Kernel Thinning

2/26/2024

Geometry @ uw mmp

$\subseteq$ finding core-sets

$\neq$ data thinning !!! > sample splitting

Problem: approximate distribution μ on $\mathcal{X}$ by a sample

—n— a large sample μ_N by smaller sample

$y_{1:N} \sim iid\ \mu$

empirical distribution

$Q = \{x_{1:n} \in \mathcal{X}\} \longrightarrow \mu_Q$ = approximator

so that $\mu_Q \approx \mu_N \approx \mu$

Plan

- Kernel Thinning problem
- Ideas for solving it ← Herding, K. Thinning, weighted Quad.
- (Basic) RKHS facts ← geometric
  Important / useful
- The methods : H, KT, WQ (if time)

Problem: approximate distribution μ on $\mathcal{X}$ by a sample

—"— a large sample μ_N by smaller sample

$y_{1:N} \sim iid\ \mu$ → empirical distribution

$Q = \{x_{1:n} \in \mathcal{X}\} \longrightarrow \mu_Q$ = approximator

so that $\mu_Q \approx \mu_N \approx \mu$ **More precisely** → $\mu f \approx \mu_N f \approx \mu_Q f$ for all $f \in \mathcal{H}_k$ (RKHS) uniformly

$$\mu f \equiv E_\mu[f(x)]$$

More precisely

- $MMD_k(\nu, \mu) = \sup_{\|f\|_{\mathcal{H}_k} \leq 1} |\nu f - \mu f|$

Maximum Mean Discrepancy

Wanted $Q = \{x_{1:n}\}$ so that

$$MMD_k(\mu_N, \mu_Q) < \varepsilon_n$$

$$MMD_k(\mu, \mu_Q) < \varepsilon_n$$

Why RKHS ?

- compact specification of $\{f's\}$ - no need to enumerate!
- nice, elegant, well understood math : Wahba
  - complete
  - normed
  - "Euclidean" (intuitive)
  - tractable computations
  - non-parametric
  - general
  - ....
- for ex:
  - polynomials $k = (1 + x^T x')^d$
  - kernel regr, classification
    - k = Gaussian, Matérn, ...
  - string, tree kernels

Why core-sets / thinning ?

want to do ML - with large data
- non-parametric
- f not known yet

$\Rightarrow$ save computation time
tractability

$\rightsquigarrow$ Kernel $k(\,)$ must match problem !!

Why expectations $E_\mu[f]$

many functionals (criteria of quality) are expectations
- $P[\text{err}]$
- $E[\text{err}^2]$
- quantile

Problem (rephrased)

given
- $y_{1:N} = \mu_N$ large sample
- kernel k = RKHS $\mathcal{H}_k$

wanted
- $x_{1:n} \equiv \mu_Q$ small sample

so that $\mathrm{MMD}_k(\mu_N, \mu_Q) \leq \varepsilon_n$

What is a reasonable ε_n?

- $\mu_N \sim iid\ \mu$

  $\mu_N \approx \mu \Rightarrow$ want same rate for $\mu_Q \approx \mu_N$

- $E_\mu[x] = \mu \in \mathcal{H}_k$

$$\hat{\mu} = \frac{1}{N}\sum_{i=1}^{N} y_i \rightarrow \mathrm{Var}\,\hat{\mu} \propto \frac{1}{N}$$

$$\mathrm{std}\,\hat{\mu} \propto \frac{1}{\sqrt{N}} = N^{-1/2}$$

$N^{-1/2}$ ← benchmark rate

MMD_k rate no faster than $\hat{\mu}$ rate

Problem (rephrased)

given • $y_{1:N} \equiv \mu_N$ large sample

• kernel $k = \text{RKHS } \mathcal{H}_k$

wanted • $x_{1:n} \equiv \mu_Q$ small sample

so that $\text{MMD}_k(\mu_N, \mu_Q) \leq \varepsilon_n$

Solutions

- grid on $\mathcal{X}$ ← optimal N^{-2r}
- MC = iid sampling $N^{-1/2}$
- MCMC $\sim N^{-\alpha}$ $\alpha < \frac{1}{2}$
- Quasi MC $\sim N^{-\alpha'}$ $\alpha' > \alpha$
- Herding
- DPP (Determinantal Point P)
- Thinning
- Weighted Quadrature

dependent (not iid)

neg. corr.

