Bayesian Interpretations of RKHS Embedding Methods

David Duvenaud

Cambridge University
Computational and Biological Learning Lab

December 8, 2012

Outline

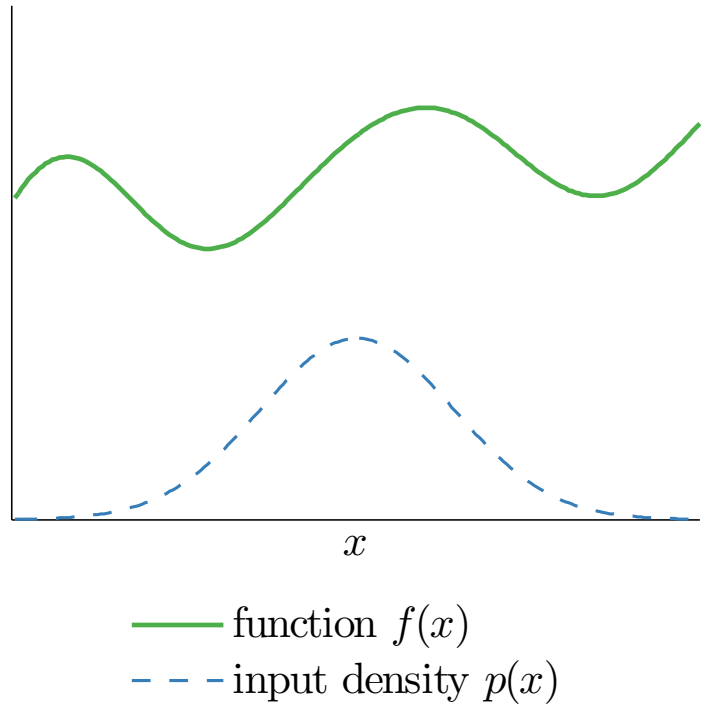
- Optimally-weighted Herding is Bayesian Quadrature
  - Kernel Herding
  - Bayesian Quadrature
  - Unifying Results
  - Demos
- Frequentist Methods, Bayesian Takeaways
  - Kernel Herding
  - Mean Embeddings
  - Kernel Two-sample Test
  - Hilbert-Schmidt Independence Criterion
  - Determinantal Point Processes

The Quadrature Problem

- We want to estimate an integral

$$Z = \int f(x)p(x)dx$$

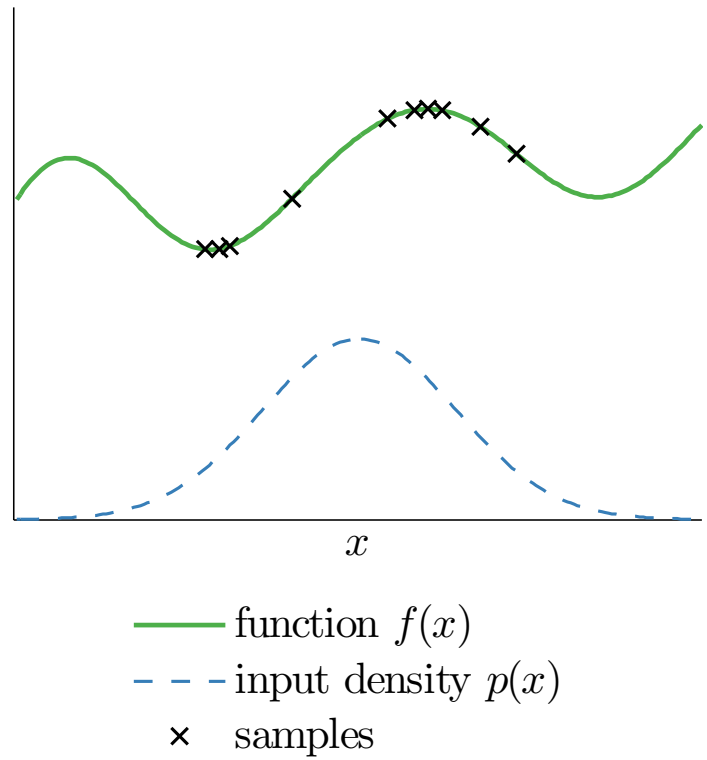
- Most computational problems in Bayesian inference correspond to integrals:
  - Expectations
  - Marginal distributions
  - Integrating out nuisance parameters
  - Normalization constants



Sampling Methods

- Monte Carlo methods: Sample from $p(x)$, take empirical mean:

$$\hat{Z} = \frac{1}{N} \sum_{i=1}^{N} f(x_i)$$

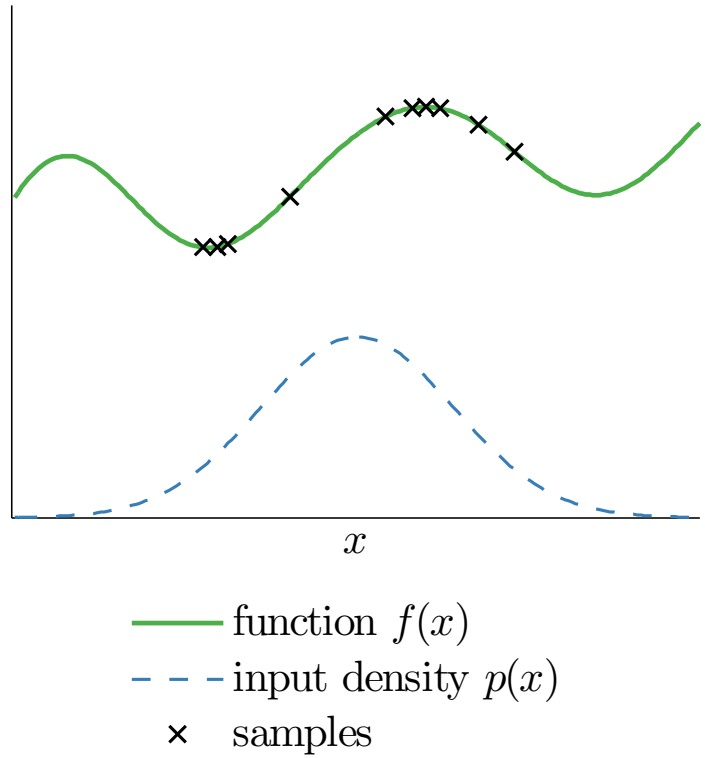


Sampling Methods

- Monte Carlo methods: Sample from $p(x)$, take empirical mean:

$$\hat{Z} = \frac{1}{N} \sum_{i=1}^{N} f(x_i)$$

- Possibly sub-optimal for two reasons:

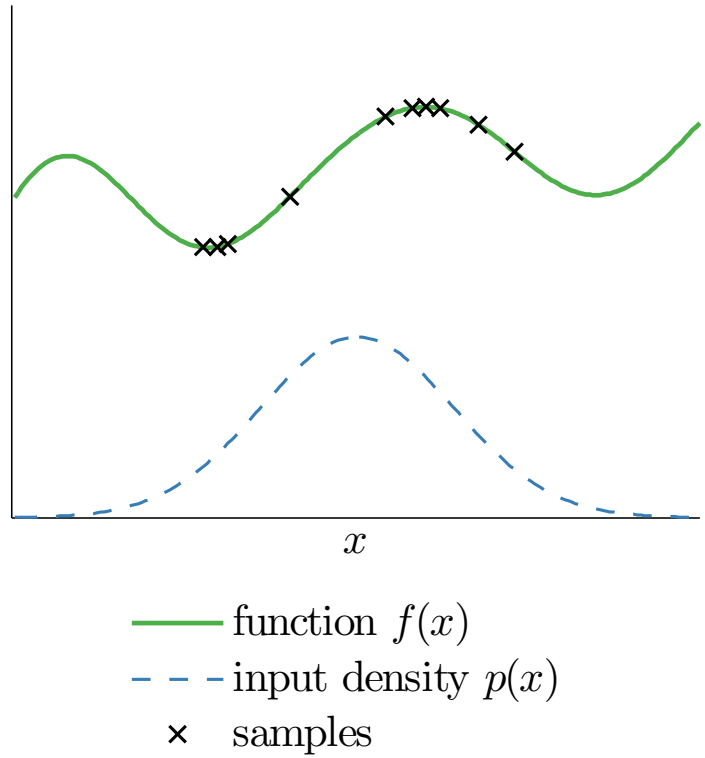


Sampling Methods

- Monte Carlo methods: Sample from $p(x)$, take empirical mean:

$$\hat{Z} = \frac{1}{N} \sum_{i=1}^{N} f(x_i)$$

- Possibly sub-optimal for two reasons:
  - Random bunching up

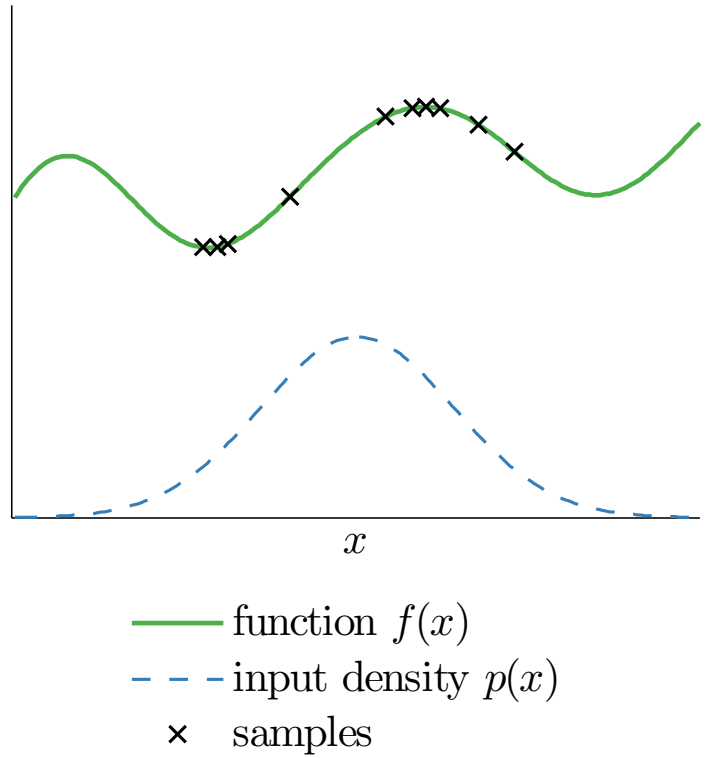


Sampling Methods

- Monte Carlo methods: Sample from $p(x)$, take empirical mean:

$$\hat{Z} = \frac{1}{N} \sum_{i=1}^{N} f(x_i)$$

- Possibly sub-optimal for two reasons:
  - Random bunching up
  - Often, nearby function values will be similar

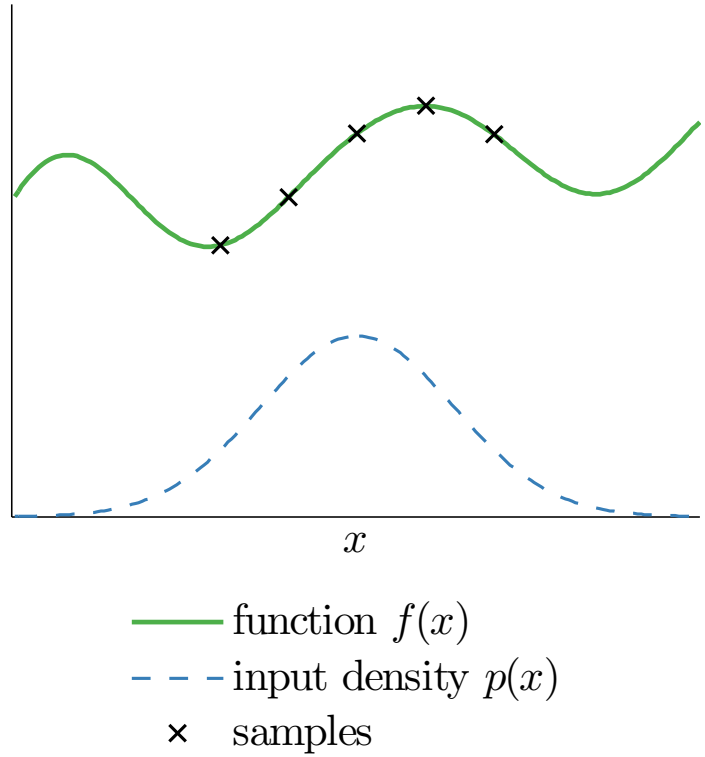


Sampling Methods

- Monte Carlo methods: Sample from $p(x)$, take empirical mean:

$$\hat{Z} = \frac{1}{N} \sum_{i=1}^{N} f(x_i)$$

- Possibly sub-optimal for two reasons:
  - Random bunching up
  - Often, nearby function values will be similar
- Quasi-Monte Carlo methods spread out samples to achieve faster convergence.



