Kernel Bayes' Rule

K. Fukumizu, L. Song, A. Gretton,
"Kernel Bayes' rule: Bayesian inference with positive definite kernels"
Journal of Machine Learning Research, vol. 14, Dec. 2013.

Yan Xu
yxu15@uh.edu

Kernel based automatic learning workshop
University of Houston
April 24, 2014

Bayesian inference

Bayes' rule

likelihood prior

$$q(x|y) = \frac{p(y|x)\pi(x)}{\int p(y|x)\pi(x)dx}$$

posterior

- PROS
  - Principled and flexible method for statistical inference.
  - Can incorporate prior knowledge.
- CONS
  - Computation: integral is needed
    - » Numerical integration: Monte Carlo etc
    - » Approximation: Variational Bayes, belief propagation etc.

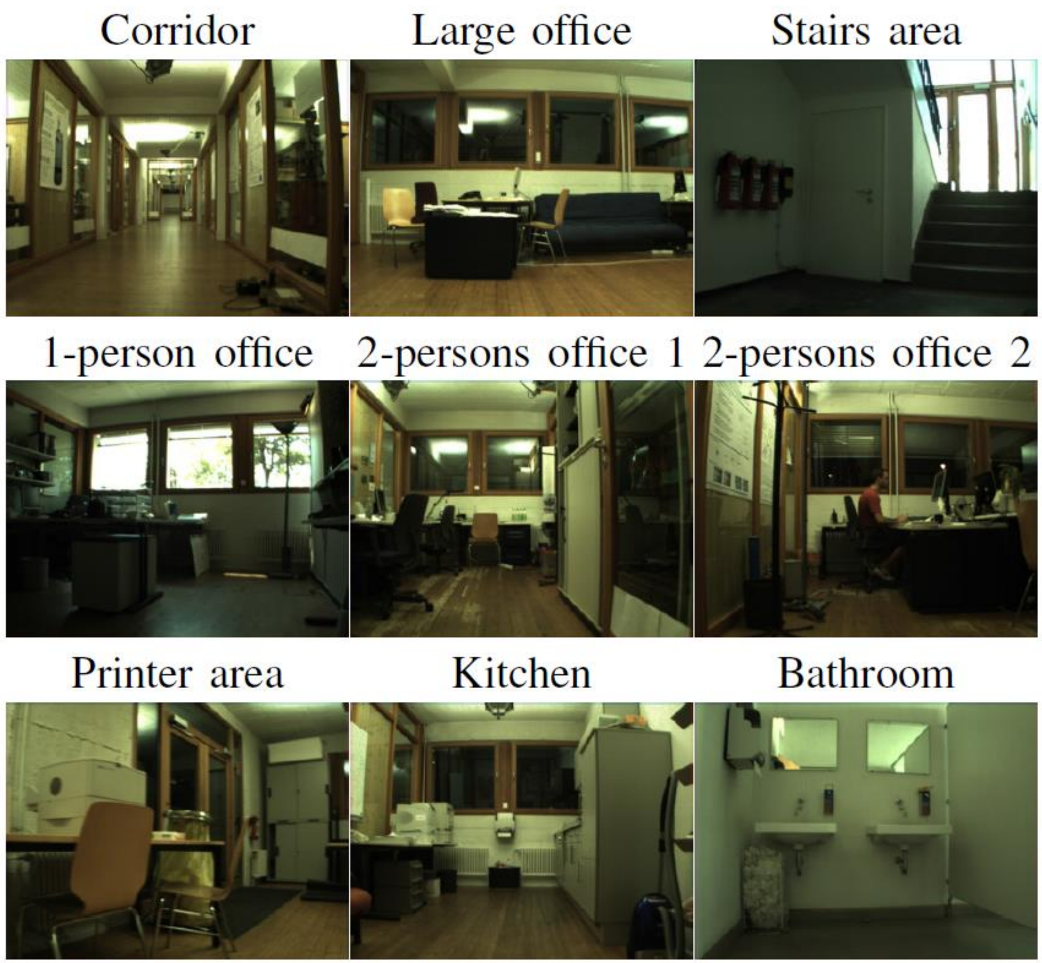
Motivating Example: Robot location

Kanagawa et al. Kernel Monte Carlo Filter, 2013

State $X_t \in \mathbf{R}^3$:
2-D coordinate and orientation of a robot

Observation Z_t :
image SIFT features (Scale Invariant Feature Transform, 4200dim）

Goal:
Estimate the location of a robot from image sequences

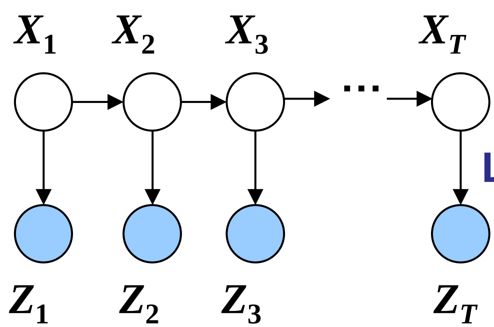


COLD: Cosy Location Database

– Hidden Markov Model

Sequential application of Bayes' rule solves the task.

