Optimal Quantisation of Probability Measures Using Maximum Mean Discrepancy

ONUR TEYMUR

University of Kent & Alan Turing Institute

Institut Henri Poincaré

2021

Optimal Quantisation of Probability Measures Using Maximum Mean Discrepancy

PMLR 130:1027–1035 / arXiv 2010.07064

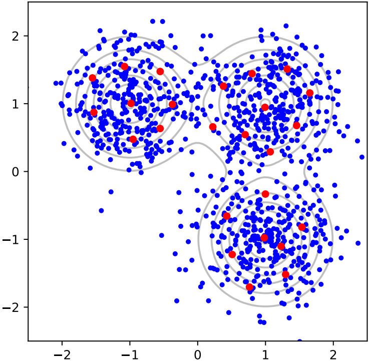
Jackson Gorham
Whisper.ai

Marina Riabiz
King's College London
Alan Turing Institute

Chris Oates
Newcastle University
Alan Turing Institute

CORE AIM

Aim: to `optimally approximate' a probability measure $P$ on $\mathcal{X}$ by a discrete distribution $Q_m = \frac{1}{m}\sum_{i=1}^{m}\delta(X_i)$. Our starting point is a *discrete* set $X_1, \ldots, X_n$ of points in $\mathcal{X}$ ($n \gg m$) that we aim to `thin'.



BASIC PROBLEM SETUP

We have: a target distribution P, and a set of samples X_i – from the samples we form an empirical distribution Q_n:

$$P \qquad\qquad Q_n := \frac{1}{n}\sum_{i=1}^{n}\delta(X_i)$$

But: n is large and it is expensive to compute with all n samples.
Or: Q_n is not actually that good an approximation to P.

So: We choose $m \ll n$ and then try to find a representative subset of samples of size m that minimises

$$\text{“difference”}\left(P\,,\ \frac{1}{m}\sum_{i=1}^{m}\delta(X_i)\right) \quad \text{given } m\,;\ \ \{X_i\}_{i=1}^{m} \subset \{X_i\}_{i=1}^{n}$$

BASIC PROBLEM SETUP

We refer to this as optimal quantisation.

Our task will be to define some appropriate measure of ``difference" to do this, and thereby to find an appropriate representative subset.

BASIC PROBLEM SETUP

Note: We are concerned with (somehow) minimising:

$$\text{“difference”}\left(P\,,\ \frac{1}{m}\sum_{i=1}^{m}\delta(X_i)\right)\quad \text{given } m\,;\ \ \{X_i\}_{i=1}^{m}\subset\{X_i\}_{i=1}^{n}$$

This is not the same as:

- $\text{“diff.”}\left(\frac{1}{n}\sum_{i=1}^{n}\delta(X_i)\,,\ \frac{1}{m}\sum_{i=1}^{m}\delta(X_i)\right)\ ;\ \{X_i\}_{i=1}^{m}\subset\{X_i\}_{i=1}^{n}$
- $\text{“diff.”}\left(P\,,\ \frac{1}{m}\sum_{i=1}^{m}\delta(X_i)\right)\ ;\ \{X_i\}_{i=1}^{m}\subset\mathcal{X}$

BASIC PROBLEM SETUP

$$\underset{\substack{S\subset\{1,\ldots,n\}\\|S|=m}}{\operatorname{argmin}}\ \mathrm{MMD}\left(P\,,\ \frac{1}{m}\sum_{i=1}^{m}\delta(X_i)\right)$$

SOME RELATED APPROACHES

- Minimise Wasserstein distance [Graf & Luschgy 2007]
- Minimise `power function' (worse-case interpolation error) [de Marchi et al. 2005; Santin & Haasdonk 2017]
- Support points (minimise `energy distance') [Mak & Joseph 2018]
- Minimise Stein discrepancy to optimally *weight* $\{X_i\}_{i=1}^n$ [Liu & Lee 2017]
- Minimum energy designs [Joseph 2015,2019]
- ``Kernel herding'' [Chen et al. 2010; Lacoste-Julien et al. 2015]
- ``Stein points'' [Chen et al. 2018]
- Stein Variational Gradient Descent [Liu & Wang 2019; Duncan et al. 2018]
- ``Kernel Thinning'' [Dwivedi & Mackey 2021]
- ``Cube Thinning'' [Chopin & Ducrocq 2021]

OUTLINE

- Motivating examples
- Introduction to Maximum mean discrepancy (MMD)
- Algorithms for selecting points
- Some theory
- Heuristics

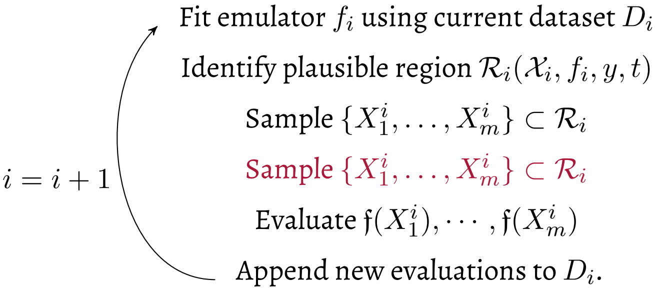
EXAMPLE 1: HISTORY MATCHING

Core idea: establish which inputs $x \in \mathcal{X}$ to a computer model $\mathfrak{f}(\cdot)$ could have generated a known output y.

