Kernel Filtering

Benjamin Connault*

January 2018

Abstract

The paper describes a new approximate nonlinear filtering technique. Strengths of the technique include: (1) it can be used as long as one can simulate from the model, without the need to evaluate measurement densities, (2) it is easy to implement, yet competitive with state-of-the-art alternative techniques in terms of speed and accuracy, (3) it can be used with some models that include infinite-dimensional state variables. The main theoretical result of the paper is that the approximation error of the technique goes to zero with computational power, and that it does so uniformly with respect to the time horizon of the data.

*University of Pennsylvania, Department of Economics, connault@econ.upenn.edu
I thank Tim Christensen, Ian Dew-Becker, Leland Farmer, Jesus Fernandez-Villaverde, Ed Herbst, Frank Schorfheide, Neil Shephard and other participants at seminars where I presented preliminary versions of this work, as well as my colleagues at UPenn, and the Cowles foundation for its hospitality.

1 Introduction

Given a theory about the stochastic evolution of a set of economic variables $z_t = (x_t, y_t)$ and observations $y_{1:t}$, the *nonlinear filter* $\mu_t := q(x_t|y_{1:t})$ captures all we can know about the current unobserved variable x_t. In empirical applications, it is a quantity of foremost interest both in and of itself and as a stepping stone towards statistical estimation of the model.

Structural dynamic economic models are rarely, if ever, linear. Nonlinear filtering – the exercise of computing μ_t in a nonlinear model – remains challenging despite the existence of a few techniques, most notably particle filtering.

This paper describes a new nonlinear filtering technique called *kernel filtering*. Among the advantages of kernel filtering are: (1) the ability to use kernel filtering as long as one can simulate from the model, without the need to evaluate measurement densities, (2) easy implementation and computational speed, (3) guaranteed numerical accuracy properties, proved in this paper, at the level of state-of-art particle filtering, (4) applicability to some models with infinite-dimensional state variables such as heterogeneous agent models in macroeconomics.

The paper does not cover statistical estimation: the data generating process for z_t is assumed to be "known", "calibrated" or estimated by other means. Like any other approximate nonlinear filtering technique, kernel filtering could also be used within a likelihood-based estimation strategy, but this is left for future work.

Like most other nonlinear filtering techniques, kernel filtering follows the recursive relationship $\mu^\star_{t+1} = \Phi^\star_{t+1}(\mu^\star_t)$ satisfied by the true population nonlinear filter, by tracking an approximate nonlinear filter along approximate updates $\hat{\mu}_{t+1} = \hat{\Phi}_{t+1}\left(\hat{\mu}_t\right)$. The key ingredient of kernel filtering is a representation of probability measures as elements in certain vector spaces that make the approximations $\hat{\mu}_t$ and $\hat{\Phi}_{t+1}$ both computationally tractable and theoretically justified. The vector spaces in question are specific reproducing *kernel* Hilbert spaces, as in Guilbart (1979), which motivates the name *kernel* filtering.

This paper takes the perspective of a two-stage algorithm. In a pre-data stage, an approximation of the update function $\hat{\Phi}$ is built. This will typically involve stochastic approximation techniques, such as simulations. The second stage actually "runs the filter" on the data, using the approximate update function $\hat{\Phi}$. This is a deterministic stage: running the filter twice on the same data will give the same result.

The paper describes two flavors of kernel filtering. Full-rank kernel filtering (Algorithm 1) is simpler and more accurate for a fixed tuning parameter n that can be interpreted as a "grid size" or a "number of particles", but requires user-provided "grids" and does not scale well with n. Low-rank kernel filtering (Algorithm 2) adds one layer of approximation but picks grids adaptively and scales very well. All told low-rank kernel filtering (Algorithm 2) will be the preferred option in most configurations.

The theoretical part of the paper provides guarantees about the accuracy of kernel filtering. Specifically, Theorem 1 proves time-uniform error bounds that demonstrate that the technique does not accumulate approximation errors despite its recursive nature.

Simulation evidence demonstrates that kernel filtering compares favorably to the particle filter, both computationally and in terms of accuracy. This is despite the fact that the particle filter uses knowledge of the measurement density, a priori a strong advantage.

I present one example application of kernel filtering in a stylized dynamic supply/demand model. By design the model is easily simulated, but measurement densities are unavailable, making kernel filtering particularly suitable. The unobserved variable is a demand curve and the observed variable is the clearing price. Random independent supply curves play the role of "innovations". While price in a given period carries little information about the underlying demand curve, the time-series of prices *together with a model that disciplines how the demand curve may change from one period to the next* can have nontrivial information content.

The fact that probability measures can be fruitfully embedded in particular reproducing kernel Hilbert spaces has been known since at least Guilbart (1979). Kernel embeddings are related to the "kernel trick" of machine learning and have been used for various purposes, see eg. Berlinet and Thomas-Agnan (2003) or Steinwart and Christmann (2008). This paper makes a few contributions of independent interest to the theory of kernel embeddings, see eg. Lemma 2, Lemma 4 or Lemma 9. This paper's "kernel disintegrations" are similar to the "kernel Bayes rule" of Fukumizu et al. (2013). The kernel Bayes rule was used in the context of nonlinear filtering in two papers related to the current paper, Fukumizu et al. (2013) and Song et al. (2013). The proof technique used in the theoretical part of this paper follows the classical telescopic-sum-of-geometric-bounds tradition, see eg. Moral and Guionnet (2001). There are several alternative techniques to do approximate nonlinear filtering including particle filtering techniques (Pitt and Shephard (1999), Kantas et al. (2015)), and techniques based on discretization of x_t, see Farmer (2017) and references therein.

2 Kernel Filtering

This section introduces the (full-rank) kernel filtering algorithm. It starts with a brief motivation and then describes the algorithm in pseudo-code. Low-rank kernel filtering, a high-performance variant of the algoritm, is described in section 3. Formal statements and technical assumptions are given in the theoretical part of the paper, section 4.

A fixed hidden Markov data generating process $q^\star$ for an unobserved variable $x_t \in E_x$ and an observed variable $y_t \in E_y$ is given, in the form of a one-step-ahead Markov transition kernel $Q^\star(z, dz')$ for $z_t = (x_t, y_t)$. There is no unknown parameter to be estimated. Kernel filtering can be used as soon as the user knows how to simulate from $Q^\star(z, dz')$. In this section we assume that the Markov kernel factorizes as:

$$Q^\star(z, dz') = A^\star(x, dx')B^\star(x', dy') = A^\star(x, dx')b^\star(x', y')\lambda(dy')$$

The restriction on the dynamics is for ease of exposition only: kernel filtering can be used without any restriction on the dynamics.

Starting from an initial value $\mu_1^\star = q^\star(dx_1|y_1)$ and once the data $y_{1:t}$ has been realized and observed, the (population) nonlinear filter $\mu_t^\star = q^\star(dx_t|y_{1:t})$ satisfies a deterministic recursive equation. Writing $\eta_{t+1}^\star = q^\star(dx_{t+1}|y_{1:t})$ for the intermediary "predictive density", one moves from $\mu_t^\star$ to $\eta_{t+1}^\star$ by applying the Markov kernel $A^\star(x, dx')$, and moves from $\eta_{t+1}^\star$ to $\mu_{t+1}^\star$ by reweighting the measure $\eta_{t+1}^\star(dx')$ by the positive function $x' \to b_{t+1}^\star(x') := b^\star(x', y_{t+1})$, ie. by taking a Bayes step with prior $\eta_{t+1}^\star(dx')$, conditional $B^\star(x', dy')$, and data y_{t+1}.

The key idea of kernel filtering is to interpret these probabilistic operations in a functional-analytic framework that allows both for computationally tractable approximations and for formal analysis of the approximation errors. In other words we want to interpret the operation $\mu(dx) \to \lambda(dy) = \mu(dx)A(x, dy)$ as a continuous linear operation between two suitable vector spaces that include $\mathcal{P}(E_x)$ and $\mathcal{P}(E_y)$, and similarly for other operations such as marginalizations, forming joints, conditioning, etc. The theory of *kernel embeddings* provides us with such a context. Here we give a brief informal summary of this theory, and we postpone a formal exposition to section 4.

A kernel embedding space on a ground space E is a particular Hilbert space H of functions on E, such that any probability measure $\mu(dx)$ on E is represented[1] by some function f_μ in H. We will freely abuse notation by writing μ instead of f_μ. The space H is characterized by its *kernel function* $k(x, x') := \langle f_{\delta_x}, f_{\delta'_x} \rangle$, hence the name of *kernel embedding space*. Reciprocally, a suitable kernel function $k(x, x')$ induces a kernel embedding space H on E. In a kernel embedding space, one can explicitly compute the inner product of two measures as:

$$\langle \mu, \mu' \rangle_H = \mu(dx)\mu(dx')k(x, x')$$

[1] "Represented" in the following sense: for any test function $g \in H$, we can compute the integral of $g(x)$ with respect to $\mu(dx)$ by taking the inner product of $f_\mu(x)$ and $g(x)$ in H, $\mu(dx)g(x) = \langle f_\mu, g \rangle_H$. f_μ is *not* a density with respect to some dominating measure.

Two examples of kernels on $E = [0, 1]$ are:

$$k(x, x') = e^{-|x-x'|} \text{ (the Laplace kernel)} \qquad \text{and} \qquad k(x, x') = e^{-(x-x')^2} \text{ (the Gauss kernel)}$$

Kernel filtering approximates the true nonlinear filter $\mu_t^\star$, seen as a vector in H, by a finite linear combination of basis vectors $B_x = \nu^x_{1:n_x}$. Every basis vector $\nu_i^x \in H$ is chosen to be a probability measure, and the linear combinations are constrained to be simplex combinations, so that $\hat{\mu}_t = \sum_{i=1}^{n_x} w_{ti}\nu_i^x$ can be interpreted as a mixture of probabilities. The approximation error is measured by $\|\mu_t^\star - \hat{\mu}_t\|_H$. For suitable kernels, this is a meaningful way to measure the error – specifically $\|\cdot\|_H$ is a metric for the standard weak topology on probability measures. Theorem 1 in section 4 shows that the approximation error of the kernel filtering algorithm goes to zero as computing power increases.

We now turn to describing the kernel filtering algorithm. Following the structure of the true population update function $\Phi_t^\star$, the approximate update function $\hat{\Phi}_t$ happens in two steps: a Markov step, and a Bayes step. We start with $\hat{\mu}_t$ with coordinates w_t in B_x.

In the Markov step, the transition $A^\star(x, dx')$ is approximated by a Markov transition matrix $\hat{A}$ in B_x. We simply define $\hat{A}$ row-by-row: $\hat{A}\nu_i^x := \hat{a}_i$, where $\hat{a}_i(dx') = \sum_{j=1}^{n_x} a_{ij}\nu_j^x(dx')$ is close to $a_i^\star(dx') := \nu_i^x(dx)A^\star(x, dx')$. One way to compute the a_{ij} weights is to simulate a large sample $\tilde{x}_{1:L}$ from $a_i^\star(dx')$ and to compute the orthogonal projection of $\tilde{a}_i = \sum_l \frac{1}{L}\delta_{\tilde{x}_l}$ on B_x, followed by a simplex normalization. The finite-rank approximation of $\mu \to \mu A$ in the B_x basis is simply $w \to w\hat{A}$. This is exactly like a finite-state Markov transition.

The Bayes step works as follows. Think of Bayes rule with prior η, conditional B and data y_{t+1} as forming the joint $J(dx, dy) = \eta(dx)B(x, dy)$, disintegrating J as $m(dy)\pi(y, dx)$ and finally conditioning on the realized data value $\pi(y_{t+1}, dx)$. Kernel filtering starts from an approximation $\hat{B}$ of $B^\star$ computed similarly to $\hat{A}$, computes an approximate joint $\hat{\eta} \to \hat{J} = \hat{\eta}\hat{B}$, and finally realizes an approximate disintegration using a regularization strategy. The in-basis expression for computing kernel disintegrations is given in pseudo-code below. The theoretical study of kernel disintegrations and how they contribute to the approximation error of kernel filtering is one of the core contributions of the paper, see in particular Theorem 1, Lemma 9 and Lemma 14.