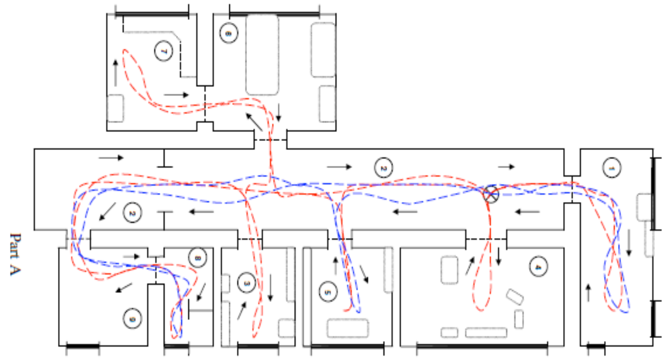
Transition of state



location & orientation

Location & orientation → image

image of the environment



location & orientation

$p(Z_t|X_t)$

$p(X_t|Z_{1:t})$

image of the environment

– Nonparametric approach is needed:

Observation process: $p(Z_t|X_t)$ is very difficult to model with a simple parametric model.

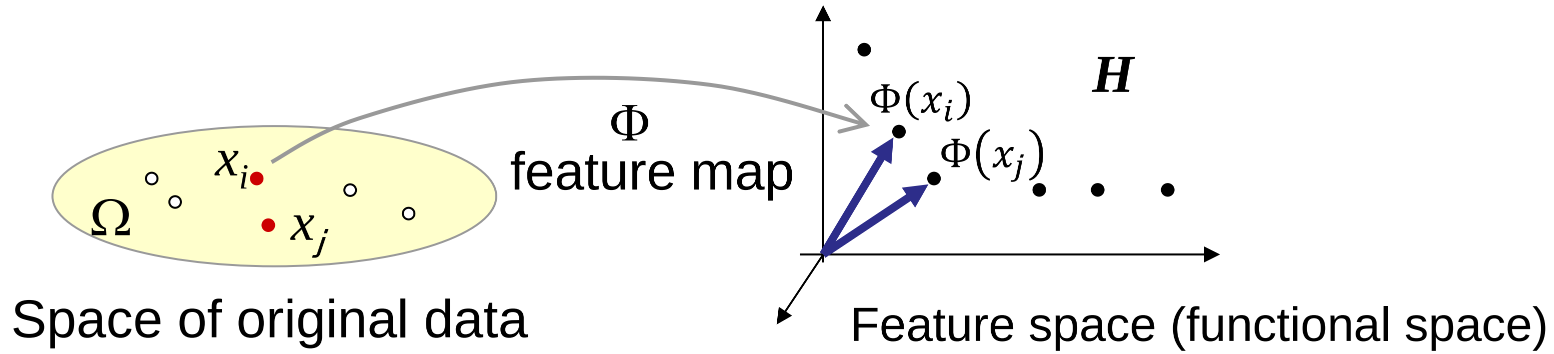
"Nonparametric" implementation of Bayesian inference

Kernel method for Bayesian inference

A new nonparametric / kernel approach to Bayesian inference

- Using positive definite kernels to represent probabilities.
  - Kernel mean embedding is used.
- “Nonparametric” Bayesian inference
  - No density functions are needed, but data are needed.
- Bayesian inference with matrix computation.
  - Computation is done with Gram matrices.
  - No integral, no approximate inference.

Kernel methods: an overview



Do linear analysis in the feature space.

$$\Phi: \quad \Omega \quad \rightarrow \; H, \qquad x \mapsto \Phi(x)$$

Kernel PCA, kernel SVM, kernel regression etc.

Positive semi-definite kernel

Def. Ω: set; $k: \Omega \times \Omega \to \mathbf{R}$ $\quad k(X_i, X_j) = \langle \Phi(X_i), \Phi(X_j) \rangle$

k is positive semi-definite if k is symmetric, and for any $n \in \mathbf{N}$, $x_1, \ldots, x_n \in \Omega$, $c = [c_1, \ldots, c_n]^T \in R^n$, the matrix G_X: $\left(k(X_i, X_j)\right)_{ij}$ (Gram matrix) satisfies

$$c^T G_X c = \sum_{i,j=1}^{n} c_i c_j k(X_i, X_j) \geq 0.$$

positive definite: $c^T G_X c > 0.$

– Examples on $\mathbf{R}^m$:
  - Gaussian kernel $\quad k_G(x, y) = \exp\left(-\frac{1}{2\sigma^2} ||x - y||^2\right) \quad (\sigma > 0)$
  - Laplace kernel $\quad k_L(x, y) = \exp\left(-\alpha \sum_{i=1}^{m} |x_i - y_i|\right) \quad (\alpha > 0)$
  - Polynomial kernel $\quad k_P(x, y) = (x^T y + c)^d \quad (c \geq 0, d \in \mathbf{N})$

Reproducing Kernel Hilbert Space

"Feature space" = Reproducing kernel Hilbert space (RKHS)

A positive definite kernel k on Ω uniquely defines a RKHS H_k (Aronzajn 1950).

