A. Appendix

This appendix complements the paper "On the sampling problem for kernel quadrature". Section A.1 discusses the potential lack of robustness of greedy optimization methods, which motivated the development of SMC−KQ. Sections A.2 and A.3 discuss some of the theoretical aspects of KQ, whilst Section A.4 and A.5 presents additional numerical experiments and details for implementation. Finally, Section A.6 provides detailed pseudo-code for all algorithms used in this paper.

A.1. Lack of Robustness of Optimisation Methods

To demonstrate the non-robustness to mis-specified kernels, that is a feature of optimisation-based methods, we considered integration against $\Pi = \mathrm{N}(0,1)$ for functions that can be approximated by the kernel $k(x,x') = \exp(-(x-x')^2/\ell^2)$. An initial state x_1 was fixed at the origin and then for $n = 2,3,\ldots$ the state x_n was chosen to minimise the error criterion $e_n(\boldsymbol{w}; \{x_j\}_{j=1}^n)$ given the location of the $\{x_j\}_{j=1}^n$. This is known as 'sequential Bayesian quadrature' (SBQ; Huszar and Duvenaud, 2012; Gunter et al., 2014; Briol et al., 2015a). The kernel length scale was fixed at $\ell = 0.01$ and we consider (as a thought experiment, since it does not enter into our selection of points) a more regular integrand, such as that shown in Fig. 5 (top). The location of the states $\{x_j\}_{j=1}^n$ obtained in this manner are shown in Fig. 5 (bottom). It is clear that SBQ is not an efficient use of computation for integration of the integrand against $\mathrm{N}(0,1)$. Of course, a bad choice of kernel length scale parameter ℓ can in principle be alleviated by kernel learning, but this will not be robust the case where n is very small.

This example motivates sampling-based methods as an alternative to optimisation-based methods. Future work will be required to better understand when methods such as SBQ can be reliable in the presence of unknown kernel parameters, but this was beyond the scope of this work.

A.2. Additional Definitions

The space $L_2(\Pi)$ is defined to be the set of Π-measurable functions $f : \mathcal{X} \to \mathbb{R}$ such that the Lebesgue integral

$$\int_{\mathcal{X}} f^2 \, \mathrm{d}\Pi$$

exists and is finite.

For a multi-index $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_d)$ define $|\boldsymbol{\alpha}| = \alpha_1 + \cdots + \alpha_d$. The (standard) Sobolev space of order $s \in \mathbb{N}$ is denoted

$$\mathbb{H}_s(\Pi) = \{f : \mathcal{X} \to \mathbb{R} \text{ s.t. } (\partial x_1)^{\alpha_1} \ldots (\partial x_d)^{\alpha_d} f \in L_2(\Pi) \; \forall \, |\boldsymbol{\alpha}| \leq s\}.$$

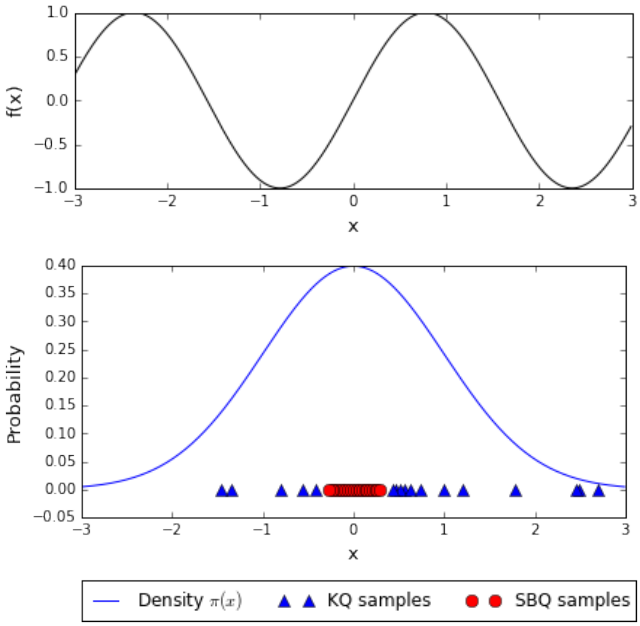

Figure 5. Sequential minimisation of the error criterion $e_n(\boldsymbol{w}; \{\boldsymbol{x}_j\}_{j=1}^n)$, denoted SBQ, does not lead to adequate placement of points $\{\boldsymbol{x}_j\}_{j=1}^n$ when the kernel is mis-specified. [Here the kernel length scale was fixed to $\ell = 0.01$. Selected points x_j are represented as red. For comparison, a collection of draws from Π, as used in KQ, are shown as blue points.]

This space is equipped with norm

$$\|f\|_{\mathbb{H}_s(\Pi)} = \left(\sum_{|\boldsymbol{\alpha}| \leq s} \|(\partial x_1)^{\alpha_1} \ldots (\partial x_d)^{\alpha_d} f\|_{L_2(\Pi)}^2 \right)^{1/2}.$$

Two normed spaces $(\mathcal{F}, \|\cdot\|)$ and $(\mathcal{F}, \|\cdot\|')$ are said to be 'norm equivalent' if there exists $0 < c < \infty$ such that

$$c^{-1}\|f\|' \leq \|f\| \leq c\|f\|'$$

for all $f \in \mathcal{F}$.

A.3. Theoretical Results

A.3.1. Proof of Theorem 1

Proof. From Thm. 11.13 in Wendland (2004) we have that there exist constants $0 < c_k < \infty$, $h_0 > 0$ such that

$$|\hat{f}(\boldsymbol{x}) - f(\boldsymbol{x})| \leq c_k h_n^s \|f\|_{\mathcal{H}} \tag{7}$$

for all $\boldsymbol{x} \in \mathcal{X}$, provided $h_n < h_0$, where

$$h_n = \sup_{\boldsymbol{x} \in \mathcal{X}} \min_{i=1,\ldots,n} \|\boldsymbol{x} - \boldsymbol{x}_i\|_2.$$

Under the hypotheses, we can suppose that the deterministic states $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m$ ensure $h_m < h_0$. Then Eqn. 7 holds

for all $n > m$, where the $\boldsymbol{x}_{m+1}, \dots, \boldsymbol{x}_n$ are independent draws from Π'. It follows that

$$\begin{aligned} |\hat{\Pi}(f) - \Pi(f)| &\leq \sup_{\boldsymbol{x} \in \mathcal{X}} |\hat{f}(\boldsymbol{x}) - f(\boldsymbol{x})| \\ &\leq c_k h_n^s \|f\|_{\mathcal{H}}. \end{aligned}$$

Next, Lem. 1 in Oates et al. (2016) establishes that, under the present hypotheses on $\mathcal{X}$ and Π', there exists $0 < c_{\Pi',\epsilon} < \infty$ such that

$$\mathbb{E}[h_n^{2s}] \leq c_{\Pi',\epsilon} m^{-2s/d+\epsilon}$$

for all $\epsilon > 0$, where $c_{\Pi',\epsilon}$ is independent of n.

