Kernel Embeddings for Inference with Intractable Likelihoods

Dino Sejdinovic

Department of Statistics
University of Oxford

The Institute of Statistical Mathematics, Tokyo, 30/03/2016

Intractable Likelihood

Interested in Bayesian posterior inference:

$$\mathsf{p}(\theta|\mathcal{D}) = \frac{\mathsf{p}(\theta)\mathsf{p}(\mathcal{D}|\theta)}{\mathsf{p}(\mathcal{D})}$$

Intractable Likelihood

Interested in Bayesian posterior inference:

$$\mathsf{p}(\theta|\mathcal{D}) = \frac{\mathsf{p}(\theta)\mathsf{p}(\mathcal{D}|\theta)}{\mathsf{p}(\mathcal{D})}$$

- The case of intractable $\mathsf{p}(\mathcal{D}) = \int \mathsf{p}(\theta)\mathsf{p}(\mathcal{D}|\theta)d\theta$: while posterior density is intractable, can use MCMC to generate an (asymptotically exact) sample from $\mathsf{p}(\theta|\mathcal{D})$

Intractable Likelihood

Interested in Bayesian posterior inference:

$$\mathsf{p}(\theta|\mathcal{D}) = \frac{\mathsf{p}(\theta)\mathsf{p}(\mathcal{D}|\theta)}{\mathsf{p}(\mathcal{D})}$$

- The case of intractable $\mathsf{p}(\mathcal{D}) = \int \mathsf{p}(\theta)\mathsf{p}(\mathcal{D}|\theta)d\theta$: while posterior density is intractable, can use MCMC to generate an (asymptotically exact) sample from $\mathsf{p}(\theta|\mathcal{D})$
- The case of intractable $\mathsf{p}(\mathcal{D}|\theta)$: doubly intractable posterior which cannot be evaluated even up to a normalising constant.

Intractable Likelihood

Interested in Bayesian posterior inference:

$$\mathsf{p}(\theta|\mathcal{D}) = \frac{\mathsf{p}(\theta)\mathsf{p}(\mathcal{D}|\theta)}{\mathsf{p}(\mathcal{D})}$$

- The case of intractable $\mathsf{p}(\mathcal{D}) = \int \mathsf{p}(\theta)\mathsf{p}(\mathcal{D}|\theta)d\theta$: while posterior density is intractable, can use MCMC to generate an (asymptotically exact) sample from $\mathsf{p}(\theta|\mathcal{D})$
- The case of intractable $\mathsf{p}(\mathcal{D}|\theta)$: doubly intractable posterior which cannot be evaluated even up to a normalising constant.

Two situations where (approximate) posterior inference is still possible:

- Can simulate from $\mathsf{p}(\cdot|\theta)$ for any $\theta \in \Theta$:
  Approximate Bayesian Computation (ABC)
  [Tavaré et al, 1997; Beaumont et al, 2002]
- Can construct an unbiased estimator of $\mathsf{p}(\mathcal{D}|\theta)$:
  Pseudo-Marginal MCMC [Beaumont, 2003; Andrieu & Roberts, 2009]

Motivating Example I: Bayesian GP Classification

- Given: covariates $\mathbf{X}$ and labels $\mathbf{y} = [y_1, \ldots, y_n]$.
- Model: y depends on $\mathbf{X}$ via latent Gaussian process $\mathbf{f} = [f(\mathbf{x}_1), \ldots, f(\mathbf{x}_n)]$, with covariance parametrised by $\theta \in \Theta$
  - $f|\theta \sim \mathcal{GP}(0, \kappa_\theta)$ has a covariance function κ_θ.
  - Logistic link $p(\mathbf{y}|\mathbf{f}) = \prod_{i=1}^{n} \frac{1}{1+\exp(-y_i f_i)}$, $y_i \in \{-1, 1\}$.
  - κ_θ: Automatic Relevance Determination (ARD) covariance function:

$$\kappa_\theta(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\frac{1}{2}\sum_{s=1}^{d}\frac{(x_{i,s} - x_{j,s})^2}{\exp(\theta_s)}\right)$$

- Goal: For a prior $p(\theta)$, sample from $p(\theta|\mathbf{y})$ [Williams & Barber, 1998; Filippone & Girolami, 2014]
  - Likelihood $p(\mathbf{y}|\theta) = \int p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\theta)d\mathbf{f}$ is intractable but can be unbiasedly estimated (by e.g. importance sampling $\mathbf{f}$).

Motivating Example I: Bayesian GP Classification

- Given: covariates $\mathbf{X}$ and labels $\mathbf{y} = [y_1, \ldots, y_n]$.
- Model: y depends on $\mathbf{X}$ via latent Gaussian process $\mathbf{f} = [f(\mathbf{x}_1), \ldots, f(\mathbf{x}_n)]$, with covariance parametrised by $\theta \in \Theta$
  - $f|\theta \sim \mathcal{GP}(0, \kappa_\theta)$ has a covariance function κ_θ.
  - Logistic link $p(\mathbf{y}|\mathbf{f}) = \prod_{i=1}^n \frac{1}{1+\exp(-y_i f_i)}$, $y_i \in \{-1, 1\}$.
  - κ_θ: Automatic Relevance Determination (ARD) covariance function:

$$\kappa_\theta(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\frac{1}{2}\sum_{s=1}^{d}\frac{(x_{i,s} - x_{j,s})^2}{\exp(\theta_s)}\right)$$

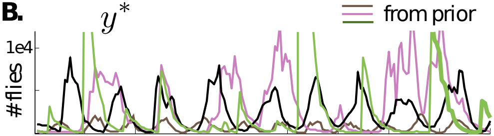
- Goal: For a prior $p(\theta)$, sample from $p(\theta|\mathbf{y})$ [Williams & Barber, 1998; Filippone & Girolami, 2014]
  - Likelihood $p(\mathbf{y}|\theta) = \int p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\theta)d\mathbf{f}$ is intractable but can be unbiasedly estimated (by e.g. importance sampling $\mathbf{f}$).
  - Posterior of θ can have tightly coupled and nonlinearly dependent dimensions - how to sample from it efficiently without gradients?

Motivating example II: ABC for modelling ecological dynamics

- Given: a time series $\mathbf{Y} = (Y_1, \dots, Y_T)$ of population sizes of a blowfly.
- Model: A dynamical system for blowfly population (a discretised ODE) [Nicholson, 1954; Gurney et al, 1980; Wood, 2010; Meeds & Welling, 2014]

$$Y_{t+1} = P Y_{t-\tau} \exp\left(-\frac{Y_{t-\tau}}{Y_0}\right) e_t + Y_t \exp(-\delta \epsilon_t),$$

where $e_t \sim \text{Gamma}\left(\frac{1}{\sigma_P^2}, \sigma_P^2\right)$, $\epsilon_t \sim \text{Gamma}\left(\frac{1}{\sigma_d^2}, \sigma_d^2\right)$.
Parameter vector: $\theta = \{P, Y_0, \sigma_d, \sigma_p, \tau, \delta\}$.



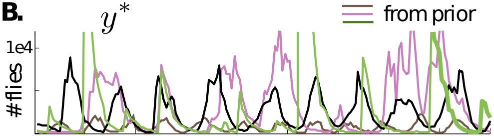
- Goal: For a prior $p(\theta)$, sample from $p(\theta|\mathbf{Y})$.
  - Cannot evaluate $p(\mathbf{Y}|\theta)$. But, can sample from $p(\cdot|\theta)$.

Motivating example II: ABC for modelling ecological dynamics

- Given: a time series $\mathbf{Y} = (Y_1, \ldots, Y_T)$ of population sizes of a blowfly.
- Model: A dynamical system for blowfly population (a discretised ODE) [Nicholson, 1954; Gurney et al, 1980; Wood, 2010; Meeds & Welling, 2014]

$$Y_{t+1} = PY_{t-\tau} \exp\left(-\frac{Y_{t-\tau}}{Y_0}\right) e_t + Y_t \exp(-\delta \epsilon_t),$$

where $e_t \sim$ Gamma $\left(\frac{1}{\sigma_P^2}, \sigma_P^2\right)$, $\epsilon_t \sim$ Gamma $\left(\frac{1}{\sigma_d^2}, \sigma_d^2\right)$.
Parameter vector: $\theta = \{P, Y_0, \sigma_d, \sigma_p, \tau, \delta\}$.



- Goal: For a prior $p(\theta)$, sample from $p(\theta|\mathbf{Y})$.
  - Cannot evaluate $p(\mathbf{Y}|\theta)$. But, can sample from $p(\cdot|\theta)$.
  - For $\mathbf{X} = (X_1, \ldots, X_T) \sim p(\cdot|\theta)$, how to measure distance $\rho(\mathbf{X}, \mathbf{Y})$?