We can now summarize the kernel filtering algorithm. In a pre-data stage, we pick bases B_x and B_y for probability measures on E_x and E_y. Then we build the matrices $\hat{A}$ and $\hat{B}$ that approximate $A^\star(x, dx)$ and $B^\star(x, dy)$ in those bases. In a post-data stage, we actually "run the filter" by applying successive Markov and Bayes steps, starting from the coordinates w_1 in B_x of an initial $\hat{\mu}_1$. These two stages are described in pseudo-code below. `gram(`$u_{1:m}$`,`$v_{1:n}$`)` means the Gram matrix $G_{ij} = \langle u_i, v_j\rangle_H$ and `gram(`$u_{1:m}$`,`$x_{1:n}$`)` is shorthand for `gram(`$u_{1:m}$`,`$(\delta_{x_i})_{i\in 1:n}$`)`. `probnorm(`$\cdot$`)` is a simplex normalization.

```
tuning parameter: m
output: Â

Â = zeros(n_x,n_x)
for i=1:n_x
  x2 = zeros(m)
  for j=1:m
    x = rand(ν_i^x)
    x2[j] = rand(A*,x)
  end
  w2 = ones(m)/m
  ai = gram(B_x,B_x)^{-1} * gram(B_x,x2) * w2
  Â[i,:] = probnorm(ai)
end
return Â
```

```
tuning parameter: m
output: B̂

B̂ = zeros(n_x,n_y)
for i=1:n_x
  y2 = zeros(m)
  for j=1:m
    x = rand(ν_i^x)
    y2[j] = rand(B*,x)
  end
  w2 = ones(m)/m
  bi = gram(B_y,B_y)^{-1} * gram(B_y,y2) * w2
  B̂[i,:] = probnorm(bi)
end
return B̂
```

Algorithm 1: kernel filtering. Pre-data stage.

```
tuning parameter: τ
other inputs: Â, B̂, w_1
output: μ̂_{1:T}

w = zeros(n_x,T)
w[:,1] = w_1
for t=1:T-1
  η_{t+1} = w[:,t]' * Â
  J = diagm(η_{t+1}) * B̂
  D = diagm(η_{t+1} * B̂)
  m = J * (gram(B_y,B_y) * D + τ I)^{-1} * gram(B_y,y_{t+1})
  w[:,t+1] = probnorm(m)
end
return w
```

Algorithm 1: kernel filtering. Post-data stage.

Kernel smoothing is also available, using similar techniques. Using suitable combinations of Markov steps and Bayes steps, kernel filtering and smoothing can accommodate models with arbitrary hidden Markov dynamics.

We make a few comments on the algorithm's parameters.

Suitable kernel functions on E_x and E_y need to be chosen. Lemma 4 identifies three properties that make a kernel function "suitable" for the purpose of kernel filtering[2]. Lemma 5 proves that a modified Laplace kernel $k(x, x') = 0.9e^{-|x-x'|} + 0.1$ is suitable on $\mathbb{R}$ and its subsets and that the Laplace kernel itself $k(x, x') = e^{-|x-x'|}$ is suitable on bounded subsets of $\mathbb{R}$. In applications, x or y often have several subcomponents. For instance, x may include

[2] $\|\cdot\|_H$ metrizes weak convergence for probability measures; the constant functions belong to H; and multiplication $(f, f') \to ff'$ is a bounded bilinear operation from $H \times H$ to H

a discrete "regime" variable x_1, and two types of continuous variables, $x_2 \in \mathbb{R}^+$ and $x_3 \in \mathbb{R}^2$. Lemma 6 shows that the kernel $k((x,y),(x',y')) = k_x(x,x')k_y(y,y')$ is suitable on $E_x \times E_y$ when k_x, k_y are suitable on E_x, E_y. Finding suitable kernels for more exotic spaces, such as infinite-dimensional state spaces, is a topic of interest. For example, Lemma 7 suggests to use $\hat{k}(\mu,\mu') = 0.9e^{-\|\mu-\mu'\|_k} + 0.1$ on $\mathcal{P}(E)$, where k is a suitable kernel on E, and makes partial progress towards proving suitability of $\hat{k}$.

The choice of bases B_x and B_y matters, as expressed in assumptions (A1-3) of Theorem 1. Domain-specific knowledge should be used when possible. A basis of Dirac probability measures δ_{x_i} – a "grid" x_i – taken from a long simulated time-series is a reasonable default option. The Gram matrix of a basis δ_{x_i} is simply the matrix $G_{ij} = k(x_i, x_j)$. See also after Theorem 1 for additional comments. The number of simulation draws m used to approximate each row of $\hat{A}$ or $\hat{B}$ can typically be taken large enough to make the simulation error small compared to the approximation error induced by the basis projections.

The regularization parameter τ is a bandwidth-type parameter. It should be taken of order $\epsilon_2^{\frac{1}{1+\alpha}}$, where ϵ_2 measures the approximation error in $\hat{B}$ and α is a smoothing parameter between 0 (light smoothing) and 1 (strong smoothing), see Theorem 1 for a precise statement. Note that ϵ_2 can typically be computed with good accuracy in the process of building $\hat{B}$. Standard considerations about bandwidth choice rules apply, although they are out-of-scope for this paper.

Under the assumptions of Theorem 1, the initialization error in $\hat{\mu}_1$ disappears quickly.

Kernel filtering (Algorithm 1) has the following computational complexity, as a function of the dimensions n_x and n_y of the bases B_x and B_y. The $(n_x \times n_x)$ and $(n_x \times n_y)$ matrices $\hat{A}$ and $\hat{B}$ must be kept in memory, and repeated multiplication by $\hat{A}$ and $\hat{B}$ means that the post-data stage has computational cost $O(Tn_x(n_x + n_y))$. In practice this means that Algorithm 1 is not practical beyond one or two thousands basis elements. One way to mitigate the computational complexity of the post-data stage is to compute low-rank factorizations of $\hat{A}$ and $\hat{B}$, eg. using truncated singular value decomposition. This brings the running time of the post-data stage to $O(Tr(n_x + n_y))$ for some low rank r. However the matrices $\hat{A}$ and $\hat{B}$ still need to be computed and held in memory beforehand, making this solution not entirely satisfying. Algorithm 2 in the next section uses an alternative pre-data stage to obtain low-rank approximations $\hat{A}_r$ and $\hat{B}_r$ directly, without the need to compute (full-rank) $\hat{A}$ and $\hat{B}$ in the first place.

3 High Performance Variant: Low-Rank Kernel Filtering

This section presents the low-rank kernel filtering algorithm, a high-performance variant of kernel filtering. Low-rank kernel filtering is the better option in most configurations. First, it can computationally scale up to many more "grid points" n at the cost of a small accuracy loss for a fixed n, with net effect typically being much higher accuracy. Second the "grid points" are selected automatically by the algorithm, and the bandwidth parameter τ is replaced by a rank parameter $r \in \mathbb{N}$ that is directly tied to the computational complexity, making low-rank kernel filtering also more user-friendly.

Low-rank kernel filtering is similar to kernel filtering (Algorithm 1) except that $\hat{A}$ and $\hat{B}$ are approximated by low-rank matrices $\hat{A}_r$ and $\hat{B}_r$. $\hat{A}_r$ and $\hat{B}_r$ are computed directly in a modified pre-data stage: (1) simulate a sample $\hat{J} = (x_i, x_i')_{1:n_x}$ from a joint distribution $\nu(dx)A^\star(x, dx')$, for $\nu(dx)$ a probability measure that covers E_x in a suitable fashion, (2) compute $\hat{A}_r$ by kernel disintegration from $\hat{J}$, together with low-rank approximations of the relevant Gram matrices. Good quality low-rank approximations for Gram matrices can be obtained using the Nyström method, see Gittens and Mahoney (2016). Explicit pseudo-code is given below:

```
tuning parameters: ν, n, r
output: G_10, G_20, G_40, B_x, B_y

x1 = zeros(n)
bx = zeros(n)
by = zeros(n)
for i=1:n
  x1[i] = rand(ν)
  bx[i] = rand(A*,x1[i])
  by[i] = rand(B*,x1[i])
end
x0 = sample(x1,r,replace = false)
y0 = sample(by,r,replace = false)
G_10 = gram(x1,x0)
G_20 = gram(bx,x0)
G_40 = gram(by,y0)
return G_10, G_20, G_40, bx, by, y0
```

Algorithm 2: low-rank kernel filtering. Pre-data stage.

```
tuning parameter: none
other inputs: w_1
output: \hat{\mu}_{1:T}

w = zeros(n_x,T)
w[:,1] = w_1
for t=1:T-1
  η_{t+1} = w[:,t]' * G_20 * (G_01 G_10)^{-1} * G_01
  J = diagm(η_{t+1}) * G_20 * (G_01 G_10)^{-1} * G_01
  D = diagm(η_{t+1} * G_20 * (G_01 G_10)^{-1} * G_01)
  G_05 = gram(y0,y_{t+1})
  pi = J * G_40* (G_04 D^2 G_40)^{-1} * (G_04 D G_40) * (G_04 G_40)^{-1} * G_50
  w[:,t+1] = probnorm(pi)
end
return w
```

Algorithm 2: low-rank kernel filtering. Post-data stage.

For computational efficiency the low-rank matrices $\hat{A}_r$ and $\hat{B}_r$ are never stored in memory in Algorithm 2, but their expression would be (see appendix section 8 for a derivation):

$$\hat{A}_r = G_{20}(G_{10}G_{01})^{-1}G_{01} \qquad \text{and} \qquad \hat{B}_r = G_{20}(G_{10}G_{01})^{-1}G_{01}$$

A heuristic description of low-rank kernel filtering at a more abstract level is as follows. The true transition kernels $A^\star(x, dx')$ and $B^\star(x', dy')$, seen as operators between the relevant kernel embedding spaces, admit finite-rank approximations $A^\star_r$ and $B^\star_r$ that are of good quality if $A^\star$ and $B^\star$ are smooth enough. An unfeasible but very high-quality kernel filter would use $A^\star_r$, $B^\star_r$ and the bases B_x, B_y spanned by the corresponding leading singular vectors. Algorithm 2 is an approximate, randomized version of this unfeasible, high-quality kernel filter.

4 Theory: accuracy of the Kernel Filtering Algorithm

In section 4.1, we describe the theory of kernel embeddings. Kernel embeddings go back to at least Guilbart (1979). This paper makes some contributions of independent interest to this theory, eg. Lemma 2, Lemma 4 or Lemma 9. In section 4.2 we present the paper's main theorem, time-uniform error bounds for kernel filtering, Theorem 1. First we agree on some notation.

E is a Polish metric space considered with its Borel σ-algebra. $\mathcal{M}(E)$, $\mathcal{M}^+(E)$, $\mathcal{P}(E)$ are the spaces of (signed) bounded measures, positive bounded measures, probability measures on E. $B(E)$ is the space of bounded measurable functions on E. We may drop E from the notation when the context is clear. For Hilbert spaces H_x, H_y, $B(H_x, H_y)$, $B_2(H_x, H_y)$, $B_1(H_x, H_y)$ are the spaces of bounded, Hilbert-Schmidt, trace-class linear operators with the operator norm $\|\cdot\|$, the Hilbert-Schmidt norm $\|\cdot\|_2$ and the trace norm $\|\cdot\|_1$. $K(H_x, H_y)$ is the compact operators, a closed subspace of $B(H_x, H_y)$. We may write $B(H)$ for $B(H, H)$ and similarly for $B_2(H)$, $B_1(H)$, $K(H)$. For two Hilbert spaces H_x and H_y, $H_x \otimes H_y$ means the Hilbert tensor product. We will frequently use the canonical isomorphism between $t \in H_1 \otimes H_2$ and $T \in B_2(H_1, H_2)$ given by $\langle t, f \otimes g\rangle = \langle Tf, g\rangle$ (the equivalent of vectorizing a matrix in finite dimensions). We will use $\langle \cdot, \cdot \rangle$ as general notation for duality brackets, eg. the inner product in a Hilbert space or $\langle \mu, f\rangle = \mu(dx)f(x)$ for $\mu \in \mathcal{M}(E)$ and $f \in B(E)$.