Combining the above results produces

$$\begin{aligned} \mathbb{E}[\hat{\Pi}(f) - \Pi(f)]^2 &\leq c_k^2 \mathbb{E}[h_n^{2s}] \|f\|_{\mathcal{H}}^2 \\ &\leq c_k^2 c_{\Pi',\epsilon} m^{-2s/d+\epsilon} \|f\|_{\mathcal{H}}^2 \end{aligned}$$

as required, with $c_{k,\Pi',\epsilon} = c_k c_{\Pi',\epsilon}^{1/2}$. □

A.3.2. Proof of Theorem 2

Proof. The Cauchy-Schwarz result for kernel mean embeddings (Smola et al., 2007) gives

$$\begin{aligned} &|\hat{\Pi}(f) - \Pi(f)| \qquad (8) \\ \leq &\left\| \sum_{i=1}^n w_i k(\cdot, \boldsymbol{x}_i) - \int_{\mathcal{X}} k(\cdot, \boldsymbol{x}) \Pi(\mathrm{d}\boldsymbol{x}) \right\|_{\mathcal{H}} \|f\|_{\mathcal{H}}. \end{aligned}$$

Consider the first term above. Since $\mathcal{H}$ is dense in $L_2(\Pi)$, it follows that $\Sigma^{1/2}$ (the unique positive self-adjoint square root of Σ) is an isometry from $L_2(\Pi)$ to $\mathcal{H}$. Now, since $k(\cdot, \boldsymbol{x}) \in \mathcal{H}$, there exists a unique element $\psi(\cdot, \boldsymbol{x}) \in L_2(\Pi)$ such that $\Sigma^{1/2}\psi(\cdot, \boldsymbol{x}) = k(\cdot, \boldsymbol{x})$. Then we have that

$$\begin{aligned} &\left\| \sum_{i=1}^n w_i k(\cdot, \boldsymbol{x}_i) - \int_{\mathcal{X}} k(\cdot, \boldsymbol{x}) \Pi(\mathrm{d}\boldsymbol{x}) \right\|_{\mathcal{H}} \\ = &\left\| \sum_{i=1}^n w_i \Sigma^{1/2} \psi(\cdot, \boldsymbol{x}_i) - \int_{\mathcal{X}} \Sigma^{1/2} \psi(\cdot, \boldsymbol{x}) \Pi(\mathrm{d}\boldsymbol{x}) \right\|_{\mathcal{H}} \\ = &\left\| \sum_{i=1}^n w_i \psi(\cdot, \boldsymbol{x}_i) - \int_{\mathcal{X}} \psi(\cdot, \boldsymbol{x}) \Pi(\mathrm{d}\boldsymbol{x}) \right\|_{L_2(\Pi)}. \end{aligned}$$

For $f \in L_2(\Pi)$, we have $f \in \mathcal{H}$ if and only if

$$f = \int_{\mathcal{X}} g(\boldsymbol{x}) \psi(\cdot, \boldsymbol{x}) \Pi(\mathrm{d}\boldsymbol{x}) \qquad (9)$$

for some $g \in L_2(\Pi)$, in which case $\|f\|_{\mathcal{H}}$ is equal to the infimum of $\|g\|_{L_2(\Pi)}$ under all such representations g. In particular, it follows that $\|f\|_{\mathcal{H}} = 1$ for the particular choice with $g(\boldsymbol{x}) = 1$ for all $\boldsymbol{x} \in \mathcal{X}$.

Under the hypothesis on n, Prop. 1 of Bach (2015) established that when $\boldsymbol{x}_1, \dots, \boldsymbol{x}_n \sim \Pi_{\mathrm{B}}$ are independent, then

$$\sup_{\|f\|_{\mathcal{H}} \leq 1} \inf_{\|\boldsymbol{\beta}\|_2^2 \leq \frac{4}{n}} \left\| \sum_{i=1}^n \frac{\beta_i}{\pi_{\mathrm{B}}(\boldsymbol{x}_i)^{1/2}} \psi(\cdot, \boldsymbol{x}_i) - f \right\|_{L_2(\Pi)}^2 \leq 4\lambda$$

with probability at least $1 - \delta$. Fixing the function f in Eqn. 9 leads to the statement that

$$\inf_{\|\boldsymbol{\beta}\|_2^2 \leq \frac{4}{n}} \left\| \sum_{i=1}^n \frac{\beta_i}{\pi_{\mathrm{B}}(\boldsymbol{x}_i)^{1/2}} \psi(\cdot, \boldsymbol{x}_i) - \int_{\mathcal{X}} \psi(\cdot, \boldsymbol{x}) \Pi(\mathrm{d}\boldsymbol{x}) \right\|_{L_2(\Pi)}^2$$

is at most 4λ with probability at least $1 - \delta$. The infimum over $\|\boldsymbol{\beta}\|_2^2 \leq 4/n$ can be replaced with an unconstrained infimum over $\mathbb{R}^n$ to obtain the weaker statement that

$$\inf_{\boldsymbol{\beta} \in \mathbb{R}^n} \left\| \sum_{i=1}^n \frac{\beta_i}{\pi_{\mathrm{B}}(\boldsymbol{x}_i)^{1/2}} \psi(\cdot, \boldsymbol{x}_i) - \int_{\mathcal{X}} \psi(\cdot, \boldsymbol{x}) \Pi(\mathrm{d}\boldsymbol{x}) \right\|_{L_2(\Pi)}^2$$

is at most 4λ with probability at least $1 - \delta$. Now, recall from Sec. 2.1 that the KQ weights $\boldsymbol{w}$ are characterised through the solution $\boldsymbol{\beta}^*$ to this optimisation problem as $w_i = \beta_i^* \pi_{\mathrm{B}}(\boldsymbol{x}_i)^{-1/2}$. It follows that

$$\left\| \sum_{i=1}^n w_i \psi(\cdot, \boldsymbol{x}_i) - \int_{\mathcal{X}} \psi(\cdot, \boldsymbol{x}) \Pi(\mathrm{d}\boldsymbol{x}) \right\|_{L_2(\Pi)}^2 \leq 4\lambda$$

with probability at least $1 - \delta$. Combining this fact with Eqn. 8 completes the proof. □

A.3.3. Π_{B} for the Example of Figure 1

In this section we consider scope to derive Π_{B} in closed-form for the example of Fig. 1. The following will be used:

Proposition 1 (Prop. 1 in Shi et al. (2009))**.** *Let $\mathcal{X} = \mathbb{R}$, $\Pi = \mathrm{N}(\mu, \sigma^2)$ and $k(x, x') = \exp(-(x - x')^2/\ell^2)$. Define $\beta = 4\sigma^2/\ell^2$ and denote the jth Hermite polynomial as $H_j(x)$. Then the eigenvalues μ_j and corresponding eigenfunctions e_j of the integral operator Σ are*

$$\mu_j = \sqrt{\frac{2}{(1 + \beta + \sqrt{1 + 2\beta})}} \times \Big(\frac{\beta}{1 + \beta + \sqrt{1 + 2\beta}} \Big)^j$$

and

$$\begin{aligned} e_j(x) = \frac{(1 + 2\beta)^{1/8}}{\sqrt{2^j j!}} \exp\Big(-\frac{(x - \mu)^2}{2\sigma^2} \frac{\sqrt{1 + 2\beta} - 1}{2} \Big) \\ \times H_j\Big(\Big(\frac{1}{4} + \frac{\beta}{2} \Big)^{1/4} \frac{x - \mu}{\sigma} \Big) \end{aligned}$$

for $j \in \{0, 1, 2, \dots\}$.

Proposition 2 (Ex. 6.8 in Temme (1996), p.167)**.** *The bilinear generating function for Hermite polynomials is*

$$\sum_{j=0}^{\infty} \frac{t^j}{j!} H_j(x) H_j(z) = \frac{1}{\sqrt{1-4t^2}} \exp\left(x^2 - \frac{(x-2zt)^2}{1-4t^2}\right).$$

Proposition 3. *For the example in Fig. 1 we have*

$$\pi_B(x;\lambda) \propto \exp(-x^2) \sum_{j=0}^{\infty} \frac{1}{1+\lambda 2^{j+1}} \frac{1}{2^j j!} H_j^2\left(\sqrt{\frac{3}{2}}x\right).$$

Proof. For the example of Fig. 1, in the notation of Prop. 1, we have $\mu = 0$, $\sigma = 1$, $\ell = 1$ and $\beta = 4$. Thus

$$\mu_j = \left(\frac{1}{2}\right)^{j+1}$$
$$e_j(x)^2 = \sqrt{3}\exp(-x^2)\frac{1}{2^j j!} H_j^2\left(\sqrt{\frac{3}{2}}x\right)$$

and so

$$\pi_{\mathrm{B}}(x;\lambda) \propto \sum_j \frac{\mu_j}{\mu_j + \lambda} e_j^2(x) \propto \exp(-x^2) \sum_{j=0}^{\infty} \frac{1}{1+\lambda 2^{j+1}} \frac{1}{2^j j!} H_j^2\left(\sqrt{\frac{3}{2}}x\right)$$

as required. □

To the best of our knowledge, the expression for Π_{B} in Prop. 3 does not admit a closed form. This poses a practical challenge. However, some limited insight is available through basic approximations:

- For large values of λ we have $1 + \lambda 2^{j+1} \approx \lambda 2^{j+1}$ for all $j \in \{0, 1, 2, \dots\}$, from which we obtain

$$\pi_{\mathrm{B}}(x;\lambda) \propto\!\!\propto \exp(-x^2) \sum_{j=0}^{\infty} \frac{1}{4^j j!} H_j^2\left(\sqrt{\frac{3}{2}}x\right) \propto \exp(-x^2)\exp(x^2) = 1,$$

  where the second step made use of Prop. 2. Thus when large integration errors are tolerated, Π_{B} requires that we take the states $\boldsymbol{x}_i$ to be approximately uniform over $\mathcal{X}$ (of course, this limiting distribution is improper and serves only for illustration).