- Function space: functions on Ω.
- Very special inner product: for any $f \in H_k$

$$\langle f, k(\cdot, x)\rangle_{H_k} = f(x)$$ (reproducing property)

- Its dimensionality may be infinite (Gaussian, Laplace).

Mapping data into RKHS

$$\Phi : \Omega \to H_k, \quad x \mapsto k(\cdot, x)$$

$X_1, \ldots, X_n \quad \mapsto \quad \Phi(X_1), \ldots, \Phi(X_n)$: functional data

Basic statistics on Euclidean space	Basic statistics on RKHS
Probability	Kernel mean
Covariance	Covariance operator
Conditional probability	Conditional kernel mean

Mean on RKHS

X: random variable taking value on a measurable space Ω, $\sim P$.

k: pos.def. kernel on Ω. H_k: RKHS defined by k.

Def. kernel mean on H :

$$m_P := E[\Phi(X)] = E[k(\cdot, X)] = \int k(\cdot, x) dP(x) \in H_k$$

– Kernel mean can express higher-order moments of X.

Suppose $k(u,x) = c_0 + c_1 ux + c_2 (ux)^2 + \cdots \quad (c_i \geq 0)$, e.g., e^{ux}

$$m_P(u) = c_0 + c_1 E[X] u + c_2 E[X^2] u^2 + \cdots$$

– Reproducing expectations

$$\langle f, m_P \rangle = E[f(X)] \qquad \text{for any } f \in H_k.$$

Characteristic kernel

(Fukumizu et al. JMLR 2004, AoS 2009; Sriperumbudur et al. JMLR2010)

<u>Def.</u> A bounded pos. def. kernel k is called characteristic if

$$\mathcal{P} \to H_k, \quad P \mapsto m_P$$

is injective, i.e., $E_{X\sim P}[k(\cdot, X)] = E_{Y\sim Q}[k(\cdot, Y)] \longleftrightarrow P = Q$.

m_P with a characteristic kernel uniquely determines a probability.

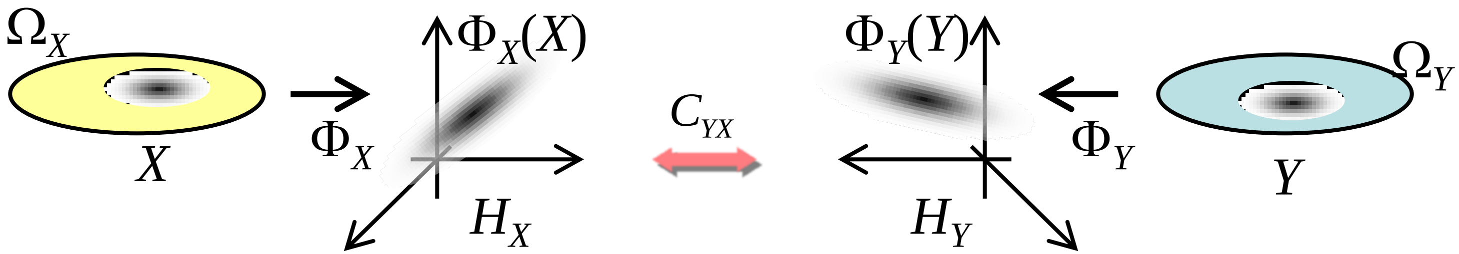
Examples: Gaussian, Laplace kernel

Polynomial kernel: not characteristic.

Covariance

(X, Y) : random vector taking values on $\Omega_X \times \Omega_Y$.

(H_X, k_X), (H_Y, k_Y): RKHS on Ω_X and Ω_Y, resp.