A kernel k is a function from $E \times E$ to $\mathbb{R}$ that is symmetric positive definite, i.e. for any x, x' in E, $k(x, x') = k(x', x)$, and for any $x_{1:n} \in E$ and $a_{1:n} \in \mathbb{R}$, $\sum_{i,j} a_i k(x_i, x_j) a_j \geq 0$. Recall that a reproducing kernel Hilbert space H on E is a Hilbert space of functions on E such that evaluation at any point $x \in E$ is a bounded linear operation on H. By Riesz representation, evaluation at any point x can be obtained by inner product with some element $k_x \in H$: for any $f \in H$, $f(x) = \langle k_x, f\rangle$. The function $k(x, x') = \langle k_x, k_{x'}\rangle$ is a kernel, called the reproducing kernel of H. The linear span of the k_x's is a dense subset of H. Reciprocally, the Moore-Aronzajn theorem states that a kernel k induces a unique reproducing kernel Hilbert space H of functions on E with k as reproducing kernel. In this paper, we will consider only bounded, (jointly) continuous kernel, in which case all the elements of H are continuous bounded functions on E. See Saitoh (1997) for more details.

4.1 Kernel embeddings

If k is a bounded continuous kernel, $\mathcal{M}(E)$ can be "embedded" in H in the following sense. First note that for $f \in H$:

$$\|f\|_\infty = \sup_{x \in E} |\langle \delta_x, f\rangle| \leq \gamma_k \|f\|_H \qquad \text{for any } \gamma_k \text{ such that } \|k_x\| = \sqrt{k(x,x)} \leq \gamma_k$$

As a consequence any $\nu \in \mathcal{M}(E)$ induces a bounded linear functional on H:

$$\langle \nu, f\rangle \leq \|\nu\|_{TV} \|f\|_\infty \leq \|\nu\|_{TV} \gamma_k \|f\|_H$$

By Riesz representation, there is a (unique) $f_\nu \in H$ such that for any $f \in H$, $\nu(dx)f(x) = \langle f_\nu, f\rangle$. We may sometimes abuse notation and write ν for f_ν. The embedding induces a topology on $\mathcal{M}(E)$ and its subsets via the pseudometric $d_H(\nu, \nu') = \|\nu - \nu'\|_H$.

Kernel embeddings are appealing because they give access to a computationally tractable "probabilistic calculus": marginalizations, forming joints or applying Markov kernels are continuous linear operations, there are natural notions of orthogonal projections, basis representations, finite-rank approximations, etc. In this section we develop the main tools of this calculus. We start by identifying three regularity properties that make a kernel embedding well-behaved. Then we give examples of kernels that satisfy all three regularity conditions and a recipe to build new kernels (Lemma 5). Finally we consider embeddings of Markov transition kernels and kernel disintegrations. All proofs are given in section 7.

The first and foremost regularity condition that we require is that the topology induced by d_H on $\mathcal{P}(E)$ be statistically meaningful. In general a kernel embedding needs not even be injective, meaning $d_H(\mu, \mu')$ could be zero for two distinct $\mu \neq \mu' \in \mathcal{P}(E)$, or equivalently μ and μ' would be represented by the same function $f_\mu \in H$ – there would not be enough test functions in H to tell apart μ and μ'. Guilbart (1979) proved the following:

Lemma 1:
Let E be a Polish metric space.

(i) There exists a bounded continuous kernel k on E such that k induces the weak topology on $\mathcal{P}(E)$ (Théorème d'existence 4.5).

(ii) If k is bounded continuous, the topology induced on $\mathcal{P}(E)$ is weaker than weak (Théorème de faible comparaison 1.5).

We will say that k is *characteristic* if it induces the weak topology on $\mathcal{P}(E)$ – the finest we can hope for. For a subset Ω of $\mathcal{M}(E)$, we will also say that k is Ω-*nondegenerate* if for $\nu, \nu' \in \Omega$, $d_H(\nu, \nu') = 0$ implies $\nu = \nu'$. At face-value $\mathcal{P}(E)$-nondegeneracy is much weaker than the characteristic property, but it turns out that both properties are equivalent if E is locally compact Hausdorff, see Theorem 55 in Simon-Gabriel and Schölkopf (2016), quoted as Lemma 11 in section 7. For translation-invariant kernels $k(x, x') = \phi(x - x')$ on Euclidean spaces, there is an explicit criterion to detect whether a kernel is characteristic, see Corollary 33 in Simon-Gabriel and Schölkopf (2016), quoted as Lemma 10 in section 7.

We know turn to kernel marginalizations and a second regularity property. For two reproducing kernel Hilbert spaces H_x, H_y with kernels k_x, k_y on ground spaces E_x, E_y, the Hilbert tensor product $H = H_x \otimes H_y$ is a reproducing kernel Hilbert space on $E_x \times E_y$ with kernel $k((x, y), (x', y')) = k_x(x, x')k_y(y, y')$ (see Saitoh (1997)). If k_x and k_y are continuous and bounded then so is k, and H embeds joint probability measures. For joints $\mu(dx, dy)$, $\mu'(dx, dy)$ with corresponding marginals $\mu_x(dx)$, $\mu'_x(dx)$, we would like $\|\mu_x - \mu'_x\|_{H_x}$ to be small when $\|\mu - \mu'\|_H$ is small, in other words it is natural to require that the operation of marginalizing joint probability measures admit a continuous linear extension from H to H_x. The following lemma gives a necessary and sufficient condition for this to be the case, thereby identifying a second regularity condition for kernel embeddings.

Lemma 2: Kernel marginalizations and embedding spaces having constants
Let H_x and H_y embed probability measures. Marginalization extends to a bounded linear operator Ψ_M from $H_x \otimes H_y$ to H_x if, and only if, the constant function 1 belongs to H_y. When this is the case, the extension is unique and $\|\Psi_M\| = \|1\|_{H_y}$.

The kernel embedding of a probability measure μ in H comes attached with a covariance operator D_μ that will play an important role further down.

Lemma 3: Covariance operators
Let $\mu \in H$ be the kernel embedding of a probability measure. The expression $\langle D_\mu f, g\rangle = \mu(dx)f(x)g(x)$ defines a bounded operator D_μ on H which is positive self-adjoint and trace-class, with $\|D_\mu\|_1 = \mu(dx)k(x, x)$.

In particular, kernel functions identically equal to one on their diagonal, ie. $k(x,x) = 1$, provide an appealing normalization $\|D_\mu\|_1 = 1$. It will be useful to be able to control the error $\|D_\mu - D_{\mu'}\|$ when $\|\mu - \mu'\|$ is small, in other words it is natural to require that the operation $\mu \to D_\mu$ extend to a continuous linear operator from H to $B(H)$. The following lemma gives a necessary and sufficient condition for this to hold, thereby identifying a third regularity condition.

Lemma 4: Continuity of the covariance operator map, embedding spaces having products
The map $\mu \to D_\mu$ extends to a bounded linear operator Ψ_D from H to $B(H)$ if, and only if, multiplication is a bounded bilinear operation M on $H \times H$, meaning thar for any $f, g \in H$:

$$\|fg\|_H = \|M(f,g)\|_H \leq \|M\| \, \|f\|_H \, \|g\|_H$$

When this is the case, the extension is unique and $\|\Psi_D\| = \|M\|$.

We have identified three regularity conditions that make kernel embeddings well-behaved with respect to basic probabilistic operations: being characteristic, having constants, and having products. Lemma 1 only guarantees the existence of kernels with the first property. Do kernels that satisfy all three regularity properties even exist? For concrete common ground spaces E, do we know how to pick a kernel that satisfies all three regularity properties? The next lemma provides a positive answer to both questions:

Lemma 5:

(i) $k_1(x,x') = e^{-|x-x'|}$ *is characteristic, has constants and has products on $E = [0,1]$.*

(ii) $k_2(x,x') = 0.9e^{-|x-x'|} + 0.1$ *is characteristic, has constants and has products on $E = \mathbb{R}$.*

(iii) $k_3(x,x') = e^{-(x-x)^2}$ *does not have constants and does not have products on $E = [0,1]$ nor on $E = \mathbb{R}$.*

If k_x and k_y have all three regularity properties, does $k((x,y),(x',y')) = k_x(x,x')k_y(y,y')$ do as well on $E_x \times E_y$? Not quite, but almost, as proved in the following lemma. This allows one to obtain a wide range of well-behaved kernels from a few standard well-behaved kernels, such as those of Lemma 5.

Lemma 6:

(i) If k_x, k_y are $\mathcal{P}$-nondegenerate and have constants, then k is $\mathcal{P}$-nondegenerate and has constants.

(ii) If k_x, k_y are characteristic and have constants, and if $E_x \times E_y$ is locally compact Hausdorff, then k is characteristic and has constants.

(iii) If k_x is such that $f \otimes f' \to ff'$ can be extended to a continuous linear operation from $H_x \otimes H_x$ to H_x, and similarly for k_y, then k also has this property.

The assumption in (iii) is slightly stronger than having products. Lemma 12 in the appendix gives a criterion to detect whether a kernel has such "strong products". An inspection of the proof of Lemma 5 shows that the kernels k_1 and k_2 of Lemma 5 have "strong products".

What about more exotic ground spaces, in particular infinite-dimensional state spaces? We would like to have access to well-behaved kernels on those spaces in order, for example, to apply kernel filtering techniques when one of the state variables is infinite-dimensional. In most cases this remains an open question. As long as E is Polish – which is enough for almost all applications – Lemma 1 guarantees the existence of a characteristic kernel k, and Lemma 13 in section 7.2 shows that k can be taken with constants without loss of generality. However this does not tells us how to pick a kernel in practice. The following lemma suggests a kernel for the ground space $\hat{E} = \mathcal{P}(E)$ and makes partial progress towards proving that it has good properties.

Lemma 7:
Let k induce a kernel embedding on a ground space E. Consider $\hat{E} = \mathcal{P}(E)$ and define the kernel $\hat{k}$ on $\hat{E}$ by:

$$\hat{k}(\mu, \mu') = 0.9e^{-\|\mu-\mu'\|_k} + 0.1$$

$\hat{k}$ is continuous bounded and induces a kernel embedding on $\hat{E}$. Furthermore:

(i) $\hat{k}$ has constants.

(ii) If k is $\mathcal{P}$-nondegenerate, then $\hat{k}$ is fa-nondegenerate, where fa is the set of finite measures, ie. finite linear combinations of Dirac measures δ_{μ_i}, $\mu_i \in \mathcal{P}(E)$.

So far we have seen how to embed probability measures as vectors in particular vector spaces. Now we turn to Markov transition kernels $Q(x, dy)$, which we would like to see as linear operators between vector spaces $\mu \in H_x \to T_Q\mu \in H_y$, where of course $T_Q\mu \in H_y$ would also be the embedding of $\mu(dx)Q(x, dy)$ in H_y.

Definition 1:
Let H_x, H_y kernel embedding spaces and $Q(x, dy)$ a Markov transition kernel. We will say that Q has a kernel embedding *if $\delta_x \in H_x \to Q(x, dy) \in H_y$ can be extended continuously as a linear operator from H_x to H_y.*

The assumption in Definition 1 can be seen as a type of Feller continuity assumption. By duality, Q has a kernel embedding if the conditional expectation of a test function $g \in H_y$ belongs to H_x. As an example, if $E_x = E_y = [0, 1]$ and $k(x, x') = e^{-|x-x'|}$, then $H_x = H_y$ is the Sobolev space $W_{21}(0, 1)$, see Saitoh (1997).