- Construct a *surrogate* or *emulator* f trained on evaluations of $\mathfrak{f}$ – use this to determine which x are compatible with y, while accounting for uncertainty introduced by the emulation.
- Fix a t, and say x is *plausible* if $\|\mathbb{C}[f(x)]^{-1/2}(\mathbb{E}[f(x)] - y)\| < t$.
- This defines a *plausible region*

$$\mathcal{R}(\mathcal{X}, f, y, t) = \{x \in \mathcal{X} \ : \ \|\mathbb{C}[f(x)]^{-1/2}\{\mathbb{E}[f(x)] - y\}\| < t\}$$

EXAMPLE 1: HISTORY MATCHING



Fit emulator f_i using current dataset D_i

Identify plausible region $\mathcal{R}_i(\mathcal{X}_i, f_i, y, t)$

Sample $\{X_1^i, \ldots, X_m^i\} \subset \mathcal{R}_i$

Sample $\{X_1^i, \ldots, X_m^i\} \subset \mathcal{R}_i$

Evaluate $\mathfrak{f}(X_1^i), \cdots, \mathfrak{f}(X_m^i)$

Append new evaluations to D_i.

$i = i + 1$

EXAMPLE 1: HISTORY MATCHING

In practical settings $\mathcal{R}_i$ can be very complicated – non-convex, highly curved boundaries, disconnected regions etc.

Checking whether a point x is in $\mathcal{R}_i$ is easy. But characterising the entire region is very difficult.

Depending on which statistical model one uses for f, there might be different optimal ways of selecting the new points (Uniform sampling? Space-filling design? Sobol' sequence? ...)

Some of these may be difficult to do given the form of $\mathcal{R}_i$ (Taking the intersection of an existing low-discrepancy set with $\mathcal{R}_i$? Does this guarantee anything?)

EXAMPLE 1: HISTORY MATCHING

Many good reasons to use MMD instead:

- it doesn't care about the region $\mathcal{R}_i$ being really complicated,
- it can `correct' statistical inaccuracies caused by having to perform rejection sampling when first identifying $\mathcal{R}_i$,
- theoretical guarantees in some settings:

  If f is a Gaussian process (very common), a natural quantification of the uncertainty present is via the *maximum eigenvalue of its integrated covariance*

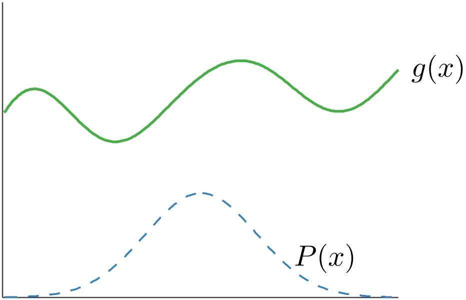
  $$U(\Sigma, \mathcal{X}) = \lambda_{\max}\left\{\iint_{\mathcal{X}\times\mathcal{X}} \Sigma(x, x')\mathrm{d}\mathcal{U}(x)\mathrm{d}\mathcal{U}(x')\right\},$$

  and using MMD sampling we have $U(\Sigma, \mathcal{X}) = O(m^{-1})$.

EXAMPLE 2: BAYESIAN QUADRATURE

Let g be a function that we would like to integrate against P.

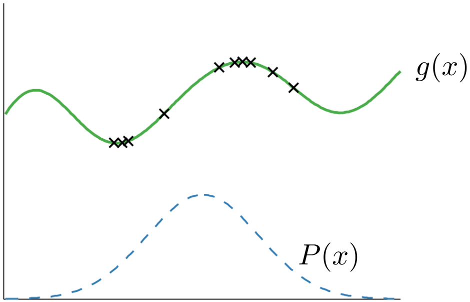
$$Z = \int g(x)\,\mathrm{d}P(x)$$



EXAMPLE 2: BAYESIAN QUADRATURE

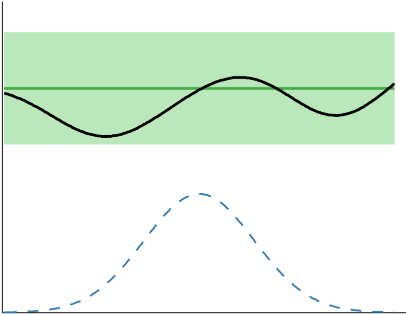
Let g be a function that we would like to integrate against P.