- For small values of λ, the series in Prop. 3 is dominated by the first m terms such that $j < m$ if and only if $\lambda 2^{j+1} < 1$. Indeed, for $j \leq m$ we have $1 + \lambda 2^{j+1} \approx 1$. Thus we have a computable approximation

$$\pi_{\mathrm{B}}(x;\lambda) \propto\!\!\propto \exp(-x^2) \sum_{j=0}^{m} \frac{1}{2^j j!} H_j^2\left(\sqrt{\frac{3}{2}}x\right)$$

  where $m = \lceil -\log_2(\lambda) \rceil$. Empirical results (not shown) indicate that this is not a useful approximation from a practical standpoint, since at finite m the tails of the approximation are explosive (due to the use of a polynomial basis).

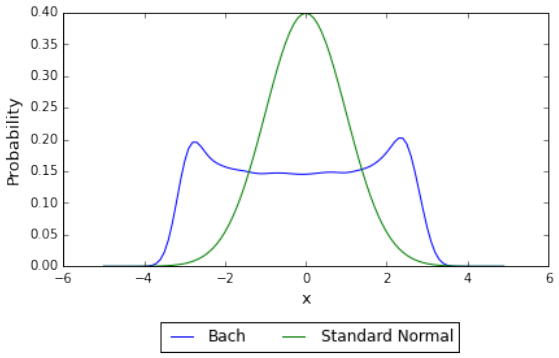

Figure 6. Numerical approximation of Π_{B} for the running illustration. Here the regularisation parameter was $\lambda = 10^{-15}$.

The approximation method in Bach (2015) was also used to obtain the numerical approximation to Π_{B} shown in Fig. 6. This appears to support the intuition that it is beneficial to over-sample from the tails of Π.

To finish, we remark that Prop. 3 implies that the integration error in this example scales as

$$\sqrt{\mu_n} \sim 2^{-n/2}$$

as $n \to \infty$ when samples are drawn from Π_{B}. This agrees with both intuition and empirical results that concern approximation with exponentiated quadratic kernels.

A.3.4. Additional Theoretical Material

As mentioned in the Main Text, the worst-case error $e_n(\{\boldsymbol{x}_j\}_{j=1}^n)$ can be computed in closed form:

$$e_n(\{\boldsymbol{x}_j\}_{j=1}^n)^2 = \Pi \otimes \Pi(k) - 2\boldsymbol{w}^\top \mathbf{K}\boldsymbol{z} + \boldsymbol{w}^\top \mathbf{K}\boldsymbol{w}$$

Here we have defined

$$\Pi \otimes \Pi(k) = \iint_{\mathcal{X}\times\mathcal{X}} k(\boldsymbol{x}, \boldsymbol{x}')\, \Pi \otimes \Pi(\mathrm{d}\boldsymbol{x} \times \mathrm{d}\boldsymbol{x}')$$

where $\Pi \otimes \Pi$ is the product measure of Π with itself.

Next, we report a result which does not address KQ itself, but considers importance sampling methods for integration

of functions in a Hilbert space. The following is due to Plaskota et al. (2009); Hinrichs (2010) and we provide an elementary proof of their result:

Theorem 3. *The assumptions of Sec. 2.4 are taken to hold. In addition, we assume that distributions Π, Π' admit densities π, π'. Introduce importance sampling estimators of the form*

$$\hat{\Pi}_{\mathrm{IS}}(f) = \frac{1}{n}\sum_{i=1}^{n} f(\boldsymbol{x}_i)\frac{\pi(\boldsymbol{x}_i)}{\pi'(\boldsymbol{x}_i)},$$

where $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \sim \Pi'$ are independent, and consider the distribution Π' that minimises

$$\sup_{f\in\mathcal{F}} \sqrt{\mathbb{E}[\hat{\Pi}_{\mathrm{IS}}(f) - \Pi(f)]^2}.$$

For $\mathcal{F} = \{f\}$ we have that Π' is $\pi'(\boldsymbol{x}) \propto |f(\boldsymbol{x})|\pi(\boldsymbol{x})$, while for $\mathcal{F} = \{f \in \mathcal{H} \,:\, \|f\|_{\mathcal{H}} \leq 1\}$ we have that Π' is $\pi'(\boldsymbol{x}) \propto \sqrt{k(\boldsymbol{x},\boldsymbol{x})}\pi(\boldsymbol{x})$.

Proof. The first result, for $\mathcal{F} = \{f\}$ is well-known; e.g. Thm. 3.3.4 in Robert and Casella (2013).

For the second case, where $\mathcal{F}$ is the unit ball in $\mathcal{H}$, we start by establishing a (tight) upper bound for the supremum of f^2 over $f \in \mathcal{F}$:

$$\begin{aligned} |f(\boldsymbol{x})| &= \left|\langle f, k(\cdot,\boldsymbol{x})\rangle_{\mathcal{H}}\right| \\ &\leq \|f\|_{\mathcal{H}}\|k(\cdot,\boldsymbol{x})\|_{\mathcal{H}} \\ &= \|f\|_{\mathcal{H}}\sqrt{\langle k(\cdot,\boldsymbol{x}), k(\cdot,\boldsymbol{x})\rangle_{\mathcal{H}}} \\ &= \|f\|_{\mathcal{H}}\sqrt{k(\boldsymbol{x},\boldsymbol{x})} \end{aligned}$$

where the inequality here is Cauchy-Schwarz. Squaring both sides and taking the supremum over $f \in \mathcal{F}$ gives

$$\sup_{f\in\mathcal{F}} f(\boldsymbol{x})^2 \;\leq\; \sup_{f\in\mathcal{F}} \|f\|_{\mathcal{H}}^2\, k(\boldsymbol{x},\boldsymbol{x}) \;=\; k(\boldsymbol{x},\boldsymbol{x}). \tag{10}$$

This is in fact an equality, since for given $\boldsymbol{x} \in \mathcal{X}$ we can take $f(\boldsymbol{x}') = k(\boldsymbol{x}',\boldsymbol{x})/\sqrt{k(\boldsymbol{x},\boldsymbol{x})}$ which has $\|f\|_{\mathcal{H}} = 1$ and $f(\boldsymbol{x})^2 = k(\boldsymbol{x},\boldsymbol{x})$.

Our objective is expressed as

$$\sup_{f\in\mathcal{F}} \sqrt{\mathbb{E}[\hat{\Pi}_{\mathrm{IS}}(f) - \Pi(f)]^2} = \sup_{f\in\mathcal{F}} \frac{1}{\sqrt{n}}\,\mathrm{Std}\Big(\frac{f\pi}{\pi'};\Pi'\Big)$$

and since

$$\mathrm{Std}\Big(\frac{f\pi}{\pi'};\Pi'\Big)^2 \;=\; \Pi'\Big(\Big(\frac{f\pi}{\pi'}\Big)^2\Big) - \Pi'\Big(\frac{f\pi}{\pi'}\Big)^2$$

we thus aim to minimise

$$\sup_{f\in\mathcal{F}} \Pi'\Big(\Big(\frac{f\pi}{\pi'}\Big)^2\Big)$$

over $\Pi' \in \mathcal{P}(\mathcal{F}\cdot \mathrm{d}\Pi/\mathrm{d}\Pi')$. (Here $\mathcal{F}\cdot \mathrm{d}\Pi/\mathrm{d}\Pi'$ denotes the set of functions of the form $f\cdot \mathrm{d}\Pi/\mathrm{d}\Pi'$ such that $f \in \mathcal{F}$.)

