Sequential Core-Set Monte Carlo

Boyan Beronov[1] **Christian Weilbach**[1] **Frank Wood**[1] **Trevor Campbell**[2]

[1]Dept. of Computer Science, University of British Columbia, Vancouver, Canada
[2]Dept. of Statistics, University of British Columbia, Vancouver, Canada

Abstract

Sequential Monte Carlo (SMC) is a general-purpose methodology for recursive Bayesian inference, and is widely used in state space modeling and probabilistic programming. Its *resample-move* variant reduces the variance of posterior estimates by interleaving Markov chain Monte Carlo (MCMC) steps for particle "rejuvenation"; but this requires accessing all past observations and leads to linearly growing memory size and quadratic computation cost. Under the assumption of exchangeability, we introduce *sequential core-set Monte Carlo (SCMC)*, which achieves constant space and linear time by rejuvenating based on sparse, weighted subsets of past data. In contrast to earlier approaches, which uniformly subsample or throw away observations, SCMC uses a novel online version of a state-of-the-art *Bayesian core-set* algorithm to incrementally construct a nonparametric, data- and model-dependent variational representation of the unnormalized target density. Experiments demonstrate significantly reduced approximation errors at negligible additional cost.

1 INTRODUCTION

Sequential Monte Carlo (SMC) [Del Moral et al., 2006, Naesseth et al., 2019, Chopin and Papaspiliopoulos, 2020] is a class of algorithms for sampling from a sequence of target probability distributions that are each known up to normalization. Originally developed for filtering of time series [Handschin and Mayne, 1969], SMC has become a mainstay in a wide variety of applications, from phylogenetic inference to universal probabilistic programming languages [van de Meent et al., 2018, Bouchard-Côté et al., 2019]. SMC creates a set of samples (or *particles*) representing each distribution in the sequence by iterating three key operations: particles may be *reweighted* to account for the next distribution in the sequence, *resampled* to remove particles of very low weight, and *rejuvenated* (or *moved*) by a random perturbation from a Markov kernel to ensure that the population of particles is representative of the next distribution in the sequence.

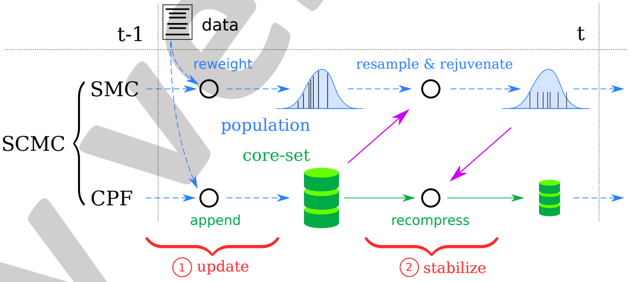


Figure 1: *Sequential core-set Monte Carlo* (SCMC) maintains a hybrid belief representation, consisting of a particle population (blue) and a core-set memory (green). Their coupling (purple) establishes two innovations: the *core-set rejuvenation* kernel for SMC (Section 3.2) and the *core-set projection filter* (CPF) – an online version of Bayesian core-set construction (Section 3.1). During each observation step, (1) the *update* phase takes in new data, reweights the particles and expands the core-set; while (2) the *stabilization* phase resamples and rejuvenates particles to counteract degeneracy and impoverishment, and recompresses the core-set to bound time and space complexity.

In this work, we consider SMC in the setting where there is an additional challenge: the sequence of probability densities $(p_t)_{t=0}^{T}$ becomes more expensive to evaluate as the length T increases. This is the case, for example, in sequential Bayesian inference [Chopin, 2002], where the sequence of probability densities is a set of posterior densities p_t given t batches of data $(X_s)_{s=1}^{t}$. In this setting, the reweighting and resampling steps are typically still straightforward to implement and computationally inexpensive. They both incur a constant $O(K)$ cost for each step in the sequence, given K particles; equivalently, their overall cost scales linearly as $O(KT)$ with the number of steps in the distribution sequence T (see Section 2 for details). Therefore, these steps typically remain tractable as T increases. In contrast, the cost of rejuvenation generally accumulates over time, because the Markov kernel needs to adapt to the target distributions p_t. This effect is demonstrated in Figure 5 for the sequential Bayesian inference setting, where the rejuvenation kernel accesses all past observations. Past work addresses this issue generally by approximate rejuvenation based on a uniformly weighted subset of data [Börschinger and Johnson, 2012, May et al., 2014]; but the subset is selected and constructed without consideration of the model or data itself, leading to poor posterior approximations. While adaptive subsampling ideas [Bardenet et al., 2017, Quiroz et al., 2019] have been applied in the context of pseudo-marginal methods such as particle MCMC, this work focuses on SMC because of its suitability for streaming inference.

We propose a novel variant, *sequential core-set Monte Carlo (SCMC)*, which rejuvenates based on a *Bayesian core-set*—i.e., an optimized weighted subset [Huggins et al., 2016, Campbell and Broderick, 2019])—of all past observations (Figure 1, right-pointing purple arrow). In contrast to previous core-set methods formulated as pre-processing steps for Bayesian inference (see Figure 2), we use the SMC particles themselves to construct the core-set *online* in a "distributionally-aware" manner (Figure 1, left-pointing purple arrow), which minimizes the overall posterior error incurred by the approximate rejuvenation. As the number of observations grows, we iteratively recompress the core-set, resulting in bounded space and time complexity per step. In practice, this enables a consistently better trade-off between computational cost and inference accuracy.

The remainder of the paper proceeds as follows. In Section 2, we first provide a brief overview of SMC. Section 3 introduces the *core-set projection filter (CPF)*, the main technical contribution of this work and key component of SCMC, which enables *core-set rejuvenation*, with bounded resources. After a brief discussion of related work in Section 4, we provide empirical results on both synthetic and real data that demonstrate that SCMC significantly improves posterior approximation accuracy in comparison to subsampling baselines[1].