$$Z \approx \frac{1}{n} \sum_{i=1}^{n} g(X_i) \, , \quad (X_i \sim P)$$



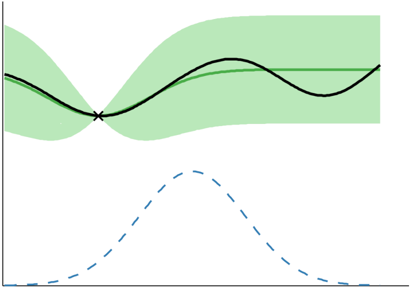
EXAMPLE 2: BAYESIAN QUADRATURE

Instead: Model $g(x)$ *a priori* as a Gaussian process with covariance k, then condition on `data' $\mathcal{D} = \{g(X_i)\}_{i=1}^n$.



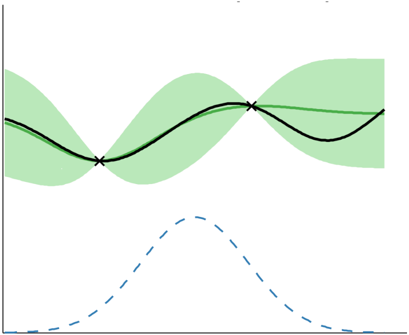
EXAMPLE 2: BAYESIAN QUADRATURE

Instead: Model $g(x)$ *a priori* as a Gaussian process with covariance k, then condition on `data' $\mathcal{D} = \{g(X_i)\}_{i=1}^n$.



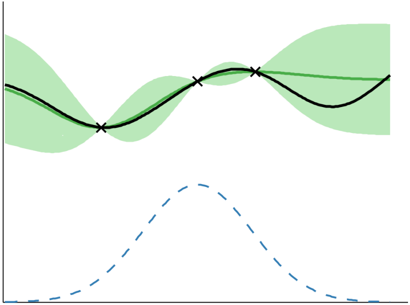
EXAMPLE 2: BAYESIAN QUADRATURE

Instead: Model $g(x)$ *a priori* as a Gaussian process with covariance k, then condition on `data' $\mathcal{D} = \{g(X_i)\}_{i=1}^{n}$.



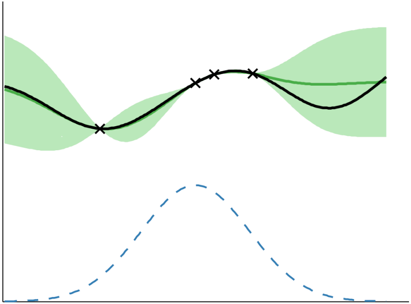
EXAMPLE 2: BAYESIAN QUADRATURE

Instead: Model $g(x)$ *a priori* as a Gaussian process with covariance k, then condition on `data' $\mathcal{D} = \{g(X_i)\}_{i=1}^{n}$.



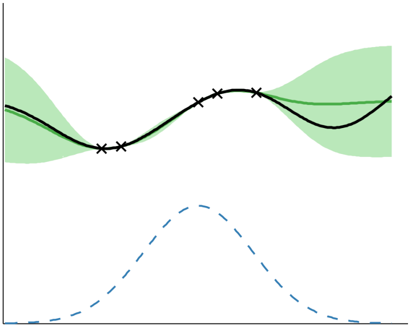
EXAMPLE 2: BAYESIAN QUADRATURE

Instead: Model $g(x)$ *a priori* as a Gaussian process with covariance k, then condition on `data' $\mathcal{D} = \{g(X_i)\}_{i=1}^n$.



EXAMPLE 2: BAYESIAN QUADRATURE

Instead: Model $g(x)$ *a priori* as a Gaussian process with covariance k, then condition on `data' $\mathcal{D} = \{g(X_i)\}_{i=1}^{n}$.



EXAMPLE 2: BAYESIAN QUADRATURE

This gives a *probabilistic* approximation to Z – its (Gaussian) posterior distribution $p(Z|\mathcal{D})$.

$$\mathrm{std}[Z|\mathcal{D}] = \min_{\substack{w_1,\ldots,w_m \\ \in \mathbb{R}}} \mathrm{MMD}_{P,k}\left(\sum_{i=1}^{m} w_i \delta(X_i)\right)$$

This is bounded above by $\mathrm{MMD}_{P,k}\left(\frac{1}{m}\sum_{i=1}^{m} \delta(X_i)\right)$

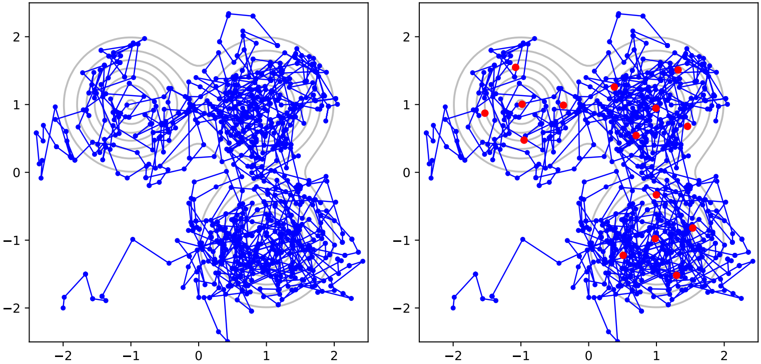
[Huszár & Duvenaud (2012), Briol et al. (2015)]

EXAMPLE 2: BAYESIAN QUADRATURE

It's quite common here that g is complicated and expensive to evaluate, but P is something straightforward like a Gaussian or uniform distribution (against which the integration of k required to calculate MMD is easy).