Combining Eqns. 10 and A.3.4, we have

$$\begin{aligned} \sup_{f\in\mathcal{F}} \Pi'\Big(\Big(\frac{f\pi}{\pi'}\Big)^2\Big) &\leq \Pi'\Big(\sup_{f\in\mathcal{F}}\Big(\frac{f\pi}{\pi'}\Big)^2\Big) \\ &= \Pi'\Big(k(\cdot,\cdot)\Big(\frac{\pi(\cdot)}{\pi'(\cdot)}\Big)^2\Big) \end{aligned}$$

As before, this is in fact an equality, as can be seen from $f(\boldsymbol{x}) = \sqrt{k(\boldsymbol{x},\boldsymbol{x})}$.

From Jensen's inequality,

$$\begin{aligned} \Pi'\Big(k(\cdot,\cdot)\Big(\frac{\pi(\cdot)}{\pi'(\cdot)}\Big)^2\Big) &\geq \Big(\Pi'\Big(\sqrt{k(\cdot,\cdot)}\frac{\pi(\cdot)}{\pi'(\cdot)}\Big)\Big)^2 \quad (11) \\ &= \big(\Pi\big(\sqrt{k(\cdot,\cdot)}\big)\big)^2. \end{aligned}$$

Since the right hand side is independent of Π', a choice of Π' for which Eqn. 11 is an equality must be a minimiser of Eqn. A.3.4. It remains just to verify this fact for $\pi'(\boldsymbol{x}) = \sqrt{k(\boldsymbol{x},\boldsymbol{x})}\pi(\boldsymbol{x})/C$, where the normalising constant is $C = \Pi(\sqrt{k(\cdot,\cdot)})$. For this choice

$$\begin{aligned} \Pi'\Big(k(\cdot,\cdot)\Big(\frac{\pi(\cdot)}{\pi'(\cdot)}\Big)^2\Big) &= \Pi'(C^2) \\ &= (\Pi(\sqrt{k(\cdot,\cdot)}))^2 \end{aligned}$$

as required. □

A.4. Implementation of `test`$(R < R_{\min})$

Here we provide details for how the criterion $R < R_{\min}$ was tested. The problem with the naive approach of comparing R estimated at t_{i-1} directly with R estimated at t_i is that Monte Carlo error can lead to an incorrect impression that R is increasing, when it is in fact decreasing, and cause the algorithm to terminate when estimation is poor (see Fig. 7 and note the jaggedness of the estimated R curve as a function of inverse temperature t). Our solution was to apply a least-squares linear smoother to the estimates for R over 5 consecutive temperatures. This approach, denoted `test`, illustrated in Fig. 7, determines whether the gradient of the linear smoother is positive or negative, and in this way we are able to provide robustness to Monte Carlo error in the termination criterion. To be precise, the algorithm requires at least 5 temperature evaluations before termination is considered (Fig. 7; left) and terminates when the gradient of the linear smoother becomes positive for the first time (Fig. 7; right). The success of this strategy was established in Fig. 9 later in the Appendix.

A.5. Experimental Results

A.5.1. Implementation of Simulation Study

Denote by $\mathrm{N}(\boldsymbol{x}|\boldsymbol{\mu},\boldsymbol{\Sigma})$ the p.d.f. of the multivariate Gaussian distribution with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$. Furthermore, we denote by $\boldsymbol{\Sigma}_\sigma$ the diagonal covariance matrix

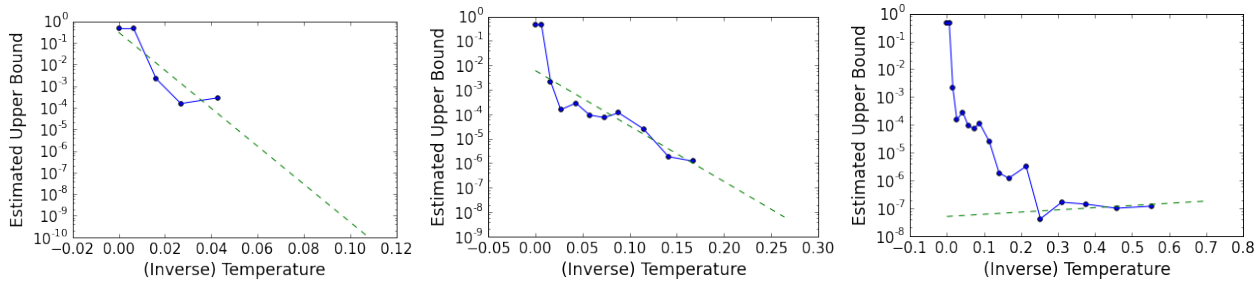

Figure 7. Implementation of $\texttt{test}(R < R_{\min})$. A linear smoother (dashed line) was based on 5 consecutive (inverse) temperature parameters $t_{i-4}, t_{i-3}, t_{i-2}, t_{i-1}, t_i$. To begin it is required that 5 temperatures are considered (left panel). The algorithm terminates on the first occasion when the linear smoother takes a positive gradient (right panel).

with diagonal element σ^2. Then elementary manipulation of Gaussian densities produces:

$$\begin{aligned} k(\boldsymbol{x},\boldsymbol{y}) &:= \exp\Big(-\frac{\sum_{j=1}^d (x_j - y_j)^2}{l^2}\Big) \\ &= (\sqrt{\pi}l)^d \phi\big(\boldsymbol{x}|\boldsymbol{y}, \Sigma_{l/\sqrt{2}}\big) \\ \nabla_l k(x,y) &:= \frac{2\sum_{j=1}^d (x_j - y_j)^2}{l^3} k(\boldsymbol{x},\boldsymbol{y}) \\ \Pi[k(\cdot,\boldsymbol{x})] &:= (\sqrt{\pi}l)^d \mathrm{N}\big(\boldsymbol{x}|\boldsymbol{0}, \Sigma_\sigma + \Sigma_{l/\sqrt{2}}\big) \\ \Pi\otimes\Pi(k) &:= (\sqrt{\pi}l)^d \mathrm{N}\big(\boldsymbol{0}|\boldsymbol{0}, \Sigma_{\sqrt{2}\sigma} + \Sigma_{l/\sqrt{2}}\big) \end{aligned}$$

A.5.2. Dependence on Parameters for the Simulation Study

For the running illustration with $f(x) = 1 + \sin(x)$, $\Pi = \mathrm{N}(0,1)$, $\Pi' = \mathrm{N}(0,\sigma^2)$ and $k(x,x') = \exp(-(x - x')^2/\ell^2)$, we explored how the RMSE of KQ depends on the choice of both σ and ℓ. Here we go beyond the results presented in Fig. 2, which considered fixed n, to now consider the simultaneous choice of both σ, ℓ for varying n. Note that in these numerical experiments the kernel matrix inverse $\mathbf{K}^{-1}$ was replaced with the regularised inverse $(\mathbf{K} + \lambda\mathbf{I})^{-1}$ that introduces a small 'nugget' term $\lambda > 0$ for stabilisation. Results, shown in Fig. 8, demonstrate two principles that guided the methodological development in this paper:

- Length scales ℓ that are 'too small' to learn from n samples do not permit good approximations $\hat{f}$ and lead in practice to high RMSE. At the same time, if ℓ is taken to be 'too large' then efficient approximation at size n will also be sacrificed. This is of course well understood from a theoretical perspective and is borne out in our empirical results. These results motivated extension of `SMC-KQ` to `SMC-KQ-KL`.
- In general the 'sweet spot', where σ and ℓ lead to minimal RMSE, is quite small. However, the problem of optimal choice for σ and ℓ does not seem to become more or less difficult as n increases. This suggests that a method for selection of σ (and possibly also of ℓ) ought to be effective regardless of the number n of states that will be used.