Outline

1. Preliminaries on Kernel Embeddings
2. Gradient-free kernel-based proposals in adaptive Metropolis-Hastings
3. Using Kernel MMD as a criterion in ABC
4. (Conditional) distribution regression for semi-automatic ABC

Outline

1. Preliminaries on Kernel Embeddings
2. Gradient-free kernel-based proposals in adaptive Metropolis-Hastings
3. Using Kernel MMD as a criterion in ABC
4. (Conditional) distribution regression for semi-automatic ABC

Reproducing Kernel Hilbert Space (RKHS)

Definition ([Aronszajn, 1950; Berlinet & Thomas-Agnan, 2004])

Let $\mathcal{X}$ be a non-empty set and $\mathcal{H}$ be a Hilbert space of real-valued functions defined on $\mathcal{X}$. A function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is called *a reproducing kernel* of $\mathcal{H}$ if:

1. $\forall x \in \mathcal{X}, \quad k(\cdot, x) \in \mathcal{H}$, and
2. $\forall x \in \mathcal{X}, \forall f \in \mathcal{H}, \quad \langle f, k(\cdot, x) \rangle_{\mathcal{H}} = f(x)$.

If $\mathcal{H}$ has a reproducing kernel, it is said to be *a reproducing kernel Hilbert space*.

Reproducing Kernel Hilbert Space (RKHS)

Definition ([Aronszajn, 1950; Berlinet & Thomas-Agnan, 2004])

Let $\mathcal{X}$ be a non-empty set and $\mathcal{H}$ be a Hilbert space of real-valued functions defined on $\mathcal{X}$. A function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is called *a reproducing kernel* of $\mathcal{H}$ if:

1. $\forall x \in \mathcal{X}, \quad k(\cdot, x) \in \mathcal{H}$, and
2. $\forall x \in \mathcal{X}, \forall f \in \mathcal{H}, \quad \langle f, k(\cdot, x)\rangle_{\mathcal{H}} = f(x)$.

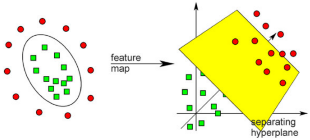
If $\mathcal{H}$ has a reproducing kernel, it is said to be *a reproducing kernel Hilbert space*.

In particular, for any $x, y \in \mathcal{X}$,
$k(x, y) = \langle k(\cdot, y), k(\cdot, x)\rangle_{\mathcal{H}} = \langle k(\cdot, x), k(\cdot, y)\rangle_{\mathcal{H}}$. Thus $\mathcal{H}$ servers as a canonical *feature space* with feature map $x \mapsto k(\cdot, x)$.

- Equivalently, all evaluation functionals $f \mapsto f(x)$ are continuous (norm convergence implies pointwise convergence).
- **Moore-Aronszajn Theorem**: every positive semidefinite $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is a reproducing kernel and has a unique RKHS $\mathcal{H}_k$.

Kernel Trick and Kernel Mean Trick

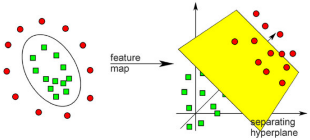
- implicit feature map $x \mapsto k(\cdot, x) \in \mathcal{H}_k$
  replaces $x \mapsto [\varphi_1(x), \dots, \varphi_s(x)] \in \mathbb{R}^s$
- $\langle k(\cdot, x), k(\cdot, y) \rangle_{\mathcal{H}_k} = k(x, y)$
  inner products readily available
    - nonlinear decision boundaries, nonlinear regression functions, learning on non-Euclidean/structured data



[Cortes & Vapnik, 1995; Schölkopf & Smola, 2001]

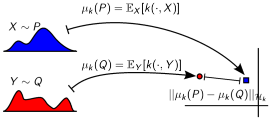
Kernel Trick and Kernel Mean Trick

- implicit feature map $x \mapsto k(\cdot, x) \in \mathcal{H}_k$
  replaces $x \mapsto [\varphi_1(x), \dots, \varphi_s(x)] \in \mathbb{R}^s$
- $\langle k(\cdot, x), k(\cdot, y) \rangle_{\mathcal{H}_k} = k(x, y)$
  inner products readily available
  - nonlinear decision boundaries, nonlinear regression functions, learning on non-Euclidean/structured data



[Cortes & Vapnik, 1995; Schölkopf & Smola, 2001]

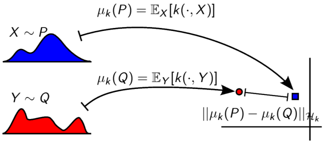
- **RKHS embedding**: implicit feature mean
  [Smola et al, 2007; Sriperumbudur et al, 2010]
  $P \mapsto \mu_k(P) = \mathbb{E}_{X \sim P} k(\cdot, X) \in \mathcal{H}_k$
  replaces $P \mapsto [\mathbb{E}\varphi_1(X), \dots, \mathbb{E}\varphi_s(X)] \in \mathbb{R}^s$
- $\langle \mu_k(P), \mu_k(Q) \rangle_{\mathcal{H}_k} = \mathbb{E}_{X \sim P, Y \sim Q} k(X, Y)$
  inner products easy to estimate
  - nonparametric two-sample, independence, conditional independence, interaction testing, learning on distributions

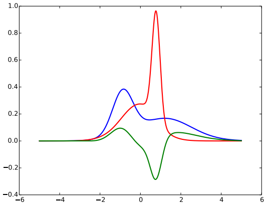


[Gretton et al, 2005; Gretton et al, 2006; Fukumizu et al, 2007; DS et al, 2013; Muandet et al, 2012; Szabo et al, 2015]

Maximum Mean Discrepancy

- **Maximum Mean Discrepancy (MMD)** [Borgwardt et al, 2006; Gretton et al, 2007] between P and Q:

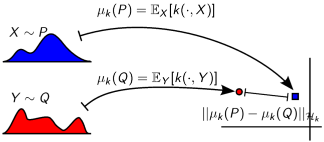


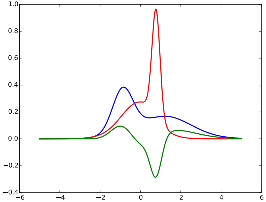


$$\text{MMD}_k(P, Q) = \|\mu_k(P) - \mu_k(Q)\|_{\mathcal{H}_k} = \sup_{f \in \mathcal{H}_k : \|f\|_{\mathcal{H}_k} \le 1} |\mathbb{E}f(X) - \mathbb{E}f(Y)|$$

Maximum Mean Discrepancy

- **Maximum Mean Discrepancy (MMD)** [Borgwardt et al, 2006; Gretton et al, 2007] between P and Q:





$$\text{MMD}_k(P, Q) = \|\mu_k(P) - \mu_k(Q)\|_{\mathcal{H}_k} = \sup_{f \in \mathcal{H}_k :\, \|f\|_{\mathcal{H}_k} \leq 1} |\mathbb{E}f(X) - \mathbb{E}f(Y)|$$

- **Characteristic** kernels: $\text{MMD}_k(P, Q) = 0$ iff $P = Q$.
  - Gaussian RBF $\exp(-\frac{1}{2\sigma^2}\|x - x'\|_2^2)$, Matérn family, inverse multiquadrics.
- For characteristic kernels on LCH $\mathcal{X}$, MMD metrizes weak* topology on probability measures [Sriperumbudur,2010],

$$\text{MMD}_k(P_n, P) \to 0 \Leftrightarrow P_n \rightsquigarrow P.$$

Some uses of MMD

within-sample average similarity

−

between-sample average similarity

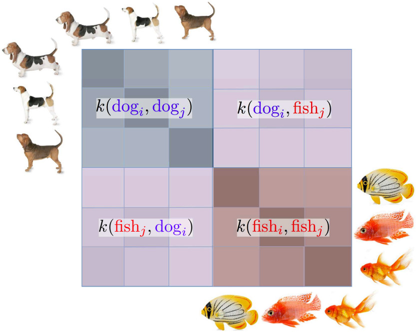


Figure by Arthur Gretton

MMD has been applied to:

- independence tests [Gretton et al, 2009]
- two-sample tests [Gretton et al, 2012]
- training generative neural networks for image data [Dziugaite, Roy and Ghahramani, 2015]
- traversal of manifolds learned by convolutional nets [Gardner et al, 2015]
- similarity measure between observed and simulated data in ABC [Park, Jitkrittum and DS, 2015]

$$\text{MMD}_k^2(P,Q) = \mathbb{E}_{X,X' \overset{i.i.d.}{\sim} P} k(X,X') + \mathbb{E}_{Y,Y' \overset{i.i.d.}{\sim} Q} k(Y,Y') - 2\mathbb{E}_{X\sim P, Y\sim Q} k(X,Y).$$