So MMD can be tractable and useful in this setting.

EXAMPLE 3: THINNING OF MCMC OUTPUT



See Marina's talk :)

MAXIMUM MEAN DISCREPANCY

Let
- $\mathcal{X}$ be a measurable space,
- $\mathcal{P}(\mathcal{X})$ be the set of probability distributions on $\mathcal{X}$,
- $\mathcal{F}$ be some set of bounded real-valued functions on $\mathcal{X}$.

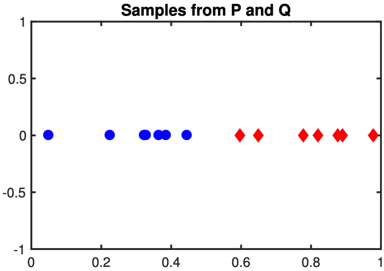
For $P, Q \in \mathcal{P}(\mathcal{X})$, a discrepancy is a quantity of the form

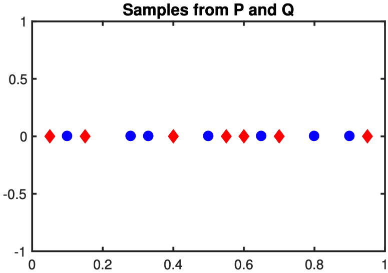
$$D_{\mathcal{F}}(P,Q) = \sup_{f \in \mathcal{F}} \left| \int f \, \mathrm{d}P - \int f \, \mathrm{d}Q \right|$$

(If $D_{\mathcal{F}}(P,Q) = 0$ implies $P = Q$ then $\mathcal{F}$ is called *measure-determining*, and $D_{\mathcal{F}}$ is also called an integral probability metric.)

MAXIMUM MEAN DISCREPANCY – INTUITION

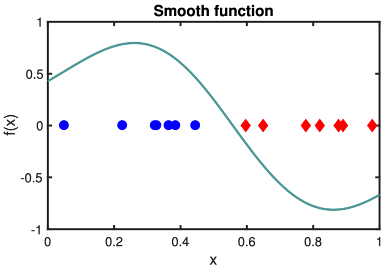
(Images borrowed/looted from Arthur Gretton)

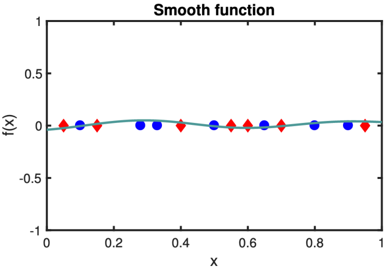




$$\left|\int f\,\mathrm{d}P - \int f\,\mathrm{d}Q\right|$$

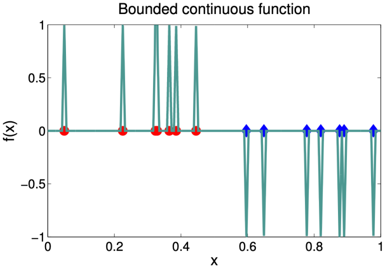
MAXIMUM MEAN DISCREPANCY – INTUITION

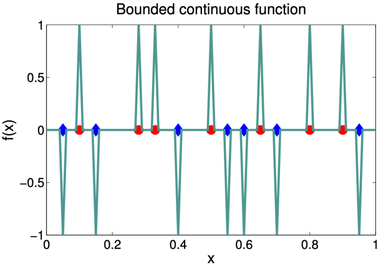

Smooth function


Smooth function

$$\sup_{f \in \mathcal{F}_1} \left| \int f \, \mathrm{d}P - \int f \, \mathrm{d}Q \right|$$

MAXIMUM MEAN DISCREPANCY – INTUITION





$$\sup_{f\in\mathcal{F}_2}\left|\int f\,\mathrm{d}P - \int f\,\mathrm{d}Q\right|$$

MAXIMUM MEAN DISCREPANCY

Recall the general form of an integral probability metric:

$$D_{\mathcal{F}}(P,Q) = \sup_{f\in\mathcal{F}} \left| \int f \,\mathrm{d}P - \int f \,\mathrm{d}Q \right|$$

Choose a kernel k and consider its RKHS $\mathcal{H}(k)$.