Kernel Herding [Welling et. al., 2009, Chen et. al., 2010]

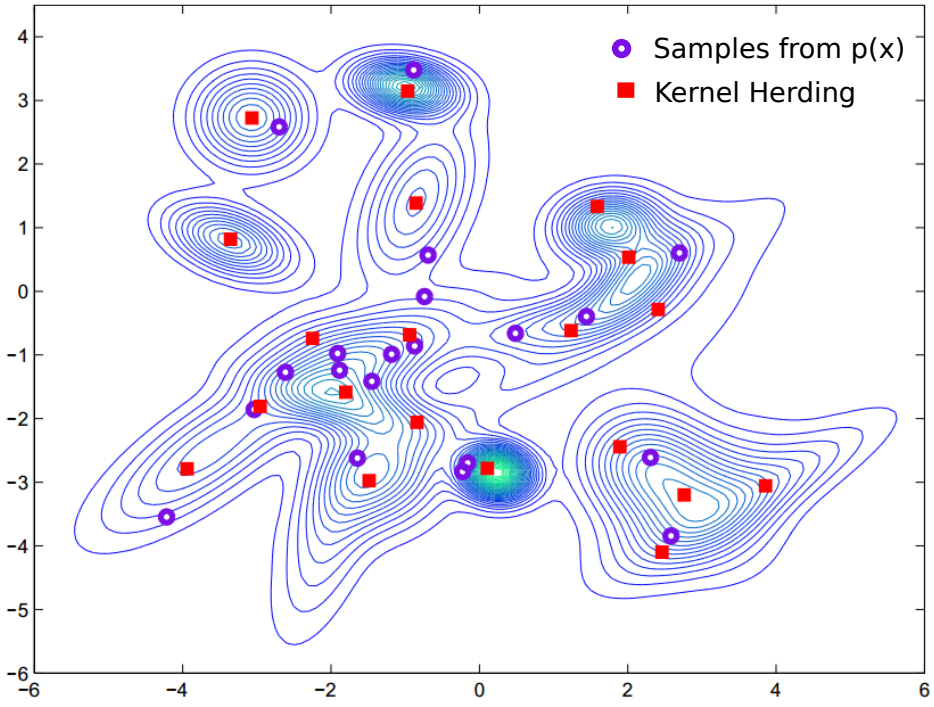
- A sequential procedure for choosing sample locations, depending on previous locations.

Kernel Herding [Welling et. al., 2009, Chen et. al., 2010]

- A sequential procedure for choosing sample locations, depending on previous locations.
- Keeps estimate rule $\hat{Z} = \frac{1}{N} \sum_{i=1}^{N} f(x_i)$

Kernel Herding [Welling et. al., 2009, Chen et. al., 2010]

- A sequential procedure for choosing sample locations, depending on previous locations.
- Keeps estimate rule $\hat{Z} = \frac{1}{N} \sum_{i=1}^{N} f(x_i)$
- Almost $\mathcal{O}(1/N)$ convergence instead of $\mathcal{O}(1/\sqrt{N})$ typical of random sampling, by spreading out samples.



Kernel Herding Objective

KH was found to minimize Maximum Mean Discrepancy:

$$\mathrm{MMD}_{\mathcal{H}}(p, q) = \sup_{\substack{f \in \mathcal{H} \\ \|f\|_{\mathcal{H}}=1}} \left| \int f(x)p(x)dx - \int f(x)q(x)dx \right|$$

Kernel Herding Objective

KH was found to minimize Maximum Mean Discrepancy:

$$\mathrm{MMD}_{\mathcal{H}}(p, q) = \sup_{\substack{f \in \mathcal{H} \\ \|f\|_{\mathcal{H}}=1}} \left| \int f(x)p(x)dx - \int f(x)q(x)dx \right|$$

In KH, $p(x)$ is true distribution, and $q(x)$ is a set of point masses at sample locations $\{x_1, \ldots, x_N\}$:

$$\epsilon_{KH}(\{x_1, \ldots, x_N\}) = \mathrm{MMD}_{\mathcal{H}}\left(p, \underbrace{\frac{1}{N}\sum_{n=1}^{N}\delta_{x_n}}_{q(x)}\right)$$

Kernel Herding

- Assuming function is in a Reproducing Kernel Hilbert Space defined by $k(\cdot, \cdot)$, MMD has closed form.

Kernel Herding

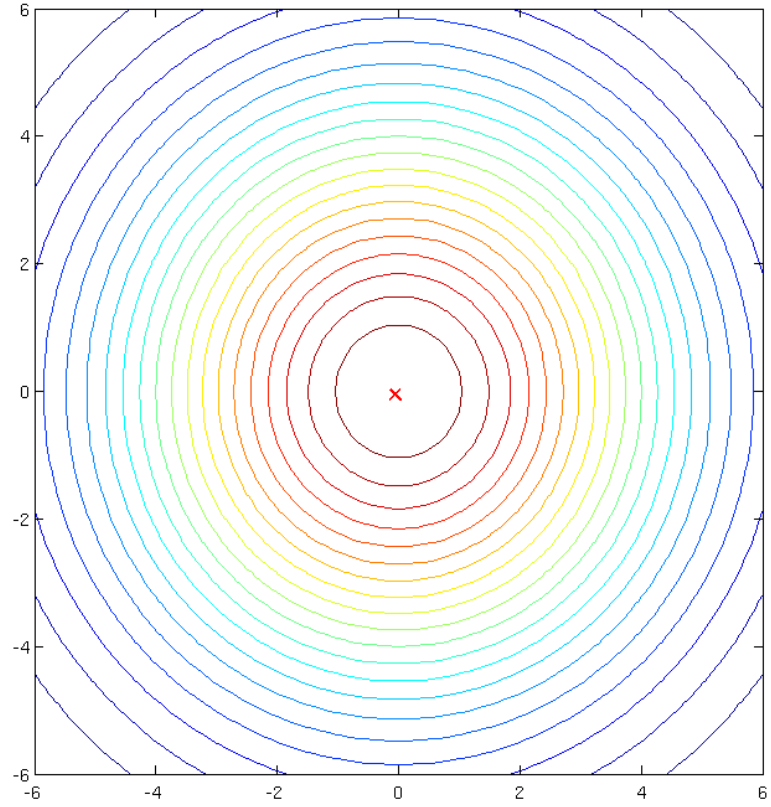
- Assuming function is in a Reproducing Kernel Hilbert Space defined by $k(\cdot, \cdot)$, MMD has closed form.
- When sequentially minimizing MMD, new point is added at:

$$x_{N+1} = \underset{x \in \mathcal{X}}{\text{argmax}} \left[2 \int k(x, x') p(x') dx' - \frac{1}{N+1} \sum_{m=1}^{N} k(x, x_m) \right]$$

Kernel Herding in Action

New point is added at:

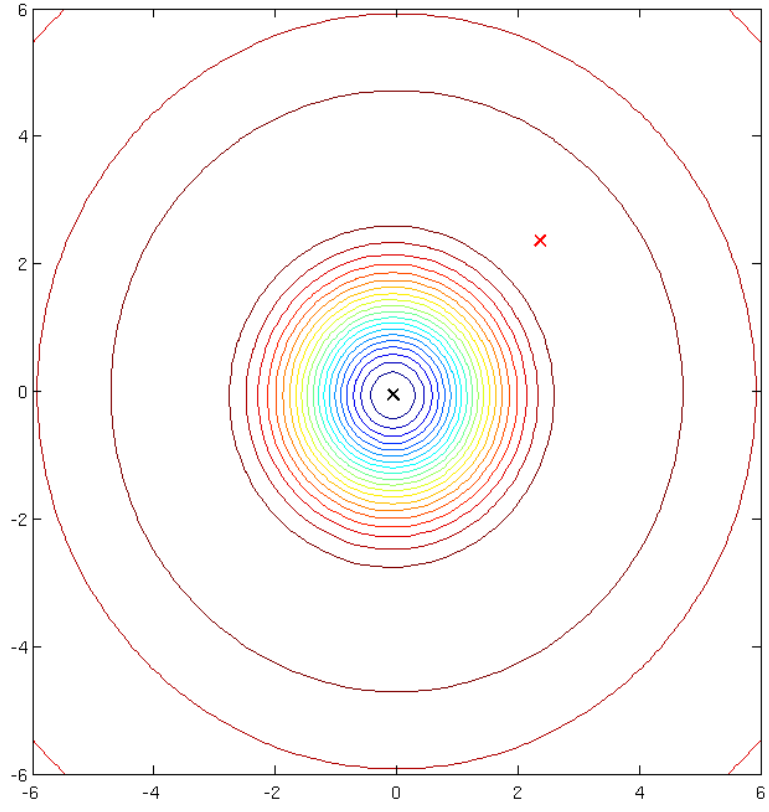
$$x_{N+1} = \operatorname*{argmax}_{x \in \mathcal{X}} \left[2 \int k(x, x') p(x') dx' - \frac{1}{N+1} \sum_{m=1}^{N} k(x, x_m) \right]$$



Kernel Herding in Action

New point is added at:

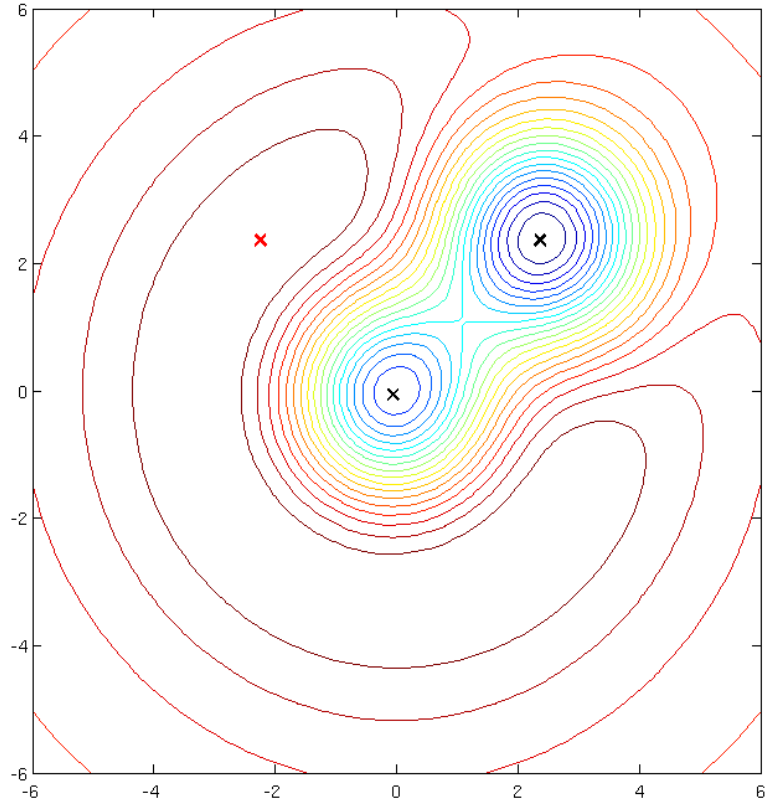
$$x_{N+1} = \operatorname*{argmax}_{x \in \mathcal{X}} \left[2 \int k(x, x') p(x') dx' - \frac{1}{N+1} \sum_{m=1}^{N} k(x, x_m) \right]$$



Kernel Herding in Action

New point is added at:

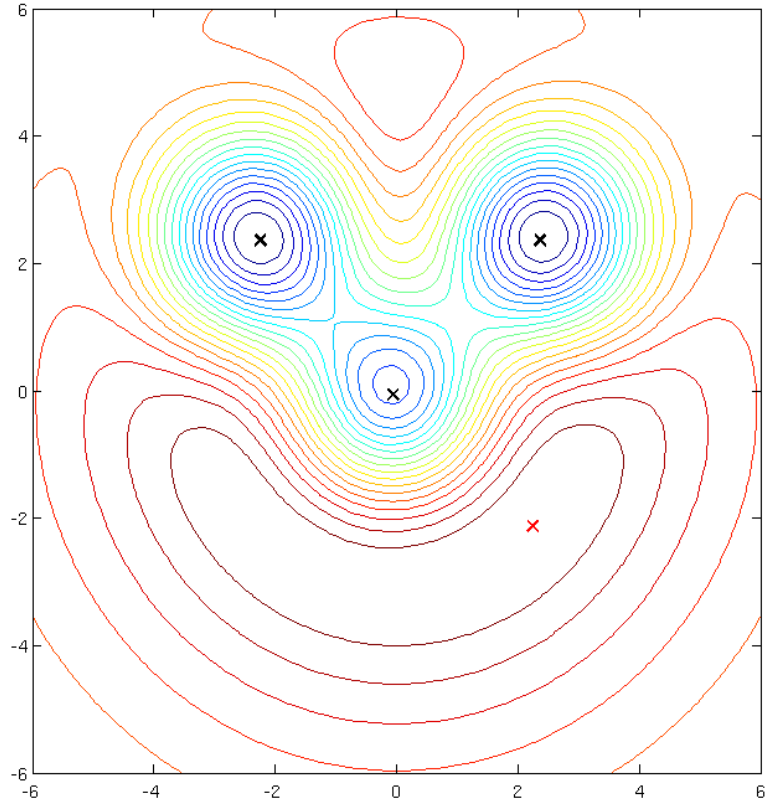
$$x_{N+1} = \operatorname*{argmax}_{x \in \mathcal{X}} \left[2 \int k(x, x') p(x') dx' - \frac{1}{N+1} \sum_{m=1}^{N} k(x, x_m) \right]$$



Kernel Herding in Action

New point is added at:

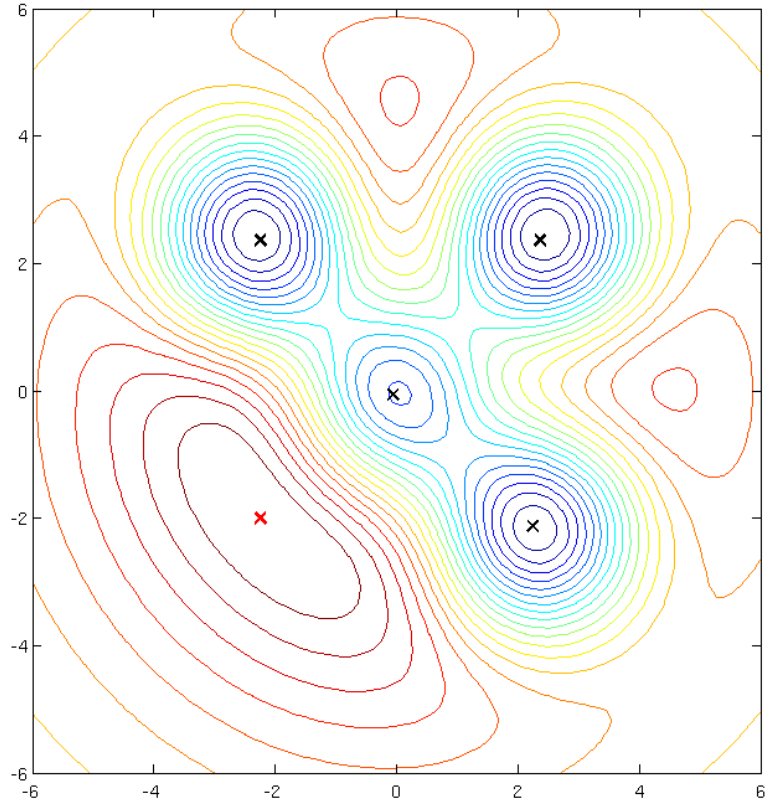
$$x_{N+1} = \operatorname*{argmax}_{x \in \mathcal{X}} \left[2 \int k(x, x') p(x') dx' - \frac{1}{N+1} \sum_{m=1}^{N} k(x, x_m) \right]$$



Kernel Herding in Action

New point is added at:

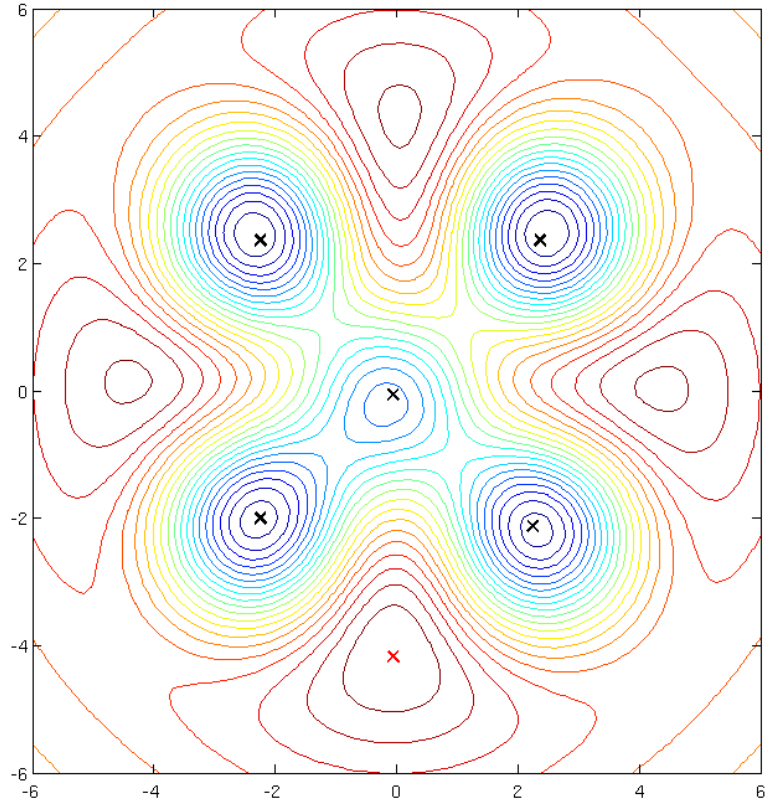
$$x_{N+1} = \operatorname*{argmax}_{x \in \mathcal{X}} \left[2 \int k(x, x') p(x') dx' - \frac{1}{N+1} \sum_{m=1}^{N} k(x, x_m) \right]$$



Kernel Herding in Action

New point is added at:

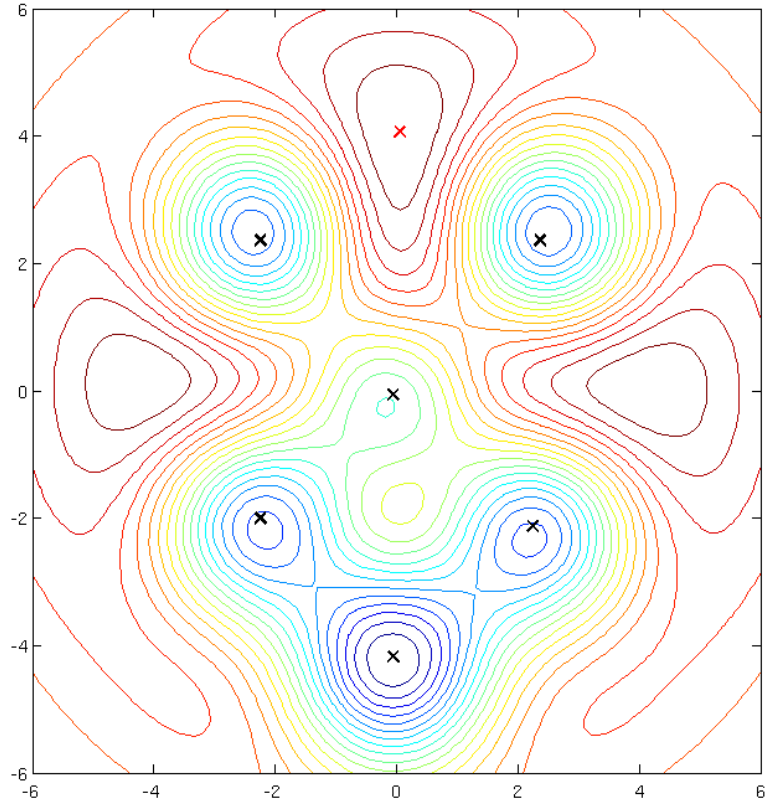
$$x_{N+1} = \operatorname*{argmax}_{x \in \mathcal{X}} \left[2 \int k(x, x') p(x') dx' - \frac{1}{N+1} \sum_{m=1}^{N} k(x, x_m) \right]$$



Kernel Herding in Action

New point is added at:

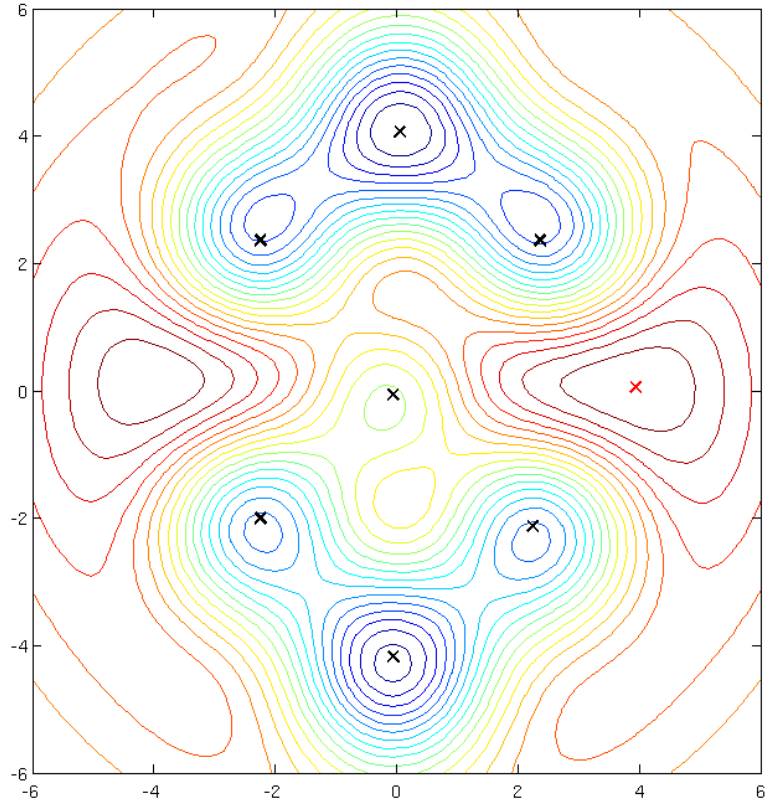
$$x_{N+1} = \underset{x \in \mathcal{X}}{\operatorname{argmax}} \left[2 \int k(x, x') p(x') dx' - \frac{1}{N+1} \sum_{m=1}^{N} k(x, x_m) \right]$$



Kernel Herding in Action

New point is added at:

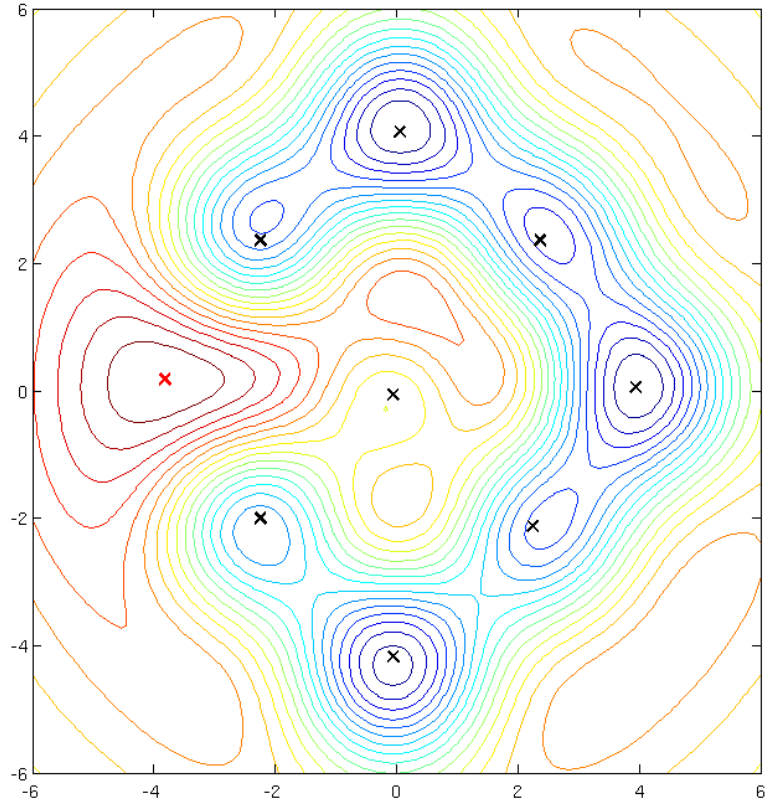
$$x_{N+1} = \operatorname*{argmax}_{x \in \mathcal{X}} \left[2 \int k(x, x') p(x') dx' - \frac{1}{N+1} \sum_{m=1}^{N} k(x, x_m) \right]$$



Kernel Herding in Action

New point is added at:

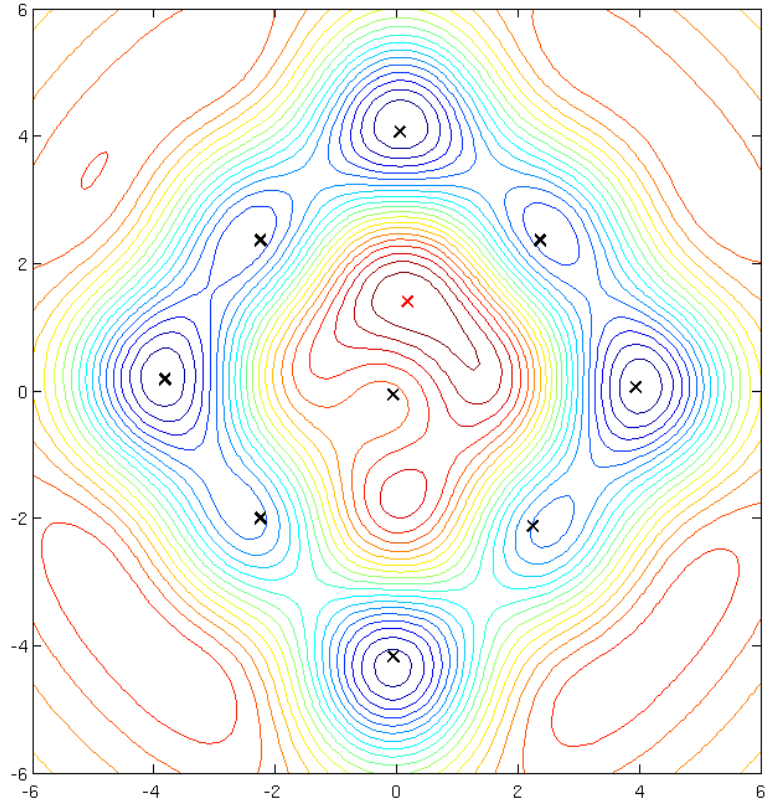
$$x_{N+1} = \operatorname*{argmax}_{x \in \mathcal{X}} \left[2 \int k(x, x') p(x') dx' - \frac{1}{N+1} \sum_{m=1}^{N} k(x, x_m) \right]$$