2 SEQUENTIAL MONTE CARLO

In the setting of the present work, the goal is to take samples from a sequence of probability distributions with densities $(p_t)_{t=0}^T$ starting from a base density p_0, where each probability update $\log(p_t/p_{t-1})$ stems from a log-density or *potential* ℓ_t. The moving target is hence expressed as an

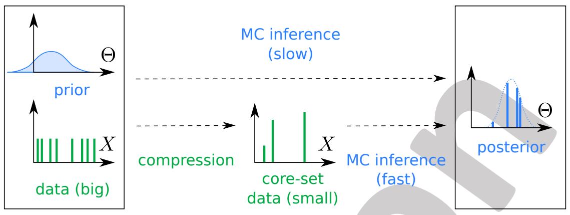


Figure 2: Traditional view of Bayesian core-set compression as a pre-processing step for Monte Carlo inference. Instead, as depicted in Figure 1, we interpret populations and core-sets as complementary online belief representations.

increasing sum across time t,

$$p_t(\theta) \propto p_0(\theta) \exp\left(\sum_{s=1}^{t} \ell_s(\theta)\right). \qquad (1)$$

This general setting includes Bayesian posterior inference in a wide range of models with prior $p_0(\theta)$ on a latent parameter θ. For example, it encompasses all models with conditionally independent data x_t having log-likelihoods $\ell_t(\theta) = \log p(x_t \mid \theta)$, such as logistic regression, mixture models and conjugate exponential family models. It also generalizes many models with non-i.i.d. data, such as order-k auto-regressive time series data x_t with transition log-likelihoods $\ell_t(\theta) = \log p(x_t \mid x_{t-k:t-1}, \theta)$ [Robert and Casella, 2013], and edge-exchangeable network sequences with edges $x_{t,\text{in}}, x_{t,\text{out}}$ having log-likelihoods $\ell_t(\theta) = \log p(x_{t,\text{in}}, x_{t,\text{out}} \,|\, \theta)$ [Cai et al., 2016]. In the technical discourse we will focus on the conditionally i.i.d. data setting for concreteness.

A popular family of methods for sampling from a sequence of unnormalized distributions such as $(p_t)_{t=0}^T$ is sequential Monte Carlo (SMC). Such algorithms generally follow the same pattern, summarized in Section 1.1 and described here specifically for the setting of conditionally i.i.d. data. One begins by simulating or sampling a collection of K parameter values from p_0—often referred to as "particles"—and endowing them with uniform weights,

$$\forall\, k \in [K]: \qquad \pi_k^{(0)} = \frac{1}{K}, \qquad \theta_k^{(0)} \sim p_0, \qquad (2)$$

where $[K] := \{1, \ldots, K\}$. This is generally assumed to be tractable, e.g., when p_0 is a pre-defined Bayesian prior distribution. For each new observation batch $X^{(t)} = (x_b^{(t)})_{b=1}^B$, whose size B reflects a trade-off between computation and posterior estimate variance, three steps occur in sequence. First, the particles are *reweighted* by the batch likelihood to account for the information in the new data, and the weights

[1] Our code is available at `https://github.com/plai-group/scmc`

are renormalized,

$$\forall\, k \in [K]: \quad \bar{\pi}_k'^{(t)} = \pi_k^{(t-1)} \prod_{b=1}^{B} p(x_b^{(t)} \,|\, \theta_k^{(t-1)}) \,,$$

$$\bar{\pi}_k^{(t)} = \frac{\bar{\pi}_k'^{(t)}}{\sum_{l=1}^{K} \bar{\pi}_l'^{(t)}} \,. \qquad (3)$$

This step requires $O(KB)$ computation and memory, which does not increase over time and remains tractable. Next, we check for *particle degeneracy*—i.e., a severe imbalance in particle weights—using a criterion such as *effective sample size* (ESS) [Liu, 2008]. If the check passes, we simply keep the same particles and weights,

$$\forall\, k \in [K]: \qquad \widehat{\pi}_k^{(t)} = \bar{\pi}_k^{(t)} \,, \qquad \widehat{\theta}_k^{(t)} = \theta_k^{(t-1)} \,, \qquad (4)$$

and otherwise *resample* them uniformly,

$$\forall\, k \in [K]: \quad \widehat{\pi}_k^{(t)} = \frac{1}{K} \,, \quad \widehat{\theta}_k^{(t)} \sim \sum_{j=1}^{K} \bar{\pi}_j^{(t)} \delta_{\theta_j^{(t-1)}} \,, \quad (5)$$

where δ_θ is the Dirac measure at θ. This step requires $O(K)$ computation, which again remains tractable. Finally, in order to ensure that the set of particles is always able to represent the current distribution in the sequence p_t, we allow them to *rejuvenate* (or *move* or *mutate*) via a Markov kernel κ applied to each particle,

$$\forall\, k \in [K]: \quad \pi_k^{(t)} = \widehat{\pi}_k^{(t)} \,, \quad \theta_k^{(t)} \sim \kappa(\cdot \mid \widehat{\theta}_k^{(t)}) \,, \quad (6)$$

completing the recursion. Rejuvenation is a critical variance reduction technique for SMC in this setting [Gilks and Berzuini, 2001]. Ideally, one would choose the Markov kernel κ to consist of some number of steps of any *Markov chain Monte Carlo* (MCMC) algorithm with stationary distribution equal to the target p_t at the current step t. This is motivated by the fact that in the limit of infinitely many steps, the particle set will converge to an i.i.d. sample from p_t [Gilks and Berzuini, 2001]. However, this is computationally intractable as it requires access to all previous data even for a single step, and rejuvenation gets more expensive over time. In particular, applying rejuvenation at a single step s requires both $O(s)$ computation and memory, yielding a growing memory cost of $O(t)$ and computation cost of $O(t^2)$ to obtain samples from p_t when starting from p_0. For example, one step of *Metropolis-Hastings* (Section 1.2) for a single particle involves computing the log-joint probability ratio of two parameters θ, θ' across *all past data*,

$$\log \frac{p_t(\theta')}{p_t(\theta)} = \log \frac{p_0(\theta')}{p_0(\theta)} + \sum_{s=1}^{t} \sum_{b=1}^{B} \log \frac{p(x_b^{(s)} \,|\, \theta')}{p(x_b^{(s)} \,|\, \theta)} \,. \quad (7)$$

In this work, we address this computational challenge by reducing the number of observations involved in evaluating the data-dependent term in Equation (7). It is common practice in prior work on rejuvenation to use a subset of data rather than all past data. The data points are usually chosen by random subsampling or reservoir sampling algorithms [May et al., 2014]. Denoting $M \ll TB$ to be the available computational and memory budget for rejuvenation, and $u_1, \ldots, u_M$ to be the selected data subset, all of these cases amount to the following approximation:

$$\log \frac{p_t(\theta)}{p_0(\theta)} \;\approx\; \sum_{m=1}^{M} \frac{tB}{M} \log p(u_m \,|\, \theta) \,. \qquad (8)$$

While computationally efficient, note that the uniform weighting by tB/M in past approaches ignores both the data (some data may better represent the larger collection than others) and the model (some data may not be relevant for inference), resulting in poor approximations in practice. In contrast, in Section 3.1, we provide a method that both selects and weights the data subset adaptively, with the goal of better approximating the total data log-likelihood.