$\Bigg($ A kernel is a symmetric, positive definite function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$. k *reproduces* a Hilbert space $\mathcal{H}(k) : \mathcal{X} \to \mathbb{R}$ for which (i) for all $x \in \mathcal{X}$, $k(\cdot, x) \in \mathcal{H}(k)$, and (ii) for all $x \in \mathcal{X}$ and $f \in \mathcal{H}(k)$, $\langle k(\cdot,x), f\rangle_{\mathcal{H}(k)} = f(x)$. $\Bigg)$

Let $\mathcal{B}(k) = \{f \in \mathcal{H}(k) : \langle f, f\rangle_{\mathcal{H}(k)} \leq 1\}$ be the unit ball in $\mathcal{H}(k)$.

Setting $\mathcal{F}$ to be $\mathcal{B}(k)$ defines the maximum mean discrepancy.

MAXIMUM MEAN DISCREPANCY

Why this choice?

- Opportunity to easily enforce different degrees of smoothness through choice of the kernel k.
- If P and/or Q are empirical distributions, we can write $\mathcal{D}_{\mathcal{H}}(k)(P,Q)$ in closed form using only kernel evaluations.
- Furthermore in this setting $\mathcal{D}_{\mathcal{B}(k)}(P_n, Q_m)$ is a consistent estimator of $\mathcal{D}_{\mathcal{B}(k)}(P,Q)$ and the rate of convergence is *independent* of d.
- Since kernels can be defined on arbitrary domains $\mathcal{X}$, MMD can used to measure distances between measures on eg. graphs, strings, etc., (not just $\mathbb{R}^d$).

OTHER USES OF MMD

Has been used for (amongst other things):

- Hypothesis testing [Fukumizu et al. 2008; Gretton et al. 2012; Doran et al. 2014, Chwialkowski and Gretton 2014]
- Density estimation [Song et al. 2007,2008; Sriperumbudur 2011]
- Clustering [Jegelka et al. 2009]
- Causal discovery [Sgouritsa et al. 2013; Chen et al. 2014; Schölkopf et al. 2015]
- Statistical model criticism [Lloyd & Ghahramani 2015; Kim et al. 2016]
- MCMC [Sejdinovic et al. 2014]
- ABC [Park et al. 2016]
- Training generative models [Li et al. 2015; Dziugaite et al. 2015]

MAXIMUM MEAN DISCREPANCY

MMD can be written *in closed form* (without the supremum).

$$D_{\mathcal{B}(k)}(P,Q)^2 = \iint k(x,y)\,\mathrm{d}P(x)\,\mathrm{d}P(y) \\ - 2\iint k(x,y)\,\mathrm{d}P(x)\,\mathrm{d}Q(y) + \iint k(x,y)\,\mathrm{d}Q(x)\,\mathrm{d}Q(y)$$

With $Q = \dfrac{1}{m}\displaystyle\sum_{i=1}^{m} \delta(X_i)$, this becomes

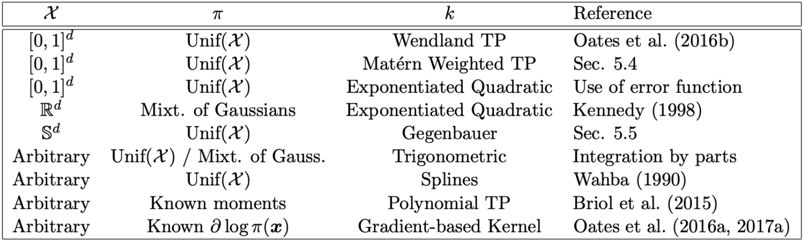
$$\mathrm{MMD}_{P,k}(Q)^2 = \frac{1}{m^2}\sum_{i,j=1}^{m} k(X_i, X_j) - \frac{2}{m}\sum_{i=1}^{m}\int k(X_i, x)\,\mathrm{d}P(x) \\ + \iint k(x,y)\,\mathrm{d}P(x)\,\mathrm{d}P(y)$$

TRACTABILITY OF MMD

``But what about the integral with respect to P?"

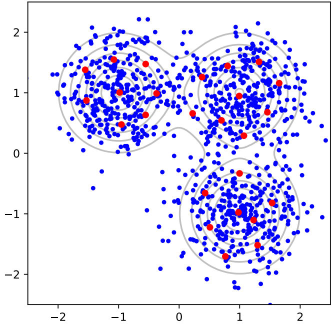
For general or unknown P is this not usually possible.

- In many useful cases, combinations of P and k *are* in fact tractable. (table below from [Briol et al. 2015])
- Otherwise consider using Kernel Stein Discrepancy instead.