Def. (uncentered) covariance operators $C_{YX}: H_X \to H_Y$, $C_{XX}: H_X \to H_X$

$$C_{YX} := E\left[\Phi_Y(Y)\langle \Phi_X(X), \cdot \rangle_{H_X}\right], \qquad C_{XX} = E\left[\Phi_X(X)\langle \Phi_X(X), \cdot \rangle_{H_X}\right]$$

$$C_{YX} f = \int k_Y(\cdot, y) f(x) dP(x,y), \quad C_{XX} f = \int k_X(\cdot, x) f(x) dP_X(x)$$

Reproducing property

$$\langle g, C_{YX} f \rangle_{H_Y} = E[f(X)g(Y)] \qquad \text{for all } f \in H_X, g \in H_Y.$$

Empirical Estimator: Given $(X_1, Y_1,), \ldots, (X_n, Y_n) \sim P$, i.i.d.,

$$\hat{C}_{YX} f = \frac{1}{n}\sum_{i=1}^{n} k_Y(\cdot, Y_i)\langle k_X(\cdot, X_i), f\rangle = \frac{1}{n}\sum_{i=1}^{n} k_Y(\cdot, Y_i) f(X_i)$$

Conditional kernel mean

- X, Y: Centered gaussian random vectors ($\in R^m, R^\ell$, resp.)

$$E[Y|X=x] = V_{YX}V_{XX}^{-1}x$$

$$\underset{A\in R^{\ell\times m}}{\operatorname{argmin}} \int \|Y - AX\|^2 dP(X,Y) = V_{YX}V_{XX}^{-1}$$

V : Covariance matrix

- With characteristic kernels, for general X and Y,

$$\underset{F\in H_X\otimes H_Y}{\operatorname{argmin}} \int \|\Phi_Y(Y) - \underline{F(X)}\|_{H_Y}^2 dP(X,Y) = C_{YX}C_{XX}^{-1}$$

$\langle F, \Phi_X(X)\rangle$

$$E[\Phi(Y)|X=x] = C_{YX}C_{XX}^{-1}\Phi_X(x)$$

In practice:

$$\widehat{m}_{Y|X=x} := \hat{C}_{YX}\left(\hat{C}_{XX} + \varepsilon_n I\right)^{-1}\Phi_X(x)$$

Kernel realization of Bayes' rule

- Bayes' rule

$$q(x|y) = \frac{p(y|x)\pi(x)}{q(y)}, \qquad q(y) = \int p(y|x)\pi(x)dx.$$

Π: prior with p.d.f π

$p(y|x)$: conditional probability (likelihood).

- Kernel realization:

Goal: estimate the kernel mean of the posterior

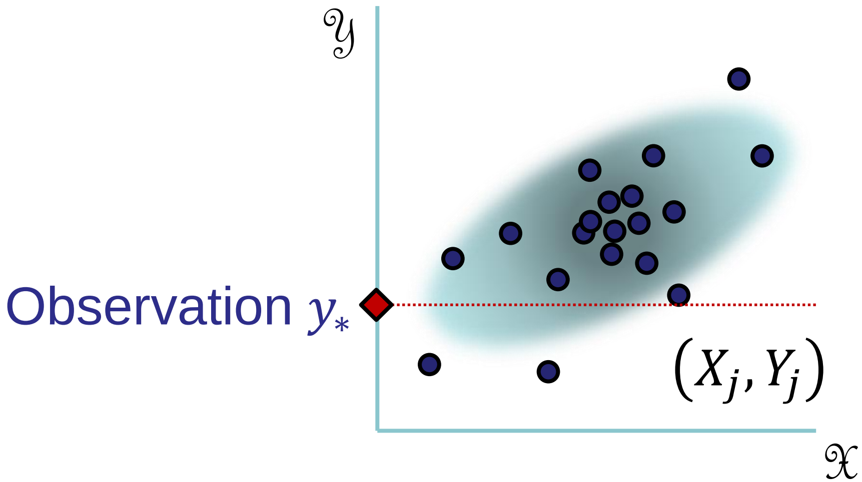
$$m_{Q_{x|y*}} := \int k_X(\cdot, x) q(x|y_*) dx$$

given

- m_Π: kernel mean of prior Π,
- C_{XX}, C_{YX}: covariance operators for $(X, Y) \sim Q$,

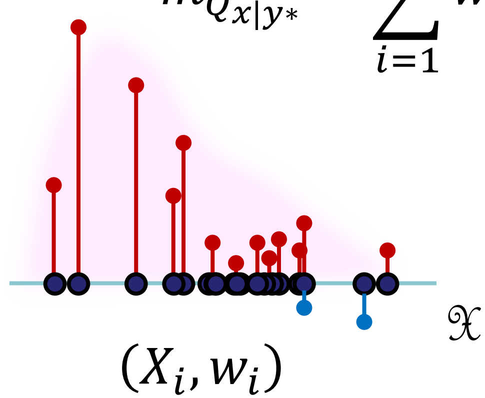
Kernel realization of Bayes' rule

$(X_1, Y_1), \ldots, (X_n, Y_n)$: (joint) sample ~ Q



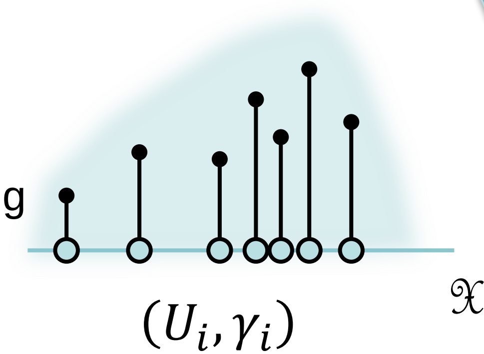
Posterior

$$\widehat{m}_{Q_{x|y*}} = \sum_{i=1}^{n} w_i(y_*)\Phi_X(X_i)$$



Prior $\widehat{m}_\Pi = \sum_{j=1}^{\ell} \gamma_j \Phi_X(U_j)$

$(U_1, \gamma_1), \ldots, (U_\ell, \gamma_\ell)$:
weighted sample
expression from
importance sampling