3 SEQUENTIAL CORE-SET MONTE CARLO

3.1 CORE-SET PROJECTION FILTER

In this section, we introduce the key technical innovation that maintains and updates an approximation to the total data log-likelihood — the *core-set projection filter* (CPF), and specify in Algorithm 1 its precise interleaving with SMC. We are given a memory budget $M \in \mathbb{N}$, representing how many total data points we are able to store persistently as SMC proceeds, and we will use core-set techniques to weight them so as to approximate the full likelihood for efficient rejuvenation. We begin with an empty core-set memory of size

$$C^{(0)} = 0 \,. \qquad (9)$$

Now suppose that at time $t-1$, we have a core-set memory with $C^{(t-1)} \le M$ data points $u_j^{(t-1)}$ and weights $w_j^{(t-1)} \ge 0$, $j = 1, \ldots, C^{(t-1)}$, that approximates the log-likelihood of past data, i.e.,

$$\log \frac{p_{t-1}(\theta)}{p_0(\theta)} \;\approx\; \sum_{j=1}^{C^{(t-1)}} w_j^{(t-1)} \log p(u_j^{(t-1)} \,|\, \theta) \,. \qquad (10)$$

Equivalently, $w^{(t-1)} \in \mathbb{R}^{C^{(t-1)}}$ and the component potentials $(\log p(u_j^{(t-1)} \,|\, \cdot))_j^{C^{(t-1)}}$ can be interpreted as the natural parameter and the sufficient statistic of an exponential family, the *core-set posterior family*, which includes the exact posterior at $w^\star = \vec{1} \in \mathbb{R}^{(t-1)\cdot B}$.

Much like in SMC, upon receiving the next batch of data $(x_b^{(t)})_{b=1}^{B}$, two steps occur in sequence. First, the core-set memory is temporarily *expanded* to encompass the new

Algorithm 1 Sequential core-set Monte Carlo (SCMC) for data tempering in exchangeable models

```
procedure SCMC(pop. size $K$, mem. size $M$, proc. REJUV, proc. SNNLS)
        ▷ initialize particles & core-set memory
    $(\pi_k^{(0)}, \theta_k^{(0)}),\ C^{(0)} \leftarrow$ Equations (2) and (9)
    for $t \leftarrow 1, \ldots, T$ do
            ▷ reweight particles & expand core-set
            ▷ with new data batch $X^{(t)}$
        $(\bar{\pi}_k^{(t)})_{k=1}^{K} \leftarrow$ Equation (3)
        $\bar{C}^{(t)},\ (\bar{w}_j^{(t)}, \bar{u}_j^{(t)})_{j=1}^{\bar{C}^{(t)}} \leftarrow$ Equation (11)
            ▷ resample particles
        if ESS$(\bar{\pi}^{(t)}) \geq$ threshold then
            $(\widehat{\pi}_k^{(t)}, \widehat{\theta}_k^{(t)})_{k=1}^{K} \leftarrow$ Equation (4)
        else
            $(\widehat{\pi}_k^{(t)}, \widehat{\theta}_k^{(t)})_{k=1}^{K} \leftarrow$ Equation (5)
        end if
            ▷ rejuvenate particles using core-set
            ▷ (Section 3.2)
        $(\pi_k^{(t)}, \theta_k^{(t)})_{k=1}^{K} \leftarrow$ REJUV$\big((\widehat{\pi}_k^{(t)}, \widehat{\theta}_k^{(t)})_{k=1}^{K}, (\bar{w}_j^{(t)}, \bar{u}_j^{(t)})_{j=1}^{\bar{C}^{(t)}}\big)$
            ▷ recompress core-set using particles
        if $\bar{C}^{(t)} \leq M$ then
            $C^{(t)},\ (w_j^{(t)}, u_j^{(t)})_{j=1}^{C^{(t)}} \leftarrow$ Equation (12)
        else
            $A^{(t)}, b^{(t)}, \widehat{w}^{(t)} \leftarrow$ Equations (14) to (16)
            $C^{(t)},\ (w_j^{(t)}, u_j^{(t)})_{j=1}^{C^{(t)}} \leftarrow$ Equation (17)
        end if
    end for
    return $(\pi_k^{(T)}, \theta_k^{(T)})_{k=1}^{K},\ (w_j^{(T)}, u_j^{(T)})_{j=1}^{C^{(T)}}$
end procedure
```

observations with unit weights,

$$\bar{C}^{(t)} = C^{(t-1)} + B\,, \tag{11}$$

$$(\bar{w}_j^{(t)}, \bar{u}_j^{(t)}) = \begin{cases} (w_j^{(t-1)}, u_j^{(t-1)}) & : 1 \leq j \leq C^{(t-1)} \\ (1,\, x_{j-C^{(t-1)}}^{(t)}) & : C^{(t-1)} < j \leq \bar{C}^{(t)}\,, \end{cases}$$

such that the expanded memory maintains the approximation in Equation (10) for p_t. Second, the core-set is *recompressed* to fit within the memory budget M. If we are able to store all of the new data in addition to the previous core-set memory — i.e., $\bar{C}^{(t)} \leq M$ — then recompression involves no operation,

$$C^{(t)} = \bar{C}^{(t)}\,,$$
$$\forall\, j \in [C^{(t)}]: \quad (w_j^{(t)}, u_j^{(t)}) = (\bar{w}_j^{(t)}, \bar{u}_j^{(t)})\,. \tag{12}$$

On the other hand, if $\bar{C}^{(t)} > M$, we are required to reduce the amount of stored data before continuing. As opposed to past subsampling methods, we formulate this recompression as a sparse nonnegative least-squares (SNNLS) optimization problem proposed in earlier work on data summarization via *Bayesian core-sets* [Huggins et al., 2016, Campbell and Broderick, 2018, 2019],

$$\operatorname*{argmin}_{\widehat{w}\in\mathbb{R}^{\bar{C}^{(t)}}} \mathbb{E}_{p_t}\left[\left(\sum_{j=1}^{\bar{C}^{(t)}} (\widehat{w}_j - \bar{w}_j^{(t)}) \cdot f_j^{(t)}(\cdot\,; p_t)\right)^2\right] \tag{13}$$
$$\text{s.t.} \quad \widehat{w} \geq 0\,, \quad \|\widehat{w}\|_0 \leq M\,,$$
$$f_j^{(t)}(\theta\,; q) \coloneqq \log p(\bar{u}_j^{(t)}\,|\,\theta) - \mathbb{E}_q\Big[\log p(\bar{u}_j^{(t)}\,|\,\cdot)\Big]\,,$$

where the objective is the squared L^2-norm w.r.t. the current posterior p_t, the number of nonzero entries in a vector is denoted as $\|\cdot\|_0$, and $f_j^{(t)}(\cdot\,; q)$ represents the log-likelihood function for the data point $\bar{u}_j^{(t)}$, centered in terms of some weighting distribution q. Note that subsampling and reweighting methods of the form in Equation (8) generate a feasible (but suboptimal) solution to this optimization problem. Intuitively, the optimization problem in Equation (13) attempts to approximate the current estimate of the total log-likelihood function, as represented by weights $\bar{w}_j^{(t)}$, with a sparser set of weights $\hat{w}_j$, while prioritizing regions of the latent space where *the current posterior p_t has significant mass* and where *the potential deviates strongly from its posterior marginal*; Campbell and Beronov [2019] have provided an information-geometric foundation for this type of approach.