$\mathcal{X}$	π	k	Reference
$[0,1]^d$	Unif($\mathcal{X}$)	Wendland TP	Oates et al. (2016b)
$[0,1]^d$	Unif($\mathcal{X}$)	Matérn Weighted TP	Sec. 5.4
$[0,1]^d$	Unif($\mathcal{X}$)	Exponentiated Quadratic	Use of error function
$\mathbb{R}^d$	Mixt. of Gaussians	Exponentiated Quadratic	Kennedy (1998)
$\mathbb{S}^d$	Unif($\mathcal{X}$)	Gegenbauer	Sec. 5.5
Arbitrary	Unif($\mathcal{X}$) / Mixt. of Gauss.	Trigonometric	Integration by parts
Arbitrary	Unif($\mathcal{X}$)	Splines	Wahba (1990)
Arbitrary	Known moments	Polynomial TP	Briol et al. (2015)
Arbitrary	Known $\partial \log \pi(\boldsymbol{x})$	Gradient-based Kernel	Oates et al. (2016a, 2017a)

BACK TO OUR PROBLEM



Given $m \ll n$, we'd like to find a minimiser of $\text{MMD}_{P,k}(Q)$ over size-m subsets of $\{X_1, \ldots, X_n\}$.

$$\underset{\substack{S\subset\{1,\ldots,n\}\\|S|=m}}{\operatorname{argmin}} \ \text{MMD}_{P,k}\left(\frac{1}{m}\sum_{i=1}^{m}\delta(X_i)\right)$$

A SIMPLE SEQUENTIAL ALGORITHM

Given a set of samples $X_1, \ldots, X_{i-1}$ that we use to form a measure $Q_{i-1} = \frac{1}{i-1} \sum_{j=1}^{i-1} \delta(X_j)$ that minimises $\mathrm{MMD}_{P,k}(Q_{i-1})$ over all possible size $(i-1)$ subsets, we select for the next point X_i that which minimises $\mathrm{MMD}_{P,k}(Q_i)$, where $Q_i = \frac{1}{i} \sum_{j=1}^{i} \delta(X_j)$.

Notation: The indices of the points we select will be written:

$$\pi(1), \pi(2), \ldots, \pi(m) \quad , \quad \pi(\cdot) \in \{1, \ldots, n\}$$

A SIMPLE SEQUENTIAL ALGORITHM

$$M_i = \tfrac{1}{i^2} \sum_{j,j'}^{i} k(X_j, X_{j'}) - \tfrac{2}{i} \sum_{j}^{i} \int k(X_j, y)\, \mathrm{d}P(y) + \iint k(x,y)\, \mathrm{d}P(x)\, \mathrm{d}P(y)$$

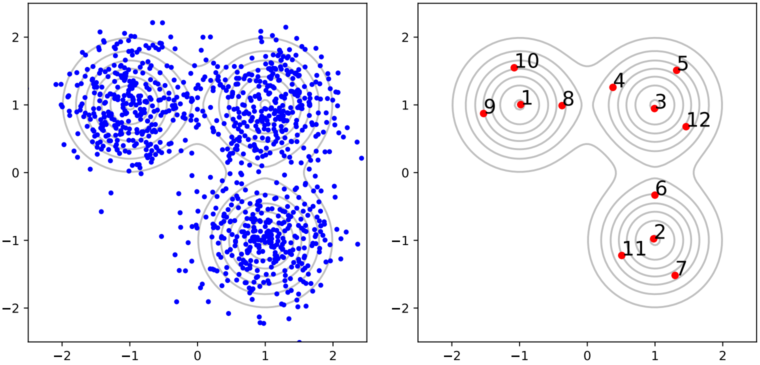
$$M_{i-1} = \tfrac{1}{(i-1)^2} \sum_{j,j'}^{i-1} k(X_j, X_{j'}) - \tfrac{2}{i-1} \sum_{j}^{i-1} \int k(X_j, y)\, \mathrm{d}P(y) + \iint k(x,y)\, \mathrm{d}P(x)\, \mathrm{d}P(y)$$

$$\begin{aligned} M_i - M_{i-1} = \left(\tfrac{1}{i^2} - \tfrac{1}{(i-1)^2}\right) \sum_{j,j'}^{i-1} k(X_j, X_{j'}) + \tfrac{2}{i^2} \sum_{j}^{i-1} k(X_j, X_i) + \tfrac{1}{i^2} k(X_i, X_i) \\ - \left(\tfrac{2}{i} - \tfrac{2}{i-1}\right) \sum_{j}^{i-1} \int k(X_j, y)\, \mathrm{d}P(y) - \tfrac{2}{i} \int k(X_i, y)\, \mathrm{d}P(y) \end{aligned}$$

A SIMPLE SEQUENTIAL ALGORITHM

$$\pi(i) \in \underset{j \in \{1,\ldots,n\}}{\operatorname{argmin}} \left[\frac{1}{2} k(X_j, X_j) + \sum_{i'=1}^{i-1} k(X_{\pi(i')}, X_j) - i \int k(x, X_j) \mathrm{d}P(x) \right]$$

A SIMPLE SEQUENTIAL ALGORITHM



A SIMPLE SEQUENTIAL ALGORITHM

Issues:

- This algorithm is greedy. (It scans through, and calculates with, all n points at each iteration).