Problem (rephrased)

given • $y_{1:N} \equiv \mu_N$ large sample

• kernel k = RKHS $\mathcal{H}_k$

wanted • $x_{1:n} \equiv \mu_Q$ small sample

so that $MMD_k(\mu_N, \mu_Q) \leq \varepsilon_n$

→ • all select subsample Q of $y_{1:N}$

→ • $n = \sqrt{N}$

rate $\sim n^{-1} = N^{-1/2}$ but with $\sqrt{N}$ points

Solutions

- grid on $\mathcal{X}$ ← optimal N^{-2r}
- MC = iid sampling $N^{-1/2}$
- MCMC $\sim N^{-\alpha}$ $\alpha < \frac{1}{2}$
- Quasi MC $\sim N^{-\alpha'}$ $\alpha' > \alpha$
- Herding [Yutian Chen, Welling, Smola '12] (sequential, greedy)
- DPP (Determinantal Point P) [Belhadji ...]
- Thinning [Dwivedi, Mackey 21] (recursive vector balancing (paired comparisons) + square root kernel)
- Weighted Quadrature [Hayakawa, Oberhauser, Lyons 21] (weighted sample RKHS approximation + cubature algorithm (Caratheodory))

What makes thinning (core-set) pb. hard? $\varepsilon_n \sim c n^{-1}$

rate of decay of kernel, μ

For ex:

μ compact support
$\uparrow$
μ subgaussian
$\uparrow$
μ heavy tailed

- similar for k : compact support, ...

Weighted Quadrature : $(\sigma_{1,2,\dots}, e_{1,2,\dots}(x))$ = e-values, e-functions of operator

$$(kf)(x) = \int_{\mathcal{X}} f(x') k(x', x) \, d\mu(x')$$ ← typically convolution op.

$\Rightarrow$ err depends on σ_n, $r_{n+1} = \sum_{j=n+1}^{\infty} \sigma_j$ (residual spectrum)

Kernel thinning

- $R_{k,n} = \inf \{ r : \sup_{\|x-y\|_2 \geq r} |k(x,y)| \leq \frac{1}{n} \|k\|_\infty \}$ ← kernel decay radius

- $\tau_k(r) = \sup_x \int_{\|y\|_2 \geq r} k^2(x, x-y)\, dy$ ← tail weight of $k^2(\cdot)$ at r

Ex: $k(x,y) = N(x-y; 0, \sigma^2 I) \Rightarrow \tau_k(r) = \int_{\|y\| \geq r} e^{-2 \frac{\|x-(x-y)\|^2}{2\sigma^2}} dy$ = tail prob for $N(0, \frac{\sigma^2}{2} I)$ at $r \cdot \sqrt{2}$

Gaussian kernel

for $\sigma = 1$: $\|k\|_\infty = \frac{1}{(2\pi)^{d/2}}$

$\|x-y\| \geq r \Rightarrow k(x,y) \leq \frac{1}{(2\pi)^{d/2}} e^{-\frac{r^2}{2}}$

$\Rightarrow e^{-\frac{R^2}{2}} = \frac{1}{n} \Rightarrow \boxed{R_{k,n} = \sqrt{2 \ln n}}$

It helps if these are known analytically ← otherwise

or exact, tractably

- kernel tail probabilities $\bar{c}_k(r)$ computable — they must be approximated, majorized, ...

  decay radii $R_{k,n}$

  (+ decay rates of μ)

  (Thinning)