The following lemma shows that in kernel embedding spaces you form the joint of $\mu(dx)$ and $Q(x, dy)$ by the linear operation $J = D_\mu Q$.

Lemma 8: Kernel joints
Let H_x and H_y kernel embedding spaces, $\mu_x(dx)$ a probability measure, $Q(x, dy)$ a Markov

transition kernel with kernel embedding and $\mu(dx, dy) = \mu_x(dx)Q(x, dy)$. Then $J = D_{\mu_x}Q$ is the kernel embedding of μ in $H_1 \otimes H_2$.

Finally consider the operation of disintegrating (or "conditioning"). Conditioning is *not* a continuous operation. To see this, suppose $J = D_\mu Q$ as in Lemma 8. Heuristically $Q = D_\mu^{-1} J$, but D_μ is no invertible since it is trace-class, as shown in Lemma 3. Instead we compute kernel disintegrations using a regularization strategy. The following lemma is one of the key technical lemmas of the paper:

Lemma 9: Kernel disintegrations
Let H_x and H_y be kernel embedding spaces, H_x with a bounded multiplication operator M^x and H_y with constants. Let $\mu(dx, dy)$ be a joint probability distribution on $E_x \times E_y$ with disintegration $\mu_x(dx)Q_\mu(x, dy)$. Let $J \in H_x \otimes H_y$ be the kernel embedding of μ and $D \in B(H_x)$ be the covariance operator of μ_x. Assume that Q_μ has a kernel embedding Q that satisfies the following smoothness condition: there is $W \in B_2(H_y, H_x)$, $c_W > 0$ and $0 < s \leq 1$ such that:

$$Q = D^s W \qquad \text{and} \qquad \|W\|_2 \leq c_W \tag{1}$$

By Lemma 8, it holds:

$$DQ = J$$

Assume also that D is injective. Let $\hat{\mu}(dx, dy)$ be another joint probability distribution on $E_x \times E_y$ with kernel embedding $\hat{J}$. Define:

$$\delta := \left\| \hat{J} - J \right\|_2$$

Finally, write $\hat{D}$ for the covariance operator of the marginal $\hat{\mu}_x$ of $\hat{\mu}$ on E_x and define:

$$\hat{Q} = \left(\hat{D} + \tau I\right)^{-1} \hat{J} \qquad \text{with} \qquad \tau = \delta^{\frac{1}{1+s}}$$

Then:

$$\left\| \hat{Q} - Q \right\|_2 \leq (1 + \|M^x\| \, \|1\|_{H_y} c_W + c_W)\delta^{\frac{s}{1+s}}$$

4.2 Accuracy of the kernel filter

We now state the main result of this paper. Theorem 1 shows not only that the approximation error induced by kernel filtering goes to zero with computing power, but also that it does so in a time-uniform manner. In other words, kernel filtering does not accumulate numerical error across time despite its recursive structure.

Theorem 1: Time-uniform error bounds

Let E_x, E_y be Polish metric spaces and $(X_t, Y_t) \in E_x \times E_y$ follow (strict) hidden Markov dynamics with transition kernel $A(x, dx)$ and measurement kernel $B(x, dy)$. Let H_x and H_y be embedding spaces with products and constants. Compute an approximate nonlinear filter $\hat{\mu}_t$ via the kernel filtering algorithm described in section 2, with $\tau = \epsilon_2^{\frac{1}{1+\alpha}}$ where ϵ_2 and α are defined below. Assume:

(A1) $\hat{A}$ is such that:

$$\max_{\nu_i^x \in B_x} \|\nu_i^x(dx)A(x, dx) - \hat{a}_i(dx)\|_{H_y} \leq \epsilon_1$$

(A2) $\hat{B}$ is such that:

$$\max_{\nu_i^x \in B_x} \left\|\nu_i^x(dx)B(x, dy) - \nu_i^x(dx) \otimes \hat{b}_i(dy)\right\|_{H_x \otimes H_y} \leq \epsilon_2$$

(A3) The distance between δ_{y_t} and the linear span of B_y is bounded by ϵ_3 (almost surely).

(A4) The initialization error $\|\hat{\mu}_1 - \mu_1\|$ is bounded by ϵ_4.

(A5) The conditional distribution $(X_{t+1}|Y_{t+1}, y_{1:t})$ has a kernel embedding $Q \in B(H_y, H_x)$ that satisfies the following smoothness assumption. Write D for the covariance operator of the distribution of $Y_{t+1}|y_{1:t}$. There is $W \in B_2(H_y, H_x)$, $c_W > 0$ and $0 < \alpha \leq 1$ such that:

$$Q = D^\alpha W \qquad \text{and} \qquad \|W\|_2 \leq c_W \qquad\qquad y_{1:t} - a.s.$$

(A6) $B(x, dy)$ has density $b(x, y)$ with respect to some dominating measure $\lambda(dy)$ and there is $c_b > 0$ such that $b_{t+1}(x) := b(x, y_{t+1}) \in H_x$ and $\frac{\|b_{t+1}\|_{H_x}}{\inf_{x \in E_x} b_{t+1}(x)} < c_b$ (a.s.).

(A7) Call $\hat{p}_{s+1:t}$ the density of $(Y_{s+1:t}|X_s)$ with respect to λ^t and $p_{s+1:t}(x) := \hat{p}_t(y_{s+1:t}|X_s = x)$. There is $c_p > 0$ such that for any s, t, $p_{s+1:t} \in H_x$ and $\frac{\|p_{s+1:t}\|_{H_x}}{\inf_{x \in E_x} p_{s+1:t}(x)} < c_p$ (a.s.).

(A8) For any t, $(X_1, dX_t|y_{1:t})$ has a kernel embedding Q_t and there is $c_Q > 0$ and $\rho < 1$ such that, for any t, $\|Q_t\| \leq c_Q \rho^t$ (a.s.).

Then there are constants γ_1, γ_2, γ_3, γ_4, independent of t, such that:

$$\|\mu_t - \hat{\mu}_t\|_{H_x} \leq \gamma_1 \epsilon_1 + \gamma_2 \epsilon_2^{\frac{\alpha}{1+\alpha}} + \gamma_3 \epsilon_3 + \gamma_4 \rho^t \epsilon_4$$

Proof. Following a proof technique going back to at least Moral and Guionnet (2001), we decompose the approximation error as follows:

$$\begin{aligned}
\hat{\mu}_t - \mu_t &= \hat{\Phi}_t\left(\hat{\mu}_{t-1}\right) - \Phi_t\left(\hat{\mu}_{t-1}\right) + \Phi_t\left(\hat{\mu}_{t-1}\right) - \Phi_t\left(\mu_{t-1}\right) \\
&= \ldots \\
&= \sum_{s=2}^{t}\left(\Phi_{s+1:t}\left(\hat{\Phi}_s\left(\hat{\mu}_{s-1}\right)\right) - \Phi_{s+1:t}\left(\Phi_s\left(\hat{\mu}_{s-1}\right)\right)\right) \\
&\quad + \left(\Phi_{2:t}\left(\hat{\mu}_1\right) - \Phi_{2:t}\left(\mu_1\right)\right)
\end{aligned}$$

The proof is done in two parts: (1) controlling the one-step-ahead approximation error and (2) controlling the many-steps-ahead propagation error. The one-step-ahead approximation error is controlled as follows:

$$\begin{aligned}
\left\|\hat{\Phi}_s\left(\hat{\mu}_{s-1}\right) - \Phi_s\left(\hat{\mu}_{s-1}\right)\right\| &= \left\|\hat{M}_s\left(\hat{A}\left(\hat{\mu}_{s-1}\right)\right) - M_s\left(A\left(\hat{\mu}_{s-1}\right)\right)\right\| \\
&\leq \left\|\hat{M}_s\left(\hat{A}\left(\hat{\mu}_{s-1}\right)\right) - M_s\left(\hat{A}\left(\hat{\mu}_{s-1}\right)\right)\right\| \\
&\quad + \left\|M_s\left(\hat{A}\left(\hat{\mu}_{s-1}\right)\right) - M_s\left(A\left(\hat{\mu}_{s-1}\right)\right)\right\|
\end{aligned}$$

The first term is the approximation error in the Bayes step and the second term is the approximation error in the Markov step propagated through one Bayes step. The Bayes step error is controlled by Lemma 14 using (A2), (A3) and (A5):

$$\left\|\hat{M}_s\left(\hat{A}\left(\hat{\mu}_{s-1}\right)\right) - M_s\left(A\left(\hat{\mu}_{s-1}\right)\right)\right\| \leq (1 + \|M^y\|\,\|1\|_{H_x} c_W + c_W)\epsilon_2^{\frac{\alpha}{1+\alpha}} + c_W\epsilon_3$$

The Markov step error is controlled by Lemma 15 using (A1):

$$\left\|\hat{A}\left(\hat{\mu}_{s-1}\right) - A\left(\hat{\mu}_{s-1}\right)\right\| \leq \epsilon_1$$

The propagation error is controlled by Lemma 16 using (A6):

$$\left\|M_s\left(\hat{A}\left(\hat{\mu}_{s-1}\right)\right) - M_s\left(A\left(\hat{\mu}_{s-1}\right)\right)\right\| \leq \|M^x\|\left(c_b^2 + c_b\right)\epsilon_1$$

The many-steps-ahead propagation error is controlled by proving a geometric contraction property of the population filter. This is independent of the approximation technique being used. Under (A7) and (A8), it is proved in Lemma 17 that there is $\kappa > 0$ such that:

$$\|\Phi_{s+1:t}(\mu) - \Phi_{s+1:t}(\mu')\|_H \leq \kappa\rho^{t-s}\,\|\mu - \mu'\|_H$$

A geometric sum of the one-step-ahead error bounds concludes the proof:

$$\begin{aligned}
\|\hat{\mu}_t - \mu_t\| \leq \frac{\kappa}{1-\rho}\,\|M^x\|\left(c_b^2 + c_b\right)\epsilon_1 + \frac{\kappa}{1-\rho}(1 + \|M^y\|\,\|1\|_{H_x} c_W + c_W)\epsilon_2^{\frac{\alpha}{1+\alpha}} \\
+ \frac{\kappa}{1-\rho}c_W\epsilon_3 + \frac{\kappa}{1-\rho}\epsilon_4
\end{aligned}$$

□

We make a few comments.

(A1-4) measure the "input" approximation errors while (A5-A8) control how those input errors propagate to the output error. For a non-Dirac basis B_x, (A2) can be much stronger than (A1): asking that the independent product $\nu_i^x(dx) \otimes \left(\sum_i b_{ij}\nu_j^y(dy)\right)$ be a good approximation of the joint $\eta_i^\star(dx,dy) = \nu_i(dx)B(x,dy)$ is much stronger than asking that $\sum_i b_{ij}\nu_j^y(dy)$ be a good aproximation of the marginal $b_i^\star(dy) = \nu_i(dx)B(x,dy)$, except if $\nu_i(dx)$ is a Dirac. (A3) is quite strong and would imply that very large bases must be used for a small ϵ_3. However an inspection of the proof of Lemma 14 shows that the relevant error is $\left\|\tilde{Q}_t\left(\hat{\delta}_{y_t} - \delta_{y_t}\right)\right\|$ where $\tilde{Q}_t$ is the kernel embedding of $(X_t|Y_t, y_{1:t-1})$: this error can be small even with small bases if $\tilde{Q}_t$ has good continuity properties.