Although evaluating the objective in Equation (13) is typically intractable, a key insight in this work is that we can obtain a tractable Monte Carlo approximation from the current rejuvenated SMC posterior approximation with particles $(\theta_k^{(t)})_{k=1}^{K}$ and weights $(\pi_k^{(t)})_{k=1}^{K}$. This corresponds to the sum of squared errors of the log-likelihood function at discretization points $(\theta_k^{(t)})_{k=1}^{K}$. In particular, setting

$$A^{(t)} \in \mathbb{R}^{K \times \bar{C}^{(t)}}, \quad A_{kj}^{(t)} = \sqrt{\pi_k^{(t)}} \cdot f_j^{(t)}(\theta_k^{(t)}\,; \pi^{(t)})\,, \tag{14}$$
$$b^{(t)} \in \mathbb{R}^{K}, \qquad b^{(t)} = A^{(t)}\bar{w}^{(t)}\,, \tag{15}$$

the Monte Carlo approximation of Equation (13), in which the population $q \coloneqq \pi^{(t)} \approx p_t$ is substituted for the true posterior p_t, is equivalent to the following sparse non-negative least squares problem:

$$\texttt{SNNLS:} \quad \operatorname*{argmin}_{\widehat{w}\in\mathbb{R}^{\bar{C}^{(t)}}} \left\| A^{(t)}\widehat{w} - b^{(t)} \right\|_2^2 \tag{16}$$
$$\text{s.t.} \quad \widehat{w} \geq 0\,, \quad \|\widehat{w}\|_0 \leq M\,.$$

Although the cardinality constraint $\|\widehat{w}\|_0 \leq M$ makes this optimization problem intractable to solve exactly, there are numerous efficient off-the-shelf algorithms such as GIGA [Campbell and Broderick, 2018] and orthogonal matching pursuit [Tropp, 2004] which provide computationally efficient approximations with theoretical guarantees, and CPF is agnostic to precisely which approximation algorithm is used.

Rejuvenation	Memory	Time	Error
full	$O(T)$	$O(T^2)$	none
subsampling	$O(T)$	$O(MT)$	stoch.
reservoir	$O(M)$	$O(MT)$	stoch.
core-set	$O(M)$	$O(MT)$	det.

Table 1: Overview of SMC rejuvenation density approximations, for T filtering steps and memory bound M.

The recursion is completed by removing the zero-weight data points from the core-set memory, i.e.,

$$C^{(t)} = \left\|\widehat{w}^{(t)}\right\|_0 ,$$
$$\forall\, j \in [C^{(t)}] : \quad (w_j^{(t)}, u_j^{(t)}) = (\widehat{w}_{i_j}^{(t)}, \bar{u}_{i_j}^{(t)}) \,, \qquad (17)$$

where $i_1, \ldots, i_{C^{(t)}}$ represent the indices of nonzero elements in $\widehat{w}^{(t)}$. Note that the potential evaluations at the particles $\theta_k^{(t)}$ in Equation (14) can be *reused* directly from the last rejuvenation step of SMC in Equation (6). The asymptotic cost of SNNLS solvers is algorithm-specific, but using GIGA for Equation (16) incurs only an additional cost of $O(KM(M + B))$ per filtering stage, *independently* of parameter and observation dimensions. This is optimal for compressing $M + B$ observations into M via pairwise comparisons on K particles. In contrast, reservoir sampling via Algorithm L achieves the data-agnostic optimum of $O(M(1 + \log(TB/M)))$ for T stages. In practice, GIGA consists of efficient linear algebra operations and was negligible in relative runtime across all of our experiments. In general, the ratio of GIGA vs. SMC wall-clock times strongly depends on the model likelihood and the efficiency of the SMC implementation.

3.2 CORE-SET REJUVENATION

Sequential core-set Monte Carlo (SCMC) replaces the total data log-likelihood in the rejuvenation step with the approximation from the expanded core-set memory defined recurrently in Equations (10) and (11). If rejuvenation occurs using the Metropolis-Hastings algorithm with proposal distribution g, for example, we use the approximate acceptance ratio

$$\alpha(\theta', \theta) := \min\left\{1,\ \frac{g(\theta\,|\,\theta')}{g(\theta'\,|\,\theta)} \cdot \frac{p_0(\theta')}{p_0(\theta)} \cdot \prod_{j=1}^{\bar{C}^{(t)}} \frac{p(\bar{u}_j^{(t)}\,|\,\theta')^{\bar{w}_j^{(t)}}}{p(\bar{u}_j^{(t)}\,|\,\theta)^{\bar{w}_j^{(t)}}}\right\} , \qquad (18)$$

see Section 1.2. This term can be evaluated in $O(\bar{C}^{(t)}) = O(M + B)$ time, independently of the data set size $O(T)$. Two details are worth highlighting regarding the numerical precision of SCMC: First, the reweighting steps are unaffected by the core-set approximation, and second, core-set compression is performed using the rejuvenated population $\pi^{(t)}$ with lower variance than $\bar{\pi}^{(t)}$ and $\hat{\pi}^{(t)}$.

Under suitable regimes, core-set rejuvenation achieves lower error than subsampling rejuvenation by introducing a bias which depends on the memory size. We defer to future work the extension of theoretical error bounds from the batch pre-processing [Campbell and Broderick, 2018, 2019] to the online setting of CPF, but note that SCMC behavior depends strongly on the intrinsic compressibility of the potential, as reflected in the constants in GIGA's convergence analysis Campbell and Broderick [2018]. Essential failure regimes for SCMC are too small memory size (projection error of SNNLS), too few particles (discretization error of potential functions) and numerical instability of SMC (e.g., maladapted transition or rejuvenation proposals). Adding memory is a simple mitigation when possible, and corresponds to an isometric immersion of the core-set posterior manifold, but the problem of estimating a sufficient core-set size is of comparable difficulty to assessing variational approximations. In practice, approximation quality can be improved during SCMC by including tempering steps, and at the end by recomputing a fresh core-set using GIGA on the final population and by adding further MCMC steps.