A.5.3. Additional Results for the Simulation Study

To understand whether the termination criterion of Sec. 3.5 was suitable (and, by extension, to examine the validity of the convexity ansatz in Sec. 3.2), in Fig. 9 we presented histograms for both estimated and actual optimal (inverse) temperature parameter t^*. Results supported the use of the criterion, in the form described above for `test`.

In Fig. 10 reports the dependence of performance on the choice of initial distribution Π_0. There was relatively little influence on the RMSE obtained by the method for this wide range of initial distribution, which supports the purported robustness of the method.

We also test the method on more complex integrands in Fig. 11: $f(x) = 1 + \sin(4\pi x)$ and $f(x) = 1 + \sin(8\pi x)$. These are more challenging for KQ compared to the illustration in the Main Text, since they are more difficult to interpolate due to their higher periodicity. However, `SMC-KQ` still manages to adapt to the complexity of the integrand and performs as well as the best importance sampling distribution ($\sigma = 2$).

As an extension, we also study the robustness to the dimensionality to the problem. In problem, we consider the generalisation of our main test function to $f : \mathbb{R}^d \to \mathbb{R}$ given by $f(\boldsymbol{x}) = 1 + \prod_{j=1}^d \sin(2\pi x_j)$. Notice that the integral can still be computed analytically and equals 1. We present results for $d = 2$ and $d = 3$ in Fig. 12. These two cases are more challenging for both the `KQ` and `SMC-KQ` methods, since the higher dimension implies a slower convergence rate. Once again, we notice that `SMC-KQ` manages to adapt to the complexity of the problem at hand, and provides improved performance on simpler sampling distributions.

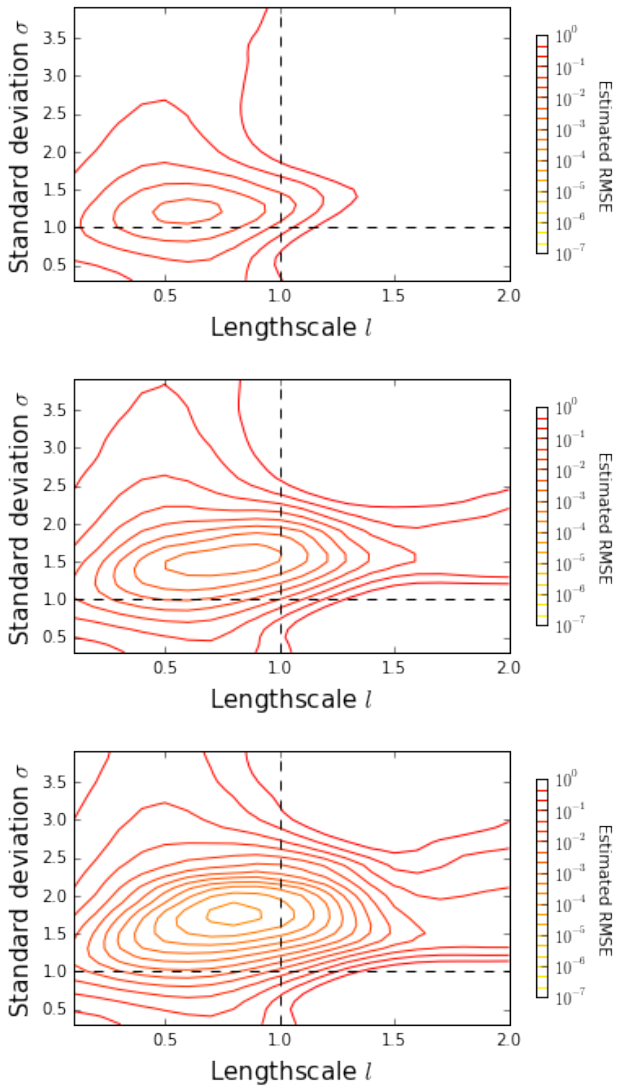

Figure 8. Example of Fig. 2, continued. Here we consider the simultaneous choice of sampling standard deviation σ and kernel length-scale ℓ, reporting empirical estimates for the estimated root mean square integration error (over $M = 300$ repetitions) in each case for sample size (a) $n = 25$ (top), (b) $n = 50$ (middle) and (c) $n = 75$ (bottom).

Finally, we considered replacing the independent samples $x_j \sim \Pi$ with samples drawn from a quasi-random point sequence. Fig. 13 reports results where draws from $\mathrm{N}(0, 1)$ were produced based on a Halton quasi-random number generator. In this case, the performance is improved by up to 10 orders of magnitude in MSE when the sampling is done with respect to a range of tempered sampling distribution (here $\mathrm{N}(0, 3^2)$). This suggests that a SQMC approach (Gerber and Chopin, 2015) could provide further improvement and this suggested for future work.

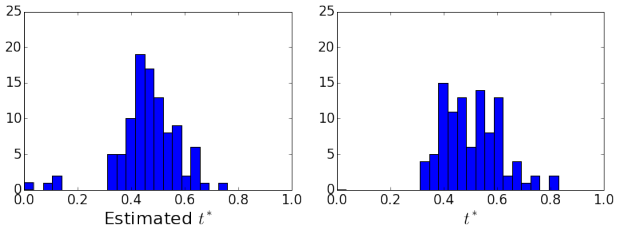

Figure 9. Histograms for the optimal (inverse) temperature parameter t^*. Left: Estimate of t^* provided under the termination criterion of Sec. 3.5. Right: Estimate of t^* obtained by estimating R over a grid for $t \in [0, 1]$ and returning the global minimum. The similarity of these histograms is supportive of the convexity ansatz in Sec. 3.2.

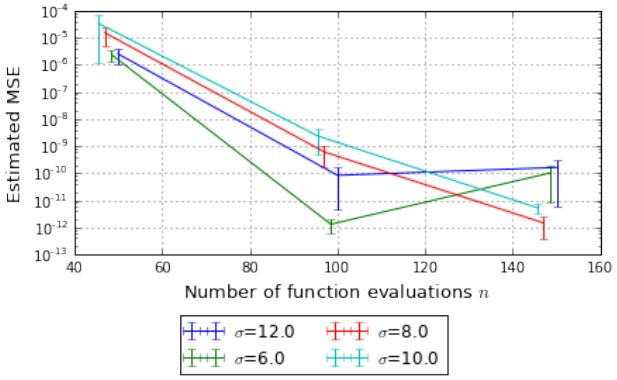

Figure 10. Comparison of the performance of `SMC-KQ` on the running illustration of Figs. 1 and 2 for varying initial distribution $\Pi_0 = \mathrm{N}(0, \sigma^2)$.