Kernel Herding in Action

New point is added at:

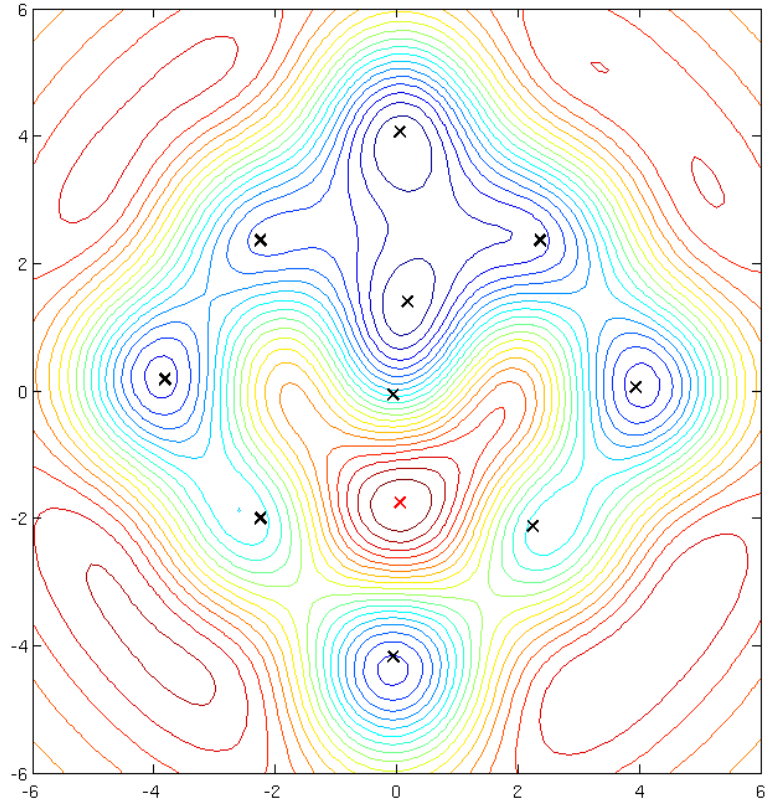
$$x_{N+1} = \operatorname*{argmax}_{x \in \mathcal{X}} \left[2 \int k(x, x') p(x') dx' - \frac{1}{N+1} \sum_{m=1}^{N} k(x, x_m) \right]$$



Kernel Herding in Action

New point is added at:

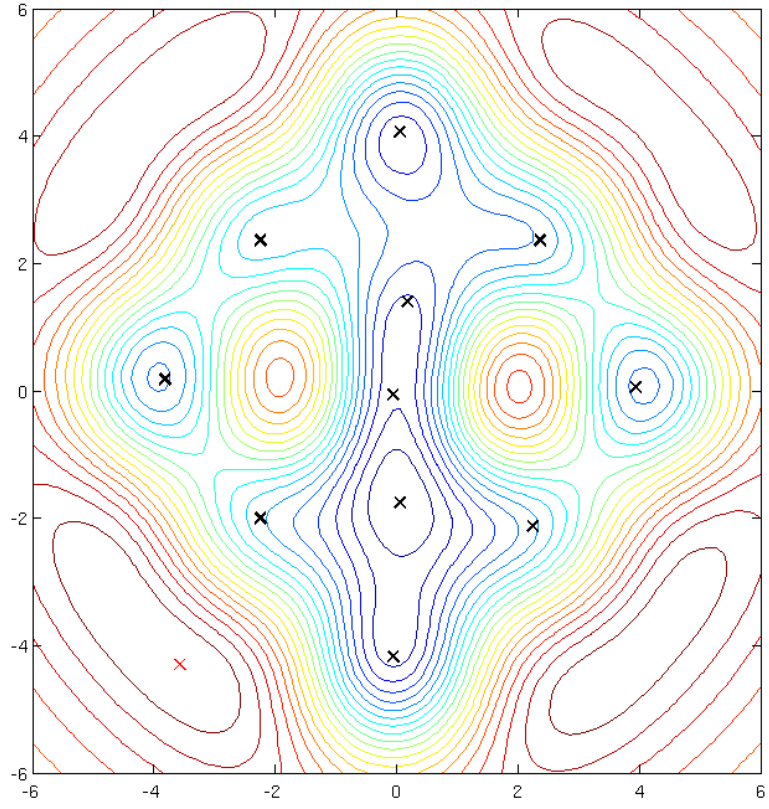
$$x_{N+1} = \operatorname*{argmax}_{x \in \mathcal{X}} \left[2 \int k(x, x') p(x') dx' - \frac{1}{N+1} \sum_{m=1}^{N} k(x, x_m) \right]$$



Kernel Herding in Action

New point is added at:

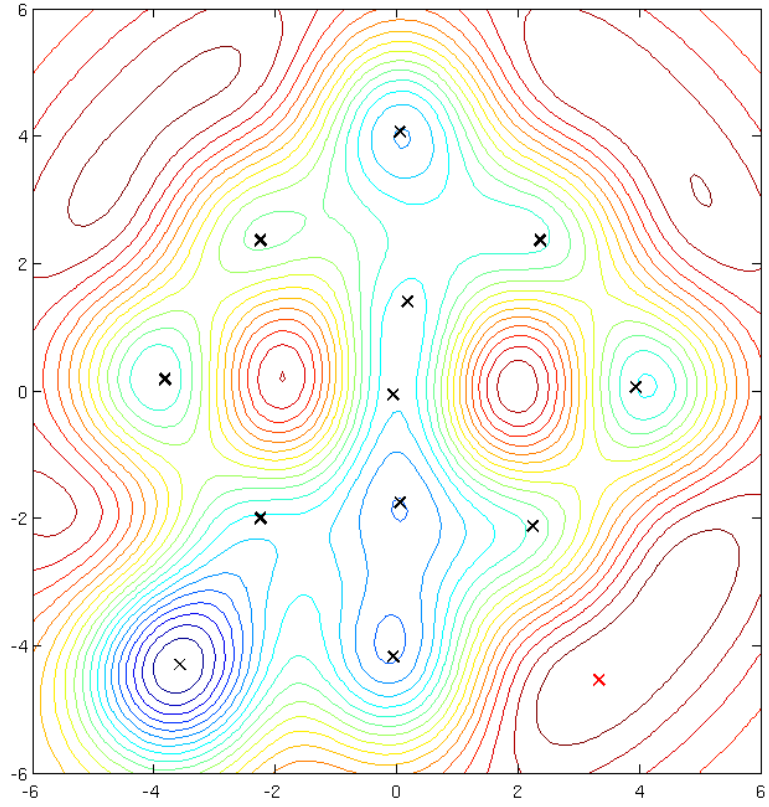
$$x_{N+1} = \operatorname*{argmax}_{x \in \mathcal{X}} \left[2 \int k(x, x') p(x') dx' - \frac{1}{N+1} \sum_{m=1}^{N} k(x, x_m) \right]$$



Kernel Herding in Action

New point is added at:

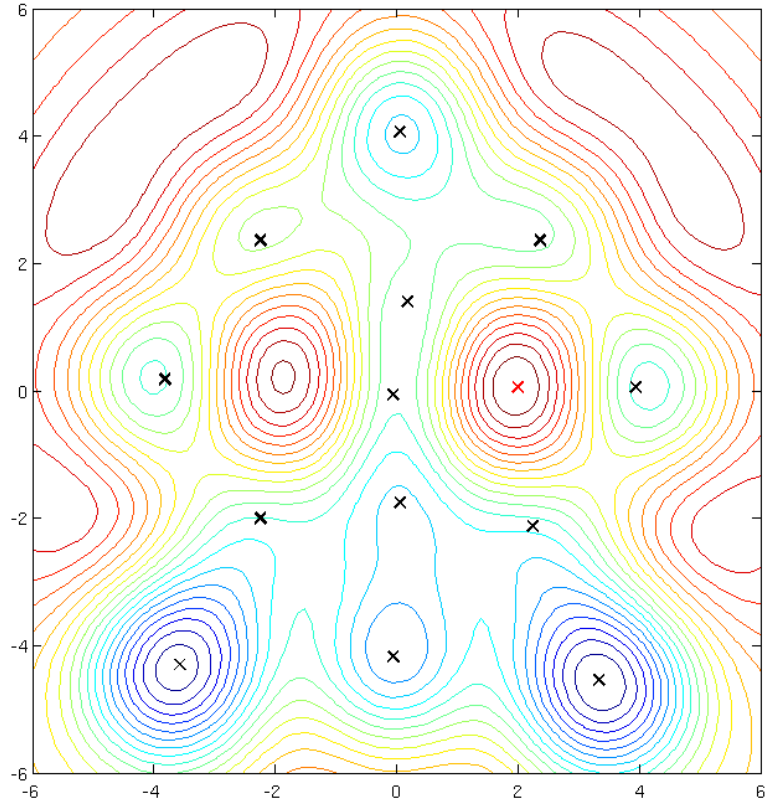
$$x_{N+1} = \operatorname*{argmax}_{x \in \mathcal{X}} \left[2 \int k(x, x') p(x') dx' - \frac{1}{N+1} \sum_{m=1}^{N} k(x, x_m) \right]$$



Kernel Herding in Action

New point is added at:

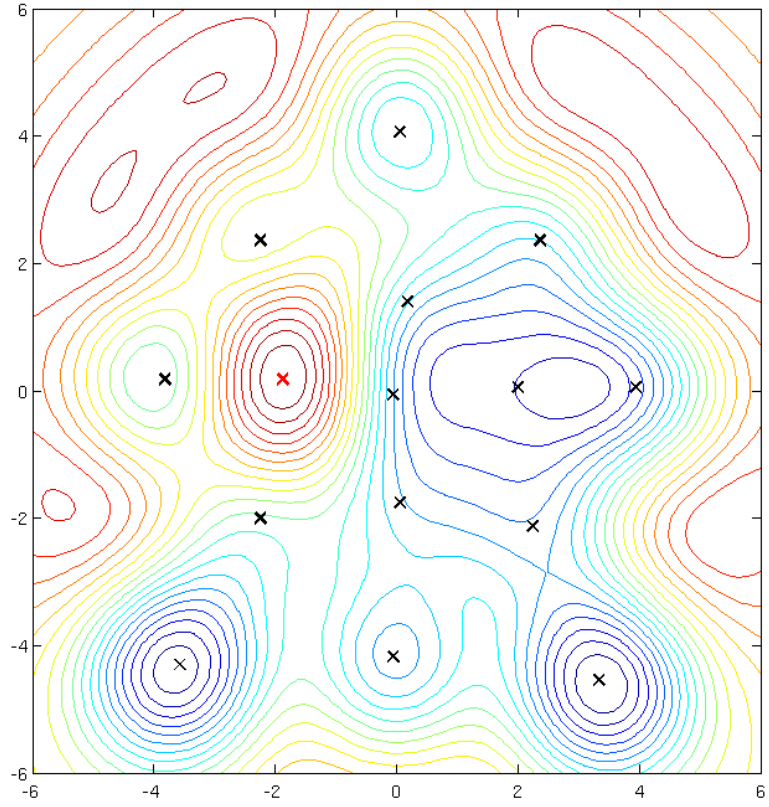
$$x_{N+1} = \operatorname*{argmax}_{x \in \mathcal{X}} \left[2 \int k(x, x') p(x') dx' - \frac{1}{N+1} \sum_{m=1}^{N} k(x, x_m) \right]$$



Kernel Herding in Action

New point is added at:

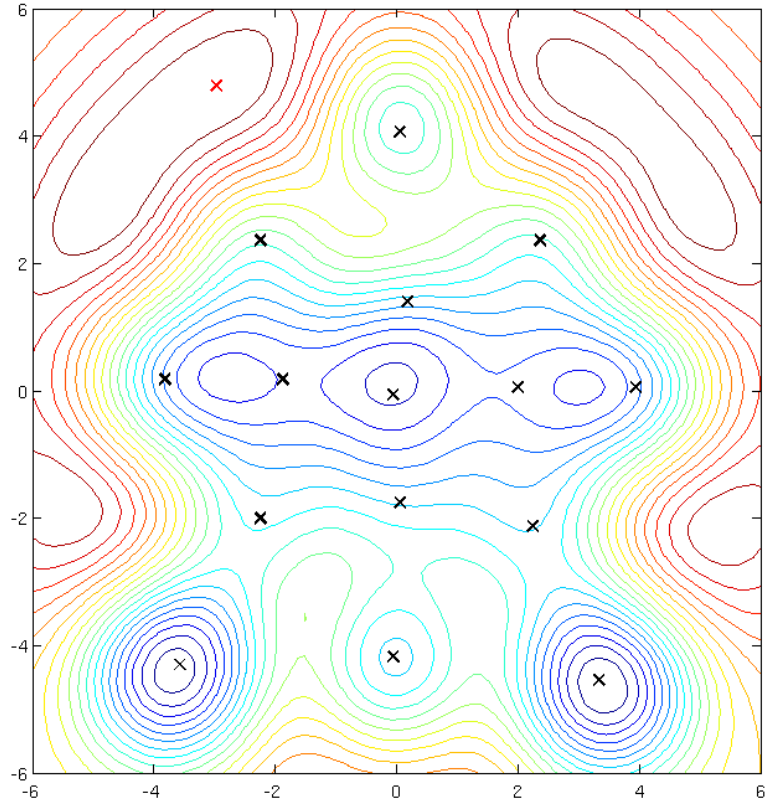
$$x_{N+1} = \operatorname*{argmax}_{x \in \mathcal{X}} \left[2 \int k(x, x') p(x') dx' - \frac{1}{N+1} \sum_{m=1}^{N} k(x, x_m) \right]$$