Assumptions (A7-8) are high-level assumptions and inconveniently involve many-steps-ahead objects whose properties can be hard to study in practice. The following assumptions imply (A7-8) (see Lemma 18 for a proof):

(A7') There is a probability measure $\omega(dx')$, C_ω and $c_\omega > 0$ such that:

$$c_\omega\omega(dx') \leq A(x,dx') \qquad \text{and} \qquad \forall f \in H_x, f \geq 0, \|Ag\|_{H_x} \leq C_\omega\omega(dx')g(x')$$

(A8') For any t, $(X_1, dX_2|y_{1:t})$ is bounded from H_x to $(\mathcal{M}(E), \|\cdot\|_{TV})$.

Theorem 1 obtains strong results from strong assumptions. All smoothness assumptions are required to hold $y_{1:t}$ almost surely. A trivial remark is that the conclusion holds on those paths $y_{1:t}$ that satisfy the assumptions. Heuristically and at a finer level, we can expect kernel filtering to have good properties on average, although this type of results is beyond the current paper.

5 Simulations

This section presents simulation results for a linear Gaussian model. In the linear Gaussian case, we can compute the exact nonlinear filter by Kalman filtering and report the root-mean-squared-error between the filter mean computed via kernel filtering and the true one. The data generating processes are AR(1) of dimension d that were drawn randomly and are fixed across simulations. The measurement equation is simply $y_{t+1} = x_{t+1} + \sigma\mathcal{N}(0, I)$, where $\sigma = 0.1$ or 0.4. The accuracy of the kernel filter is compared to that of a simple bootstrap particle filter with the number of particles n equal to the length of the basis used in kernel filtering. Table 1 reports results for the (full-rank) kernel filter (Algorithm 1), and Table 2 reports results for the low-rank kernel filter (Algorithm 2). We also present indicative timings based on Julia implementations of the particle and kernel filtering algorithms on a laptop.

	$(n,m)=(100,500)$				$(n,m)=(500,1000)$	
(d,σ)	PF	$\tau\sqrt{n}=0.1$	0.01	0.001	PF	0.01
$(1,0.4)$	0.082 (0.06 sec)	0.032 (3.38)	0.015 (3.44)	0.019 (3.48)	0.033 (0.25)	0.009 (141.45)
$(2,0.4)$	0.123 (0.07)	0.152 (19.35)	0.152 (19.4)	0.167 (18.76)	0.053 (0.3)	0.095 (562.81)
$(3,0.4)$	0.17 (0.08)	0.245 (23.14)	0.234 (21.0)	0.232 (22.1)	0.083 (0.28)	0.144 (629.11)

TABLE 1: Full-rank kernel filtering (Algorithm 1). Average RMSE over 20 Monte Carlo draws of length 200 time-series. Average computing time between parentheses. Sobol grids and m simulation draws on each row in the pre-data stage.

	$n=100$		$n=1000$		$n=10000$		
(d,σ)	PF	$r=50$	PF	50	PF	50	200
$(1,0.4)$	0.089 (0.05 sec)	0.383 (0.17)	0.027 (0.46)	0.12 (0.41)	0.015 (4.18)	0.057 (3.5)	0.068 (21.93)
$(1,0.1)$	0.078 (0.05)	0.346 (0.16)	0.021 (0.47)	0.116 (0.43)	0.005 (4.21)	0.083 (3.49)	0.063 (20.86)
$(2,0.4)$	0.122 (0.05)	0.466 (0.18)	0.039 (0.46)	0.21 (0.44)	0.013 (4.24)	0.164 (3.78)	0.124 (21.66)
$(2,0.1)$	0.152 (0.05)	0.509 (0.18)	0.051 (0.46)	0.292 (0.44)	0.017 (4.21)	0.281 (3.7)	0.211 (21.78)
$(3,0.4)$	0.169 (0.06)	0.523 (0.16)	0.059 (0.45)	0.32 (0.45)	0.023 (4.26)	0.291 (3.69)	0.197 (22.76)
$(5,0.4)$	0.428 (0.08)	0.664 (0.18)	0.145 (0.49)	0.545 (0.49)	0.055 (4.7)	0.497 (4.11)	0.341 (24.76)
$(5,0.1)$	(crash)	0.687 (0.18)	(crash)	0.589 (0.5)	0.125 (4.71)	0.554 (4.12)	0.414 (23.97)
$(8,0.4)$	0.577 (0.09)	0.81 (0.18)	0.271 (0.54)	0.754 (0.56)	0.121 (5.44)	0.722 (4.72)	0.538 (26.49)

TABLE 2: Low-rank kernel filtering (Algorithm 2). Average RMSE over 20 Monte Carlo draws of length 200 time-series. Average computing time between parentheses.

As a reminder, particle filtering has the strong advantage of exact Bayes steps via density evaluations. In low dimensions there is a regime where the full-rank kernel filter is more accurate than the particle filter, because the Markov step is more accurate. However the computational disadvantage of full-rank kernel filtering quickly increases with n. Furthermore, the accuracy of full-rank kernel filtering deteriorates quickly with the dimension. Note that the pre-data stage of Algorithm 1 has an embarrassingly parallel structure, allowing for up to 100x speed-ups (not shown here) by computing each row of $\hat{A}$ and $\hat{B}$ in parallel.

Low-rank kernel filtering is roughly 5x less accurate than particle filtering across the board. It also scales much better than Algorithm 1 both in terms of computational time and in terms of dimension, allowing reasonable accuracy in medium dimensions (5-10). Low-rank kernel filtering is not susceptible to online errors the way particle filtering is, as happened here with $(d, \sigma) = (5, 0.1)$ and 100 or 1000 particles.

6 Example: a model with latent supply and demand curves

Consider a stylized model with N_1 persistent buyers who come back to a market at each period $t \in 1{:}T$, together with N_2 fresh noise buyers and N_3 fresh sellers that join the market at each period for one period only. Buyers post bids, sellers post asks and the market clears. A persistent buyer's bid is $b_{it} = \frac{1}{1+e^{-\beta_{it}}}$ with:

$$\beta_{it+1} = \rho\beta_{it} + \underbrace{\sigma_i u_{it+1}}_{\text{idiosyncratic shock}} + \underbrace{\sigma v_{t+1}}_{\text{aggregate shock}} \qquad u_{it+1}, v_{t+1} \sim \mathcal{N}(0,1)$$

The noise buyers' bids b_{jt} are a random sample from a random Beta random variable:

$$c_{1t}, c_{2t} \sim U([0,5]) \qquad b_{jt} \sim Beta(c_{1t}, c_{2t})$$

The sellers' asks a_{it} are a random sample from a fixed $Beta(1,1)$.

From an econometric perspective, the unobserved state x_t is the persistent part of the demand curve b_{it} and the observed variable is simply the clearing price p_t. From observing only the time-series of prices, we may want to know about the current state of the persistent demand curve. The price realization in a single period obviously contains very little information about the unobserved demand curve. However a sequence of price realizations *together with a structural model that disciplines how the demand curve may change from one period to the next* may be much more informative. Suppose for instance that the price is median in period 10 but was high from period 1 to period 9. Then it is likely that the unobserved persistent demand was and is still high but that a temporary low realization of the noise part of the demand drove down the price. This is precisely the kind of question that the (population) nonlinear filter captures, in a quantitative way.

In this example kernel filtering is particularly appealing, because we can very easily simulate and we don't have access to a measurement density. Furthermore, kernel filtering can handle large values of N_1 or even a continuous demand curve – an infinite-dimensional state variable – without difficulty as long as we have a way to simulate from the model.

Kernel filtering is very easy to implement. One-step-ahead simulations are trivial. We only need to provide two things: kernel functions and bases. We can use the Laplace kernel for the observed state $y_t = p_t \in [0,1]$. For the demand curve x_t, we use $k_x(x, x') = e^{-d(x,x')}$, where $d(x, x') = \frac{1}{n}\frac{1}{n}\sum_{ij} e^{-|x_i - x'_j|}$ is the Laplace kernel distance between the distribution of bids x and x'. For B_y we use a uniform grid on $[0, 1]$, and for B_x we draw n_x demand curves $\tilde{x}_{1:n_x}$ that span a wide array of shapes:

$$\tilde{x}_i = N_1 \text{ iid draws from } Beta(c_{3j}, c_{4j}) \qquad c_{3j}, c_{4j} \sim U([0, 5])$$

We consider two DGP calibrations. In DGP1, there are 100 persistent buyers, no noise buyers, 100 sellers, the aggregate shock standard deviation is 0.2 and the idiosyncratic one is 0.01. In DGP2, there are 25 persistent buyers, 25 noise buyers, 50 sellers and both the aggregate and idiosyncratic shock standard deviations are 0.1. In both DGPs $\rho = 0.999$. In DGP1 there is very little uncertainty beyond the aggregate level of persistent demand, which will be revealed with high precision by realized prices. DGP2 has more noise. We run Algorithm 1 and report the filter quantiles for the mean of the persistent demand curve in Figures 1 and 2. For $n_x = 81$, $n_y = 40$ and $T = 1000$, the pre-data stage takes around 20 seconds and the post-data stage (actual filtering) under one second.

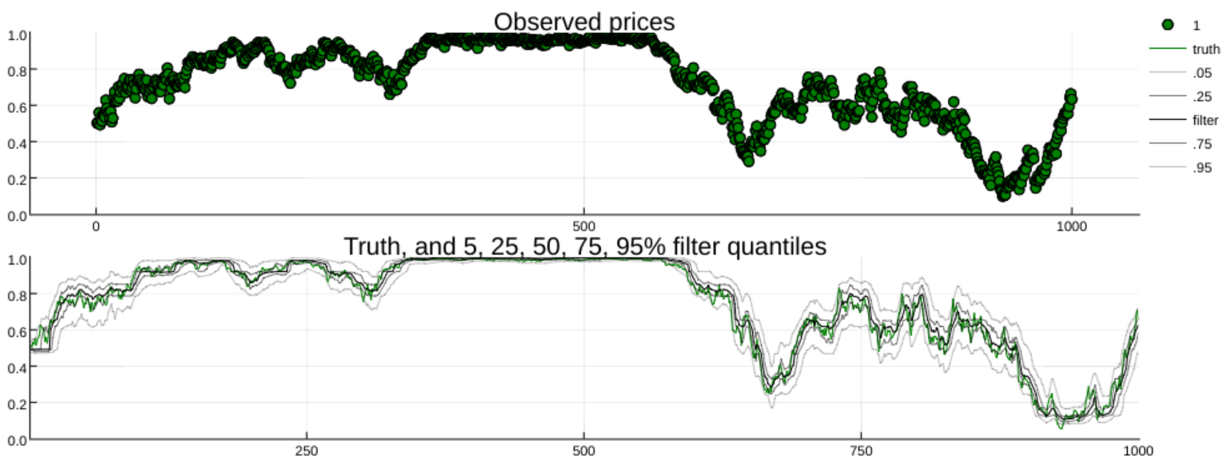

Figure 1: DGP1.

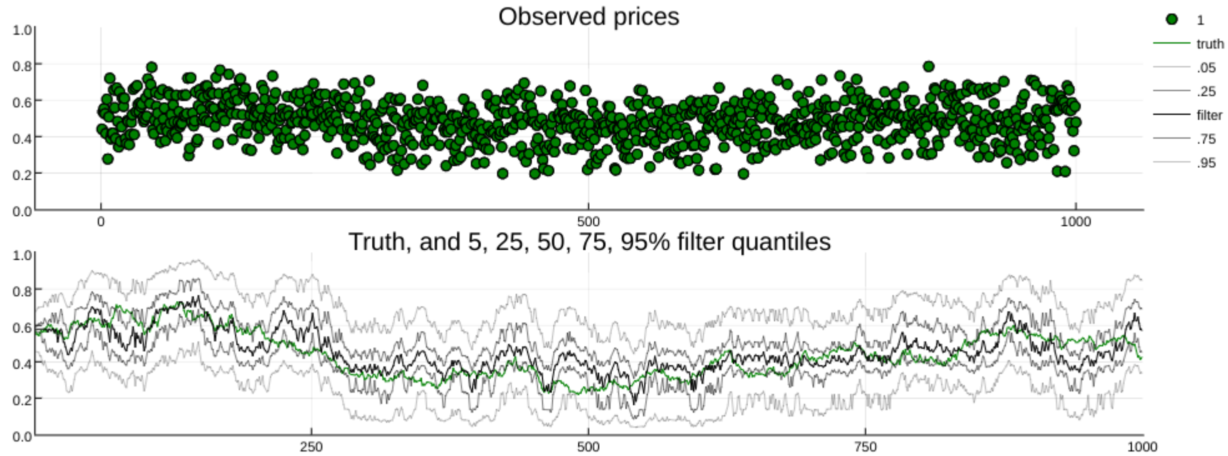

Figure 2: DGP2.