Kernel Bayes' Rule

$$\widehat{m}_{Q_{x|y*}}(\cdot) = \sum_{i=1}^{n} w_i(y_*)k_X(\cdot, X_i) = \mathbf{k}_X(\cdot)^T R_{x|y}\mathbf{k}_Y(y_*)$$

Input: $(X_1, Y_1), \dots, (X_n, Y_n) \sim$ Q, $\widehat{m}_\Pi = \left(\sum_{j=1}^{\ell} \gamma_j \mathrm{k}_X(X_i, U_j)\right)_{i=1}^{\mathrm{n}}$ (prior)

$$\mathbf{k}_Y(y_*) = (\mathbf{k}_Y(Y_i, y_*))_{i=1}^{\mathrm{n}}$$

$$R_{x|y} = \Lambda G_Y\left((\Lambda G_Y)^2 + \delta_n I_n\right)^{-1}\Lambda.$$

$\mathrm{n} \times \mathrm{n}$ $\qquad$ $\mathrm{n} \times \mathrm{n}$

$$\Lambda = \mathrm{Diag}[(G_X/n + \varepsilon_n I_n)^{-1} G_{XU}\gamma\ ]$$

$\mathrm{n} \times \mathrm{n}$ $\qquad$ $\mathrm{n} \times \mathrm{n}$ $\qquad$ $\mathrm{n} \times \ell$ $\qquad$ $\ell \times 1$

Note:
y_* : observation
G_X: $\left(k_X(X_i, X_j)\right)_{ij}$
G_{XU}: $\left(k_X(X_i, U_j)\right)_{ij}$
G_Y: $\left(k_Y(Y_i, Y_j)\right)_{ij}$

ε_n, δ_n: regularization coefficients

$$f \in H_X \qquad < f\ , \widehat{m}_{Q_{x|y*}} > = \mathbf{f}_X{}^T R_{x|y}\mathbf{k}_Y(y_*), \mathbf{f}_X = \left(f(X_1), \dots, f(X_n)\right)^T$$

Application: Bayesian Computation Without Likelihood

KBR for kernel posterior mean:

1). Generate samples $X_1, \ldots, X_n$ from the prior Π;
2). Generate a sample Y_t from $P(Y|X_t)$;
3). Compute Gram matrices G_X and G_Y with $(X_1, Y_1), \ldots, (X_n, Y_n)$;
4). $R_{x|y} = \Lambda G_Y\left((\Lambda G_Y)^2 + \delta_n I_n\right)^{-1}\Lambda$.

$$\widehat{m}_{Q_{x|y*}}(\cdot) = \mathbf{k}_X(\cdot)^T R_{x|y} \mathbf{k}_Y(y_*)$$

Only obtain expectations of functions in RKHS

ABC (Approximate Bayesian Computation):

1). Generate a sample X_t from the prior Π;
2). Generate a sample Y_t from $P(Y|X_t)$;
3). If $D(y_*, Y_t) < \tau$, accept X_t; otherwise reject;
4) Go to 1).

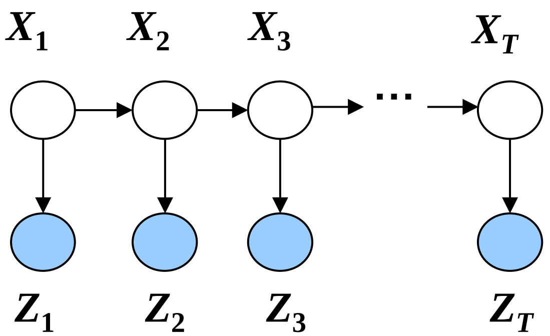
Efficiency can be arbitrarily poor for small τ.

Note: D is a distance measure in the space of Y.