Kernel Herding in Action

New point is added at:

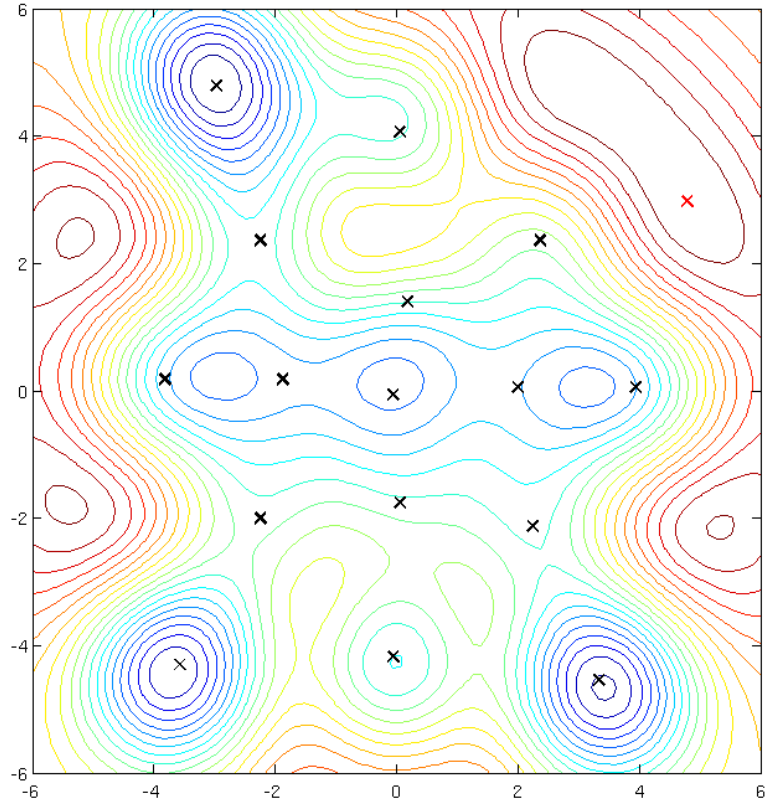
$$x_{N+1} = \operatorname*{argmax}_{x \in \mathcal{X}} \left[2 \int k(x, x') p(x') dx' - \frac{1}{N+1} \sum_{m=1}^{N} k(x, x_m) \right]$$



Kernel Herding in Action

New point is added at:

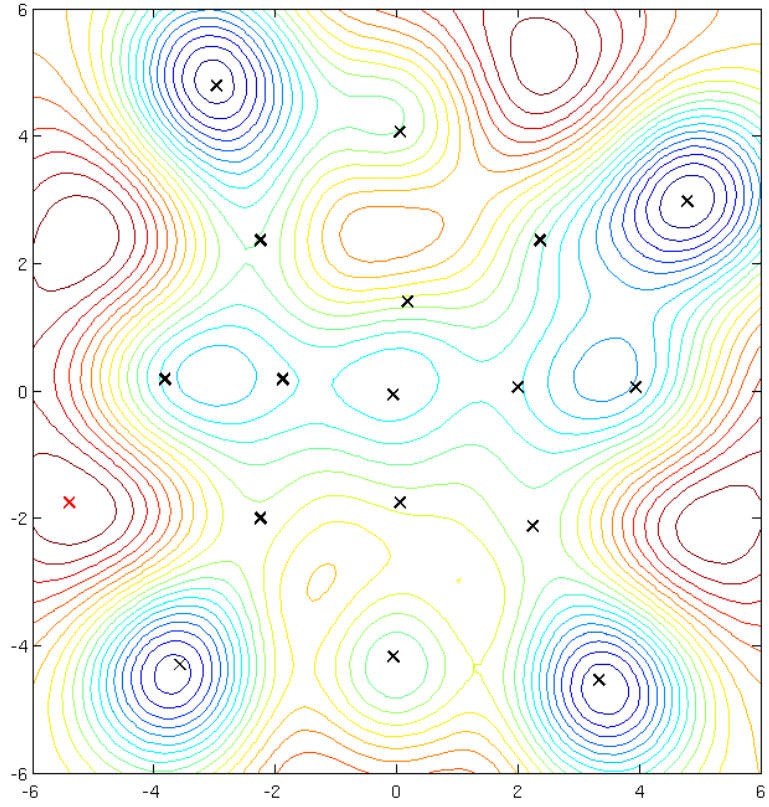
$$x_{N+1} = \operatorname*{argmax}_{x \in \mathcal{X}} \left[2 \int k(x, x') p(x') dx' - \frac{1}{N+1} \sum_{m=1}^{N} k(x, x_m) \right]$$



Kernel Herding in Action

New point is added at:

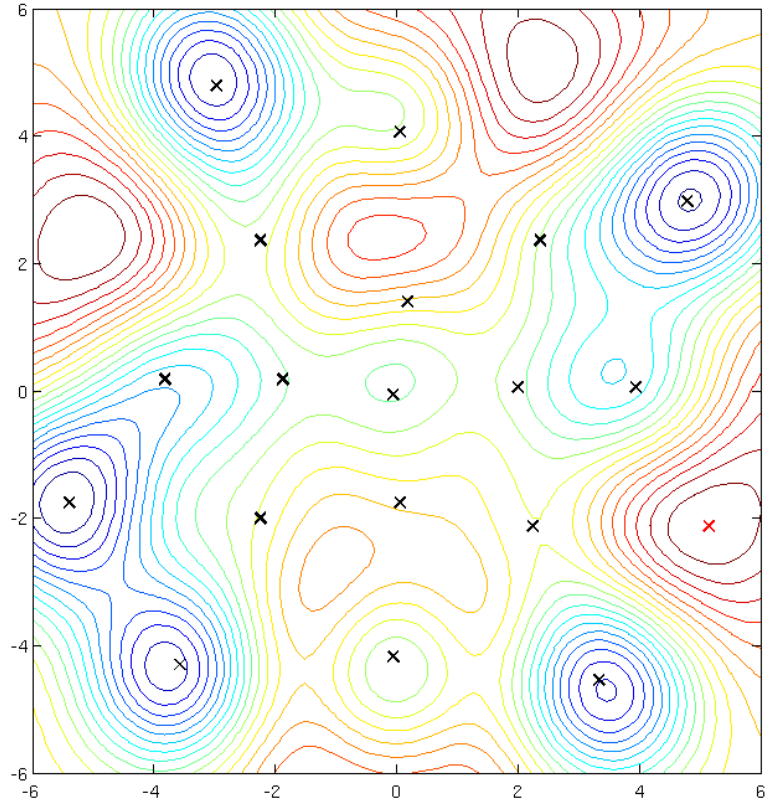
$$x_{N+1} = \operatorname*{argmax}_{x \in \mathcal{X}} \left[2 \int k(x, x') p(x') dx' - \frac{1}{N+1} \sum_{m=1}^{N} k(x, x_m) \right]$$



Kernel Herding in Action

New point is added at:

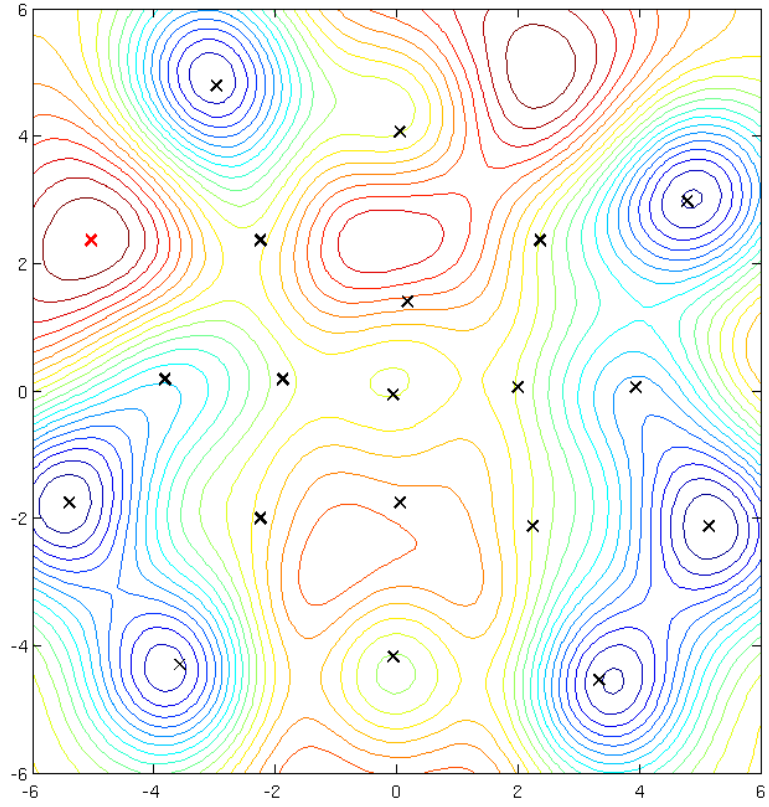
$$x_{N+1} = \operatorname*{argmax}_{x \in \mathcal{X}} \left[2 \int k(x, x') p(x') dx' - \frac{1}{N+1} \sum_{m=1}^{N} k(x, x_m) \right]$$



Kernel Herding in Action

New point is added at:

$$x_{N+1} = \operatorname*{argmax}_{x \in \mathcal{X}} \left[2 \int k(x, x') p(x') dx' - \frac{1}{N+1} \sum_{m=1}^{N} k(x, x_m) \right]$$



Kernel Herding Summary

- A sequential sampling method which minimizes a worst-case divergence, given that $f(x)$ belongs to a given RKHS.
- Like Monte Carlo, weights all samples $f(x_s)$ equally when estimating Z:

$$\hat{Z} = \sum_{i=1}^{N} \frac{1}{N} f(x_i)$$

Kernel Herding Summary

- A sequential sampling method which minimizes a worst-case divergence, given that $f(x)$ belongs to a given RKHS.
- Like Monte Carlo, weights all samples $f(x_s)$ equally when estimating Z:

$$\hat{Z} = \sum_{i=1}^{N} \frac{1}{N} f(x_i)$$

- What if we allowed different weights?

Kernel Herding Summary

- A sequential sampling method which minimizes a worst-case divergence, given that $f(x)$ belongs to a given RKHS.
- Like Monte Carlo, weights all samples $f(x_s)$ equally when estimating Z:

$$\hat{Z} = \sum_{i=1}^{N} \frac{1}{N} f(x_i)$$

- What if we allowed different weights?
- [Bach et. al. 2012] looked at weighted herding strategies, showed improvement in convergence rates.

Kernel Herding Summary

- A sequential sampling method which minimizes a worst-case divergence, given that $f(x)$ belongs to a given RKHS.
- Like Monte Carlo, weights all samples $f(x_s)$ equally when estimating Z:

$$\hat{Z} = \sum_{i=1}^{N} \frac{1}{N} f(x_i)$$

- What if we allowed different weights?
- [Bach et. al. 2012] looked at weighted herding strategies, showed improvement in convergence rates.

Can we reason about the optimal weighting strategy?