As expected, the observed prices reflect very closely the underlying state of the demand curve in DGP1. There is very little uncertainty as captured by the nonlinear filter having narrow interquantile width. In DGP2 it is hard to read anything from the time-series of prices which is very noisy, yet the filter median (black line) is able to capture movement in the true mean bid (green line). The uncertainty about the location of the unobserved mean bid is reflected by the quantile width. For both DGP1 and DGP2, the true mean bid falls in any given $z\%$ interquantile band roughly $z\%$ of the time, as it is supposed to do.

7 Proofs

7.1 Proofs for section 4.1

Proof of Lemma 2. Assume $1 \in H_y$. We define Ψ_M explicitly via its adjoint: $\Psi'_M f = f \otimes 1$. Fix μ, take any $f \in H_x$:

$$\langle \Psi_M \mu, f \rangle = \langle \mu, \Psi'_M f \rangle = \langle \mu, f \otimes 1 \rangle = \mu(dx, dy) f(x) = \mu_x(dx) f(x)$$

This proves $\Psi_M \mu = \mu_x$. Reciprocally, assume marginalization is a bounded linear operation Ψ_M. Let $f \in H$ and $x_0 \in E$ such that $f(x_0) = 1$. Define $e = (\Psi'_M f) k_{x_0} \in H_y$. We show that $e = 1$. For any $y \in E_y$:

$$\langle e, k_y \rangle = \langle (\Psi'_M f) k_{x_0}, k_y \rangle = \langle \Psi'_M f, k_{x_0} \otimes k_y \rangle = \langle f, \Psi_M (k_{x_0} \otimes k_y) \rangle = \langle f, k_{x_0} \rangle = 1$$

Then we can check $\Psi'_M f$ must be $f \otimes 1$ and $\|\Psi_M\| = \|\Psi'_M\| = \|1\|_{H_y}$ is clear. □

Proof of Lemma 3. $\mu(dx) f(x) g(x) \le \|\mu\|_{TV} \|f\|_\infty \|g\|_\infty \le \|\mu\|_{TV} \gamma_k^2 \|f\|_k \|g\|_k$ shows that $\langle D_\mu f, g \rangle = \mu(dx) f(x) g(x)$ does indeed define a bounded linear operator D_μ. Self-adjointness and positivity are clear. We explicitly compute the trace of D_μ. Fix u_i any orthonormal basis in H. By Perceval formula:

$$k(x,x) = \langle k_x, k_x \rangle = \sum_i \langle u_i, k_x \rangle^2 = \sum_i u_i(x)^2$$

Define $d_n(x) = \sum_{i=1}^n u_i(x)^2$. d_n converges pointwise from below to $d(x) := k(x,x)$. By Lebesgue dominated convergence:

$$\sum_{i=1}^n \langle D_\mu u_i, u_i \rangle = \mu(dx) d_n(x) \to \mu(dx) d(x)$$

□

Proof of Lemma 4. Assume multiplication is bounded. We define explicitly $\Psi_D : f \to D_f$ as follows:

$$\langle D_f g, g' \rangle := \langle f, M(g, g') \rangle \leq \|f\| \, \|M\| \, \|g\| \, \|g'\|$$

The inequality shows that this is a valid definition of $\Psi_D \in B(H, B(H))$. Assume $\Psi_D \in B(H, B(H))$. $\Psi_D(H)$ must in fact be contained in $K(H)$ because it sends a dense subset of H to $B_1(H)$. Use the same notation for $\Psi_D \in B(H, K(H))$ from now on. Its dual is $\Psi'_D \in B(B_1(H), H)$. Fix $f, f' \in H$. $f \otimes f'$ belongs to $B_1(H)$ as a rank one operator. We show that $\Psi'_D f \otimes f'$ is the product ff':

$$\langle \Psi'_D f \otimes f', k_x \rangle = \langle f \otimes f', \Psi_D k_x \rangle = \langle D_{k_x} f, f' \rangle = f(x) f'(x)$$

Define $M(f, f') := \Psi'_D f \otimes f'$. Let any $g \in h$.

$$\langle M(f, f'), g \rangle = \langle D_g f, f' \rangle \leq \|\Psi_D\| \, \|g\| \, \|f\| \, \|f'\|$$

□

We need four preparatory lemmas before proving Lemma 5. First, we quote the two following useful lemmas:

Lemma 10: (Corollary 33 in Simon-Gabriel and Schölkopf (2016))
Let $k(x, x')$ a kernel on $\mathbb{R}^n$ such that $k(x, x') = \phi(x - x')$ for a continuous bounded function ϕ. Then k is characteristic if, and only if, the Fourier transform of ϕ has full support.

Lemma 11: (Theorem 55 in Simon-Gabriel and Schölkopf (2016))
Let E be a locally compact Hausdorff metric space and k a bounded continuous kernel on E. k is characteristic if, and only if, k is $\mathcal{P}(E)$-nondegenerate.

Then we give a useful criterion for proving that a kernel embedding space has "strong products", ie. that multiplication is not only bilinear bounded, but in fact Hilbert-Schmidt.

Lemma 12: Criterion for strong products
Let H_k a kernel embedding space. The following are equivalent:

(i) Multiplication is a bounded linear operator from $H \otimes H$ to H.

(ii) H_{k^2} is included in H_k in the set-theoretic sense.

Proof. Assume (i) and consider $\Delta = M'$. $\langle \Delta k_x, f \otimes f' \rangle = f(x) f(x')$ so that $\Delta k_x = k_x \otimes k_x$. By continuity of Δ :

$$\begin{aligned}
\left\| \Delta \sum_i a_i k_{x_i} \right\|^2 &= \sum_{ij} a_i a_j \langle k_{x_i} \otimes k_{x_i}, k_{x_j} \otimes k_{x_j} \rangle \\
&= \sum_{ij} a_i a_j k(x_i, x_j)^2 \\
\text{must be } &\leq \|\Delta\|^2 \left\| \sum_i a_i k_{x_i} \right\|^2 \\
&\leq \|\Delta\|^2 \sum_{ij} a_i a_j k(x_i, x_j)
\end{aligned}$$

In particular $\|\Delta\|^2 k - k^2$ is a kernel function, which implies $H_{k^2} \subset H_k$ by Theorem 6 p.37 of Saitoh (1997). Each step of the argument was in fact an equivalence. □

Finally we show that the flaw of not having constants can be fixed for free. It is an open question whether not having products can similarly be fixed for free.

Lemma 13: Adding constants
Let H_k a kernel embedding space without constant. Consider the kernel $k' = k + 1$. Then:

(i) $H_{k'}$ has constants.

(ii) $H_{k'}$ induces the same topology on $\mathcal{P}(E)$ as H_k.

(iii) If multiplication is bounded on H_k, so is it on $H_{k'}$.

Proof. If H is a RKHS on E with kernel k and without constant, the standard Hilbert sum $H' = H \oplus 1$ is a RKHS on E with kernel $k' = k + 1$. Any $g \in H'$ can be uniquely written $g(x) = f(x) + \lambda$ for some $f \in H$, and $\|g\|^2_{H'} = \|f\|^2_H + \lambda^2$, see Saitoh (1997). This proves (1). Now for (ii) take two probability measures μ and μ':

$$\|\mu - \mu'\|^2_{k'} = (\mu - \mu')(dx)(\mu - \mu')(dx')k(x, x') + (\mu - \mu')(dx)(\mu - \mu')(dx')1(x, x') = \|\mu - \mu'\|^2_k$$

Now we prove (iii). Suppose H has bounded multiplication. WLOG assume $\lambda, \lambda' > 0$ in the following display:

$$\begin{aligned}
\|(f + \lambda)(f' + \lambda')\|^2_{H'} &= \|(ff' + \lambda f' + \lambda' f) + \lambda\lambda'\|^2_{H'} \\
&= \|(ff' + \lambda f' + \lambda' f)\|^2_H + (\lambda\lambda')^2 \\
&\leq (\|ff'\|_H + \lambda \|f'\|_H + \lambda' \|f\|_H + \lambda\lambda')^2 \\
&\leq \gamma_M^2 (\|f\|_H + \lambda)^2 (\|f'\|_H + \lambda')^2 \qquad \text{with } \gamma_M = \max(1, \|M\|_H) \\
&\leq 4\gamma_M^2 \left(\|f\|^2_H + \lambda^2\right)\left(\|f'\|^2_H + \lambda'^2\right) \\
&= 4\gamma_M^2 \|f + \lambda\|^2_{H'} \|f' + \lambda'\|^2_{H'}
\end{aligned}$$

□

We are now ready to prove Lemma 5. Call $k_1(x,x') = e^{-|x-x'|}$, $k_2(x,x') = 0.9e^{-|x-x'|} + 0.1$, $k_3(x,x') = e^{-(x-x')^2}$ and H_1, H_2 , H_3 their respective reproducing kernel Hilbert spaces.

Proof of Lemma 5. Lemma 10 can be used to show that k_1, k_2 and k_3 are characteristic on $\mathbb{R}$. On $E = [0,1]$, H_1 is known to be the Sobolev space $W_{21}(0,1)$ (Saitoh (1997)), which has constants. The RKHS for $k_1^2(x,x') = e^{-2|x-x'|}$ is also $W_{21}(0,1)$, so that by Lemma 12, H_1 has (strong) products. This proves (i). On $\mathbb{R}$ H_1 is also $W_{21}(\mathbb{R})$ and so has no constant, which explains why we focus on k_2 instead, using the technique of Lemma 13. k_1 has products because the RKHSs for k_1 and k_1^2 are both $W_{21}(\mathbb{R})$. Thus k_2 has products as well. This is enough to prove (ii) using Lemma 13. Finally H_3 has not constant (Theorem 2 in Minh (2010)) and $f(x) = e^{-1.5x^2} \in H_3$ but $f^2(x) = e^{-3x^2} \notin H_3$ (Theorem 3 in Minh (2010)). This proves (iii). □

Proof of Lemma 6, (i) and (ii). Let λ, $\lambda' \in \mathcal{P}(E)$ and $\|\lambda - \lambda'\|_k = 0$. The marginals on E_x are equal as measures because $\|\lambda_x - \lambda'_x\|_{H_x} = 0$ by kernel marginalization and k_x is $\mathcal{P}$-nondegenerate. Call $\lambda_1 = \lambda_x = \lambda'_x$. Consider the disintegrations $\lambda = \lambda_1(dx)q(x,dy)$ and $\lambda' = \lambda_1(dx)q'(x,dy)$. Pick u_i, v_j orthonormal bases for H_x and H_y. $w_{ij} = u_i \otimes v_j$ is an orthonormal basis for $H_x \otimes H_y$. $\|\lambda - \lambda'\|_k = 0$ implies that their coordinates in w_{ij} are equal: for any i, j:

$$\langle \lambda, w_{ij} \rangle = \lambda(dx,dy)w_{ij}(x,y) = \lambda_1(dx)u_i(x)q(x,dy)v_j(y) = \lambda_1(dx)u_i(x)q'(x,dy)v_j(y)$$