Outline

Kernel Adaptive Metropolis Hastings. **ICML 2014**.
DS, Heiko Strathmann, Maria Lomeli, Christophe Andrieu
and Arthur Gretton,
http://jmlr.org/proceedings/papers/v32/sejdinovic14.pdf
Code: https://github.com/karlnapf/kameleon-mcmc

Metropolis-Hastings MCMC

- Access to unnormalized target $\pi(\theta) \propto \mathsf{p}(\theta|\mathcal{D})$
- Generate a Markov chain with the posterior $\mathsf{p}(\cdot|\mathcal{D})$ as the invariant distribution
  - Initialize $\theta_0 \sim \mathsf{p}_0$
  - At iteration $t \geq 0$, propose to move to state $\theta' \sim \mathsf{q}(\cdot|\theta_t)$
  - Accept/Reject proposals based on the MH acceptance ratio (preserves detailed balance)

$$\theta_{t+1} = \begin{cases} \theta', & \text{w.p. } \min\left\{1, \frac{\pi(\theta')\mathsf{q}(\theta_t|\theta')}{\pi(\theta_t)\mathsf{q}(\theta'|\theta_t)}\right\}, \\ \theta_t, & \text{otherwise.} \end{cases}$$

The choice of proposal q

- What proposal $q(\cdot|\theta_t)$ to use in Metropolis-Hastings algorithms?
  - Variance of the proposal is too small:
    small increments $\rightarrow$ slow convergence
  - Variance of the proposal is too large:
    too many rejections $\rightarrow$ slow convergence

The choice of proposal q

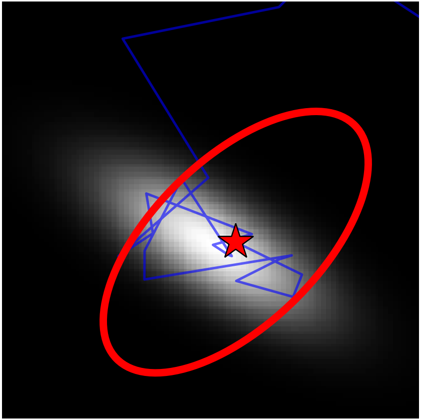
- What proposal $q(\cdot|\theta_t)$ to use in Metropolis-Hastings algorithms?
  - Variance of the proposal is too small:
    small increments $\rightarrow$ slow convergence
  - Variance of the proposal is too large:
    too many rejections $\rightarrow$ slow convergence
- In high dimensions: very different scalings along different principal directions
- [Gelman, Roberts & Gilks, 1996]: in random walk Metropolis with proposal $q(\cdot|\theta_t) = \mathcal{N}(\theta_t, \Sigma)$ on a product target π (independent dimensions):
  - $\Sigma = \frac{2.38^2}{d}\Sigma_\pi$ is shown to be asymptotically optimal as $d \to \infty$
  - Asymptotically optimal acceptance rate of 0.234.

The choice of proposal q

- What proposal $q(\cdot|\theta_t)$ to use in Metropolis-Hastings algorithms?
  - Variance of the proposal is too small:
    small increments $\rightarrow$ slow convergence
  - Variance of the proposal is too large:
    too many rejections $\rightarrow$ slow convergence
- In high dimensions: very different scalings along different principal directions
- [Gelman, Roberts & Gilks, 1996]: in random walk Metropolis with proposal $q(\cdot|\theta_t) = \mathcal{N}(\theta_t, \Sigma)$ on a product target π (independent dimensions):
  - $\Sigma = \frac{2.38^2}{d}\Sigma_\pi$ is shown to be asymptotically optimal as $d \rightarrow \infty$
  - Asymptotically optimal acceptance rate of 0.234.
- Σ_π unknown – can we learn it while running the chain?
- Assumptions not valid for complex targets – non-linear dependence between principal directions?

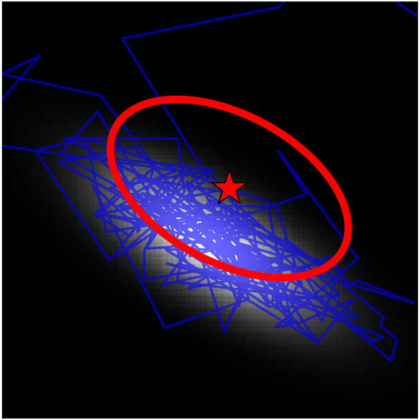
Adaptive MCMC

- **Adaptive Metropolis** [Haario, Saksman & Tamminen, 2001]: Update proposal $q_t(\cdot|\theta_t) = \mathcal{N}(\theta_t, \nu^2\hat{\Sigma}_t)$, using estimates of the target covariance



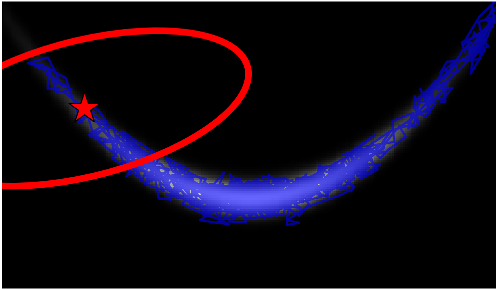
Adaptive MCMC

- **Adaptive Metropolis** [Haario, Saksman & Tamminen, 2001]: Update proposal $q_t(\cdot|\theta_t) = \mathcal{N}(\theta_t, \nu^2\hat{\Sigma}_t)$, using estimates of the target covariance



Adaptive MCMC

- **Adaptive Metropolis** [Haario, Saksman & Tamminen, 2001]: Update proposal $q_t(\cdot|\theta_t) = \mathcal{N}(\theta_t, \nu^2\hat{\Sigma}_t)$, using estimates of the target covariance



Locally miscalibrated for targets with strongly non-linear dependencies: directions of large variance depend on the current location

Intractable & Non-linear Targets?

- Efficient samplers for targets with non-linear dependencies: Hybrid/Hamiltonian Monte Carlo (HMC) or Metropolis Adjusted Langevin Algorithms (MALA) [Duane, Pendleteon & Roweth, 1987; Neal, 2011; Roberts & Stramer, 2003; Girolami & Calderhead, 2011]
  - all require target gradients and second order information.

Intractable & Non-linear Targets?

- Efficient samplers for targets with non-linear dependencies: Hybrid/Hamiltonian Monte Carlo (HMC) or Metropolis Adjusted Langevin Algorithms (MALA) [Duane, Pendleteon & Roweth, 1987; Neal, 2011; Roberts & Stramer, 2003; Girolami & Calderhead, 2011]
  - all require target gradients and second order information.
- But in **pseudo-marginal MCMC**, target $\pi(\cdot)$ cannot be evaluated - gradients typically unavailable.

Pseudo-marginal MCMC

- Posterior inference, latent process $\mathbf{f}$

$$p(\theta|\mathbf{y}) \propto p(\theta)p(\mathbf{y}|\theta) = p(\theta)\int p(\mathbf{f}|\theta)p(\mathbf{y}|\mathbf{f},\theta)d\mathbf{f} =: \pi(\theta)$$

Pseudo-marginal MCMC

- Posterior inference, latent process $\mathbf{f}$

$$p(\theta|\mathbf{y}) \propto p(\theta)p(\mathbf{y}|\theta) = p(\theta)\int p(\mathbf{f}|\theta)p(\mathbf{y}|\mathbf{f},\theta)d\mathbf{f} =: \pi(\theta)$$

- Cannot integrate out $\mathbf{f}$, so cannot compute the MH ratio:

$$\alpha(\theta,\theta') = \min\left\{1, \frac{p(\theta')p(\mathbf{y}|\theta')q(\theta|\theta')}{p(\theta)p(\mathbf{y}|\theta)q(\theta'|\theta)}\right\}$$

Pseudo-marginal MCMC

- Posterior inference, latent process $\mathbf{f}$

$$p(\theta|\mathbf{y}) \propto p(\theta)p(\mathbf{y}|\theta) = p(\theta)\int p(\mathbf{f}|\theta)p(\mathbf{y}|\mathbf{f},\theta)d\mathbf{f} =: \pi(\theta)$$

- Cannot integrate out $\mathbf{f}$, so cannot compute the MH ratio:

$$\alpha(\theta,\theta') = \min\left\{1, \frac{p(\theta')p(\mathbf{y}|\theta')q(\theta|\theta')}{p(\theta)p(\mathbf{y}|\theta)q(\theta'|\theta)}\right\}$$

- Replace $p(\mathbf{y}|\theta)$ with a Monte Carlo (typically importance sampling) estimate $\hat{p}(\mathbf{y}|\theta)$