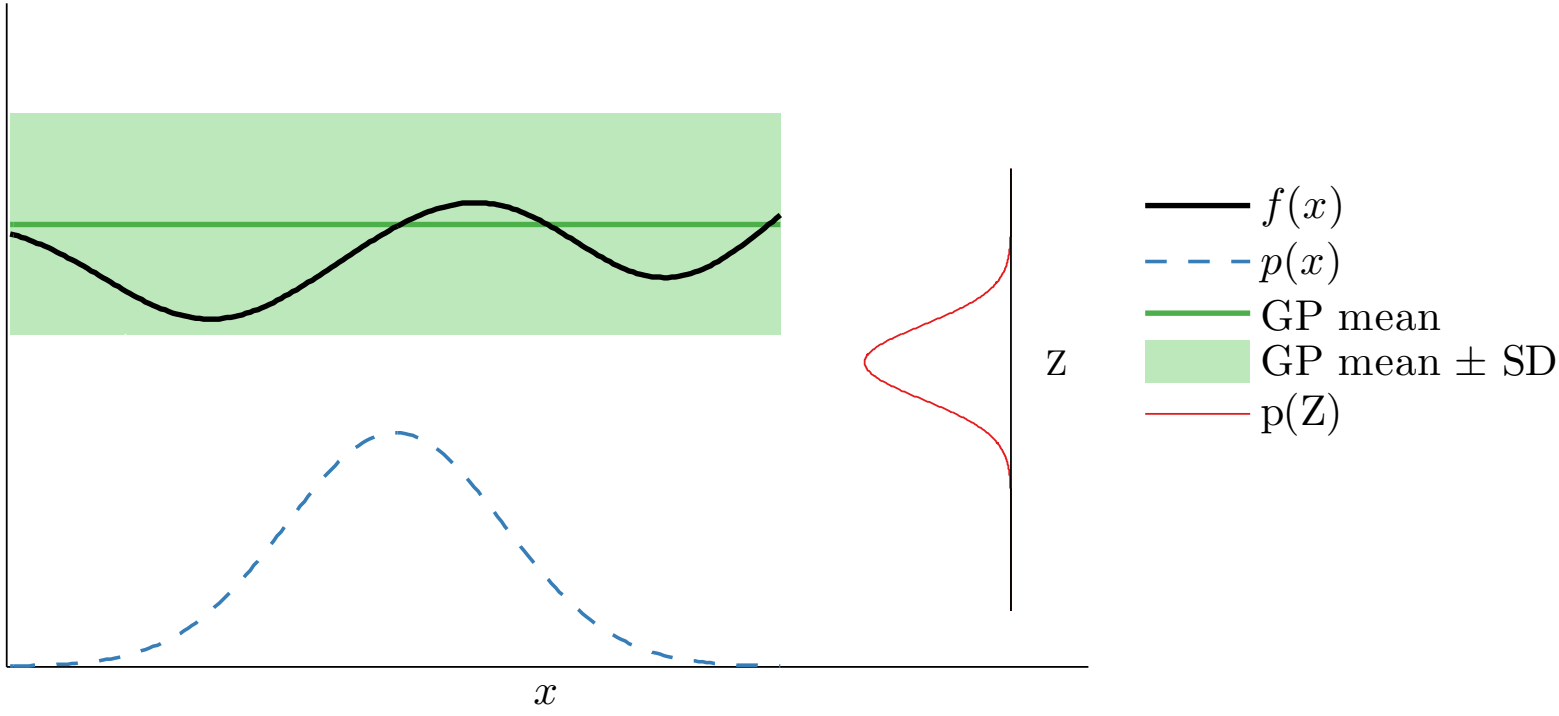
Bayesian Quadrature (a.k.a. Bayesian Monte Carlo)

[O'Hagan 1987, Diaconis 1988, Rasmussen & Ghahramani 2003]

Bayesian Quadrature (a.k.a. Bayesian Monte Carlo)

[O'Hagan 1987, Diaconis 1988, Rasmussen & Ghahramani 2003]

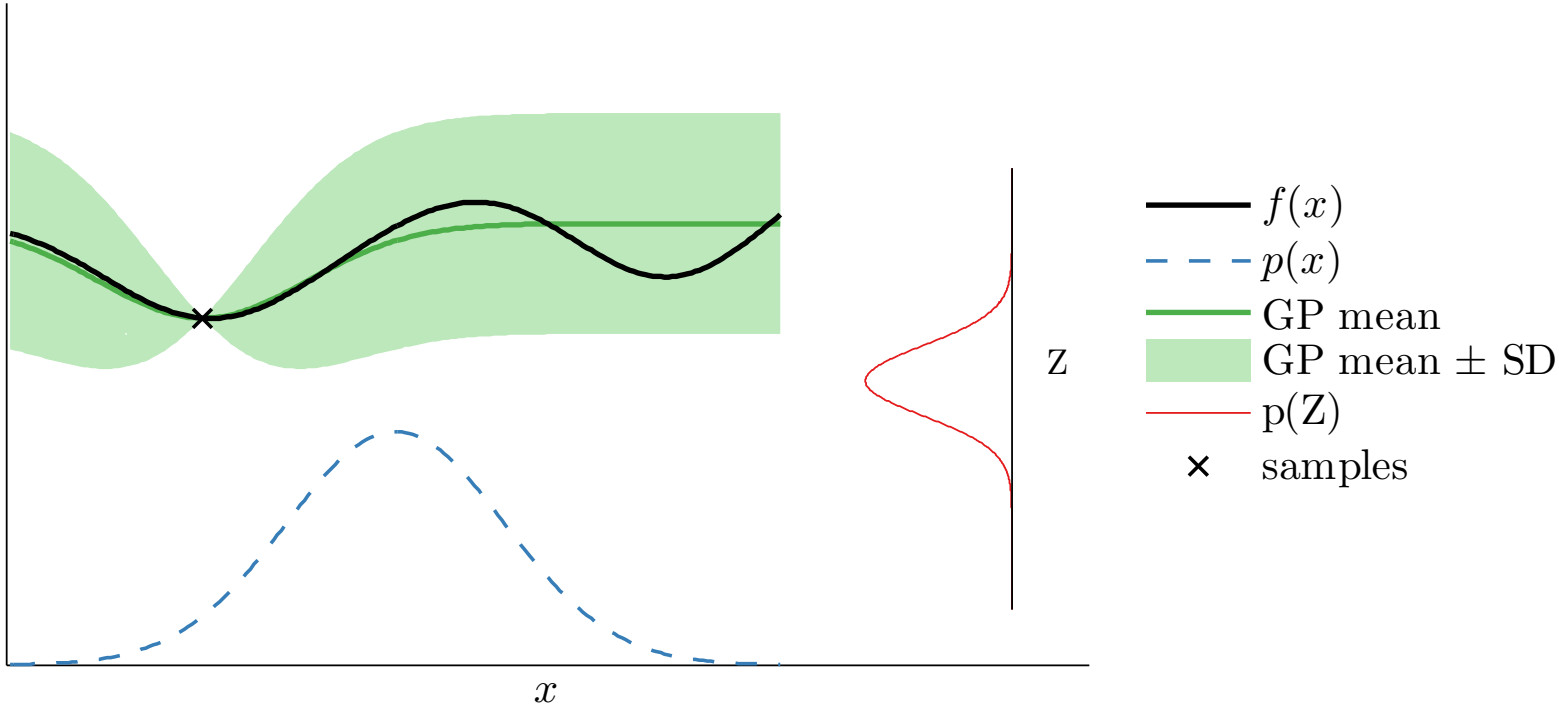
- Places a GP prior on f, defined by $k(\cdot, \cdot)$ and a mean function.
- Posterior over f implies posterior over Z.



Bayesian Quadrature (a.k.a. Bayesian Monte Carlo)

[O'Hagan 1987, Diaconis 1988, Rasmussen & Ghahramani 2003]

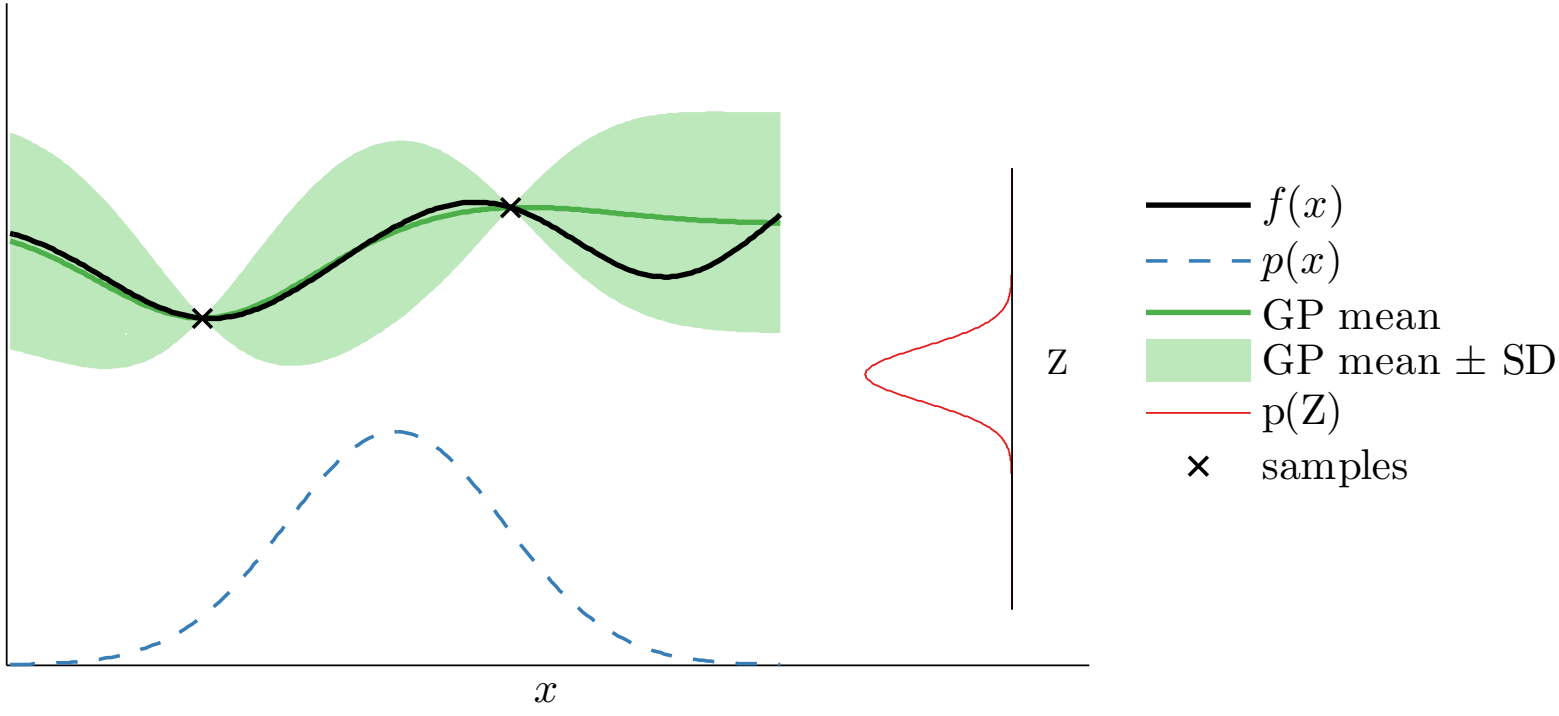
- Places a GP prior on f, defined by $k(\cdot, \cdot)$ and a mean function.
- Posterior over f implies posterior over Z.



Bayesian Quadrature (a.k.a. Bayesian Monte Carlo)

[O'Hagan 1987, Diaconis 1988, Rasmussen & Ghahramani 2003]

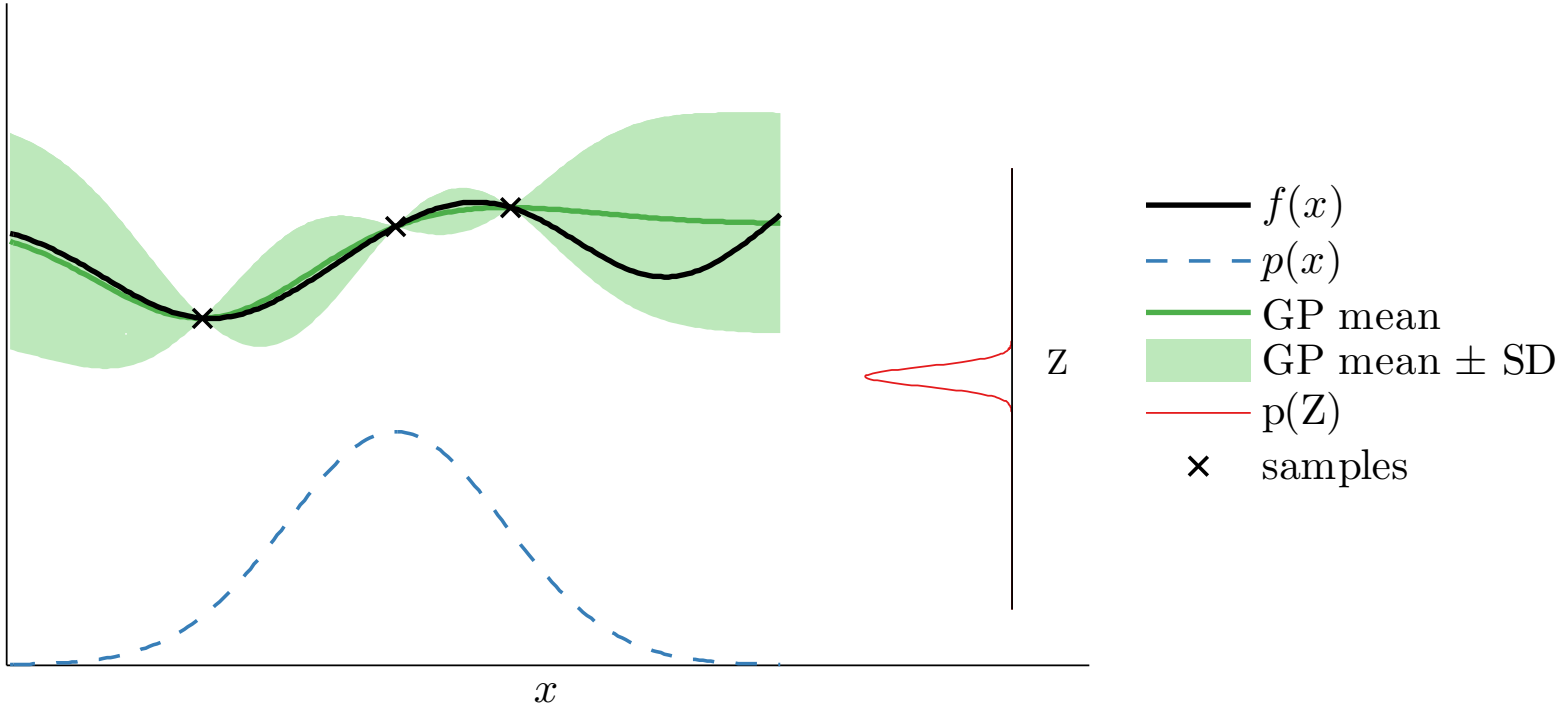
- Places a GP prior on f, defined by $k(\cdot, \cdot)$ and a mean function.
- Posterior over f implies posterior over Z.