- $\bar{\varphi} = E_\mu[\varphi(x)]$ or $\langle z, \bar{\varphi} \rangle_k \in \mathbb{R}$

  $\bar{\varphi} \in \mathcal{H}_k$; $z \in \mathcal{H}_k$

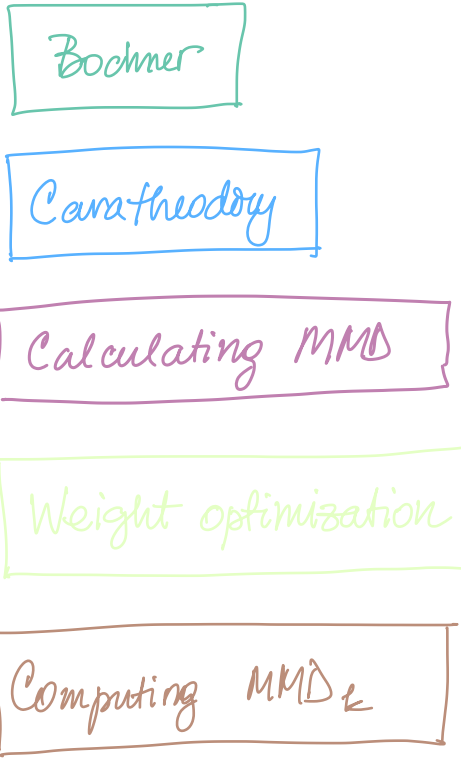
  (Herding)

  $\max_x \langle z, \varphi(x) \rangle$; $z \in \mathcal{H}_k$

- (Bochner) Fourier repn of $k(,)$ → Thinning $k_{1/2}$

  Spectrum of operator k $(\sigma_{1,2,..m}, e_{1,2,..m}(x))$

  → weighted Quadrature


Bochner
Caratheodory
Calculating MMD
Weight optimization
Computing MMD_k

Bochner

Let $k(x,y) \equiv k(x-y)$ continuous kernel on $\mathbb{R}^d$

($k(x-y)$: shift invariant)

Then

$$k(z) = \int_{\mathbb{R}^d} p(\omega)\, e^{-i\omega^T z}\, d\omega$$

Fourier transform of non-negative measure p

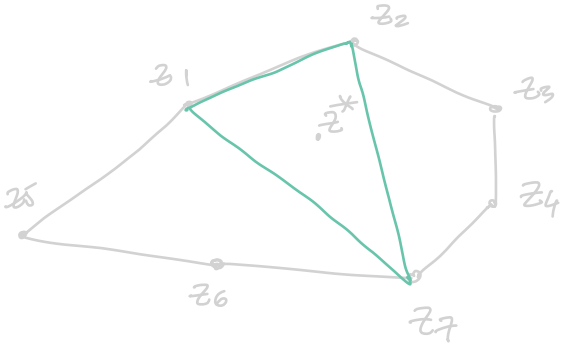
For ex:

$k(z)$	$p(\omega)$	(up to normalization)
$e^{-\|z\|^2/2}$	$e^{-\|\omega\|^2/2}$	Gaussian
$e^{-\|z\|_1}$	$\prod_{j=1}^{d} \frac{1}{1+\omega_j^2}$	Laplace
$\prod_{j=1}^{d} \frac{1}{1+z_j^2}$	$e^{-\|\omega\|_1}$	Product kernel

Caratheodory

Thm [Caratheodory] $z_{1:N} \in \mathbb{R}^{d-1}$, $z^* \in$ convex hull $(z_{1:N})$ $\Longrightarrow$