A.5.4. Implementation of Stein's Method

Following Oates et al. (2017) we considered the Stein operator

$$\mathbb{S}[f](\boldsymbol{\theta}) := [\nabla_{\boldsymbol{\theta}} + \nabla \log \pi(\boldsymbol{\theta})][f](\boldsymbol{\theta})$$

and denote the score function by $u_j(\boldsymbol{\theta}) = \nabla_{\theta_j} \log \pi(\boldsymbol{\theta})$. Here π is the p.d.f. for Π. Applying the Stein operator to each argument of a base kernel k_b, and adding a constant, gives produces the new kernel:

$$k(\boldsymbol{\theta}, \boldsymbol{\phi}) := 1 + \sum_{j=1}^{d} \begin{array}{l} [\nabla_{\theta_j} \nabla_{\phi_j} k_b(\boldsymbol{\theta}, \boldsymbol{\phi}) \\ \quad + u_j(\boldsymbol{\theta}) \nabla_{\phi_j} k_b(\boldsymbol{\theta}, \boldsymbol{\phi}) \\ \quad + u_j(\boldsymbol{\phi}) \nabla_{\theta_j} k_b(\boldsymbol{\theta}, \boldsymbol{\phi}) \\ \quad + u_j(\boldsymbol{\theta}) u_j(\boldsymbol{\phi}) k_b(\boldsymbol{\theta}, \boldsymbol{\phi})] \end{array}$$

which we will use for our KQ estimator. Using integration by parts, we can easily check that $\Pi[k(\cdot, \boldsymbol{\theta})] = 1$ and $\Pi \otimes \Pi(k) = 1$. In this experiment, the base kernel was taken to be Gaussian: $k_b(\boldsymbol{\theta}, \boldsymbol{\phi}) = \exp(-\sum_{j=1}^{d} (\theta_j - \phi_j)^2 / \ell_j^2)$. We

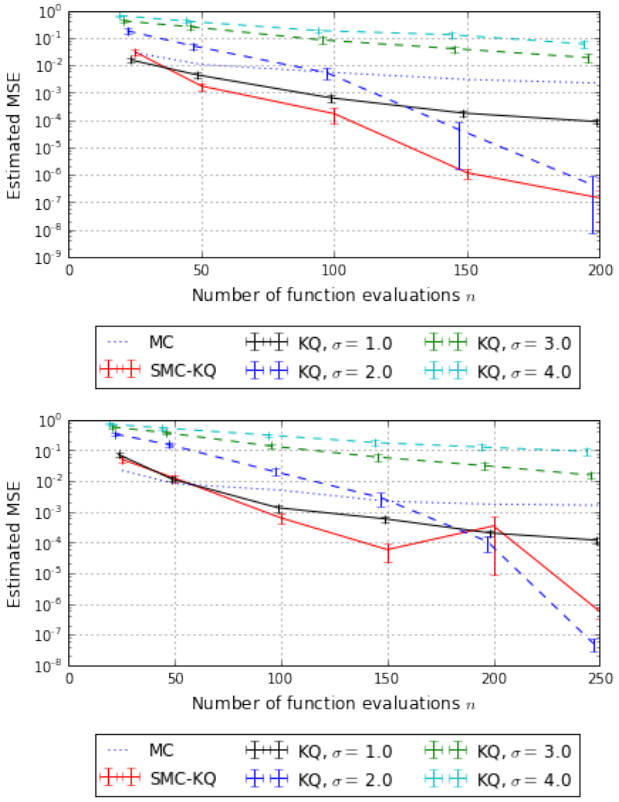

Figure 11. Performance of KQ and SMC−KQ on the integration problem with $f(x) = 1 + \sin(4\pi x)$ (top) and $f(x) = 1 + \sin(8\pi x)$ (bottom) integrated against $\mathrm{N}(0, 1)$. The SMC sampler was initiated with a $\mathrm{N}(0, 8^2)$ distribution. The kernel used was Gaussian with length scales $\ell = 0.25$ (top) and $\ell = 0.15$ (bottom) each chosen to reflect the complexity of the functions.

obtained the derivatives:

$$\begin{aligned}
\frac{\mathrm{d}k(\boldsymbol{\theta}, \boldsymbol{\phi})}{\mathrm{d}\theta_j} &= -\frac{2}{\ell_j^2}(\theta_j - \phi_j)k(\boldsymbol{\theta}, \boldsymbol{\phi}) \\
\frac{\mathrm{d}k(\boldsymbol{\theta}, \boldsymbol{\phi})}{\mathrm{d}\phi_j} &= \frac{2}{\ell_j^2}(\theta_j - \phi_j)k(\boldsymbol{\theta}, \boldsymbol{\phi}) \\
\frac{\mathrm{d}k(\boldsymbol{\theta}, \boldsymbol{\phi})}{\mathrm{d}\theta_j \mathrm{d}\phi_j} &= \frac{\left(2\ell_j^2 - 4(\theta_j - \phi_j)^2\right)}{\ell_j^4}k(\boldsymbol{\theta}, \boldsymbol{\phi})
\end{aligned}$$

Furthermore, we can obtain expressions for the score function for posterior densities as follows:

$$u_j(\boldsymbol{\theta}) = \frac{\mathrm{d}}{\mathrm{d}\theta_j}\log \pi(\boldsymbol{\theta}) + \frac{\mathrm{d}}{\mathrm{d}\theta_j}\log \pi(\boldsymbol{y}|\boldsymbol{\theta}).$$

A.6. Algorithms and Implementation

A.6.1. SMC Sampler

In Alg. 2 the standard SMC scheme is presented. Resampling occurs when the effective sample size, $\|\boldsymbol{w}\|_2^{-2}$ drops below a fraction ρ of the total number N of particles. In this work we took $\rho = 0.95$ which is a common default.

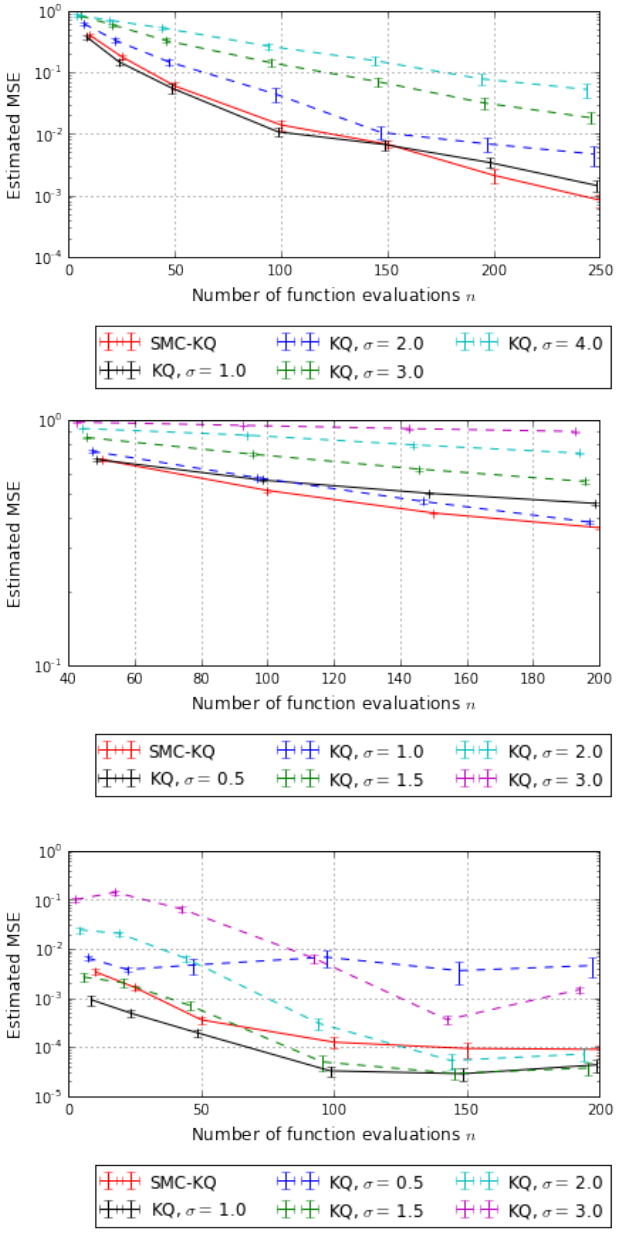

Figure 12. Performance of KQ and SMC−KQ on the integration problem with $f(\boldsymbol{x}) = 1 + \prod_{j=1}^{d} \sin(2\pi x_j)$ integrated against a $\mathrm{N}(\mathbf{0}, \mathbf{I})$ distribution for $d = 2$ (top), $d = 3$ (middle) and $d = 10$ (bottom). The SMC sampler was initiated with a $\mathrm{N}(\mathbf{0}, 8^2\mathbf{I})$ distribution. The kernel used was a (multivariate) Gaussian kernel $k(\boldsymbol{x}, \boldsymbol{y}) = \exp(-\sum_{j=1}^{d}(x_j - y_j)^2/\ell_j^2)$ with the length scales $\ell_1 = \cdots = \ell_d = 0.25$ were used.