Bayesian Quadrature (a.k.a. Bayesian Monte Carlo)

[O'Hagan 1987, Diaconis 1988, Rasmussen & Ghahramani 2003]

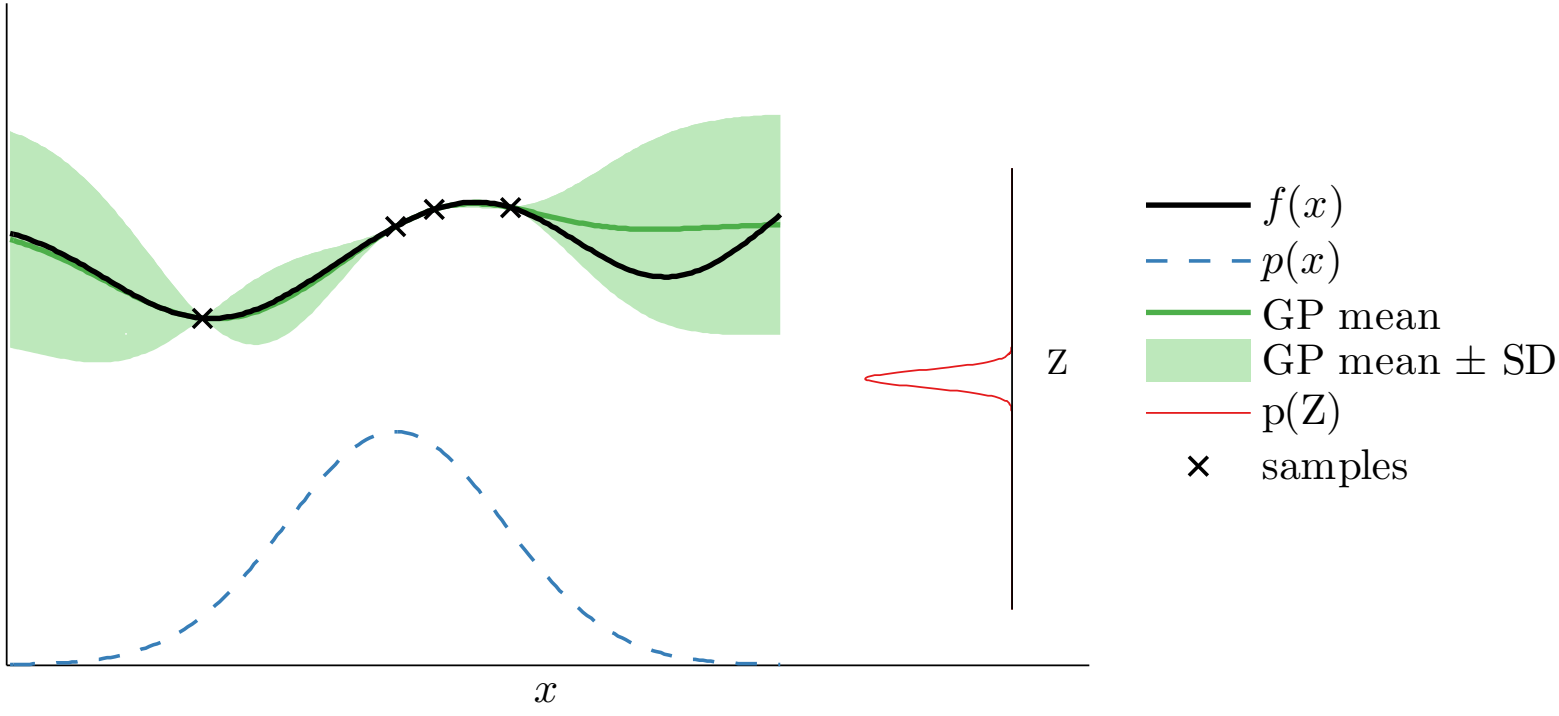
- Places a GP prior on f, defined by $k(\cdot, \cdot)$ and a mean function.
- Posterior over f implies posterior over Z.



Bayesian Quadrature (a.k.a. Bayesian Monte Carlo)

[O'Hagan 1987, Diaconis 1988, Rasmussen & Ghahramani 2003]

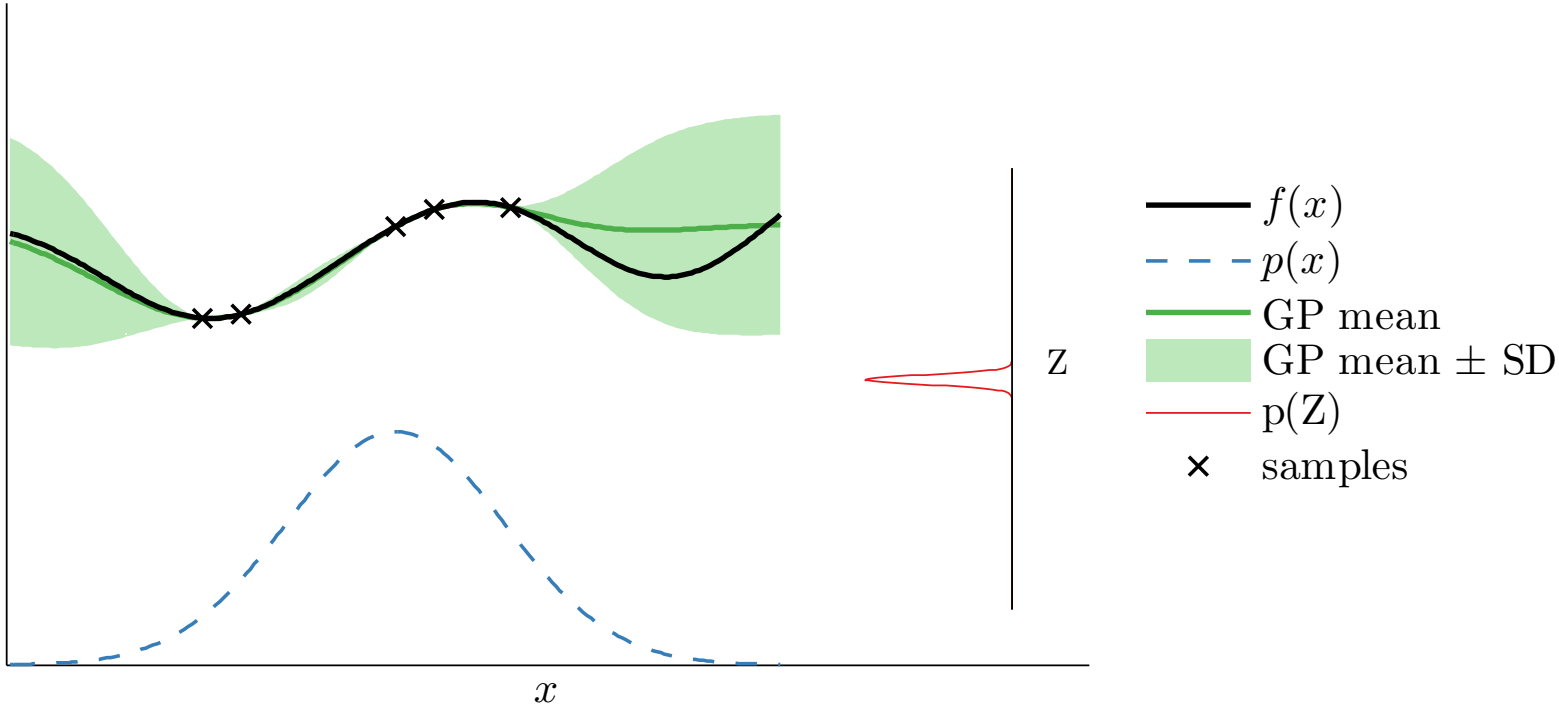
- Places a GP prior on f, defined by $k(\cdot, \cdot)$ and a mean function.
- Posterior over f implies posterior over Z.



Bayesian Quadrature (a.k.a. Bayesian Monte Carlo)

[O'Hagan 1987, Diaconis 1988, Rasmussen & Ghahramani 2003]

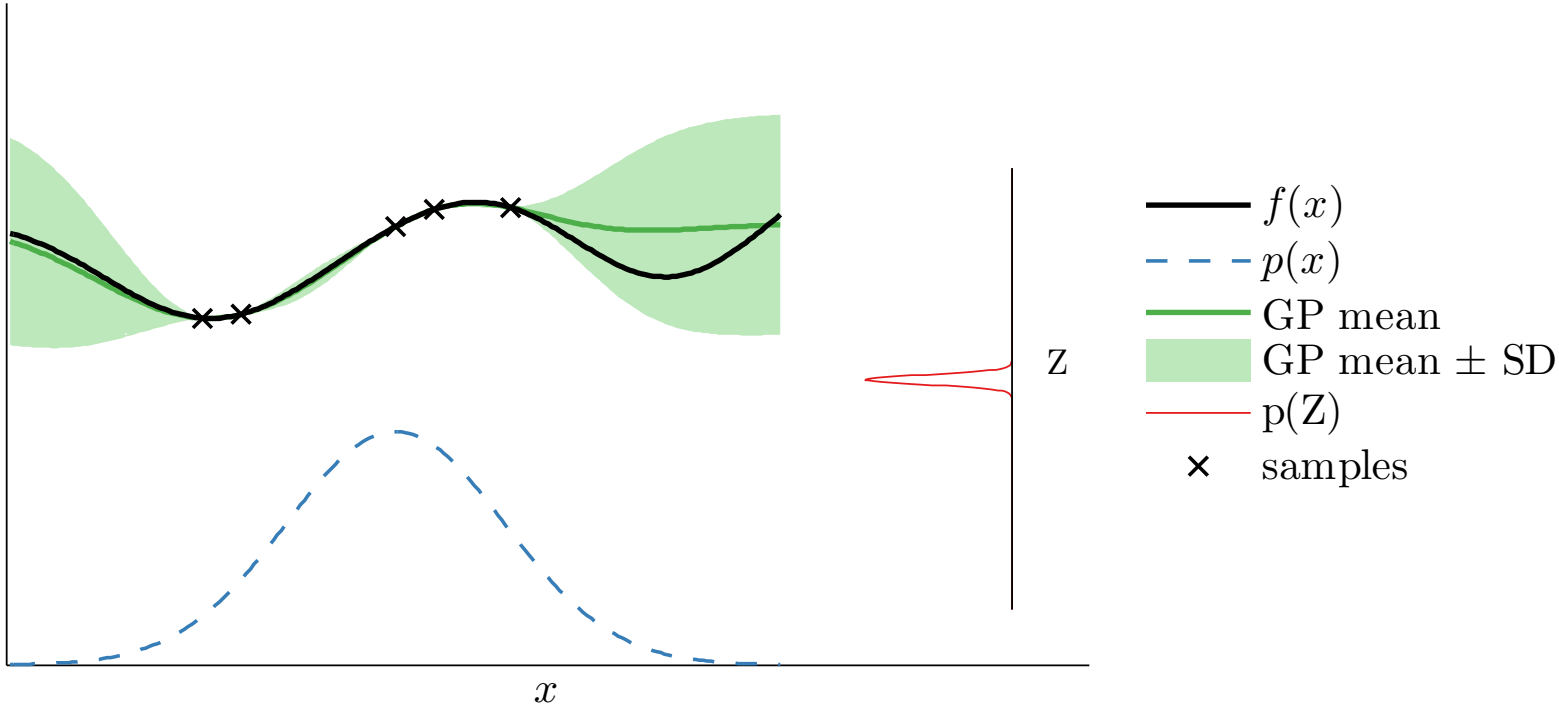
- Places a GP prior on f, defined by $k(\cdot, \cdot)$ and a mean function.
- Posterior over f implies posterior over Z.



Bayesian Quadrature (a.k.a. Bayesian Monte Carlo)

[O'Hagan 1987, Diaconis 1988, Rasmussen & Ghahramani 2003]

- Places a GP prior on f, defined by $k(\cdot, \cdot)$ and a mean function.
- Posterior over f implies posterior over Z.



- Can choose samples however we want.