Application: Kernel Monte Carlo Filter

Problem statement

Transition of state



$$p(X,Z) = \pi(X_1)\prod_{t=1}^{T} p(Z_t|X_t)\prod_{t=1}^{T-1} q(X_{t+1}|X_t)$$

Training data: $(X_1, Z_1, \ldots, X_T, Z_T)$

Kernel mean of posterior: $m_{x_t|z_{1:t}} = \int k_x(\cdot, X_i)p(x_t|z_{1:t})dx_t$

$$= \sum_{i=1}^{n} \alpha_i^t k_X(\cdot, X_i)$$

State estimation: pre-image: $\arg\min_{x\in\mathcal{X}} \|\hat{m}_{x_t|z_{1:t}} - k_{\mathcal{X}}(\cdot, x)\|_{\mathcal{H}_{\mathcal{X}}}$
or the sample point with maximum weight

Application: Kernel Monte Carlo Filter

Kanagawa et al. Kernel Monte Carlo Filter, 2013

Input: training data $\{(X_i, Z_i)\}_{i=1}^n$, test observations $\{z_j\}_{j=1}^T$, control inputs $\{u_j\}_{j=1}^T$.

set $\alpha_i^{(0)} = 1/n,\ i = 1, \ldots, n.$

for $t = 1$ to T **do**

  if $t = 1$ **then**

    generate $X_i^{(1)} \sim p_{\text{init}},\ i = 1, \ldots, n.$

  else

    generate $X_i^{(t)} \sim p(\cdot | X_i, u_t),\ i = 1, \ldots, n.$

  end if

  calculate $\mathbf{m}_{x_t|z_{1:t-1}} := (\hat{m}_{x_t|z_{1:t-1}}(X_i))_{i=1}^n \in \mathbb{R}^n$

$$\hat{m}_{x_t|z_{1:t-1}} = \sum_{i=1}^n \alpha_i^{(t-1)} k_{\mathcal{X}}(\cdot, X_i^{(t)})$$

  observe z_t and calculate $\mathbf{k}_Z(z_t) := (k(z_t, Z_i))_{i=1}^n \in \mathbb{R}^n$

  calculate $\alpha^{(t)} \in \mathbb{R}^n$

$$\Lambda = \mathrm{diag}((G_X + n\varepsilon_n I_n)^{-1} \mathbf{m}_{x_t|z_{1:t-1}})$$

$$\alpha^{(t)} = \Lambda G_Z((\Lambda G_Z)^2 + \delta_n I_n)^{-1} \Lambda \mathbf{k}_Z(z_t)$$

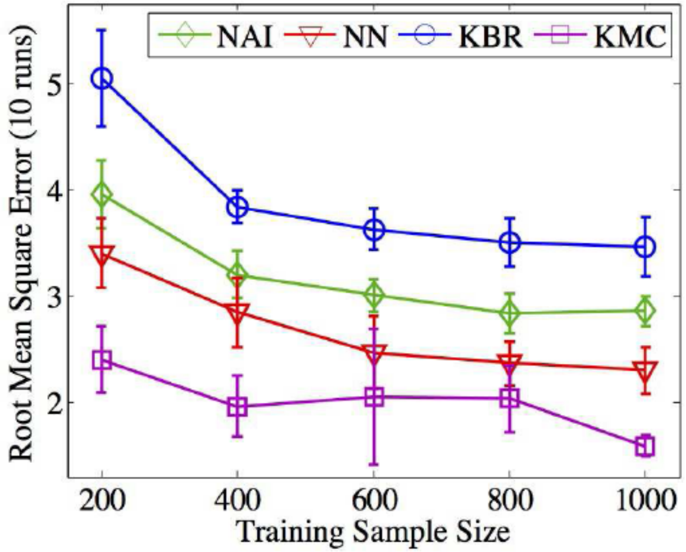
end for

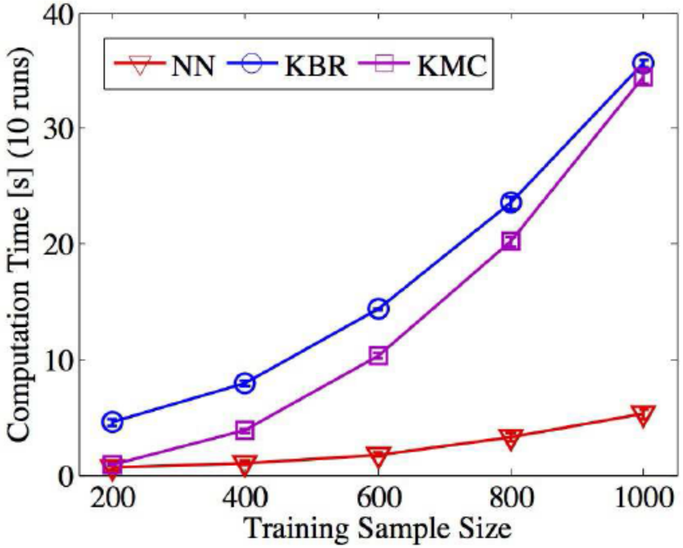
Output: kernel means of the posterior distributions $\hat{m}_{x_t|z_{1:t}} = \sum_{i=1}^n \alpha_i^{(t)} k_{\mathcal{X}}(\cdot, X_i), t = 1, \ldots, T.$