4 RELATED WORK

Bayesian Sparsity Subsampling has been proposed as a way to reduce the cost of internal representations in various approximate Bayesian inference methods [van de Meent et al., 2014, Franck and Koutsourelakis, 2016, Bardenet et al., 2017, Quiroz et al., 2019, Gunawan et al., 2020, Prangle, 2020]. Within restricted model families, several methods have been suggested for sparsity-inducing Bayesian learning by means of greedy or regularized optimization, including *sparse online Gaussian processes* [Csató and Opper, 2002], which construct representative data subsamples for GP models using an RKHS norm, *relevance vector machines* [Tipping, 2001] and *Wasserstein barycenters* [Srivastava et al., 2015] for sparse combinations of subset posteriors. In [Campbell and Beronov, 2019], Bayesian core-set construction was reformulated as a *sparse variational inference* problem suitable for Riemannian stochastic gradient descent, in the batch setting for arbitrary exchangeable models. SCMC instead is incremental both in the core-set construction and in the underlying MC method.

Recursive Inference *Streaming variational inference* methods [Friston, 2008, Broderick et al., 2013, Marino et al., 2018] achieve constant cost updates, but are asymptotically inconsistent and non-adaptive in the general setting by relying on model-specific, conjugate or stochastic gradient updates in variational families, except for non-parametric variants [Campbell et al., 2015]. SMC methods are non-parametric and asymptotically consistent, but practitioners typically need to choose between quickly diverging online methods [Kitagawa, 1998], increasing population sizes or rejuvenation costs [Chopin et al., 2011]. The closest frame-

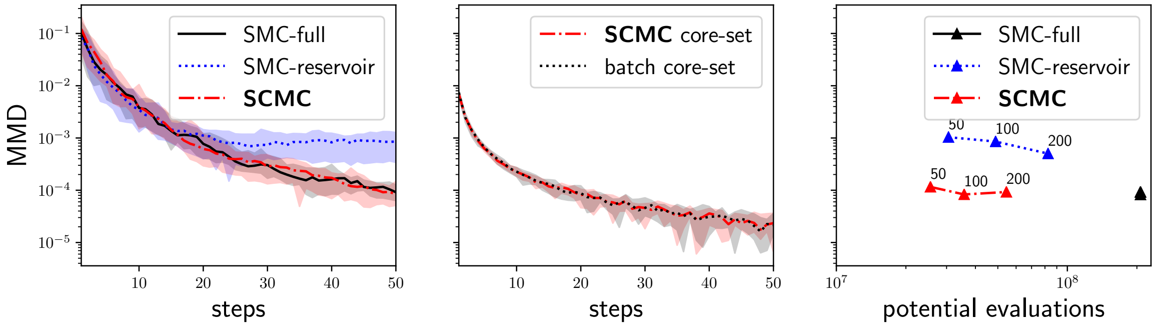


Figure 3: *(left)* Comparison of approximation errors for a 6-dim. normal-inverse-Wishart (NIW) model on 1,000 synthetic data points using 2,000 particles and a memory size of 100, depicting the median of 10 runs for each configuration, see Section 5.2. The accuracy of SCMC is indistinguishable from unbounded SMC. *(middle)* Online core-set posterior errors from SCMC match the oracle, i.e., the sequence of errors for core-sets of equal size, constructed in batch mode using true posterior samples. *(right)* Scaling of approximation error with the cost of potential evaluations for varying memory sizes. SCMC achieves higher accuracy and even lower runtime than the baseline of reservoir sampling rejuvenation, see text.

works to SCMC are *information geometric nonlinear filtering* [Kulhavý, 1996, Newton, 2018] and *projection filtering* [Brigo et al., 1995], which recursively project posterior updates onto finite-dimensional exponential or mixture families or more general statistical manifolds. The essential innovation in SCMC is the use of dual belief representations, comprising mixture (particle populations) and exponential families (core-set posteriors), which are adapted online in mutual recursion. This eliminates the need for up-front construction of approximation bases or summary statistics, and enables a flexible choice both of the rejuvenation kernel inside SMC and of the SNNLS solver for CPF.

Continual Learning Recently, several Bayesian continual learning models were proposed based on the idea of adaptively selecting past observations Nguyen et al. [2017], Pan et al. [2020]. The comparison to our method is similar as above, i.e., both SMC and CPF allow for more general model classes.

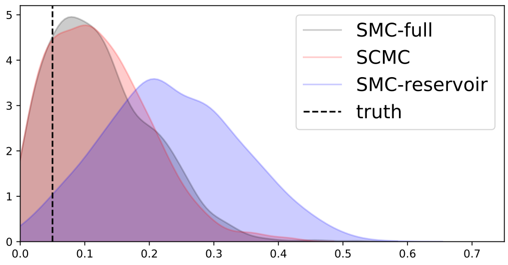


Figure 4: Comparison of approximate posteriors for the AR(1) model in Section 5.1, using Gaussian kernel density estimates from 1,000 particles. Reservoir sampling significantly deteriorates rejuvenation even in this simple scenario.

5 EXPERIMENTS

We evaluate SCMC for sequential inference, using a slight generalization of GIGA [Campbell and Broderick, 2018] as the SNNLS solver (see Section 1.3), on two illustrative synthetic experiments, on real data for Bayesian logistic regression and for filtering in stationary time series models. In all cases, our method is compared against two variants of SMC rejuvenation: the full data oracle (SMC-full) and a reservoir sampling baseline (SMC-reservoir) of the same memory size as SCMC. Details and tuning parameters for all experiments are provided in Section 2.

5.1 AUTO-REGRESSIVE PROCESS

As explained in Section 2, apart from static i.i.d. data, SCMC is also suitable for inference in more general models in which the likelihood can be factorized into exchangeable terms. We provide a simple demonstration on a 1-dim. AR(1) model [Robert and Casella, 2013],

$$\begin{aligned}\phi &\sim \mathcal{U}([0,1])\\ x_0 &\sim \mathcal{N}(0,\sigma_0)\\ \epsilon_{t+1} &\sim \mathcal{N}(0,1)\\ x_{t+1} &= \phi \cdot x_t + \epsilon_{t+1}\,.\end{aligned}$$

In this model, the posterior over ϕ is independent of the initial state x_0 and can be written as

$$\begin{aligned}&p(x_{t+1}\,|\,x_t,\phi) = \mathcal{N}(\phi\cdot x_t, 1)\\ &p(\phi\,|\,\{x_t\}_{t=0}^{T}) \propto \exp\left(\sum_{t=0}^{T}\log p(x_{t+1}\,|\,x_t,\phi)\right)\,. \qquad (19)\end{aligned}$$

Therefore, core-set compression can be applied to the pairs (x_t, x_{t+1}) of the sum in Equation (19). For simplicity, we choose the distribution over x_0 to be the stationary distribution of the process with variance $\sigma_0 = \frac{2}{1-\phi^2}$. 200 pairs

