}_k^\pi[f]$ are defined recursively with $\mathbb{V}_0^\gamma[f] = \mathsf{Var}_{\pi_0}[f]$ and

$$\mathbb{V}_k^\gamma[f] = Z_k^2 \mathsf{Var}_{\pi_k}[f] + \mathbb{V}_{k-1}^\gamma\left[Q_{k,T_k^\star}[f]\right],$$
$$\mathbb{V}_k^\pi[f] = Z_k^{-2} \mathbb{V}_k^\gamma[f - \pi_k[f]],$$

where $Q_{k,T}(x, \mathrm{d}y) := G_{k,T}(x) K_k(T(x), \mathrm{d}y)$.

Asymptotic Normality: Sketch of the proof I

- Need to show: $E_N = \sqrt{N}(\gamma_k^N[f] - \gamma_k[f]) \xrightarrow{P} \mathcal{N}\Big(0, Z_k^2 Var_{\pi_k}[f] + \mathbb{V}_{k-1}^{\gamma}\Big[Q_{k,T_k^\star}[f]\Big]\Big)$.

Proof

- Proof by induction: Assume CLT_{k-1} holds.
- Use decomposition: $E_N = \underbrace{\mathbb{E}_{k-1}[E_N]}_{P_N} + \underbrace{E_N - \mathbb{E}_{k-1}[E_N]}_{R_N}$.
- Conditionally on the past, show that: $\mathbb{E}_{k-1}[e^{itR_N}] \xrightarrow{P} \exp(-\frac{t^2}{2} Z_k^2 Var_{\pi_k}[f])$
- Need to show that P_N is normal with variance $\mathbb{V}_{k-1}^{\gamma}\Big[Q_{k,T_k^\star}[f]\Big]$.
- Can express P_N in terms of π_{k-1}^N and γ_{k-1}^N:

$$P_N = \underbrace{\sqrt{N}\gamma_{k-1}^N[1]\Big(\pi_{k-1}^N - \pi_{k-1}\Big)\Big[Q_{k,T_k^N}[f] - Q_{k,T_k^\star}[f]\Big]}_{A_N} + \underbrace{\sqrt{N}\Big(\gamma_{k-1}^N\Big[Q_{k,T_k^\star}[f]\Big] - \gamma_{k-1}\Big[Q_{k,T_k^\star}[f]\Big]\Big)}_{B_N}$$

- By induction B_N converges to a normal with variance $\mathbb{V}_{k-1}^{\gamma}[Q_{k,T_k^\star}[f]]$.
- If T_k^N was not learned and fixed to $T_k^\star$, then $A_N = 0$ and proof is similar to standard CLT results for SMC.

Asymptotic Normality: Sketch of the proof II

Recall $A_N = \sqrt{N}\gamma_{k-1}^N[1]\left(\pi_{k-1}^N - \pi_{k-1}\right)\left[Q_{k,T_k^N}[f] - Q_{k,T_k^\star}[f]\right]$. Need to show that $A_N \xrightarrow{P} 0$.

Theorem

Let f be in $\mathcal{LC}_2$ and consider the family of function $\mathcal{QG}$ of the form $S_\theta(x) = Q_{k,\tau_\theta}[f](x) - Q_{k,\tau_{\theta^\star}}[f](x)$ indexed by the parameter $\theta \in \Theta$. Under the main assumptions and for any random sequence g^N in $\mathcal{QG}$ such that $\pi_{k-1}\left[(g^N)^2\right] \xrightarrow{P} 0$ it holds that $\sqrt{N}\gamma_{k-1}^N[1]\left(\pi_{k-1}^N - \pi_{k-1}\right)\left[g^N\right] \xrightarrow{P} 0$.

- Result follows by proving asymptotic stochastic equicontinuity of a suitable empirical process.
- Chaining method does not apply here because particles are biased and not independent.
- Uniform entropy method does not require independence but does not apply directly neither because we deal with sets of functions that are not bounded.
- To make it work, we applied a localization technic to the Uniform entropy method.

Asymptotic Normality: Sketch of the proof II

Recall $A_N = \sqrt{N}\gamma_{k-1}^N[1]\left(\pi_{k-1}^N - \pi_{k-1}\right)\left[Q_{k,T_k^N}[f] - Q_{k,T_k^\star}[f]\right]$. Need to show that $A_N \xrightarrow{P} 0$.

Theorem

Let f be in $\mathcal{LC}_2$ and consider the family of function $\mathcal{QG}$ of the form $S_\theta(x) = Q_{k,\tau_\theta}[f](x) - Q_{k,\tau_{\theta^\star}}[f](x)$ indexed by the parameter $\theta \in \Theta$. Under the main assumptions and for any random sequence g^N in $\mathcal{QG}$ such that $\pi_{k-1}\left[(g^N)^2\right] \xrightarrow{P} 0$ it holds that $\sqrt{N}\gamma_{k-1}^N[1]\left(\pi_{k-1}^N - \pi_{k-1}\right)\left[g^N\right] \xrightarrow{P} 0$.

- Can apply this theorem by choosing $g^N = S_{\theta_k^N}$.
- Only remains to show that $\pi_{k-1}[(g^N)^2] \xrightarrow{P} 0$.
- Follows after proving that $\theta_k^N \xrightarrow{P} \theta_k^\star$.