KMC for Robot localization

Kanagawa et al. Kernel Monte Carlo Filter, 2013

NAI: naïve method
KBR: KBR + KBR
NN: PF + K-nearest neighbor
KMC: Kernel Monte Carlo

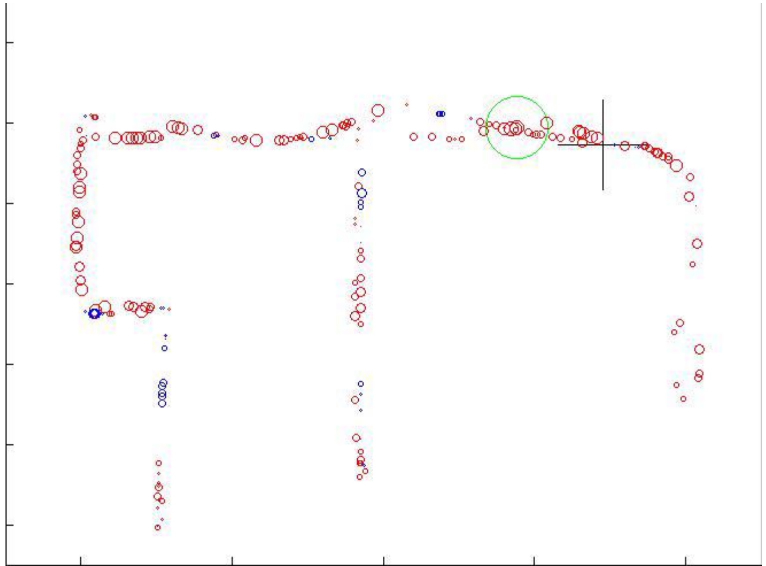




training sample = 200

+: true location

○ : estimate



Conclusions

A new nonparametric / kernel approach to Bayesian inference

- Kernel mean embedding: using positive definite kernels to represent probabilities
- "Nonparametric" Bayesian inference : No densities are needed but data.
- Bayesian inference with matrix computation.

  Computation is done with Gram matrices.

  No integral, no approximate inference.
- More suitable for high dimensional data than smoothing kernel approach.

References

- Fukumizu, K., L. Song, A. Gretton (2013) Kernel Bayes' Rule: Bayesian Inference with Positive Definite Kernels. *Journal of Machine Learning Research. 14:3753–3783.*
- Song, L., Gretton, A., and Fukumizu, K. (2013) Kernel Embeddings of Conditional Distributions. *IEEE Signal Processing Magazine 30(4), 98-111*
- Kanagawa, M., Nishiyama, Y., Gretton, A., Fukumizu. K. (2013) Kernel Monte Carlo Filter. arXiv:1312.4664

Thank you

Q&A

Appendix I. Importance sampling

$$\int f(\mathbf{x})p(\mathbf{x})d\mathbf{x} = \int f(\mathbf{x})\frac{p(\mathbf{x})}{q(\mathbf{x})}q(\mathbf{x})d\mathbf{x}$$

$$= \int f(\mathbf{x})w(\mathbf{x})q(\mathbf{x})d\mathbf{x}$$

$$\approx \frac{1}{N}\sum_{i=1}^{N} f(\mathbf{x}^{(i)})w(\mathbf{x}^{(i)}), \quad \mathbf{x}^{(i)} \sim q(\mathbf{x}).$$

$$w(\mathbf{x}) = \frac{p(\mathbf{x})}{q(\mathbf{x})}.$$

Appendix II. Simulated Gaussian data

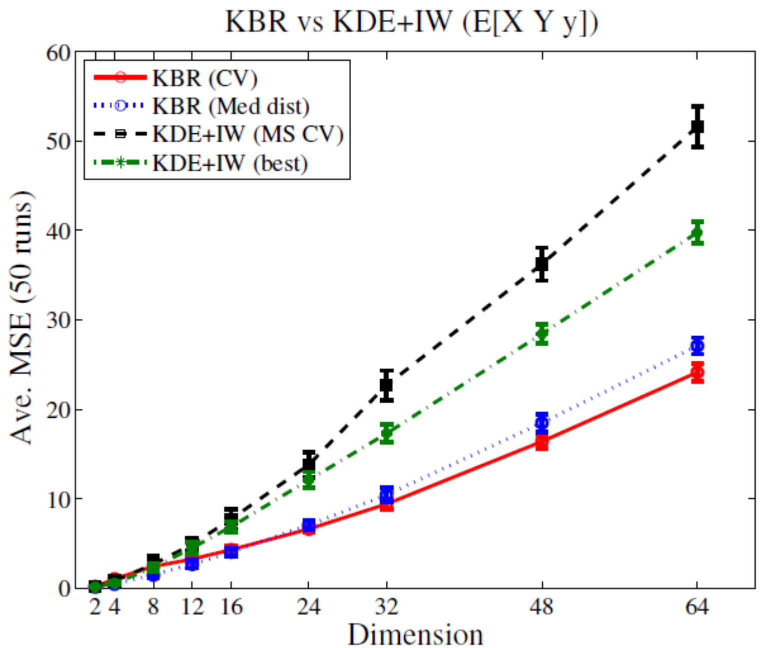
- Simulated data:

$$(X_i, Y_i) \sim N\left(\left(0_{d/2}, \mathbf{1}_{d/2}\right)^T, V\right), \quad i = 1, \dots, N$$