$z^* \in$ conv. hull $(z'_1, \dots z'_d)$ where $z'_{1:d} \subset \{z_{1:N}\}$



$\Longrightarrow z^* \in \text{conv}(z_1, z_2, z_4)$

$d=2 \Rightarrow \underbrace{d+1}$ points sufficient

For us:

functions $\varphi_{1:n-1}$, points $y_{1:N}$ $\Rightarrow$ define $z_i = \begin{bmatrix} \varphi_1(y_i) \\ \vdots \\ \varphi_{n-1}(y_i) \end{bmatrix} \in \mathbb{R}^{n-1}$

$$\bar{z} = \begin{bmatrix} \vdots \\ \frac{1}{N} \sum_{i=1}^{N} \varphi_j(y_i) \\ \vdots \end{bmatrix} \leftarrow \bar{z}_j$$

$\Rightarrow$ $\exists\, x_{1:n} \in \{y_{1:N}\}$ so that $\bar{z}_j = \sum_{i=1} w_i \varphi_j(x_i)$ for $j = 1{:}n-1$

$w_{1:n} \geq 0$, $\sum w_i = 1$

Calculating MMD

If k universal kernel $\Rightarrow |\mu f - \nu f| \leq \|f\|_{\mathcal{H}} \|\mu - \nu\|_{\mathcal{H}}$

[Koksma-Hlavka inequality]

straight-forward from $\|\mu\|_{\mathcal{H}} = \sup_{f \dots} \frac{|\mu f|}{\|f\|_{\mathcal{H}}}$

From e.g. [Gretton]

$$MMD_k(\mu, \nu) = \sup_{\|f\|_{\mathcal{H}} \leq 1} |\mu f - \nu f| = \|\mu - \nu\|_{\mathcal{H}} = \|\mu\varphi - \nu\varphi\|_{\mathcal{H}}$$

$\mu\varphi = \mathbb{E}_{x \sim \mu}[\varphi(x)] \in \mathcal{H}_k$

$$\|\mu\varphi\|^2_{\mathcal{H}} = \langle \mathbb{E}_{x \sim \mu}[\varphi(x)], \mathbb{E}_{y \sim \mu}[\varphi(y)] \rangle = \mathbb{E}_{\substack{x \sim \mu \\ y \sim \mu}} \langle \varphi(x), \varphi(y) \rangle = \mathbb{E}_{\mu \otimes \mu} k(x, y)$$

$\mu \leftarrow \mu_N$, $\nu \leftarrow \mu_Q$ $\Rightarrow$ $\|\mu_N - \mu_Q\|^2 = \frac{1}{n^2} 1^T k(Q, Q) 1 + \frac{1}{N^2} 1^T k(Y_{1:N}, Y_{1:N}) 1 - \frac{2}{nN} 1^T k(Q, Y_{1:N})$

Gram matrix

Gram matrix

Computing MMD_k

from Weighted Quadrature (14)

$$\mathrm{MMD}_k(\mu_Q, \mu)^2 = \| \mu_Q \varphi - \mu \varphi \|_{\mathcal{H}}^2$$

$\varphi: \mathcal{X} \to \mathcal{H}$ feature map

$$= w^T k(X,X) w - 2 E_y\left[w^T k(X,y) \right] + E_{y,y'}\left[k(y,y') \right]$$

weights $w_{1:n}$ or $1/n$

$\mu = \mu_N$ or need to know analytically

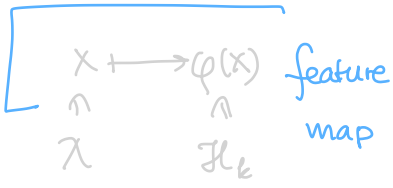
Weight optimization

Quadratic in $w \in \mathbb{R}^n$ $\Rightarrow$ $\min_w \mathrm{MMD}^2$ s.t. $w \geq 0$, $\sum w_i = 1$ with $x_{1:n}, y_{1:N}$ given

- optimizes any one-set (e.g. from QMC, Thinning, ...)

kernel Herding [Chen, Welling, Smola '12]