Pseudo-marginal MCMC

- Posterior inference, latent process $\mathbf{f}$

$$p(\theta|\mathbf{y}) \propto p(\theta)p(\mathbf{y}|\theta) = p(\theta)\int p(\mathbf{f}|\theta)p(\mathbf{y}|\mathbf{f},\theta)d\mathbf{f} =: \pi(\theta)$$

- Cannot integrate out $\mathbf{f}$, so cannot compute the MH ratio:

$$\alpha(\theta,\theta') = \min\left\{1, \frac{p(\theta')\hat{p}(\mathbf{y}|\theta')q(\theta|\theta')}{p(\theta)\hat{p}(\mathbf{y}|\theta)q(\theta'|\theta)}\right\}$$

- Replace $p(\mathbf{y}|\theta)$ with a Monte Carlo (typically importance sampling) estimate $\hat{p}(\mathbf{y}|\theta)$
- Replacing the likelihood with an *unbiased estimate* still results in the *correct invariant distribution* [Beaumont, 2003; Andrieu & Roberts, 2009]

Back to the motivating example: Bayesian GPC

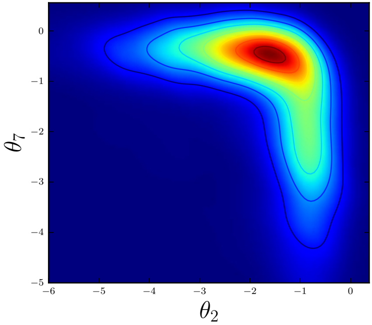
- $f|\theta \sim \mathcal{GP}(0, \kappa_\theta)$, $p(y_i|f(x_i)) = \frac{1}{1+\exp(-y_i f(x_i))}$
- Cannot use a Gibbs sampler on $p(\theta, \mathbf{f}|\mathbf{y})$, which samples from $p(\mathbf{f}|\theta, \mathbf{y})$ and $p(\theta|\mathbf{f}, \mathbf{y})$ in turns, since $p(\theta|\mathbf{f}, \mathbf{y})$ is extremely sharp.
- Use Pseudo-Marginal MCMC to sample $p(\theta|\mathbf{y}) = p(\theta) \int p(\theta, \mathbf{f}|\mathbf{y}) p(\mathbf{f}|\theta) d\mathbf{f}$.
- Unbiased estimate of $\hat{p}(\mathbf{y}|\theta)$ via importance sampling:

$$\hat{p}(\mathbf{y}|\theta) = \frac{1}{n_{\text{imp}}} \sum_{i=1}^{n_{\text{imp}}} p(\mathbf{y}|\mathbf{f}^{(i)}) \frac{p(\mathbf{f}^{(i)}|\theta)}{Q(\mathbf{f}^{(i)})}$$

- No access to the gradient or Hessian of the target.

Intractable & Non-linear Target in GPC

- Sliced posterior over hyperparameters of a Gaussian Process classifier on UCI Glass dataset obtained using Pseudo-Marginal MCMC
- Classification of window vs. non-window glass:
  - Heterogeneous structure of each of the classes (non-window glass consists of containers, tableware and headlamps): ambiguities in the set of lengthscales which determine the decision boundary



Adaptive sampler that learns the shape of non-linear targets without gradient information?

RKHS Covariance operator

Definition

The covariance operator of P is $C_P : \mathcal{H}_k \to \mathcal{H}_k$ such that $\forall f, g \in \mathcal{H}_k$, $\langle f, C_P g \rangle_{\mathcal{H}_k} = \mathrm{Cov}_P \left[f(X) g(X) \right]$.

- Covariance operator: $C_P : \mathcal{H}_k \to \mathcal{H}_k$ is given by the covariance of canonical features

$$C_P = \int (k(\cdot, x) - \mu_P) \otimes (k(\cdot, x) - \mu_P) \, \mathrm{d}P(x)$$

RKHS Covariance operator

Definition

The covariance operator of P is $C_P : \mathcal{H}_k \to \mathcal{H}_k$ such that $\forall f, g \in \mathcal{H}_k$, $\langle f, C_P g \rangle_{\mathcal{H}_k} = \text{Cov}_P [f(X)g(X)]$.

- Covariance operator: $C_P : \mathcal{H}_k \to \mathcal{H}_k$ is given by the covariance of canonical features

$$C_P = \int (k(\cdot, x) - \mu_P) \otimes (k(\cdot, x) - \mu_P) \, \mathrm{d}P(x)$$

- Empirical versions of embedding and the covariance operator:

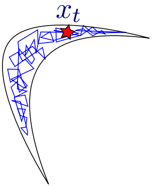
$$\mu_{\mathbf{z}} = \frac{1}{n} \sum_{i=1}^{n} k(\cdot, z_i) \qquad C_{\mathbf{z}} = \frac{1}{n} \sum_{i=1}^{n} (k(\cdot, z_i) - \mu_{\mathbf{z}}) \otimes (k(\cdot, z_i) - \mu_{\mathbf{z}})$$

The empirical covariance captures **non-linear** features of the underlying distribution, e.g. Kernel PCA [Schölkopf, Smola and Müller, 1998]

RKHS covariance informs the MH proposal

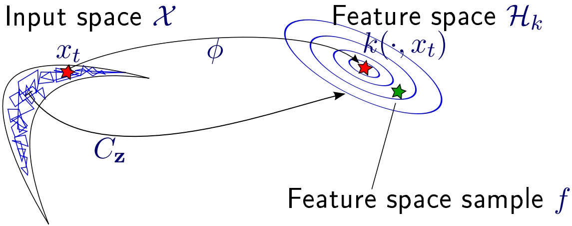
- Based on chain history $\{z_i\}_{i=1}^n$, capture non-linearities using covariance $C_{\mathbf{z}} = \frac{1}{n}\sum_{i=1}^n (k(\cdot, z_i) - \mu_{\mathbf{z}}) \otimes (k(\cdot, z_i) - \mu_{\mathbf{z}})$ in the RKHS $\mathcal{H}_k$.

Input space $\mathcal{X}$



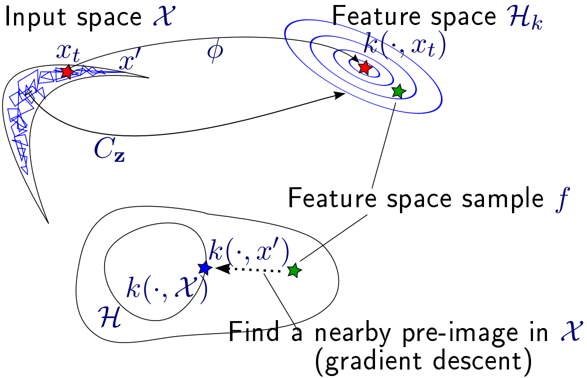
RKHS covariance informs the MH proposal

- Based on chain history $\{z_i\}_{i=1}^n$, capture non-linearities using covariance $C_{\mathbf{z}} = \frac{1}{n}\sum_{i=1}^n (k(\cdot, z_i) - \mu_{\mathbf{z}}) \otimes (k(\cdot, z_i) - \mu_{\mathbf{z}})$ in the RKHS $\mathcal{H}_k$.



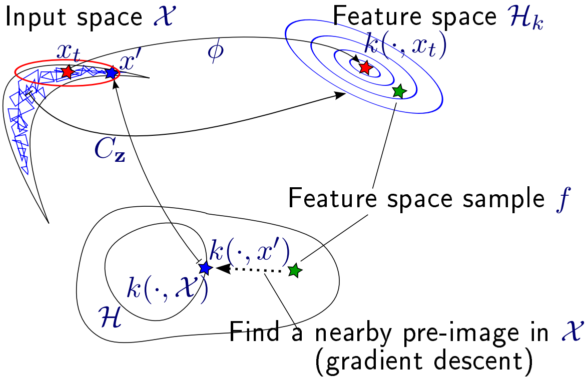
RKHS covariance informs the MH proposal

- Based on chain history $\{z_i\}_{i=1}^n$, capture non-linearities using covariance $C_{\mathbf{z}} = \frac{1}{n}\sum_{i=1}^n (k(\cdot, z_i) - \mu_{\mathbf{z}}) \otimes (k(\cdot, z_i) - \mu_{\mathbf{z}})$ in the RKHS $\mathcal{H}_k$.



RKHS covariance informs the MH proposal

- Based on chain history $\{z_i\}_{i=1}^n$, capture non-linearities using covariance $C_{\mathbf{z}} = \frac{1}{n}\sum_{i=1}^n (k(\cdot, z_i) - \mu_{\mathbf{z}}) \otimes (k(\cdot, z_i) - \mu_{\mathbf{z}})$ in the RKHS $\mathcal{H}_k$.