  This makes it (potentially) expensive.

- This algorithm is myopic. (It chooses the next point optimally, but this may not be the best long-term strategy).

  This makes it (potentially) inaccurate.

NON-MYOPIC ALGORITHM

What if we chose more than one point simultaneously?

- Greater statistical efficiency?
- Computationally favourable? (Or if not: acceptable overhead?)
- Can we implement it cleverly?

NON-MYOPIC ALGORITHM

Choose s points simultaenously. Write the index of the i'th point within iteration j as $\pi(i,j)$, i.e. $\pi(i,\cdot) \in \{1,\ldots,n\}^s$. Then pick:

index set of size s

$$\pi(i,\cdot) \in \operatorname*{argmin}_{S\in\{1,\ldots,n\}^s} \Bigg[\frac{1}{2}\sum_{j,j'\in S} k(X_j, X_{j'}) \\ + \sum_{i'=1}^{i-1}\sum_{j=1}^{s}\sum_{j'\in S} k(X_{\pi(i',j)}, X_{j'}) - is\sum_{j\in S}\int k(x, X_j)\mathrm{d}P(x)\Bigg]$$

INTEGER QUADRATIC PROGRAMMES

We can rewrite this problem as an

integer quadratic programme (IQP),

and in doing so use state-of-the-art discrete optimisation codes.

INTEGER QUADRATIC PROGRAMMES

Let $v \in \{0, \ldots, s\}^n : \sum_{j=1}^{n} v_j = s$ be a vector listing the number of copies of each sample that are selected at iteration i.

Algorithm chooses: $\{X_5, X_6, X_{10}, X_5, X_3\}$

$$\begin{array}{rccccccccccl} & X_1 & X_2 & X_3 & X_4 & X_5 & X_6 & X_7 & X_8 & X_9 & X_{10} & \\ v = (& 0 & 0 & 1 & 0 & 2 & 1 & 0 & 0 & 0 & 1 &) \end{array}$$

$(n = 10, s = 5)$

INTEGER QUADRATIC PROGRAMMES

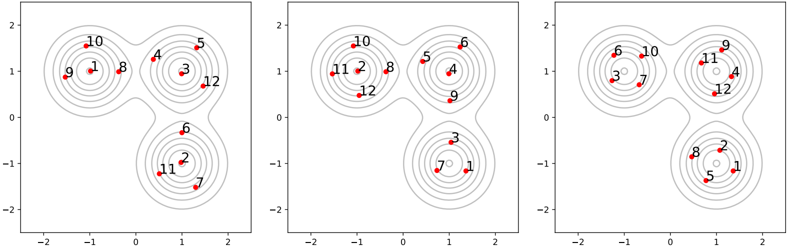
$$\operatorname*{argmin}_{S\in\{1,\dots,n\}^s}\left[\frac{1}{2}\sum_{j,j'\in S}k(X_j,X_{j'})+\sum_{i'}^{i-1}\sum_{j}^{s}\sum_{j'\in S}k(X_{\pi(i',j)},X_{j'})-is\sum_{j\in S}\int k(x,X_j)\mathrm{d}P(x)\right]$$

$$\operatorname*{argmin}_{v\in\mathbb{N}_0^s}\frac{1}{2}v^\top Kv+c^{i\top}v\quad\text{such that}\quad\mathbf{1}^\top v=s$$

$$K_{j,j'}:=k(X_j,X_{j'}),\quad\mathbf{1}_j:=1\text{ for }j=1,\dots,n,$$

$$c_j^i:=\sum_{i'=1}^{i-1}\sum_{j'=1}^{s}k(X_{\pi(i',j')},X_j)-is\int k(x,X_j)\,\mathrm{d}P(x)$$

NON-MYOPIC SELECTION

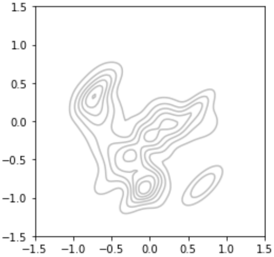


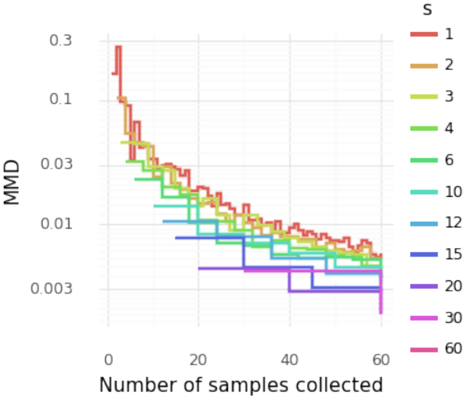
1 point at a time

4 points at a time

12 points together

NON-MYOPIC SELECTION





BATCH ALGORITHM

Myopic algorithm is $O(nm^2)$, non-myopic algorithm is $O(n^s(ms)^2)$.