- Assume $k(x,x) = 1$ (i.e. $\|\varphi(x)\|^2_{\mathcal{H}_k}$) e.g. Gaussian



① define loss $G : \mathcal{H}_k \to \mathbb{R}$

$$G(z) = \langle z, \bar{\varphi} \rangle - \max_x \langle z, x \rangle$$

concave; linear in z; ← pcwise linear, convex

$$\bar{\varphi} = \mathbb{E}_{x \sim \mu}[\varphi(x)] = \mu_\varphi$$

② gradient

$$\nabla_z G = \bar{\varphi} - \varphi(x_z)$$

$$x_z = \operatorname{argmax}_x \langle z, \varphi(x) \rangle$$

③ "gradient ascent" (step size = 1):

given z_t

$$\begin{cases} x_{t+1} = \operatorname{argmax}_x \langle z_t, \varphi(x) \rangle \\ z_{t+1} = z_t + \bar{\varphi} - \varphi(x_{t+1}) \end{cases}$$

need to solve!

need to calculate!

$\Rightarrow$ sequence $x_1, x_2, \dots x_T \dots$ = coreset $\mathbb{Q}$

But is it any good?

Kernel Herding -2-

converges to $\bar{\varphi}$?

(4) let's first look at $z_{1,2,\dots}$

$$z_T = z_{T-1} + \bar{\varphi} - \varphi(x_T) = z_0 + T\bar{\varphi} - \sum_{t=1}^{T}\varphi(x_t) \Rightarrow \boxed{\frac{1}{T}\sum_t \varphi(x_t) = \bar{\varphi} + \frac{z_0 - z_T}{T}}$$

$z_{T-1} = z_{T-2} + \bar{\varphi} - \varphi(x_{T-1})$

$\dots$

remember $\|z_0\|_{\mathcal{H}} = \|z_T\|_{\mathcal{H}} = 1 \Rightarrow \frac{\|z_0 - z_T\|}{T} \le 2T^{-1}$ Nice!

(5) now let's look at $x_{1,2,\dots}$

$$x_{T+1} = \underset{x}{\operatorname{argmax}} \left\langle \bar{\varphi} + T\bar{\varphi} - \sum_{t=1}^{T}\varphi(x_t), \varphi(x)\right\rangle =$$

$\langle \varphi(x'), \varphi(x) \rangle$

$$= \underset{x \in \mathcal{X}}{\operatorname{argmax}} \left\{ (T+1)\, \mathbb{E}_{x' \sim \mu} k(x, x') - \sum_{t=1}^{T} k(x, x_t) \right\}$$

$$= \underset{x \in \mathcal{X}}{\operatorname{argmax}} \left\{ \mathbb{E}_{x' \sim \mu} k(x, x') - \frac{1}{T+1}\sum_{t=1}^{T} k(x, x_t) \right\}$$

- initialize $z_0 = \bar{\varphi}$

(6) and now look at squared error

$$\varepsilon_T^2 = \left\| \bar{\varphi} - \frac{1}{T}\sum_{t=1}^{T}\varphi(x_t) \right\|_{\mathcal{H}}^2 = \left\langle \bar{\varphi} - \frac{1}{T}\sum \varphi(x_t), \dots \right\rangle$$

find x_{T+1} that minimizes $\mathcal{E}^2_{T+1}$ for $x_{1:T}$ fixed

$$\left\langle \bar{\varphi} - \frac{1}{T+1}\sum_{t=1}^{T+1}\varphi(x_t),\ \bar{\varphi} - \frac{1}{T+1}\sum\varphi(x_t)\right\rangle =$$

$$= -2\left\langle \bar{\varphi}, \frac{1}{T+1}\varphi(x_{T+1})\right\rangle + \frac{1}{(T+1)^2}\left\langle \varphi(x_{T+1}), \varphi(x_{T+1}) + \sum_{t=1}^{T}\varphi(x_t)\right\rangle + \dots$$

$$= -\frac{2}{T+1}\left\{ E_\mu k(x', x_{T+1}) - \frac{1}{T+1}\underbrace{\left(k(x_{T+1}, x_{T+1})\right)}_{1} + \sum_{t=1}^{T} k(x_{T+1}, x_t)\right\}$$