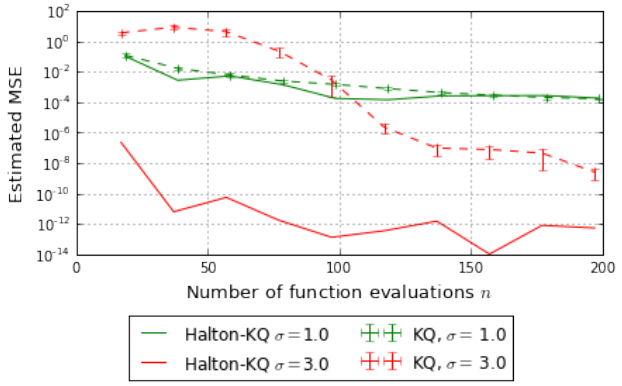

Figure 13. Comparison between KQ with $x_j \sim \mathrm{N}(0,1)$ independent and KQ with $x_j = \Phi^{-1}(u_j)$ where the $\{u_j\}_{j=1}^n$ are the first n terms in the Halton sequence and Φ is the standard Gaussian cumulative density function.

Algorithm 2 Sequential Monte Carlo Iteration

function $\mathtt{SMC}(\{(w_j, \boldsymbol{x}_j)\}_{j=1}^N, t_i, t_{i-1}, \rho)$
input $\{(w_j, \boldsymbol{x}_j)\}_{j=1}^N$ (particle approx. to Π_{i-1})
input t_i (next inverse-temperature)
input t_{i-1} (previous inverse-temperature)
input ρ (re-sample threshold)
$w'_j \leftarrow w_j \times [\pi(\boldsymbol{x}_j)/\pi_0(\boldsymbol{x}_j)]^{t_i - t_{i-1}}$ $(\forall j \in 1:N)$
$\boldsymbol{w}' \leftarrow \boldsymbol{w}'/\|\boldsymbol{w}'\|_1$ (normalise weights)
if $\|\boldsymbol{w}'\|_2^{-2} < N \cdot \rho$ **then**
  $\boldsymbol{a} \sim \mathrm{Multinom}(\boldsymbol{w}')$
  $\boldsymbol{x}'_j \leftarrow \boldsymbol{x}_{a(j)}$ (re-sample $\forall j \in 1:N$)
  $w'_j \leftarrow N^{-1}$ (reset weights $\forall j \in 1:N$)
end if
$\boldsymbol{x}'_j \sim \mathtt{Markov}(\boldsymbol{x}'_j; \Pi_i, \{(w_j, \boldsymbol{x}_j)\}_{j=1}^N)$ (Markov update $\in 1:N$)
return $\{(w'_j, \boldsymbol{x}'_j)\}_{j=1}^N$ (particle approx. to Π_i)

Denote

$$
\begin{aligned}
q(\boldsymbol{x}, \cdot; \{(w_j, \boldsymbol{x}_j)\}_{j=1}^N) &= \mathrm{N}(\cdot; \boldsymbol{\mu}, \boldsymbol{\Sigma}) \\
\boldsymbol{\mu} &= \sum_{j=1}^N w_j \boldsymbol{x}_j \\
\boldsymbol{\Sigma} &= \sum_{j=1}^N w_j (\boldsymbol{x}_j - \boldsymbol{\mu})(\boldsymbol{x}_j - \boldsymbol{\mu})^\top.
\end{aligned}
$$

The above standard adaptive independence proposal was used within a Metropolis-Hastings Markov transition:

Algorithm 3 Markov Iteration

function $\mathtt{Markov}(\boldsymbol{x}, \pi, \{(w_j, \boldsymbol{x}_j)\}_{j=1}^N)$
input $\boldsymbol{x}$ (current state)
input π (density of invar. dist.)
$\boldsymbol{x}^* \sim q(\boldsymbol{x}, \boldsymbol{x}^*; \{(w_j, \boldsymbol{x}_j)\}_{j=1}^N)$ (propose)

$$r \leftarrow \frac{\pi_i(\boldsymbol{x}^*) q(\boldsymbol{x}^*, \boldsymbol{x}; \{(w_j, \boldsymbol{x}_j)\}_{j=1}^N)}{\pi_i(\boldsymbol{x}) q(\boldsymbol{x}, \boldsymbol{x}^*; \{(w_j, \boldsymbol{x}_j)\}_{j=1}^N)}$$

$u \sim \mathrm{Unif}(0,1)$
if $u < r$ **then**
  $\boldsymbol{x} \leftarrow \boldsymbol{x}^*$ (accept)
end ifreturn $\boldsymbol{x}$ (next state)

A.6.2. Choice of Temperature Schedule

Following Zhou et al. (2016) we employed an adaptive temperature schedule construction. This was based on the conditional effective sample size of the SMC particle set, estimated as follows:

Algorithm 4 Conditional Effective Sample Size

function $\mathtt{CESS}(\{(w_j, \boldsymbol{x}_j)\}_{j=1}^N, t)$
input $\{(w_j, \boldsymbol{x}_j)\}_{j=1}^N$ (particle approx. Π_{i-1})
input t (candidate next inverse-temperature)
$z_j \leftarrow [\pi(\boldsymbol{x}_j)/\pi_0(\boldsymbol{x}_j)]^{t_i - t_{i-1}}$ $(\forall j \in 1:N)$
$E \leftarrow N \left(\sum_{j=1}^N w_j z_j\right)^2 / \left(\sum_{j=1}^N w_j z_j^2\right)$
return E (est'd. cond. ESS)

The specific construction for the temperature schedule is detailed in Alg. 5 below and makes use of a Sequential Least Squares Programming algorithm:

Algorithm 5 Adaptive Temperature Iteration

function $\mathtt{temp}(\{(w_j, \boldsymbol{x}_j)\}_{j=1}^N, t_{i-1}, \rho, \Delta)$
input $\{(w_j, \boldsymbol{x}_j)\}_{j=1}^N$ (particle approx. Π_{i-1})
input t_{i-1} (current inverse-temperature)
input ρ (re-sample threshold)
input Δ (max. grid size, default $\Delta = 0.1$)
$t \leftarrow \mathrm{solve}(\mathtt{CESS}(\{(w_j, \boldsymbol{x}_j)\}_{j=1}^N, t) = N \cdot \rho)$
(binary search in $[t_{i-1}, 1]$)
$t_i \leftarrow \min\{t_{i-1} + \Delta, t\}$ **return** t_i (next inverse-temperature)

A.6.3. Termination Criterion

For `SMC-KQ` we estimated an upper bound on the worst case error in the unit ball of the Hilbert space $\mathcal{H}$. This was computed as follows, using a bootstrap algorithm:

Algorithm 6 Termination Criterion

function `crit`$(\Pi, k, \{\boldsymbol{x}_j\}_{j=1}^N)$
input Π (target disn.)
input k (kernel)
input $\{\boldsymbol{x}_j\}_{j=1}^N$ (collection of states)
$R^2 \leftarrow 0$
$e_0 \leftarrow \iint_{\mathcal{X}\times\mathcal{X}} k(\boldsymbol{x}, \boldsymbol{x}')\Pi \otimes \Pi(\mathrm{d}\boldsymbol{x} \times \mathrm{d}\boldsymbol{x}')$ (in'l error)
for m = 1,...,M **do**
  $\tilde{\boldsymbol{x}}_j \sim \mathrm{Unif}(\{\boldsymbol{x}_j\}_{j=1}^N)$ $(\forall j \in 1:n)$
  $z_j \leftarrow \int_{\mathcal{X}} k(\cdot, \tilde{\boldsymbol{x}}_j)\mathrm{d}\Pi$ (k'l mean eval. $\forall j \in 1:n$)
  $\mathrm{K}_{j,j'} \leftarrow k(\tilde{\boldsymbol{x}}_j, \tilde{\boldsymbol{x}}_{j'})$ (kernel eval. $\forall j, j' \in 1:n$)
  $\boldsymbol{w} \leftarrow \boldsymbol{z}^T \boldsymbol{K}^{-1}$ (KQ weights)
  $e_n^2 \leftarrow \boldsymbol{w}^\top \mathbf{K} \boldsymbol{w} - 2\boldsymbol{w}^\top \boldsymbol{z} + e_0^2$
  $R^2 \leftarrow R^2 + e_n^2 M^{-1}$
end for
return R (est'd error)

Note that this could be slightly improved using a weighted bootstrap approach.