Proposal Construction Summary

1. Get a chain subsample $\mathbf{z} = \{z_i\}_{i=1}^n$
2. Construct an RKHS sample $f \sim \mathcal{N}(k(\cdot, x_t), \nu^2 C_{\mathbf{z}})$
3. Propose x' such that $k(\cdot, x')$ is close to f (with an additional exploration term $\xi \sim \mathcal{N}(0, \gamma^2 I_d)$).

Proposal Construction Summary

1. Get a chain subsample $\mathbf{z} = \{z_i\}_{i=1}^n$
2. Construct an RKHS sample $f \sim \mathcal{N}(k(\cdot, x_t), \nu^2 C_{\mathbf{z}})$
3. Propose x' such that $k(\cdot, x')$ is close to f (with an additional exploration term $\xi \sim \mathcal{N}(0, \gamma^2 I_d)$).

This gives:

$$x'|x_t, f, \xi = x_t - \eta \nabla_x \|k(\cdot, x) - f\|_{\mathcal{H}_k}^2 |_{x=x_t} + \xi$$

Proposal Construction Summary

1. Get a chain subsample $\mathbf{z} = \{z_i\}_{i=1}^n$
2. Construct an RKHS sample $f \sim \mathcal{N}(k(\cdot, x_t), \nu^2 C_{\mathbf{z}})$
3. Propose x' such that $k(\cdot, x')$ is close to f (with an additional exploration term $\xi \sim \mathcal{N}(0, \gamma^2 I_d)$).

This gives:

$$x'|x_t, f, \xi = x_t - \eta \nabla_x \|k(\cdot, x) - f\|_{\mathcal{H}_k}^2 |_{x=x_t} + \xi$$

Integrate out RKHS samples f, gradient step, and ξ to obtain marginal Gaussian proposal on the input space:

$$q_{\mathbf{z}}(x'|x_t) = \mathcal{N}(x_t, \gamma^2 I_d + \nu^2 M_{\mathbf{z},x_t} H M_{\mathbf{z},x_t}^\top),$$

$$M_{\mathbf{z},x_t} = [\nabla_x k(x, z_1)|_{x=x_t}, \dots, \nabla_x k(x, z_n)|_{x=x_t}].$$

MCMC Kameleon: Kernel Adaptive Metropolis Hastings

Input: unnormalized target π; subsample size n; scaling parameters ν, γ, kernel k;
update schedule $\{p_t\}_{t\geq 1}$ with $p_t \to 0$, $\sum_{t=1}^{\infty} p_t = \infty$

At iteration $t+1$,

1. With probability p_t, update a random subsample $\mathbf{z} = \{z_i\}_{i=1}^{n}$ of the chain history $\{x_i\}_{i=0}^{t-1}$,
2. Sample proposed point x' from $q_{\mathbf{z}}(\cdot|x_t) = \mathcal{N}(x_t, \gamma^2 I_d + \nu^2 M_{\mathbf{z},x_t} H M_{\mathbf{z},x_t}^{\top})$,
3. Accept/Reject with standard MH ratio:

$$x_{t+1} = \begin{cases} x', & \text{w.p. } \min\left\{1, \frac{\pi(x')q_{\mathbf{z}}(x_t|x')}{\pi(x_t)q_{\mathbf{z}}(x'|x_t)}\right\}, \\ x_t, & \text{otherwise.} \end{cases}$$

MCMC Kameleon: Kernel Adaptive Metropolis Hastings

Input: unnormalized target π; subsample size n; scaling parameters ν, γ, kernel k; update schedule $\{p_t\}_{t\geq 1}$ with $p_t \rightarrow 0$, $\sum_{t=1}^{\infty} p_t = \infty$

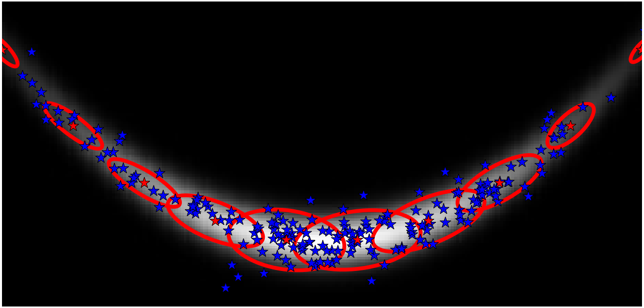
At iteration $t+1$,

1. With probability p_t, update a random subsample $\mathbf{z} = \{z_i\}_{i=1}^{n}$ of the chain history $\{x_i\}_{i=0}^{t-1}$,
2. Sample proposed point x' from $q_{\mathbf{z}}(\cdot|x_t) = \mathcal{N}(x_t, \gamma^2 I_d + \nu^2 M_{\mathbf{z},x_t} H M_{\mathbf{z},x_t}^{\top})$,
3. Accept/Reject with standard MH ratio:

$$x_{t+1} = \begin{cases} x', & \text{w.p. } \min\left\{1, \frac{\pi(x')q_{\mathbf{z}}(x_t|x')}{\pi(x_t)q_{\mathbf{z}}(x'|x_t)}\right\}, \\ x_t, & \text{otherwise.} \end{cases}$$

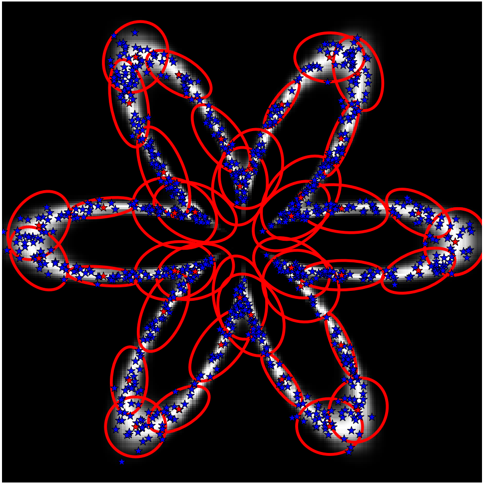
Convergence to target π preserved as long as $p_t \rightarrow 0$ [Roberts & Rosenthal, 2007].

Locally aligned covariance



Kameleon proposals capture local covariance structure

Locally aligned covariance



Examples of Covariance Structure for Standard Kernels

- **Linear kernel:** $k(x, x') = x^\top x'$

$$q_{\mathbf{z}}(\cdot|y) = \mathcal{N}(y, \gamma^2 I + 4\nu^2 \mathbf{Z}^\top H \mathbf{Z})$$

which is classical Adaptive Metropolis [Haario et al 1999;2001].

Examples of Covariance Structure for Standard Kernels

- **Linear kernel:** $k(x, x') = x^\top x'$

$$q_{\mathbf{z}}(\cdot|y) = \mathcal{N}(y, \gamma^2 I + 4\nu^2 \mathbf{Z}^\top H \mathbf{Z})$$

which is classical Adaptive Metropolis [Haario et al 1999;2001].

- **Gaussian RBF kernel:** $k(x, x') = \exp\left(-\frac{1}{2\sigma^2}\|x - x'\|_2^2\right)$

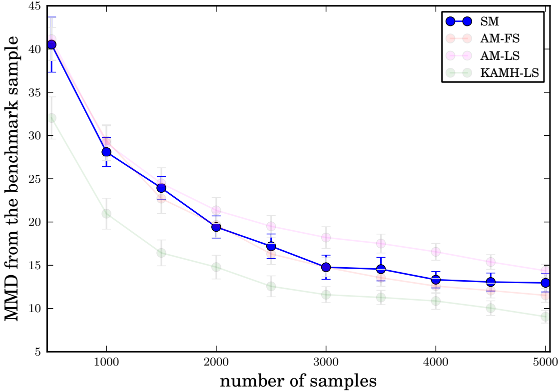
$$\left[\text{cov}[q_{\mathbf{z}(\cdot|x_t)}]\right]_{ij} = \gamma^2 \delta_{ij} + \frac{4\nu^2}{\sigma^4} \sum_{\ell=1}^{n} [k(y, z_\ell)]^2 (z_{\ell,i} - x_{t,i})(z_{\ell,j} - x_{t,j}) + \mathcal{O}\left(\frac{1}{n}\right).$$

Influence of previous points z_ℓ on the proposal covariance is weighted by the similarity $k(x_t, z_\ell)$ to the current location x_t.