Same as (5)

$\Rightarrow$ x_{T+1} greedily minimizes $\mathcal{E}_{T+1}$

Remarks:

- sequence $x_{1,2,\dots}$ NOT Markov
- elegant! but restrictive $k(x,x)=1$, needs (estimate of) $\langle \bar{\varphi}, z\rangle$, $\operatorname{argmax}_x \langle \varphi(x), z\rangle$ (not all kernels and μ's)
- how about MMD? Calculating MMD

Weighted Kernel Quadrature [Hayakawa, Oberhauser, Lyons '21]

Given $y_{1:N} \in \mathcal{X}$, find $x_{1:n} \subset y_{1:N}$, weights $w_{1:n}$ with $w_i \geq 0$, $\sum_{i=1}^n w_i = 1$

so that $\mu_Q = \sum_{i=1}^n w_i \delta_{x_i}$ and $\mathrm{MMD}_k(\mu_N, \mu_Q) \to$ with n fast

how fast? depends on k spectrum

Idea (1) approximate $\mathcal{H}_k$ by some $\mathcal{H}_{k_0}$ ← finite dimensional decay

then match $\mathcal{H}_{k_0}$ exactly (2) ← Caratheodory

(1.1) Representer Theorem (Mercer): $k(x,y) = \sum_{j=1}^{\infty} \lambda_j e_j(x) e_j(y)$

λ_j: e-value, e_j: eigenfunction of k

Then, let $k_0(x,y) = \sum_{j=1}^{n-1} \lambda_j e_j(x) e_j(y)$

$\lambda_1 \geq \lambda_2 \geq \cdots \geq 0$

Nice: $k \geq k_0 \geq 0$

but must know spectral decomp of k

(1.2) Nyström to the rescue

1. sample ℓ points $\tilde{y}_{1:\ell}$ (different from $y_{1:n}$)
2. construct $W = \left[k(\tilde{y}_i, \tilde{y}_j)\right]_{i,j=1:\ell}$ ← assume rank $= \ell$
3. SVD: $W = U \Lambda V^T$
4. keep components $1{:}s$, $s \ll \ell$

$$k_0 \equiv k_s^{\tilde{y}}(x,y) = \sum_{j=1}^{s} \frac{1}{\lambda_j} \underbrace{u_j^T k(\tilde{y},x)}_{\mathbb{R}} \underbrace{u_j^T k(\tilde{y},y)}_{\mathbb{R}}$$

Note: $k \approx k^{\tilde{y}}(x,y) = k(x,\tilde{y}) W^{\dagger} k(\tilde{y},y) \Rightarrow K \gtrsim K_0 \gtrsim 0$

(1.3) Theorem

If • $k - k_0 = k_1$ is Mercer kernel then $\exists \mu_Q$ (as above) with

• $\dim \mathcal{H}_{k_0} \leq n-1$

$$\mathop{\mathbb{E}}_{y_{1:N}}\left[\mathrm{MMD}_k^2(\mu, \mu_Q)\right] \leq 8 \underbrace{\int k_1(x,x)\, d\mu(x)}_{\text{residual kernel}} + \frac{1}{N} c_{k,\mu}$$

residual kernel → within

must be also $\frac{1}{N} \sim \frac{1}{n^2}$

how diagonal is k

$\int_X k(x,x)\, d\mu - \int k(x,y)\, d\mu_x\, d\mu_y$

(2) given k_1, $y_{1:N}$, k_0, functions $\varphi_{1:n-1}$

with $\mathcal{H}_{k_0} = \text{span}\{\varphi_{1:n-1}\}$

find $x_{1:n}$, $w_{1:n}$

now supported only on $y_{1:N}$, measure is μ_N

Alg 1. $\tilde{\varphi}_{1:n-1} = \varphi_{1:n-1}\begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix}$, $\tilde{\varphi}_0 = \begin{bmatrix} k_1(y_1,y_1) \\ \vdots \end{bmatrix}$

2. "Recombination algo" $\Rightarrow$ obtain $\tilde{x}_{1:n+1}$, $v_{1:n+1}$ so that $\tilde{\varphi}_{0:n-1}^T(\tilde{x})\, v = \tilde{\tilde{\varphi}}^T$

3. ....