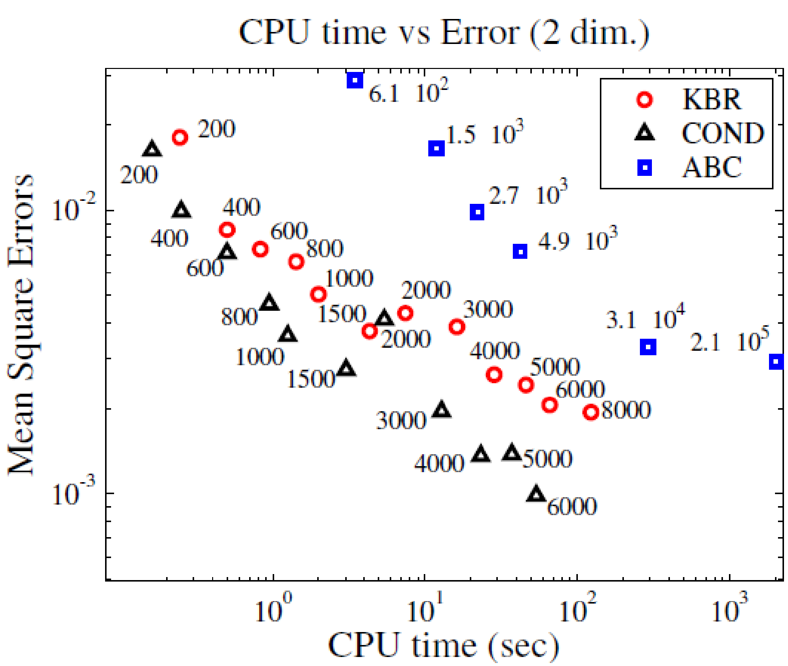
$$V \sim A^T A + 2I_d,\ A \sim N(0, I_d)\ ,\ N = 200$$

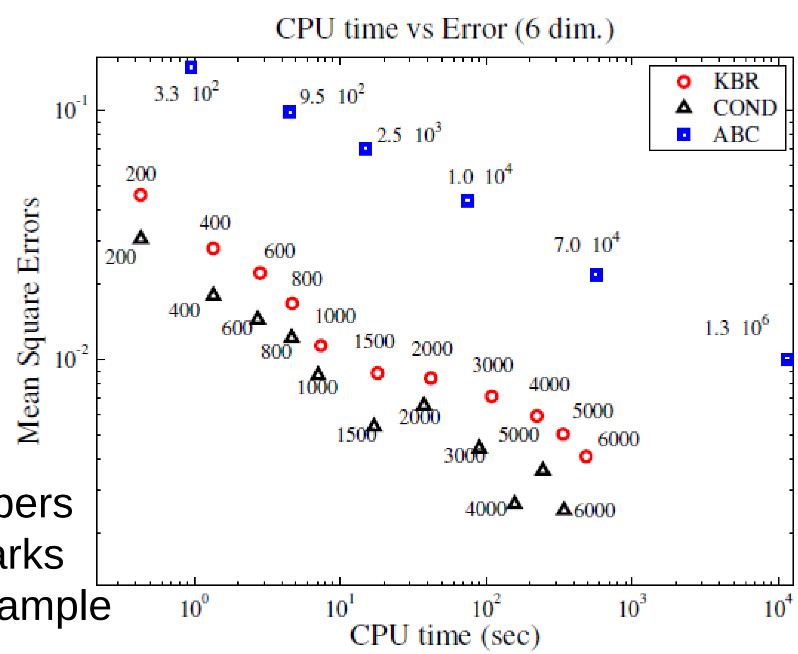
- Prior Π: $U_j \sim N(0; 0.5 * V_{XX}), \quad j = 1, \dots, L,\ L = 200$
- Dimension: $d = 2, \dots, 64$
- Gaussian kernels are used for both methods $h_X = h_Y$
- Bandwidth parameters are selected with CV or the median of the pair-wise distances

Validation: Mean square errors (MSE) of the estimates of $\int xq(x|y)dx$ over 1000 random points $y \sim N(0, V_{YY})$.



KBR: Kernel Bayes Rule
KDE+IW:
Kernel density estimation +
Importance weighting.
COND: belonging to KBR
ABC:
Approximate Bayesian Computation





Numbers at marks are sample sizes