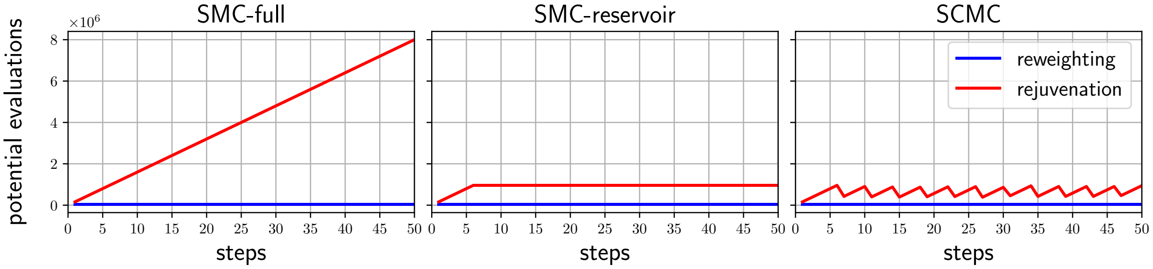


Figure 5: Potential evaluations per filtering stage of the NIW experiment in Section 5.2, which occur only in the reweighting and rejuvenation steps of SMC, as the SNNLS problem construction in Equation (14) reuses the rejuvenated samples. *(left)* unbounded, *(middle)* reservoir sampling, and *(right)* core-set rejuvenation. See text for an explanation of the jigsaw pattern.

(x_t, x_{t+1}) are sampled from the model with true parameter $\phi = 0.05$, and inference is performed using $1{,}000$ particles and a memory size of only 5. Figure 4 demonstrates that on this model, SCMC is very close in performance to SMC with full rejuvenation, and the reservoir sampling baseline performs considerably worse.

5.2 NORMAL-INVERSE-WISHART

The next synthetic experiment is performed on a 6-dim. normal-inverse-Wishart model with 27 parameters, using 1,000 observations in batches of $B = 20$, which are sampled from a ground truth distribution in the conjugate family. SCMC is allocated $K = 2{,}000$ particles and memory sizes $M = 50, 100, 200$. Approximation error is measured in terms of maximum mean discrepancy (MMD) [Gretton et al., 2012] using a Gaussian RBF kernel. See Section 2.2.3 for details.

Figure 3 displays the posterior approximation error of the sequential inference methods, as a function both of the number of batches and of the cumulative number of log-likelihood evaluations. As expected, in early steps with few total observations, all methods perform identically; whereas after about 20 steps, the reservoir sampling baseline becomes visibly less accurate than both full-rejuvenation SMC and SCMC. Moreover, the chain of core-set posteriors from SCMC obtains significantly lower MMD than the particle population it was constructed with, and is indistinguishable from the behaviour of batch core-set construction on the entire data set when provided with exact posterior samples.

Figure 5 highlights the significant cost reduction provided by SCMC for this experiment in terms of log-likelihood evaluations. While the cost of the full rejuvenation kernel dominates SMC and grows linearly with the number of observations, resampling rejuvenation and SCMC respect a constant bound. As evidenced by the jigsaw pattern alternating between linear filling up and recompression steps, as well as by the lower number of potential evaluations in Figure 3, the computation time in SCMC sometimes can be reduced even further, as core-set compression reaches its numerical tolerance threshold before saturating the memory budget. Such behavior is observable in the regimes of simple models, high data redundancy or too few particles.

5.3 LOGISTIC REGRESSION

We compare the same procedures on a Bayesian logistic regression model for the `ChemReact`[2] binary classification data set (10 features, 25,000 entries). Inference is performed in batches of size $B = 50$, using population sizes $K = 1{,}500; 3{,}000; 6{,}000$, and memory sizes $M = 75; 150; 300$. Approximation error is measured in terms of the symmetric KL divergence between Gaussian fits to the SCMC posterior populations and to the ground truth posteriors obtained using 10,000 MCMC samples from STAN [Carpenter et al., 2017].

The empirical results in Figure 6 indicate asymptotic consistency both in population and memory size under suitable conditions. While increasing amounts of data will inevitably introduce increasing approximation error into SMC rejuvenation for any fixed-size memory compression scheme, CPF demonstrates a significant improvement over reservoir sampling, establishing a new trade-off between inference accuracy and computation cost. Notably, the overall numerical accuracy shows a much stronger dependency on memory size than on population size, and in contrast to the high variance in the unbiased reservoir sampling baseline, SCMC converges gracefully towards the SMC oracle.

5.4 MLP TIME SERIES PREDICTION

We now extend the demonstration in Section 5.1 of inference over stationary time series model parameters to a more challenging scenario: online posterior inference in a multi-layer perceptron (MLP) model on real data. Specifically, we provide a sequence of time windows to SCMC, performing inference over parameters of the linear output layer, based on a feed-forward NN feature embedding trained beforehand by MLE with an L2 loss. Furthermore, rejuvenation uses weighted NUTS instead of weighted MH, showcasing the

[2] `http://komarix.org/ac/ds/`

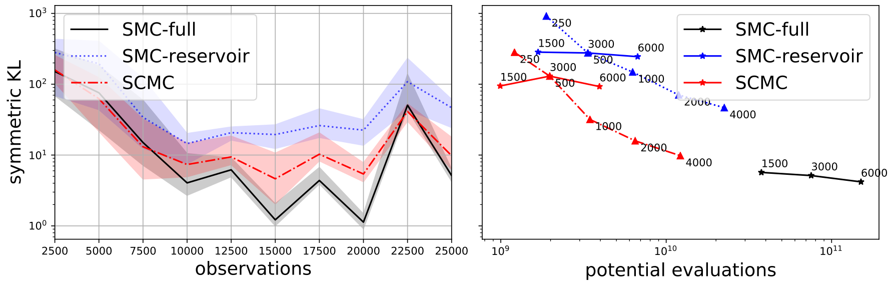
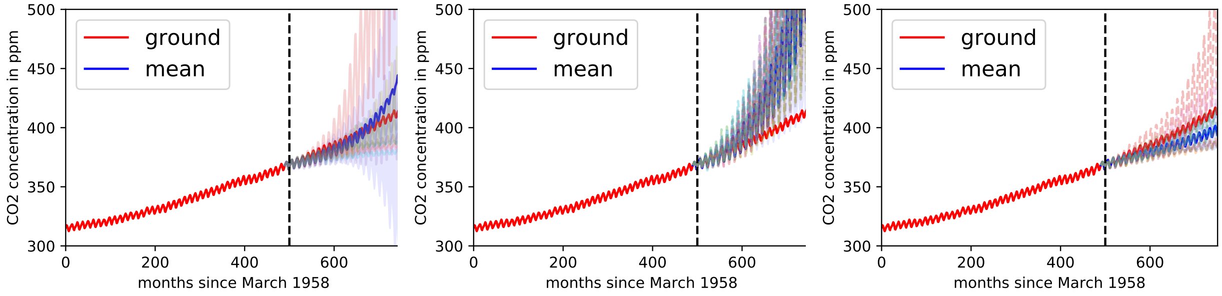


Figure 6: *(left)* Comparison akin to Figure 3 for 10-dim. Bayesian logistic regression on the `ChemReact` data set with 25,000 entries, using 3,000 particles and a memory size of 4,000, depicting the median of 10 runs for each configuration, see Section 5.3. SCMC achieves significantly lower error and variance than reservoir sampling rejuvenation. *(right)* Scaling of approximation error with the cost of potential evaluations for memory sizes 250; 500; 1,000; 2,000; 4,000 (*steep lines*, at population size of 3,000) and for population sizes 1,500, 3,000 and 6,000 (*flat lines*, at memory size of 500).