Setup

- (**SM**) Standard Metropolis with the isotropic proposal $q(\cdot|x_t) = \mathcal{N}(x_t, \nu^2 I)$ and scaling $\nu = 2.38/\sqrt{d}$ [Gelman, Roberts & Gilks, 1996].
- (**AM-FS**) Adaptive Metropolis with a learned covariance matrix and fixed global scaling $\nu = 2.38/\sqrt{d}$
- (**AM-LS**) Adaptive Metropolis with a learned covariance matrix and global scaling ν learned to bring the acceptance rate close to $\alpha^* = 0.234$ [Gelman, Roberts & Gilks, 1996].
- (**KAMH-LS**) MCMC Kameleon with the global scaling ν learned to bring the acceptance rate close to $\alpha^* = 0.234$

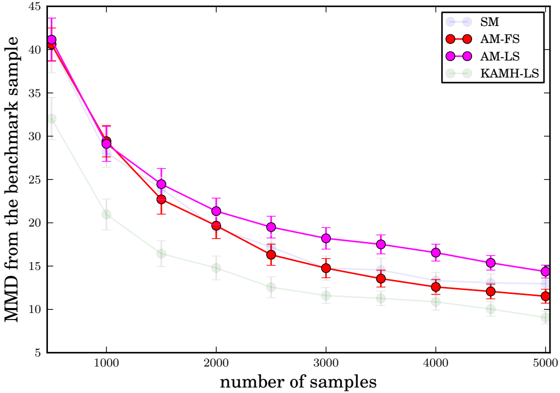
UCI Glass dataset



comparison in terms of all mixed moments up to order 3

8-dimensional non-linear posterior $p(\theta|\mathbf{y})$: no ground truth, performance with respect to a long-run, heavily thinned benchmark sample.

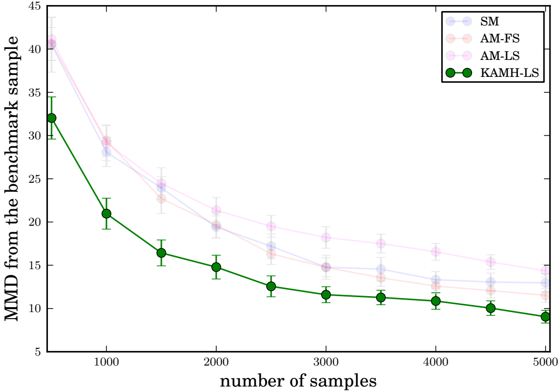
UCI Glass dataset



comparison in terms of all mixed moments up to order 3

8-dimensional non-linear posterior $p(\theta|\mathbf{y})$: no ground truth, performance with respect to a long-run, heavily thinned benchmark sample.

UCI Glass dataset



comparison in terms of all mixed moments up to order 3

8-dimensional non-linear posterior $p(\theta|\mathbf{y})$: no ground truth, performance with respect to a long-run, heavily thinned benchmark sample.

Random Fourier features: Inverse Kernel Trick

Bochner's representation: any positive definite **translation-invariant** kernel on $\mathbb{R}^p$ can be written as

$$\begin{aligned} k(x,y) &= \int_{\mathbb{R}^p} \exp\left(i\omega^\top (x-y)\right) d\Lambda(\omega) \\ &= \int_{\mathbb{R}^p} \left\{\cos\left(\omega^\top x\right)\cos\left(\omega^\top y\right) + \sin\left(\omega^\top x\right)\sin\left(\omega^\top y\right)\right\} d\Lambda(\omega) \end{aligned}$$

for some positive measure (w.l.o.g. a probability distribution) Λ.

- Sample m frequencies $\{\omega_j\} \sim \Lambda$ and use a Monte Carlo estimator of the kernel function instead [Rahimi & Recht, 2007]:

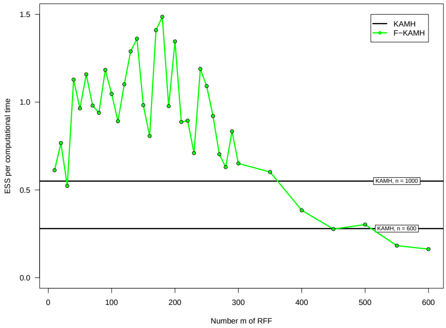
$$\begin{aligned} \hat{k}(x,y) &= \frac{1}{m}\sum_{j=1}^{m} \left\{\cos\left(\omega_j^\top x\right)\cos\left(\omega_j^\top y\right) + \sin\left(\omega_j^\top x\right)\sin\left(\omega_j^\top y\right)\right\} \\ &= \langle \varphi_\omega(x), \varphi_\omega(y)\rangle_{\mathbb{R}^{2m}}, \end{aligned}$$

  with an explicit set of features $x \mapsto \sqrt{\frac{1}{m}}\left[\cos\left(\omega_1^\top x\right), \sin\left(\omega_1^\top x\right), \ldots\right]$.

- How fast does m need to grow with n? Sublinear for regression [Bach, 2015; Rudi et al, 2016]

RFF Kameleon

- Kameleon updates cost $O(np^2 + p^3)$ where p is the ambient dimension and n is the number of samples used to estimate the RKHS covariance
- A version based on random Fourier features allows online updates independent of n, costing $O(m^2p + mp^2 + p^3)$: preserves the benefits of capturing nonlinear covariance structure with no limit on the number of samples that can be used – *better estimation of covariance in the "wrong" RKHS*.



8-dimensional synthetic Banana distribution

[A. Kotlicki, MSc Thesis, Oxford, 2015]

Summary

- A family of simple, versatile, gradient-free adaptive MCMC samplers.
- Proposals automatically conform to the local covariance structure of the target distribution at the current chain state.
- Outperforming existing approaches on intractable target distributions with nonlinear dependencies.
- Random Fourier feature expansions: tradeoffs between the computational and statistical efficiency

- **code: https://github.com/karlnapf/kameleon-mcmc**

Outline

K2-ABC: Approximate Bayesian Computation with Kernel Embeddings. **AISTATS 2016**
Mijung Park, Wittawat Jitkrittum, and DS.
http://arxiv.org/abs/1502.02558
Code: https://github.com/wittawatj/k2abc

ABC

- Observe a dataset $\mathbf{Y}$,

$$
\begin{aligned}
p(\theta|\mathbf{Y}) &\propto p(\theta)p(\mathbf{Y}|\theta) \\
&= p(\theta)\int p(\mathbf{X}|\theta)\,\mathrm{d}\delta_{\mathbf{Y}}(\mathbf{X}) \\
&\approx p(\theta)\int p(\mathbf{X}|\theta)\kappa_\epsilon(\mathbf{X},\mathbf{Y})\,\mathrm{d}\mathbf{X},
\end{aligned}
$$

where $\kappa_\epsilon(\mathbf{X},\mathbf{Y})$ defines similarity of $\mathbf{X}$ and $\mathbf{Y}$.

ABC

- Observe a dataset $\mathbf{Y}$,

$$\begin{aligned} p(\theta|\mathbf{Y}) &\propto p(\theta)p(\mathbf{Y}|\theta) \\ &= p(\theta)\int p(\mathbf{X}|\theta)\,\mathrm{d}\delta_{\mathbf{Y}}(\mathbf{X}) \\ &\approx p(\theta)\int p(\mathbf{X}|\theta)\kappa_\epsilon(\mathbf{X},\mathbf{Y})\,\mathrm{d}\mathbf{X}, \end{aligned}$$

where $\kappa_\epsilon(\mathbf{X},\mathbf{Y})$ defines similarity of $\mathbf{X}$ and $\mathbf{Y}$.

$$\text{(ABC likelihood)}\;\; p_\epsilon(\mathbf{Y}|\theta) := \int p(\mathbf{X}|\theta)\kappa_\epsilon(\mathbf{X},\mathbf{Y})\,\mathrm{d}\mathbf{X}.$$

- Simplest choice $\kappa_\epsilon(\mathbf{X},\mathbf{Y}) := \mathbf{1}(\rho(\mathbf{X},\mathbf{Y}) < \epsilon)$
  - ρ : a distance function between observed and simulated data
  - $\mathbf{1}(\cdot) \in \{0,1\}$: indicator function

Rejection ABC Algorithm

- **Input:** observed dataset $\mathbf{Y}$, distance ρ, threshold ϵ
- **Output:** posterior sample $\{\theta_i\}_{i=1}^{M}$ from approximate posterior $p_\epsilon(\theta|\mathbf{Y}) \propto p(\theta)p_\epsilon(\mathbf{Y}|\theta)$

```
1: repeat
2:     Sample θ ~ p(θ)
3:     Sample a pseudo dataset X ~ p(·|θ)
4:     if ρ(X, Y) < ε then
5:         Keep θ
6:     end if
7: until we have M points
```

- **Notation**: $\mathbf{Y}$ = observed set. $\mathbf{X}$ = pseudo (generated) dataset.