Fix j. Define $f_j(x) = q(x,dy)v_j(y)$, $\nu_j(dx) = f_j(x)\lambda_1(dx)$ and $f'_j(x)$ and $\nu'_j(dx)$ similarly. We have:

$$\text{for any } i, \qquad \nu_j(dx)u_i(x) = \nu'_j(dx)u_i(x)$$

Because k_x is nondegenerate, this implies $\nu_j(dx) = \nu'_j(dx)$ which implies $f_j(x) = f'_j(x)$, $\lambda_1(dx)$ almost surely. This is true for any j, so that $q(x,dy) = q'(x,dy)$ ($\lambda_1(dx)$ almost surely) because k_y is nondegenerate. Finally:

$$\lambda_1(dx)q(x,dy) = \lambda_1(dx)q'(x,dy) \qquad \text{ie} \qquad \lambda = \lambda'$$

Thus k is nondegenerate. H also clearly has constants. This proves (i). (ii) is a corollary of (i) and Lemma 11. □

Proof of Lemma 6, (iii). Take $M^x \in B(H_x \otimes H_x, H_x)$. Fix $x_0 \in E_x$. $M^{x\prime}\delta_{x_0} = \delta_{x_0} \otimes \delta_{x_0}$ because:

$$\langle M^{x\prime}\delta_{x_0}, f \otimes f' \rangle = \langle \delta_{x_0}, M(f,f') \rangle = f(x_0)f'(x_0)$$

Similarly for M^y. Now define:

$$A = M^x \otimes M^y \in B((H_x \otimes H_x) \otimes (H_y \otimes H_y), H_x \otimes H_y)$$

the standard tensor product of operators, and:

$$T : f \otimes f \otimes g \otimes g' \to f \otimes g \otimes f' \otimes g' \in B(H_x \otimes H_x \otimes H_y \otimes H_y, H_x \otimes H_y \otimes H_x \otimes H_y)$$

Finally define:

$$M = AT'$$

We show that $M(h, h') = hh'$ for $h, h' \in (H_x \otimes H_y)$, ie. that M continuously extends multiplication as required:

$$\begin{aligned}\langle Mh \otimes h', k_x \otimes k_y\rangle &= \langle Th \otimes h', (M^x \otimes M^y)'(k_x \otimes k_y)\rangle \\ &= \langle Th \otimes h', (k_x \otimes k_x) \otimes (k_y \otimes k_y)\rangle \\ &= \langle h \otimes h', (k_x \otimes k_y) \otimes (k_x \otimes k_y)\rangle \\ &= h(x, y)h'(x, y)\end{aligned}$$

□

Proof of Lemma 7. (i) follows from Lemma 13. (ii) follows from Theorem 3.3 in Meckes (2013) because under the $\mathcal{P}$-nondegeneracy assumption on k, the metric space $(\mathcal{P}(E), \|\cdot\|_H)$ is of negative type, as a subset of a Hilbert space. Note that if E is compact, then $\mathcal{P}(E)$ is compact and $\mathcal{P}$-nondegeneracy would automatically "lift" to the kernel being characteristic by Lemma 11. The open question is under which assumption fa-nondegeneracy "lifts" to $\mathcal{P}$-nondegeneracy. □

Proof of Lemma 8. For any $f \in H_x$, $g \in H_y$:

$$\langle J, f \otimes g\rangle = \langle Jf, g\rangle = \langle D_{\mu_x} Qg, f\rangle = \mu_x(dx)f(x)Q(x, dy)g(y) = \mu(dx, dy)(f \otimes g)(x, y)$$

□

Proof of Lemma 9. Write R_0 for the Moore-Penrose pseudo-inverse of D, so that, because D is injective, $Q = R_0 J$. For $\tau > 0$ define $R_\tau := (D + \tau I)^{-1}$and $\hat{R}_\tau := \left(\hat{D} + \tau I\right)^{-1}$. Decompose the error as:

$$\left\|\hat{Q} - Q\right\|_2 \leq \left\|\hat{R}_\tau \hat{J} - \hat{R}_\tau J\right\|_2 + \left\|\hat{R}_\tau J - R_\tau J\right\|_2 + \|R_\tau J - R_0 J\|_2$$

Recall that for a positive self-adjoint operator T and any $\alpha > 0$, $0 \leq s \leq 1$, $\|\alpha(T + \alpha I)^{-1} T^s\| \leq \alpha^s$. The first variance term is bounded as:

$$\left\|\hat{R}_\tau \hat{J} - \hat{R}_\tau J\right\|_2 \leq \left\|\hat{R}_\tau\right\| \left\|\hat{J} - J\right\|_2 \leq \frac{1}{\tau}\delta$$

Recall $A^{-1} - B^{-1} = A^{-1}(B - A)B^{-1}$. By Lemma 2 and Lemma 4:

$$\left\|\hat{D} - D\right\| = \left\|\Psi_D \Psi_M \left(\hat{J} - J\right)\right\| \leq \|M^x\| \, \|1\|_{H_y} \delta$$

The source condition (1) implies $\|Q\|_2 \leq \|D\|^s \|W\|_2 \leq c_W$. The second variance term is bounded as:

$$\begin{aligned}\left\|\hat{R}_\tau J - R_\tau J\right\|_2 &\leq \left\|\hat{R}_\tau D - R_\tau D\right\| \|Q\|_2 \\ &\leq \left\|\left(\hat{D} + \tau I\right)^{-1}\right\| \left\|\hat{D} - D\right\| \left\|(D + \tau I)^{-1} D\right\| \|Q\|_2 \\ &\leq \frac{1}{\tau} \cdot \left(\|M^x\| \, \|1\|_{H_y} \delta\right) \cdot 1 \cdot c_W\end{aligned}$$

Recall $I - (A + \tau I)^{-1} A = \tau (A + \tau I)^{-1}$. The bias term is bounded as:

$$\begin{aligned}
\|R_\tau J - R_0 J\|_2 &= \left\|(D + \tau I)^{-1} D Q - Q\right\|_2 \\
&\leq \left\|\tau (D + \tau I)^{-1} D^s\right\| \|W\|_2 \qquad \text{using (1) } Q = D^s W \\
&\leq \tau^s c_W
\end{aligned}$$

This concludes the proof. τ was chosen to balance the variance and bias terms. The constant has not been optimized. □

7.2 Proof of Theorem 1

Lemma 14: Approximation errors in Bayes step
Let $\mu_x(dx) = \sum_{i=1}^n a_i \nu_i(dx)$ be a mixture of probabilities, $B(x, dy)$ a Markov kernel. Write $\mu(dx, dy) = \mu_x(dx) B(x, dy)$ for the corresponding joint and $\mu(dx, dy) = \mu_y(dy) Q(y, dx)$ for its disintegration. Let $\pi_y(dy)$ another probability distribution, $\pi(dx, dy) = \pi_y(dy) Q(y, dx)$ the joint and $\pi_x(dx)$ the marginal of π. Suppose we compute an approximation $\hat{\pi}_x$ of π_x as follows:

1. *Compute $\hat{J} = \sum_{i=1}^n a_i \nu_i(dx) \otimes \hat{b}_i$, for $\hat{b}_i(dy)$ probability distributions. Define $\hat{\mu}_y(dy)$ the marginal of $\hat{J}$ and $\hat{D}$ the covariance operator of $\hat{\mu}_y$.*
2. *Compute $\hat{Q} = \hat{J} \left(\hat{D} + \tau I\right)^{-1}$, $\tau > 0$ to be defined below.*
3. *Compute $\hat{\pi}_x = \hat{Q} \hat{\pi}_y$ for some probability measure $\hat{\pi}_y$.*

Assume:

(i) H_x and H_y are embedding spaces such that H_x has constants and H_y has a bounded multiplication operator M^y.

(ii) $\max_i \|\nu_i(dx) B(x, dy) - \nu_i(dx) \otimes b_i(dy)\|_{H_x \otimes H_y} \leq \epsilon_1$

(iii) $Q(y, dx)$ has a kernel embedding Q that satisfies the following smoothness condition (write D for the covariance operator of μ_y): there is $W \in B_2(H_x, H_y)$, $c_W > 0$ and $0 < s \leq 1$ such that:

$$Q = D^s W \qquad \text{and} \qquad \|W\|_2 \leq c_W \tag{2}$$

(iv) $\|\hat{\pi}_y - \pi_y\| \leq \epsilon_2$ and D is injective.

Choose $\tau = \epsilon_1^{\frac{1}{1+s}}$. Then:

$$\|\hat{\pi}_x - \pi_x\| \leq (1 + \|M^y\| \|1\|_{H_x} c_W + c_W) \epsilon_1^{\frac{s}{1+s}} + c_W \epsilon_2$$

Proof. First by triangle inequality:

$$\left\|\hat{J} - J\right\|_{H_1 \otimes H_2} \leq \epsilon_1$$

We can then apply the disintegration lemma, Lemma 9:

$$\left\|\hat{Q} - Q\right\|_2 = (1 + \|M^x\| \, \|1\|_{H_y} c_W + c_W)\epsilon_1^{\frac{s}{1+s}}$$

Finally:

$$\begin{aligned}
\|\hat{\pi}_x - \pi_x\| &= \left\|\hat{Q}\hat{\pi}_y - Q\pi_y\right\| \\
&\leq \left\|\left(\hat{Q} - Q\right)\hat{\pi}_y\right\| + \|Q\left(\hat{\pi_y} - \pi_y\right)\| \\
&\leq (1 + \|M^x\| \, \|1\|_{H_y} c_W + c_W)\epsilon_1^{\frac{s}{1+s}} + c_W\epsilon_2
\end{aligned}$$

□

Lemma 15: Approximation errors in Markov step
Let $\mu(dx) = \sum_{i=1}^n a_i \nu_i(dx)$ be a mixture of probabilities, $Q(x, dy)$ a Markov kernel and:

$$\pi(dy) = \mu Q = \sum_{i=1}^n a_i q_i(dy) \qquad \textit{where: } q_i = \nu_i Q$$

Suppose we compute an approximation $\hat{\pi}$ of π by:

$$\hat{\pi}(dy) = \sum_{i=1}^n a_i \hat{q}_i(dy)$$

If $\|q_i - \hat{q}_i\| \leq \epsilon$, then:

$$\|\pi - \hat{\pi}\| \leq \epsilon$$

Proof. Triangle inequality. □

Lemma 16: Propagation of errors through a Bayes step
Let $R \in B(H)$ and ϕ a measurable function on E, bounded below by $\phi_l > 0$ and such that there is $c > 0$ such that for any $\mu, \mu' \in \mathcal{P}(E)$, $|\mu\phi - \mu'\phi| \leq c \|\mu - \mu'\|_H$. Then for any $\mu, \mu' \in \mathcal{P}(E)$:

$$\left\|\frac{\mu R}{\mu\phi} - \frac{\mu' R}{\mu'\phi}\right\|_H \leq \left(\|R\| \frac{c}{\phi_l^2} + \|R\| \frac{1}{\phi_l}\right) \|\mu - \mu'\|_H$$

In particular, if H has constants and products, if $R = M_f$ and $\phi = M_f 1 = f \in H$, then automatically $|\mu M_f 1 - \mu' M_f 1| \leq \|f\|_H \|\mu - \mu'\|_H$ and:

$$\left\|\frac{\mu R}{\mu\phi} - \frac{\mu' R}{\mu'\phi}\right\|_H \leq \|M\| \left(\frac{\|f\|_H^2}{f_l^2} + \frac{\|f\|_H}{f_l}\right) \|\mu - \mu'\|_H$$

Proof.

$$\begin{aligned}
\left\| \frac{\mu R}{\mu\phi} - \frac{\mu' R}{\mu'\phi} \right\|_H &= \sup_{\|g\|_H \le 1} \left\| \frac{\mu R g}{\mu\phi} - \frac{\mu' R g}{\mu'\phi} \right\|_H \\
&\le \sup_{\|g\|_H \le 1} \left\| \frac{\mu R g (\mu'\phi - \mu\phi)}{\mu\phi\mu'\phi} \right\|_H + \left\| \frac{\mu R g - \mu' R g}{\mu'\phi} \right\|_H \\
&\le \left(\|R\| \frac{c}{\phi_l^2} + \|R\| \frac{1}{\phi_l} \right) \|\mu - \mu'\|_H
\end{aligned}$$