(a)

(b)

(c)

Figure 7: Approximate samples from the posterior over linear output layers of an MLP time series model, see Section 5.4 (input/output: 10/1 months, hidden layers: 20/20, activations: ReLU, initialization: Xavier, fixed std. of Gaussian observation model / std. of input: 10^{-1}), tested on the Keeling Curve data set of CO_2 measurements from the Mauna Loa Observatory. Features (hidden layers) are trained via MLE on months 0-490, and samples are shown as expected recurrent roll-outs over months 501-750, taking as input the months 491-500. $K = 300, M = 20, B = 20$. Shown are typical behaviors of: *(a)* NUTS on core-set from SCMC, *(b)* SMC-reservoir and *(c)* SCMC with NUTS rejuvenation.

flexibility of SCMC.

For a single data set, the Keeling Curve of CO_2 measurements from the Mauna Loa Observatory, Figure 7 depicts typical behavior of the samplers under comparison, now also including as a benchmark the No-U-Turn Sampler (NUTS), on the core-set output from SCMC. Measured in terms of posterior marginal likelihood and averaged over 10 independent runs each, we observe more plausible uncertainty calibration on held-out data using SCMC ($-1.9088 \cdot 10^6 \pm 71.0 \cdot 10^3$) than using SMC-reservoir ($-1.9090 \cdot 10^6 \pm 345.7 \cdot 10^3$). SCMC with NUTS rejuvenation performs nearly identically to NUTS on the full data, whereas SMC-reservoir has higher variance, and NUTS alone on the core-set result of SCMC incurs higher error than SCMC, highlighting the benefit of our dual belief representation.

6 CONCLUSION

This work introduced *sequential core-set Monte Carlo* (SCMC), a streaming algorithm for recursive Bayesian inference in a broad class of models that significantly increases the level of accuracy obtainable within fixed resource bounds. SCMC uses a novel rejuvenation kernel of constant cost, constructed by iteratively recompressing past observations into Bayesian core-sets. This *core-set projection filtering* (CPF) procedure is the first streaming algorithm in the family of general-purpose Bayesian core-set methods. Potential directions for future work include generalizations to non-exchangeable and non-stationary Bayesian models, online adaptation based on a theoretical analysis of the cost–accuracy trade-off among belief representations in SMC and CPF, and distributed streaming inference using core-sets for communicating compressed updates.

Acknowledgements

The authors wish to express their gratitude to the anonymous reviewers, as well as to Robert Lieck, Berend Zwartsenberg and Adam Scibior for valuable feedback.

We acknowledge the support of the Natural Sciences and Engineering Research Council of Canada (NSERC) via two Discovery Grants (Contract Nos. RGPIN-2018-05022 and RGPIN-2019-03962) and an Accelerator Grant (Contract No. RGPAS-2018-522582), the Canada CIFAR AI Chairs Program, and the Intel Parallel Computing Centers program.

This material is based upon work supported by the United States Air Force Research Laboratory (AFRL) under the Defense Advanced Research Projects Agency (DARPA) Data Driven Discovery Models (D3M) program (Contract No. FA8750-19-2-0222) and Learning with Less Labels (LwLL) program (Contract No. FA8750-19-C-0515). Additional support was provided by UBC's Composites Research Network (CRN), Data Science Institute (DSI) and Support for Teams to Advance Interdisciplinary Research (STAIR) Grants. This research was enabled in part by technical support and computational resources provided by WestGrid (https://www.westgrid.ca/) and Compute Canada (www.computecanada.ca).

The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of DARPA or the U.S. Government.

References

Rémi Bardenet, Arnaud Doucet, and Chris Holmes. On Markov chain Monte Carlo methods for tall data. *The Journal of Machine Learning Research*, 18(1):1515–1557, January 2017. ISSN 1532-4435.

Benjamin Börschinger and Mark Johnson. Using rejuvenation to improve particle filtering for bayesian word segmentation. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 85–89, 2012.

Alexandre Bouchard-Côté, Kevin Chern, Davor Cubranic, Sahand Hosseini, Justin Hume, Matteo Lepur, Zihui Ouyang, and Giorgio Sgarbi. Blang: Bayesian declarative modelling of arbitrary data structures. *arXiv:1912.10396 [stat]*, December 2019.

Damiano Brigo, Bernard Hanzon, and François Le Gland. A differential geometric approach to nonlinear filtering: The projection filter. In *Proceedings of the 34th Conference on Decision and Control, New Orleans 1995*, volume 4, pages 4006–4011, 1995.

Tamara Broderick, Nicholas Boyd, Andre Wibisono, Ashia C Wilson, and Michael I Jordan. Streaming Variational Bayes. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 1727–1735. Curran Associates, Inc., 2013.

Diana Cai, Trevor Campbell, and Tamara Broderick. Edge-exchangeable graphs and sparsity. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 4249–4257. Curran Associates, Inc., 2016.

Trevor Campbell and Boyan Beronov. Sparse Variational Inference: Bayesian Coresets from Scratch. In H. Wallach, H. Larochelle, A. Beygelzimer, F. Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 11461–11472. Curran Associates, Inc., 2019.

Trevor Campbell and Tamara Broderick. Bayesian Coreset Construction via Greedy Iterative Geodesic Ascent. In *International Conference on Machine Learning*, pages 698–706, July 2018.

Trevor Campbell and Tamara Broderick. Automated scalable Bayesian inference via Hilbert coresets. *The Journal of Machine Learning Research*, 20(1):551–588, January 2019. ISSN 1532-4435.

Trevor Campbell, Julian Straub, John W Fisher III, and Jonathan P How. Streaming, Distributed Variational Inference for Bayesian Nonparametrics. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 280–288. Curran Associates, Inc., 2015.