Data Similarity via Summary Statistics

- Distance ρ is typically defined via summary statistics

$$\rho(\mathbf{X}, \mathbf{Y}) = \|s(\mathbf{X}) - s(\mathbf{Y})\|_2.$$

- How to select the summary statistics $s(\cdot)$? Unless $s(\cdot)$ is sufficient, targets the incorrect (partial) posterior $p(\theta|s(\mathbf{Y}))$ rather than $p(\theta|\mathbf{Y})$.
- Hard to quantify additional bias.
  - Adding more summary statistics decreases "information loss": $p(\theta|s(\mathbf{Y})) \approx p(\theta|\mathbf{Y})$
  - ρ computed on a higher dimensional space - without appropriate calibration of distances therein, leads to a higher rejection rate so need to increase ϵ: $p_\epsilon(\theta|s(\mathbf{Y})) \not\approx p(\theta|s(\mathbf{Y}))$

Data Similarity via Summary Statistics

- Distance ρ is typically defined via summary statistics

$$\rho(\mathbf{X}, \mathbf{Y}) = \|s(\mathbf{X}) - s(\mathbf{Y})\|_2.$$

- How to select the summary statistics $s(\cdot)$? Unless $s(\cdot)$ is sufficient, targets the incorrect (partial) posterior $p(\theta|s(\mathbf{Y}))$ rather than $p(\theta|\mathbf{Y})$.
- Hard to quantify additional bias.
  - Adding more summary statistics decreases "information loss": $p(\theta|s(\mathbf{Y})) \approx p(\theta|\mathbf{Y})$
  - ρ computed on a higher dimensional space - without appropriate calibration of distances therein, leads to a higher rejection rate so need to increase ϵ: $p_\epsilon(\theta|s(\mathbf{Y})) \not\approx p(\theta|s(\mathbf{Y}))$
- Contribution: Use a nonparametric distance (MMD) between the empirical measures of datasets $\mathbf{X}$ and $\mathbf{Y}$).
  - No need to design $s(\cdot)$.
  - Rejection rate does not blow up since MMD penalises the higher order moments via Mercer expansion.

Embeddings via Mercer Expansion

Mercer Expansion

For a compact metric space $\mathcal{X}$, and a continous kernel k,

$$k(x,y) = \sum_{r=1}^{\infty} \lambda_r \Phi_r(x) \Phi_r(y),$$

with $\{\lambda_r, \Phi_r\}_{r \geq 1}$ eigenvalue, eigenfunction pairs of $f \mapsto \int f(x) k(\cdot, x) dP(x)$ on $L_2(P)$, with $\lambda_r \to 0$, as $r \to \infty$. Φ_r are typically functions of increasing "complexity", i.e., Hermite polynomials of increasing degree.

$$\mathcal{H}_k \ni k(\cdot, x) \quad \leftrightarrow \quad \left\{ \sqrt{\lambda_r} \Phi_r(x) \right\} \in \ell_2$$

$$\mathcal{H}_k \ni \mu_k(P) \quad \leftrightarrow \quad \left\{ \sqrt{\lambda_r} \mathbb{E} \Phi_r(X) \right\} \in \ell_2$$

$$\left\| \mu_k(\hat{P}) - \mu_k(\hat{Q}) \right\|_{\mathcal{H}_k}^2 = \sum_{r=1}^{\infty} \lambda_r \left(\frac{1}{n_x} \sum_{t=1}^{n_x} \Phi_r(X_t) - \frac{1}{n_y} \sum_{t=1}^{n_y} \Phi_r(Y_t) \right)^2$$

K2-ABC (proposed method)

- **Input:** observed data $\mathbf{Y}$, threshold ϵ
- **Output:** Empirical posterior $\sum_{i=1}^{M} w_i \delta_{\theta_i}$

```
for i = 1, ..., M do
    Sample θ_i ~ p(θ)
    Sample pseudo dataset X_i ~ p(·|θ_i)
    w̃_i = κ_ε(X_i, Y) = exp( − MMD^2(X_i, Y) / ε )
end for
w_i = w̃_i / Σ_{j=1}^{M} w̃_j  for i = 1, ..., M
```

for $i = 1, \ldots, M$ **do**
Sample $\theta_i \sim p(\theta)$
Sample pseudo dataset $\mathbf{X}_i \sim p(\cdot|\theta_i)$
$\widetilde{w}_i = \kappa_\epsilon(\mathbf{X}_i, \mathbf{Y}) = \exp\left(-\frac{\widehat{\text{MMD}}^2(\mathbf{X}_i, \mathbf{Y})}{\epsilon}\right)$
end for
$w_i = \widetilde{w}_i / \sum_{j=1}^{M} \widetilde{w}_j$ for $i = 1, \ldots, M$

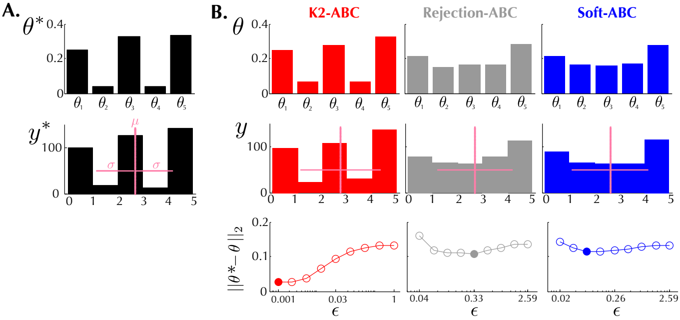
- Easy to sample from $\sum_{i=1}^{M} w_i \delta_{\theta_i}$.
- "K2" because we use two kernels. k (in MMD) and κ_ϵ.

Toy data: Failure of Insufficient Statistics

$$p(y|\theta) = \sum_{i=1}^{5} \theta_i \text{Uniform}(y; [i-1, i])$$

$$\pi(\theta) = \text{Dirichlet}(\theta; \mathbf{1})$$

$$\theta^* = \text{(see figure A)}$$



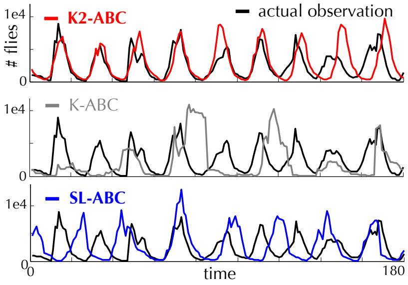
- Summary statistics $s(\mathbf{y}) = (\hat{\mathbb{E}}[\mathbf{y}], \hat{\mathbb{V}}[\mathbf{y}])^\top$ are insufficient to represent $p(\mathbf{y}|\theta)$.

Blow Fly Population Modelling

Number of blow flies over time

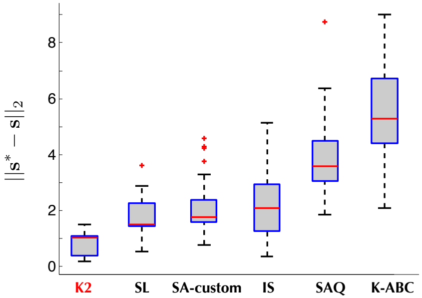
$$Y_{t+1} = PY_{t-\tau} \exp\left(-\frac{Y_{t-\tau}}{Y_0}\right) e_t + Y_t \exp(-\delta \epsilon_t)$$

- $e_t \sim \text{Gam}\left(\frac{1}{\sigma_P^2}, \sigma_P^2\right)$ and $\epsilon_t \sim \text{Gam}\left(\frac{1}{\sigma_d^2}, \sigma_d^2\right)$.
- Want $\theta := \{P, Y_0, \sigma_d, \sigma_p, \tau, \delta\}$.

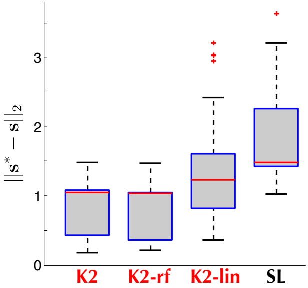


- Simulated trajectories with inferred posterior mean of θ
  - Observed sample of size 180.
  - Other methods use handcrafted 10-dimensional summary statistics $s(\cdot)$ from [Meeds & Welling, 2014]: quantiles of marginals, first-order differences, maximal peaks, etc.

Blowfly dataset



- Let $\tilde{\theta}$ be the posterior mean.
- Simulate $\mathbf{X} \sim p(\cdot|\tilde{\theta})$.
- $\mathbf{s} = s(\mathbf{X})$ and $\mathbf{s}^* = s(\mathbf{Y})$.
- Improved mean squared error on $\mathbf{s}$, even though SL-ABC, SA-custom explicitly operate on $\mathbf{s}$ while K2-ABC does not.