□

Lemma 17: Geometric contractivity of the nonlinear filter
Under (A7) and (A8) of Theorem 1, there is $\kappa > 0$ such that:

$$\|\Phi_{s+1:t}(\mu) - \Phi_{s+1:t}(\mu')\|_H \le \kappa \rho^{t-s} \|\mu - \mu'\|_H$$

Proof. We can obtain μ_t from μ_s by applying one Bayes rule with respect to the conditional density $p(y_{s+1:t}|x_s)$, followed by $t-s$ Markov transitions $Q(x_r, dx_{r+1}|y_{r:t})$, i.e. $\Phi_{s+1:t}$ can be written $\Phi_{s+1:t}(\mu) = T_{s+1:t}(\mu) S_{s+1} \dots S_t$ where $T_{s+1:t}$ is the Bayes kernel with respect to $p(y_{s+1:t}|x_s)$ and S_r is the Markov kernel $Q(x_r, dx_{r+1}|y_{r:t})$. This is a classical proof technique for the total variation norm, see eg. Künsch (2005). The Bayes step error is controlled by Lemma 16 using (A7):

$$\|T_{s+1:t}(\mu) - T_{s+1:t}(\mu')\| \le \|M^x\| \left(c_p^2 + c_p\right) \|\mu - \mu'\|$$

A direct consequence of (A8) concludes the proof:

$$\|\Phi_{s+1:t}(\mu) - \Phi_{s+1:t}(\mu')\|_H \le c_Q \rho^{t-s} \|M^x\| \left(c_p^2 + c_p\right) \|\mu - \mu'\|_H$$

□

Lemma 18:
Assumptions (A7') and (A8') together imply assumptions (A7) and (A8).

Proof. Condition (A7) is adapted from the classical condition $c_\omega \omega(dx') \le A(x, dx') \le C_\omega \omega(dx')$ which has been used in total variation contraction studies at least since Atar and Zeitouni (1997). First we prove (A7). Note:

$$p(y_{s+1:t}|x_s) = A(x_s, dx_{s+1}) b(y_{s+1}|x_{s+1}) p(y_{s+2:t}|x_{s+1})$$

By induction $p(y_{s+2:t}|x_{s+1}) \in H$ and by (A7'):

$$\|A(x_s, dx_{s+1}) b(y_{s+1}|x_{s+1}) p(y_{s+2:t}|x_{s+1})\|_H \le C_\omega \omega(dx_{s+1}) b(y_{s+1}|x_{s+1}) p(y_{s+2:t}|x_{s+1})$$

and:

$$Q(x_s, dx_{s+1}) b(y_{s+1}|x_{s+1}) p(y_{s+2:t}|x_{s+1}) \ge c_\omega \omega(dx_{s+1}) b(y_{s+1}|x_{s+1}) p(y_{s+2:t}|x_{s+1})$$

We get (A7) as a consequence:

$$\|p(y_{s+1:t}|x_s)\|_H \leq \frac{C_\omega}{c_\omega} \inf_{x_s} p(y_{s+1:t}|x_s)$$

We now turn to (A8). (A7') implies that $A(x, dx') \leq \gamma_k C_\omega \omega(dx')$. Then:

$$\begin{aligned}
Q(x_1, dx_2|y_{1:t}) &= \frac{q(y_{1:t}|x_1, x_2)A(x_1, dx_2)}{A(x_1, dx_2)q(y_{1:t}|x_1, x_2)} \\
&= \frac{q(y_1|x_1, x_2)q(y_{2:t}|x_2, x_1)A(x_1, dx_2)}{A(x_1, dx_2)q(y_1|x_1, x_2)q(y_{2:t}|x_2, x_1)} \\
&= \frac{q(y_1|x_1)q(y_{2:t}|x_2)A(x_1, dx_2)}{A(x_1, dx_2)q(y_1|x_1)q(y_{2:t}|x_2)} \\
&\geq \frac{c_\omega q(y_{2:t}|x_2)\omega(dx_2)}{\gamma_k C_\omega \omega(dx_2)q(y_{2:t}|x_2)}
\end{aligned}$$

Hence $Q(x_s, dx_{s+1}|y_{1:t})$, $s \leq t-1$, satisfies a Doeblin criterion and is contractive for the total variation norm with $\rho = 1 - \frac{c_\omega}{\gamma_k C_\omega}$. (A8') guarantees that $Q(x_1, dx_2|y_{1:t})$ is bounded from $\|\cdot\|_H$ to $\|\cdot\|_{TV}$ and $Q(x_{t-1}, dx_t|y_{1:t})$ has a kernel embedding, ie. is bounded from $\|\cdot\|_H$ to $\|\cdot\|_H$, and a fortiori from $\|\cdot\|_{TV}$ to $\|\cdot\|_H$ as well. This concludes the proof. □

8 Appendix: low-rank approximation formulas

The formulas underpinning the low-rank kernel filter (Algorithm 2) follow from Nyström approximations applied to kernel disintegrations and to changes of bases. See Gittens and Mahoney (2016).

First we look at kernel disintegrations. Start from a joint discrete probability measure $\tilde{J} = \sum_{i=1}^{n_x} \sum_{j=1}^{n_y} w_{ij} \delta_{x_i} \otimes \delta_{y_j}$. Recall that kernel disintegration is defined as $\tilde{Q} = \left(\tilde{D} + \tau I\right)^{-1} \tilde{J}$. In order to allow for cases where $\tilde{D}$ is not positive, which can arise in low-rank kernel-filtering, we consider here the alternative regularization strategy: $\tilde{Q} = \left(\tilde{D}^2 + \tau I\right)^{-1} \tilde{D}\tilde{J}$. These are coordinate-free expressions. The corresponding in-basis matrix expressions for Q is (see next paragraph for a derivation):

$$Q = (DG_{11}D + \tau I)^{-1} DG_{11}J \tag{3}$$

where D is the diagonal matrix with diagonal $d_{ii} = \sum_j w_{ij}$ (the marginal of J on x), and G_{11} is the (x, x) Gram matrix, i.e. $G_{11,ij} = k(x_i, x_j)$. The Nyström approximation uses a sample $\tilde{x}_{1:r}$, typically a subsample of $x_{1:n_x}$. Call G_{00}, G_{01} and G_{10} the $(\tilde{x}, \tilde{x})$, $(\tilde{x}, x)$ and $(x, \tilde{x})$ Gram matrices respectively. Plugging-in the Nyström approximation $G_{11} \approx G_{10}G_{00}^{-1}G_{01}$ in (3) and driving τ to 0 (which is justified whenever τ is small with respect to the lowest singular value of $DG_{10}G_{00}^{-1}G_{01}D$, or equivalently whenever n_x is large enough compared to r), we get the expression:

$$Q \approx DG_{10}(G_{01}D^2G_{10})^{-1}G_{01}J$$

To see why (3) hold, call $V_x = span\{\delta_{x_i}, i \in 1{:}n\}$. Pick a U such that $U'G_{11}U = I$. U is the coordinate matrix of an orthonormal basis $u_{1:n}$ of V_x, in the sense that $U_{ij} = u_j(x_i)$. There are at least three bases of interest for V_x: $B^\delta_x = \{\delta_{x_i}, i \in 1{:}n\}$, $B^u_x = u_{1:n}$, and $B^e_x = e_{1:n}$, where $e_i \in V_x$ is defined by $e_i(x_j) = 1[i = j]$. A Markov transition kernel $\tilde{Q}$ is more naturally expressed by a matrix Q of coordinates in (B^e_x, B^e_y) – i.e. if $\tilde{g}$ has coordinates g in B^e_y, then $\tilde{Q}\tilde{g}$ has coordinates Qg in B^e_x– where it looks like a discrete Markov transition matrix. A joint probability measure $\tilde{J}$ is more naturally expressed by a matrix J of coordinates in (B^δ_x, B^e_y), where it looks like a discrete probability matrix. With theses definitions in place, the matrix expression of $\tilde{Q} = \left(\tilde{D}^2 + \tau I\right)^{-1} \tilde{D}\tilde{J}$ in (B^e_x, B^e_y) is:

$$Q = U(U'DG_{11}U^{-1\prime}U^{-1}G_{11}DU + \tau I)^{-1}U'DG_{11}U^{-1\prime}U^{-1}G_{11}J = (DG_{11}D + \tau I)^{-1}DG_{11}J$$

Second, we turn to changes of bases. Consider three samples $x^{(0)}_{1:r}$, $x^{(1)}_{1:n_1}$ and $x^{(2)}_{1:n_2}$ with notation as above. The orthogonal projection $\tilde{P}$ from V_2 to V_1 has expression $P = G_{11}^{-1}G_{12}$ in (B^δ_1, B^δ_2). Using Nyström approximations we get:

$$P \approx G_{10}(G_{01}G_{10})^{-1}G_{02}$$

References

Atar, R. and O. Zeitouni (1997): “Exponential stability for nonlinear filtering,” *Annales de l’Institut Henri Poincare (B) Probability and Statistics*, 33, 697 – 725.

Berlinet, A. and C. Thomas-Agnan (2003): *Reproducing kernel Hilbert spaces in probability and statistics*, Springer.

Farmer, L. (2017): “The Discretization Filter: A Simple Way to Estimate Nonlinear State Space Models,” *Working Paper*, https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2780166.

Fukumizu, K., L. Song, and A. Gretton (2013): “Kernel Bayes’ Rule: Bayesian Inference with Positive Definite Kernels,” *Journal of Machine Learning Research*, 14, 3753–3783.

Gittens, A. and M. Mahoney (2016): “Revisiting the Nyström method for improved large-scale machine learning,” *The Journal of Machine Learning Research*, 17, 3977–4041.

Guilbart, C. (1979): “Produits scalaires sur l’espace des mesures,” *Ann. Inst. H. Poincaré Sect. B (N.S.)*, 15, 333–354 (1980).

Kantas, N., A. Doucet, S. S. Singh, J. Maciejowski, and N. Chopin (2015): “On Particle Methods for Parameter Estimation in State-Space Models,” *Statist. Sci.*, 30, 328–351.

Künsch, H. R. (2005): “Recursive Monte Carlo filters: Algorithms and theoretical analysis,” *Ann. Statist.*, 33, 1983–2021.

Meckes, M. W. (2013): “Positive definite metric spaces,” *Positivity*, 17, 733–757.

Minh, H. Q. (2010): “Some Properties of Gaussian Reproducing Kernel Hilbert Spaces and Their Implications for Function Approximation and Learning Theory,” *Constructive Approximation*, 32, 307–338.

Moral, P. D. and A. Guionnet (2001): “On the stability of interacting processes with applications to filtering and genetic algorithms,” *Annales de l’Institut Henri Poincare (B) Probability and Statistics*, 37, 155 – 194.

Pitt, M. K. and N. Shephard (1999): “Filtering via simulation: Auxiliary particle filters,” *Journal of the American statistical association*, 94, 590–599.

Saitoh, S. (1997): *Integral transforms, reproducing kernels and their applications*, vol. 369, CRC Press.

Simon-Gabriel, C.-J. and B. Schölkopf (2016): “Kernel Distribution Embeddings: Universal Kernels, Characteristic Kernels and Kernel Metrics on Distributions,” .

Song, L., K. Fukumizu, and A. Gretton (2013): "Kernel Embeddings of Conditional Distributions: A Unified Kernel Framework for Nonparametric Inference in Graphical Models," *IEEE Signal Processing Magazine*, 30, 98–111.

Steinwart, I. and A. Christmann (2008): *Support vector machines*, Springer.