Bob Carpenter, Andrew Gelman, Matthew D. Hoffman, Daniel Lee, Ben Goodrich, Michael Betancourt, Marcus Brubaker, Jiqiang Guo, Peter Li, and Allen Riddell. Stan: A Probabilistic Programming Language. *Journal of Statistical Software*, 76(1):1–32, January 2017. ISSN 1548-7660. doi: 10.18637/jss.v076.i01.

N. Chopin. A sequential particle filter method for static models. *Biometrika*, 89(3):539–552, August 2002. ISSN 0006-3444, 1464-3510. doi: 10.1093/biomet/89.3.539.

Nicolas Chopin and Omiros Papaspiliopoulos. *An Introduction to Sequential Monte Carlo*. Springer Series in Statistics. Springer International Publishing, 2020. ISBN 978-3-030-47844-5. doi: 10.1007/978-3-030-47845-2.

Nicolas Chopin, Pierre E. Jacob, and Omiros Papaspiliopoulos. Smc2: An efficient algorithm for sequential analysis of state-space models. *arXiv:1101.1528 [stat]*, January 2011.

Lehel Csató and Manfred Opper. Sparse on-line Gaussian processes. *Neural computation*, 14(3):641–668, 2002.

Pierre Del Moral, Arnaud Doucet, and Ajay Jasra. Sequential Monte Carlo samplers. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(3): 411–436, June 2006. ISSN 1369-7412, 1467-9868. doi: 10.1111/j.1467-9868.2006.00553.x.

Isabell M. Franck and P. S. Koutsourelakis. Sparse Variational Bayesian approximations for nonlinear inverse problems: Applications in nonlinear elastography. *Computer Methods in Applied Mechanics and Engineering*, 299:215–244, February 2016. ISSN 0045-7825. doi: 10.1016/j.cma.2015.10.015.

K.J. Friston. Variational filtering. *NeuroImage*, 41(3): 747–766, July 2008. ISSN 10538119. doi: 10.1016/j.neuroimage.2008.03.017.

Walter R. Gilks and Carlo Berzuini. Following a moving target—Monte Carlo inference for dynamic Bayesian models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(1):127–146, 2001. ISSN 1467-9868. doi: 10.1111/1467-9868.00280.

Arthur Gretton, Karsten M Borgwardt, Malte J Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *Journal of Machine Learning Research*, 13 (Mar):723–773, 2012.

David Gunawan, Khue-Dung Dang, Matias Quiroz, Robert Kohn, and Minh-Ngoc Tran. Subsampling Sequential Monte Carlo for Static Bayesian Models. *arXiv:1805.03317 [stat]*, March 2020.

J. E. Handschin and D. Q. Mayne. Monte Carlo techniques to estimate the conditional expectation in multi-stage nonlinear filtering. *International Journal of Control*, 9(5): 547–559, May 1969. ISSN 0020-7179. doi: 10.1080/00207176908905777.

Jonathan Huggins, Trevor Campbell, and Tamara Broderick. Coresets for Scalable Bayesian Logistic Regression. In *Advances in Neural Information Processing Systems*, pages 4080–4088, 2016.

Genshiro Kitagawa. A Self-Organizing State-Space Model. *Journal of the American Statistical Association*, 93(443): 1203–1215, 1998. ISSN 0162-1459. doi: 10.2307/2669862.

Rudolf Kulhavý. *Recursive Nonlinear Estimation: A Geometric Approach*, volume 216. Springer, 1996.

Jun S. Liu. *Monte Carlo Strategies in Scientific Computing*. Springer Science & Business Media, 2008. ISBN 0-387-76369-4.

Joseph Marino, Milan Cvitkovic, and Yisong Yue. A General Method for Amortizing Variational Filtering. In *Advances in Neural Information Processing Systems*, pages 7857–7868, 2018.

Chandler May, Alex Clemmer, and Benjamin Van Durme. Particle Filter Rejuvenation and Latent Dirichlet Allocation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 446–451, Baltimore, Maryland, 2014. Association for Computational Linguistics. doi: 10.3115/v1/P14-2073.

Christian A. Naesseth, Fredrik Lindsten, and Thomas B. Schön. Elements of Sequential Monte Carlo. *Foundations and Trends® in Machine Learning*, 12(3):187–306, 2019. ISSN 1935-8237, 1935-8245. doi: 10.1561/2200000074.

Nigel J. Newton. Nonlinear Filtering and Information Geometry: A Hilbert Manifold Approach. In Nihat Ay, Paolo Gibilisco, and František Matúš, editors, *Information Geometry and Its Applications*, Springer Proceedings in Mathematics & Statistics, pages 189–208, Cham, 2018. Springer International Publishing. ISBN 978-3-319-97798-0. doi: 10.1007/978-3-319-97798-0_7.

Cuong V. Nguyen, Yingzhen Li, Thang D. Bui, and Richard E. Turner. Variational Continual Learning. *arXiv:1710.10628 [cs, stat]*, October 2017.

Pingbo Pan, Siddharth Swaroop, Alexander Immer, Runa Eschenhagen, Richard E. Turner, and Mohammad Emtiyaz Khan. Continual Deep Learning by Functional Regularisation of Memorable Past. *arXiv:2004.14070 [cs, stat]*, November 2020.

Dennis Prangle. Distilling importance sampling. *arXiv:1910.03632 [stat]*, February 2020.

Matias Quiroz, Robert Kohn, Mattias Villani, and Minh-Ngoc Tran. Speeding Up MCMC by Efficient Data Subsampling. *Journal of the American Statistical Association*, 114(526):831–843, April 2019. ISSN 0162-1459. doi: 10.1080/01621459.2018.1448827.

Christian Robert and George Casella. *Monte Carlo Statistical Methods*. Springer Science & Business Media, 2013. ISBN 1-4757-3071-3.

Sanvesh Srivastava, Volkan Cevher, Quoc Dinh, and David Dunson. WASP: Scalable Bayes via barycenters of subset posteriors. In *Artificial Intelligence and Statistics*, pages 912–920, 2015.

Michael E. Tipping. Sparse Bayesian Learning and the Relevance Vector Machine. *Journal of Machine Learning Research*, 1(Jun):211–244, 2001. ISSN ISSN 1533-7928.

J.A. Tropp. Greed is Good: Algorithmic Results for Sparse Approximation. *IEEE Transactions on Information Theory*, 50(10):2231–2242, October 2004. ISSN 0018-9448. doi: 10.1109/TIT.2004.834793.

Jan-Willem van de Meent, Brooks Paige, and Frank Wood. Tempering by Subsampling. *arXiv:1401.7145 [stat]*, January 2014.

Jan-Willem van de Meent, Brooks Paige, Hongseok Yang, and Frank Wood. An Introduction to Probabilistic Programming. *arXiv:1809.10756 [cs, stat]*, September 2018.