- Computation of $\widehat{\mathrm{MMD}}^2(\mathbf{X}, \mathbf{Y})$ costs $O(n^2)$.
- Linear-time unbiased estimators of MMD^2 or random feature expansions reduce the cost to $O(n)$.

Outline

DR-ABC: Approximate Bayesian Computation with
Kernel-Based Distribution Regression
Jovana Mitrovic, DS, and Yee Whye Teh.
http://arxiv.org/abs/1602.04805

Semi-Automatic ABC

- [Fearnhead & Prangle, 2012] consider summary statistics "optimal" for Bayesian inference with respect to a particular loss function, i.e. achieves the minimum expected loss under the true posterior

$$\int L(\theta, \hat{\theta}) p(\theta|\mathbf{y}) d\theta,$$

where $\hat{\theta}$ is a point estimate under the ABC partial posterior $p_\epsilon(\theta|s(\mathbf{y}))$.

- Under the squared loss $L(\theta, \hat{\theta}) = \|\theta - \hat{\theta}\|_2^2$, and for $\hat{\theta} = \mathbb{E}_\epsilon [\theta|s(\mathbf{y})]$, the optimal summary statistic is the true posterior mean $s(\mathbf{y}) = \mathbb{E}[\theta|\mathbf{y}]$.
  - Results in ABC approximation that attempts to have the same posterior mean as the true posterior (but still returns the whole posterior).

Semi-Automatic ABC

- [Fearnhead & Prangle, 2012] consider summary statistics "optimal" for Bayesian inference with respect to a particular loss function, i.e. achieves the minimum expected loss under the true posterior

$$\int L(\theta, \hat{\theta}) p(\theta|\mathbf{y}) d\theta,$$

  where $\hat{\theta}$ is a point estimate under the ABC partial posterior $p_\epsilon(\theta|s(\mathbf{y}))$.
- Under the squared loss $L(\theta, \hat{\theta}) = \|\theta - \hat{\theta}\|_2^2$, and for $\hat{\theta} = \mathbb{E}_\epsilon[\theta|s(\mathbf{y})]$, the optimal summary statistic is the true posterior mean $s(\mathbf{y}) = \mathbb{E}[\theta|\mathbf{y}]$.
  - Results in ABC approximation that attempts to have the same posterior mean as the true posterior (but still returns the whole posterior).

SA-ABC

- Use regression on simulated $(\mathbf{x}_i, \theta_i)$ pairs to estimate the regression function $g(\mathbf{x}) = \hat{\mathbb{E}}[\theta|\mathbf{x}]$.
- Use g as the summary statistic in the usual ABC algorithm.

Regression in SA-ABC

- Linear on all concatenated dataset $\mathbf{x}_i$? Adding quadratic terms and/or basis functions? Can be extremely high-dimensional and poorly behaved.
- Target θ is not a property of the concatenated data but of its generating distribution $p(\cdot|\theta)$.

Regression in SA-ABC

- Linear on all concatenated dataset $\mathbf{x}_i$? Adding quadratic terms and/or basis functions? Can be extremely high-dimensional and poorly behaved.
- Target θ is not a property of the concatenated data but of its generating distribution $p(\cdot|\theta)$.
- Contribution: Distribution regression (for iid data from $p(\cdot|\theta)$) and conditional distribution regression (for time series or models with "auxiliary observations") to select optimal summary statistics.

Learning on Distributions

- Multiple-Instance Learning: Input is a bag of B_i vectors $\mathbf{x}_i = \{x_{i1}, \ldots, x_{iB_i}\}$, each $x_{ia} \in X$ assumed to arise from a probability distribution P_i on $\mathcal{X}$.
- Represent the i-th bag by the corresponding empirical kernel embedding w.r.t. a kernel k on $\mathcal{X}$.

$$\mathfrak{m}_i = \mathfrak{m}[\mathbf{x}_i] = \widehat{\mu_k [\mathsf{P}_i]} = \frac{1}{B_i} \sum_{a=1}^{B_i} k(\cdot, x_{ia})$$

- Now treat the problem as having inputs $\mathfrak{m}_i \in \mathcal{H}_k$: just need to define a *kernel* K on $\mathcal{H}_k$. [Muandet et al, 2012; Szabo et al, 2015].

Linear: $$K(\mathfrak{m}_i, \mathfrak{m}_j) = \langle \mathfrak{m}_i, \mathfrak{m}_j \rangle_{\mathcal{H}_k} = \frac{1}{B_i B_j} \sum_{a=1}^{B_i} \sum_{b=1}^{B_j} k(x_{ia}, x_{jb})$$

Gaussian: $$K(\mathfrak{m}_i, \mathfrak{m}_j) = \exp\left(-\frac{1}{2\gamma^2} \|\mathfrak{m}_i - \mathfrak{m}_j\|^2_{\mathcal{H}_k}\right).$$

Term $\|\mathfrak{m}_i - \mathfrak{m}_j\|^2_{\mathcal{H}_k}$ is precisely the MMD^2.

DR-ABC

Input: prior $p(\theta)$, simulator $p(\cdot|\theta)$, observed data $\mathbf{y} = \{y_i\}_i$, threshold ϵ

Step 1: Simulate training pairs $(\theta_i, \mathbf{x}_i)_{i=1}^n$, where each $\mathbf{x}_i = (x_{i1}, \ldots, x_{iB}) \overset{i.i.d.}{\sim} p(\cdot|\theta)$ and perform distribution kernel ridge regression:

$$g(\mathbf{x}) = \sum_{i=1}^{n} \alpha_i K(\mathfrak{m}[\mathbf{x}], \mathfrak{m}_i)$$

with $\alpha = (\mathbf{K} + \lambda I)^{-1}\boldsymbol{\theta}$, $\mathbf{K}_{ij} = K(\mathfrak{m}_i, \mathfrak{m}_j)$ and $\boldsymbol{\theta} = [\theta_1, \theta_2, \ldots, \theta_n]^\top$

Step 2: Run ABC with $g(\cdot)$ as the summary statistic.

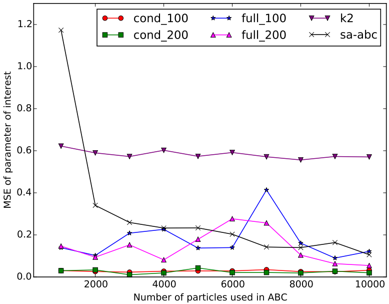
Regression from Conditional Distributions

- Often, θ models a certain transition operator, e.g. time series, or a conditional distribution of observations given certain auxiliary information $\mathbf{z}$ (e.g. a spatial location). In that case, more natural to regress from a conditional embedding operator [Fukumizu et al 2008; Song et al 2013] $C_{X|Z} : \mathcal{H}_{k_{\mathcal{Z}}} \to \mathcal{H}_{k_{\mathcal{X}}}$ of $\{P_\theta(\cdot|z)\}_{z\in\mathcal{Z}}$, such that

$$\mu_{X|Z=z} = C_{X|Z} k_{\mathcal{Z}}(\cdot, z), \quad C_{X|Z} C_{ZZ} = C_{XZ}$$

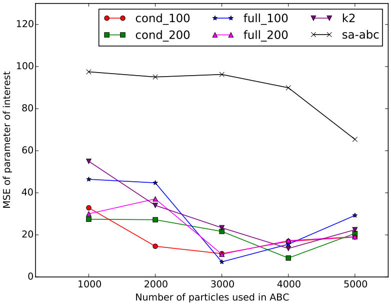
- Now simply need a kernel on the space of linear operators from $\mathcal{H}_{k_{\mathcal{Z}}}$ to $\mathcal{H}_{k_{\mathcal{X}}}$, e.g. a linear kernel $K(C, C') = Tr(C^* C')$ or any kernel that depends on $||C - C'||_{HS}$.
- Easily implementable with multiple layers of random Fourier features.

Experiments



Toy example: Gaussian hierarchical model

$$\theta \sim \mathcal{N}(2,1),$$
$$z \sim \mathcal{N}(0,2),$$
$$x|z,\theta \sim \mathcal{N}(\theta z^2, 1).$$



Blowfly data, again.

Summary

- **K2-ABC**
  - A dissimilarity criterion for ABC based on MMD between empirical distributions of observed and simulated data
  - No "information loss" due to insufficient statistics.
  - Simple and effective when parameters model marginal distribution of observations.
  - Can be thought of as kernel smoothing (Nadaraya-Watson) on the space of embeddings of empirical distributions.
- **DR-ABC**
  - When constructing a summary statistic optimal with respect to a certain loss function, supervised learning from data to parameter space can be used.
  - Distribution regression, i.e. kernel ridge regression on the space of embeddings, and conditional distribution regression natural in this context.
  - Flexible framework which allows application to time series, group-structured or spatial observations, dynamic systems etc.