In both, the algorithm scans all n possible points at every iteration.

We can mini-batch the candidate set $\{X_1, \ldots, X_n\}$ and retain $b \ll n$ at each iteration, then choose $s > 1$ samples from each batch of b.

This approach has complexity $O(b^s(ms)^2)$. In practice we find this makes it tractable in many settings.

1:

Let $\{X_i\}_{i=1}^n \subset \mathcal{X}$ be fixed. Consider an index sequence π of length m and with selection size s. Then for all $m \geq 1$ there is a C such that

$$\mathrm{MMD}_{P,k}\left(\frac{1}{ms}\sum_{i=1}^{m}\sum_{j=1}^{s}\delta(X_{\pi(i,j)})\right)^2 \leq \underbrace{\min_{\substack{1^\top w=1\\ w_i\geq 0}} \mathrm{MMD}_{P,k}\left(\sum_{i=1}^{n} w_i\delta(X_i)\right)^2}_{\substack{\text{optimal (weighted) quantisation}\\ \text{of } P \text{ achievable with the candidate set}}} + C^2\left(\frac{1+\log m}{m}\right)$$

2:

Let $\{X_i\}_{i=1}^n \subset \mathcal{X}$ be independently sampled from P. Consider an index sequence π of length m and with selection size s. Then for all $s \in \mathbb{N}$ and all $m, n \geq 1$, there are constants C, C', γ such that

$$\mathbb{E}\left[\mathrm{MMD}_{P,k}\left(\frac{1}{ms}\sum_{i=1}^{m}\sum_{j=1}^{s}\delta(X_{\pi(i,j)})\right)^2\right] \\ \leq \frac{\log(C')}{n\gamma} + 2\left(C^2 + \frac{\log(nC')}{\gamma}\right)\left(\frac{1+\log m}{m}\right).$$

3:

Consider a P-invariant, time-homogeneous, reversible Markov chain $\{X_i\}_{i\in\mathbb{N}} \subset \mathcal{X}$. Consider an index sequence π of length m and selection subset size s. Then there are constants C, C', C'', γ such that

$$\mathbb{E}\left[\mathrm{MMD}_{P,k}\left(\frac{1}{ms}\sum_{i=1}^{m}\sum_{j=1}^{s}\delta(X_{\pi(i,j)})\right)^2\right]$$
$$\leq \frac{\log(C')}{n\gamma} + \frac{C''}{n} + 2\left(C^2 + \frac{\log(nC')}{\gamma}\right)\left(\frac{1+\log m}{m}\right).$$

4:

Let each mini-batch $\{X^i_j\}_{j=1}^b \subset \mathcal{X}$ be independently sampled from μ. Consider an index sequence π of length m. Then $\forall\, m, n \geq 1$ there are constants C, C' such that

$$\mathbb{E}\left[\mathrm{MMD}_{P,k}\left(\frac{1}{ms}\sum_{i=1}^{m}\sum_{j=1}^{s}\delta(X^i_{\pi(i,j)})\right)^2\right] \\ \leq \frac{\log(C')}{b\gamma} + 2\left(C^2 + \frac{\log(bC')}{\gamma}\right)\left(\frac{1+\log m}{m}\right).$$

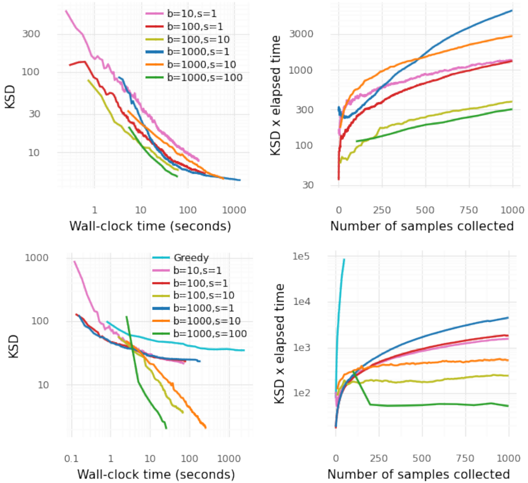
REMARKS & LACUNAE

- Bounds are all independent of s. (Does not necessarily imply that $s = 1$ is optimal; indeed experiments show otherwise.)

What's missing:

- Mini-batch result in dependent sampling context. Seems achievable but technically involved.
- Different regimes of mini-batching. (ie. non-independent mini-batches). This seems harder.
- Output from non P-stationary Markov chains. (ie. chains that have not yet converged.)

EXPERIMENTS & HEURISTICS



Optimal Quantisation of Probability Measures
Using Maximum Mean Discrepancy
PMLR 130:1027–1035 / arXiv 2010.07064

www.teymur.uk