For `SMC-KQ-KL` an empirical upper bound on integration error was estimated. This requires that the norm $\|f\|_{\mathcal{H}}$ be estimated, which was achieved as follows:

Algorithm 7 Termination Crit. + Kernel Learning

function `crit-KL`$(f, \Pi, k, \{\boldsymbol{x}_j\}_{j=1}^N)$
input f (integrand)
input Π (target disn.)
input k (kernel)
input $\{\boldsymbol{x}_j\}_{j=1}^N$ (collection of states)
$R^2 \leftarrow 0$
$e_0 \leftarrow \iint_{\mathcal{X}\times\mathcal{X}} k(\boldsymbol{x}, \boldsymbol{x}')\Pi \otimes \Pi(\mathrm{d}\boldsymbol{x} \times \mathrm{d}\boldsymbol{x}')$ (in'l error)
for m = 1,...,M **do**
  $\tilde{\boldsymbol{x}}_j \sim \mathrm{Unif}(\{\boldsymbol{x}_j\}_{j=1}^N)$ $(\forall j \in 1:n)$
  $\mathrm{f}_j \leftarrow f(\tilde{\boldsymbol{x}}_j)$ (function eval. $\forall j \in 1:n$)
  $z_j \leftarrow \int_{\mathcal{X}} k(\cdot, \tilde{\boldsymbol{x}}_j)\mathrm{d}\Pi$ (k'l mean eval. $\forall j \in 1:n$)
  $\mathrm{K}_{j,j'} \leftarrow k(\tilde{\boldsymbol{x}}_j, \tilde{\boldsymbol{x}}_{j'})$ (kernel eval. $\forall j, j' \in 1:n$)
  $\boldsymbol{w} \leftarrow \boldsymbol{z}^T \boldsymbol{K}^{-1}$ (KQ weights)
  $e_n^2 \leftarrow \boldsymbol{w}^\top \mathbf{K} \boldsymbol{w} - 2\boldsymbol{w}^\top \boldsymbol{z} + e_0^2$
  $R^2 \leftarrow R^2 + e_n^2 M^{-1}$
end for
$z_j \leftarrow \int_{\mathcal{X}} k(\cdot, \boldsymbol{x}_j)\mathrm{d}\Pi$ (kernel mean eval. $\forall j \in 1:n$)
$\mathrm{K}_{j,j'} \leftarrow k(\boldsymbol{x}_j, \boldsymbol{x}_{j'})$ (kernel eval. $\forall j, j' \in 1:n$)
$\boldsymbol{w} \leftarrow \boldsymbol{z}^T \boldsymbol{K}^{-1}$ (KQ weights)
$S^2 \leftarrow R^2 \times \boldsymbol{w}^\top \mathbf{K} \boldsymbol{w}$ **return** S (est'd error bound)

In Alg. 7 the literal interpretation, that f is re-evaluated on values of $\boldsymbol{x}_j$ which have been previously examined, is clearly inefficient. In practice such function evaluations were cached and then do not contribute further to the total number of function evaluations that are required in the algorithm.

A.6.4. Kernel Learning

A generic approach to select kernel parameters is the maximum marginal likelihood method:

Algorithm 8 Parameter Update

function `kern-param`$(\mathbf{f}, \{\boldsymbol{x}_j\}_{j=1}^n, k_\theta)$
input $\mathbf{f}$ (integrand evals.)
input $\{\boldsymbol{x}_j\}_{j=1}^n$ (associated states)
input k_θ (parametric kernel)
$\theta' \leftarrow \arg\min_\theta \mathbf{f}^\top \mathbf{K}_\theta^{-1} \mathbf{f} + \log|\mathbf{K}_\theta|$ (numer. opt.)
(s.t. $\mathrm{K}_{\theta,j,j'} = k_\theta(\boldsymbol{x}_j, \boldsymbol{x}_{j'})$) **return** θ' (optimal params)

A.6.5. Implementation of `SMC-KQ-KL`

Our final algorithm to present is the full implementation for `SMC-KQ-KL`:

Algorithm 9 SMC for KQ with Kernel Learning

function `SMC-KQ-KL`$(f, \Pi, k_\theta, \Pi_0, \rho, n, N)$
input f (integrand)
input Π (target disn.)
input k_θ (parametric kernel)
input Π_0 (reference disn.)
input ρ (re-sample threshold)
input n (num. func. evaluations)
input N (num. particles)
$i \leftarrow 0$; $t_i \leftarrow 0$; $R_{\min} \leftarrow \infty$
$\boldsymbol{x}'_j \sim \Pi_0$ (initialise states $\forall j \in 1:N$)
$w'_j \leftarrow N^{-1}$ (initialise weights $\forall j \in 1:N$)
$\theta' \leftarrow$ `kern-param`$(f, \{\boldsymbol{x}'_j\}_{j=1}^n)$ (kernel params)
$R \leftarrow$ `crit-KL`$(f, \Pi, k_{\theta'}, \{\boldsymbol{x}'_j\}_{j=1}^N)$ (est'd error)
while `test`$(R < R_{\min})$ and $t_i < 1$ **do**
  $i \leftarrow i+1$; $R_{\min} \leftarrow R$; $\theta \leftarrow \theta'$
  $\{(w_j, \boldsymbol{x}_j)\}_{j=1}^N \leftarrow \{(w'_j, \boldsymbol{x}'_j)\}_{j=1}^N$
  $t_i \leftarrow$ `temp`$(\{(w_j, \boldsymbol{x}_j)\}_{j=1}^N, t_{i-1})$ (next temp.)
  $\{(w'_j, \boldsymbol{x}'_j)\}_{j=1}^N \leftarrow$ `SMC`$(\{(w_j, \boldsymbol{x}_j)\}_{j=1}^N, t_i, t_{i-1}, \rho)$ (next particle approx.)
  $\theta' \leftarrow$ `kern-param`$(f, \{\boldsymbol{x}'_j\}_{j=1}^n)$ (kernel params)
  $R \leftarrow$ `crit-KL`$(f, \Pi, k_{\theta'}, \{\boldsymbol{x}'_j\}_{j=1}^N)$ (est'd error)
end while
$\mathrm{f}_j \leftarrow f(\boldsymbol{x}_j)$ (function eval. $\forall j \in 1:n$)
$z_j \leftarrow \int_{\mathcal{X}} k_\theta(\cdot, \boldsymbol{x}_j) \mathrm{d}\Pi$ (kernel mean eval. $\forall j \in 1:n$)
$\mathrm{K}_{j,j'} \leftarrow k_\theta(\boldsymbol{x}_j, \boldsymbol{x}_{j'})$ (kernel eval. $\forall j, j' \in 1:n$)
$\hat{\Pi}(f) \leftarrow \boldsymbol{z}^\top \mathbf{K}^{-1} \mathbf{f}$ (eval. KQ estimator) **return** $\hat{\Pi}(f)$ (estimator)

As stated here, Alg. 9 is inefficient as function evaluations that are produced in the `kern-param` and `crit-KL` components are not included in the KQ estimator $\hat{\Pi}(f)$. Thus a trivial modification is to store all function evaluations $(\mathrm{f}_j, \boldsymbol{x}_j)$ that are produced and to include all of these in the ultimate KQ estimator. This was the approach taken in our experiments that involved `SMC-KQ-KL`. However, since it is somewhat cumbersome to include in the pseudo-code, we have not made this explicit in the notation. Our reported results are on a per-function-evaluation basis and so we **do** adjust for this detail in our reported comparisons.