Kernel Thinning [Dwivedy & Mackey '21]

Idea: kernel halving

data

① $\rightarrow \min \mathrm{MMD}_k(Q_1, Q_2)$ at each step

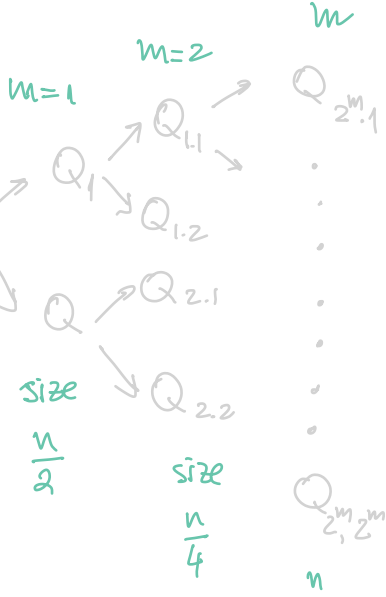
$1{:}N = Q_1 \cup Q_2$

$Q_1 \leftarrow Q_1 \cup x$ with $\{x, x'\} = \{x_{2i-1}, x_{2i}\}$

$Q_2 \leftarrow Q_2 \cup x'$

② Recursive Halving = KT-split

data $1{:}n$



- Alg that runs KT-Halving in parallel at all m levels

③ KT-SWAP $Q^{\text{prelim}} \leftarrow \underset{l=1:2^m}{\operatorname{argmin}} \mathrm{MMD}_k(\mu_N, \mu_Q)$

+ some more refinement $\Rightarrow Q$

- what should m be?

  want $|Q| = \sqrt{n} = \frac{n}{2^m} \Rightarrow n = 2^{2m} \Rightarrow m = \frac{1}{2}\log_2 n$

- run time $\sim n^2$

- Proof Ideas 1. Vector balancing

  too beam

  bounds $\|\mu_N k - \mu_Q k\|_\infty$

  Data $x_1, x_2, \dots x_n \dots$ sequential

  at time t, choose $s_t \in \pm 1$ sign so that

  $\|\sum_{t=1}^{n} s_t x_t\|_\infty$ is small

  How small? $\sim \sqrt{\log d/\delta \cdot \log n/\delta}$ w.p. $1-\delta$

  worst case $\sim n$ (when $\|x_i\| \le 1$)

2. Square root kernel trick to get bound on MMD_k

- $k_{1/2}$ is square root for k iff

  $k(x,y) = \int_{\mathcal{X}} k_{1/2}(x,z) k_{1/2}(y,z) dz$

- Theorem $MMD_k(\mu,\nu) \le c\|\mu k_{1/2} - \nu k_{1/2}\|_\infty$ + tail terms

- Algorithm: KT-Split uses $k_{1/2}$

Square root kernel

If $k(x,y) = k(x-y)$ and even

and $k_{1/2}$ —"— —"—

then $\hat{k}_{1/2} = \left(\hat{k}\right)^{1/2}$

Fourier transforms

Proof

$$k(x,y) = \int_{\mathcal{X}} k_{1/2}(z-x)\, k_{1/2}(y-z)\, dz = \int_{\mathcal{X}} k_{1/2}(t)\, k_{1/2}(y-x-t)\, dt = \left(k_{1/2} * k_{1/2}\right)(y-x)$$

$\parallel$

$k(y-x)$

t $\quad$ $y-x-t$

convolution

Fourier $\quad * \longmapsto \cdot \quad \Rightarrow \quad \hat{k} = \hat{k}_{1/2}\hat{k}_{1/2}$