Bayesian Quadrature Estimator

Posterior over Z has mean linear in $f(x_s)$:

$$\mathbb{E}_{\mathrm{GP}}\left[Z|f(x_s)\right] = \sum_{i=1}^{N} w_{BQ}^{(i)} f(x_i)$$

where

$$w_{BQ} = z^T K^{-1} \qquad \text{and} \qquad z_n = \int k(x, x_n) p(x) dx$$

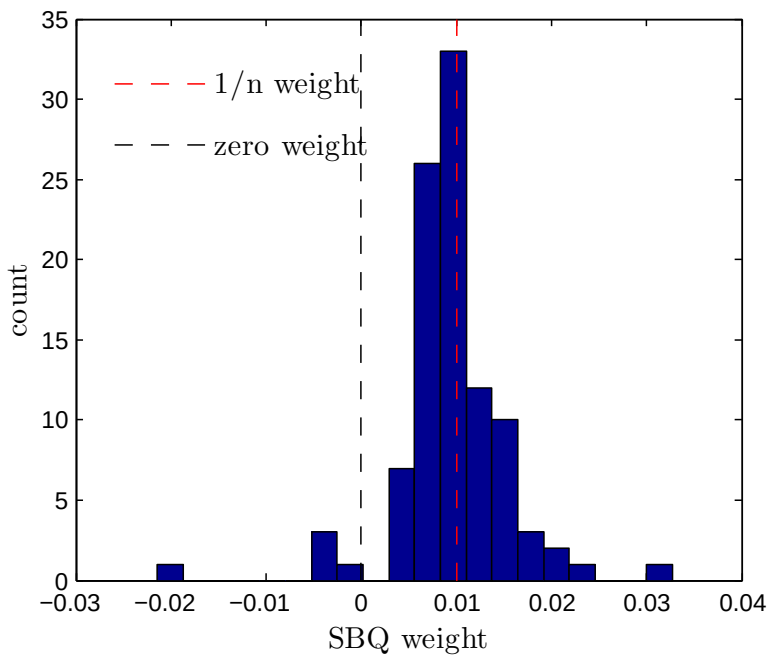
Bayesian Quadrature Estimator

Posterior over Z has mean linear in $f(x_s)$:

$$\mathbb{E}_{\mathrm{GP}}\left[Z|f(x_s)\right] = \sum_{i=1}^{N} w_{BQ}^{(i)} f(x_i)$$

where

$$w_{BQ} = z^T K^{-1} \qquad \text{and} \qquad z_n = \int k(x, x_n) p(x) dx$$



How to select samples?

How to select samples?

- Natural to minimize the posterior variance of Z:

$$\mathbb{V}\left[Z|f(x_s)\right] = \int\int k(x,x')p(x)p(x')dxdx' - z^T K^{-1} z$$

$$\text{where} \qquad z_n = \int k(x,x_n)p(x)dx$$

How to select samples?

- Natural to minimize the posterior variance of Z:

$$\mathbb{V}\left[Z|f(x_s)\right] = \int\int k(x,x')p(x)p(x')dxdx' - z^T K^{-1} z$$

$$\text{where} \qquad z_n = \int k(x,x_n)p(x)dx$$

- Favours samples in regions where $p(x)$ is high, but where covariance with other sample locations is low. Similar flavour to herding objective.

How to select samples?

- Natural to minimize the posterior variance of Z:

$$\mathbb{V}\left[Z|f(x_s)\right] = \int\int k(x,x')p(x)p(x')dxdx' - z^T K^{-1} z$$

$$\text{where} \qquad z_n = \int k(x,x_n)p(x)dx$$

- Favours samples in regions where $p(x)$ is high, but where covariance with other sample locations is low. Similar flavour to herding objective.
- Does not depend on function values

How to select samples?

- Natural to minimize the posterior variance of Z:

$$\mathbb{V}\left[Z|f(x_s)\right] = \int\int k(x,x')p(x)p(x')dxdx' - z^T K^{-1} z$$

$$\text{where} \qquad z_n = \int k(x,x_n)p(x)dx$$

- Favours samples in regions where $p(x)$ is high, but where covariance with other sample locations is low. Similar flavour to herding objective.
- Does not depend on function values
- Can choose samples sequentially: Sequential Bayesian Quadrature.

Relating Objectives

KH and BQ have completely different motivations:

- KH minimizes a worst-case bound
- BQ minimizes a posterior variance

Is there any correspondence?

Relating Objectives

KH and BQ have completely different motivations:

- KH minimizes a worst-case bound
- BQ minimizes a posterior variance

Is there any correspondence?

First Main Result

$$\mathbb{V}\left[Z|f(x_s)\right] = \mathrm{MMD}^2(p, q_{\mathrm{BQ}})$$

Where

$$q_{\mathrm{BQ}}(x) = \sum_{n=1}^{N} w_{\mathrm{BQ}}^{(n)} \delta_{x_n}(x)$$

Relating Objectives

KH and BQ have completely different motivations:

- KH minimizes a worst-case bound
- BQ minimizes a posterior variance

Is there any correspondence?

First Main Result

$$\mathbb{V}\left[Z|f(x_s)\right] = \mathrm{MMD}^2(p, q_{\mathrm{BQ}})$$

Where

$$q_{\mathrm{BQ}}(x) = \sum_{n=1}^{N} w_{\mathrm{BQ}}^{(n)} \delta_{x_n}(x)$$

BQ is minimizing KH objective

Performance

- KH and BQ are minimizing the same objective, but BQ has freedom to choose weights.

Performance

- KH and BQ are minimizing the same objective, but BQ has freedom to choose weights.
- How does this affect performance?

Performance

- KH and BQ are minimizing the same objective, but BQ has freedom to choose weights.
- How does this affect performance?

Second Main Result

BQ estimator is the optimal weighting strategy:

$$\mathbb{V}\left[Z|f(x_s)\right] = \inf_{w\in\mathbb{R}^N} \sup_{\substack{f\in\mathcal{H} \\ \|f\|_{\mathcal{H}}\mathcal{H}=1}} \left| \int f(x)p(x)dx - \sum_{n=1}^{N} w_n f(x_n) \right|^2$$

Performance

- KH and BQ are minimizing the same objective, but BQ has freedom to choose weights.
- How does this affect performance?

Second Main Result

BQ estimator is the optimal weighting strategy:

$$\mathbb{V}\left[Z|f(x_s)\right] = \inf_{w\in\mathbb{R}^N} \sup_{\substack{f\in\mathcal{H} \\ \|f\|_{\mathcal{H}}\mathcal{H}=1}} \left| \int f(x)p(x)dx - \sum_{n=1}^{N} w_n f(x_n) \right|^2$$

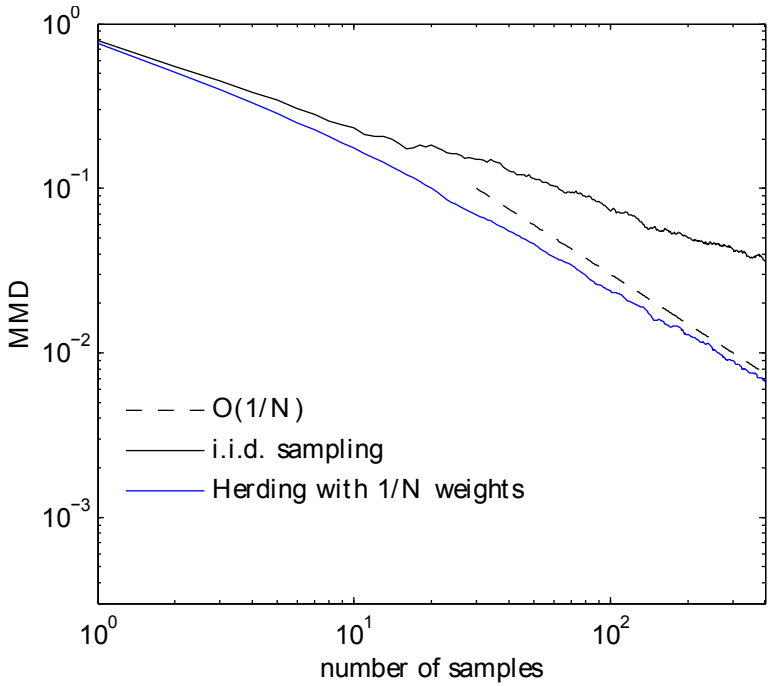
$\mathbb{V}\left[Z|f(x_s)\right]$ has two interpretations:

- Bayesian: posterior variance of Z under a GP prior.
- Frequentist: tight bound on estimation error of Z.

Rates of Convergence

What is rate of convergence of BQ?

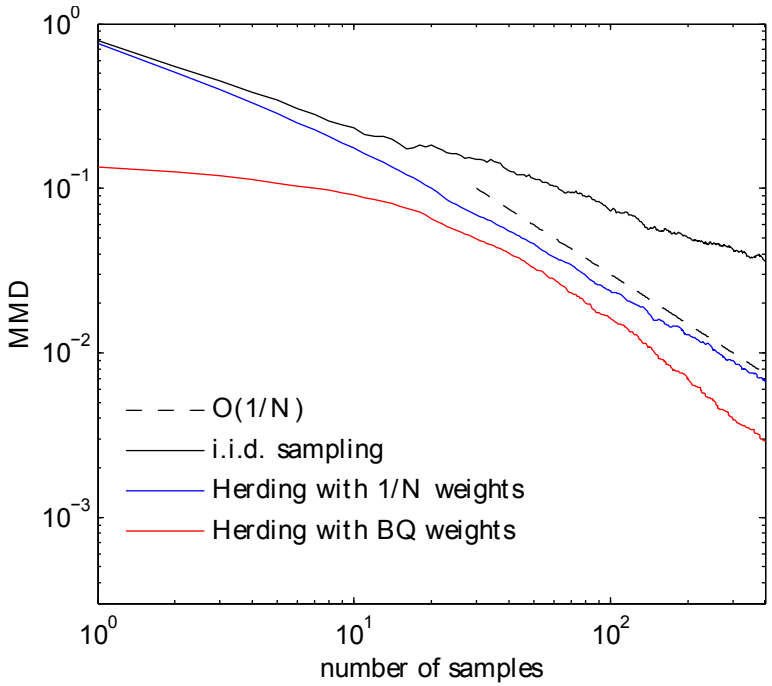
Expected Variance / MMD



Rates of Convergence

What is rate of convergence of BQ?

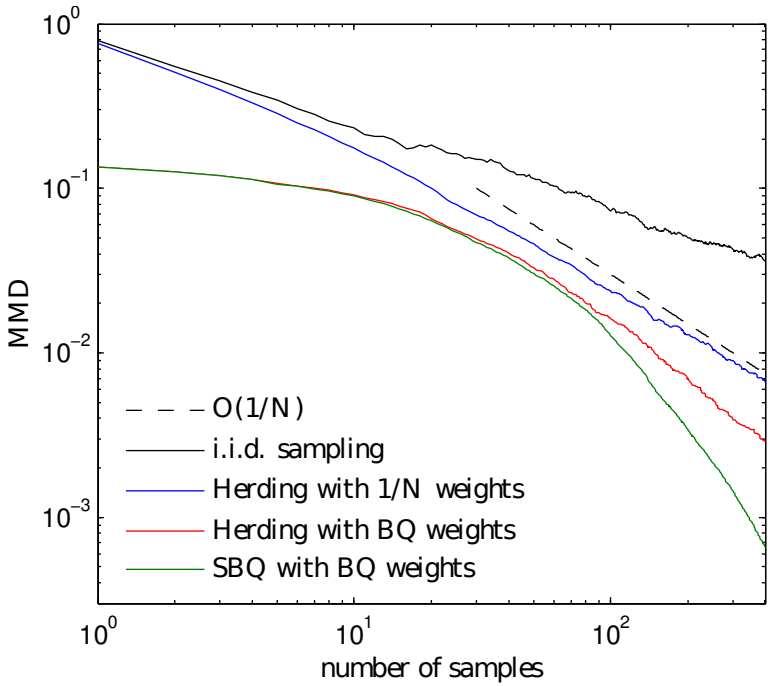
Expected Variance / MMD



Rates of Convergence

What is rate of convergence of BQ?

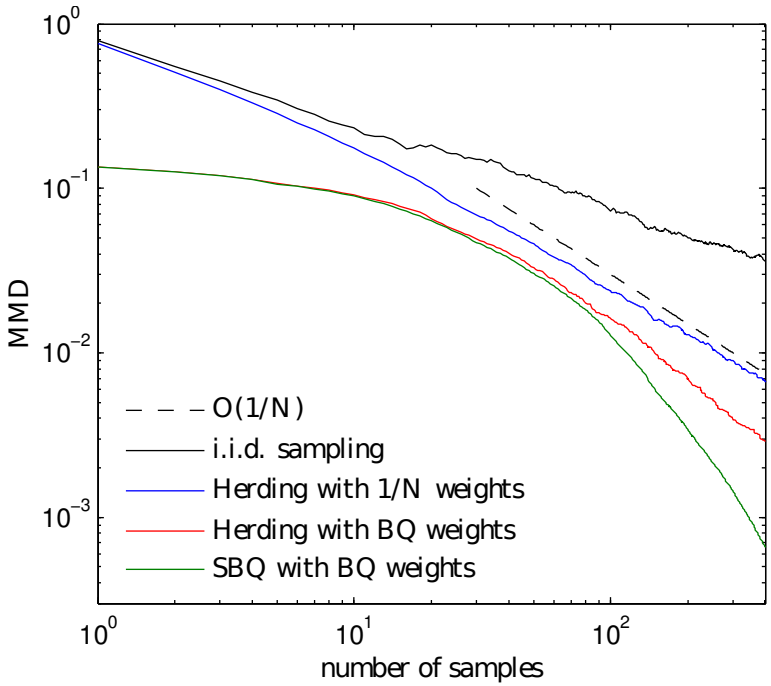
Expected Variance / MMD



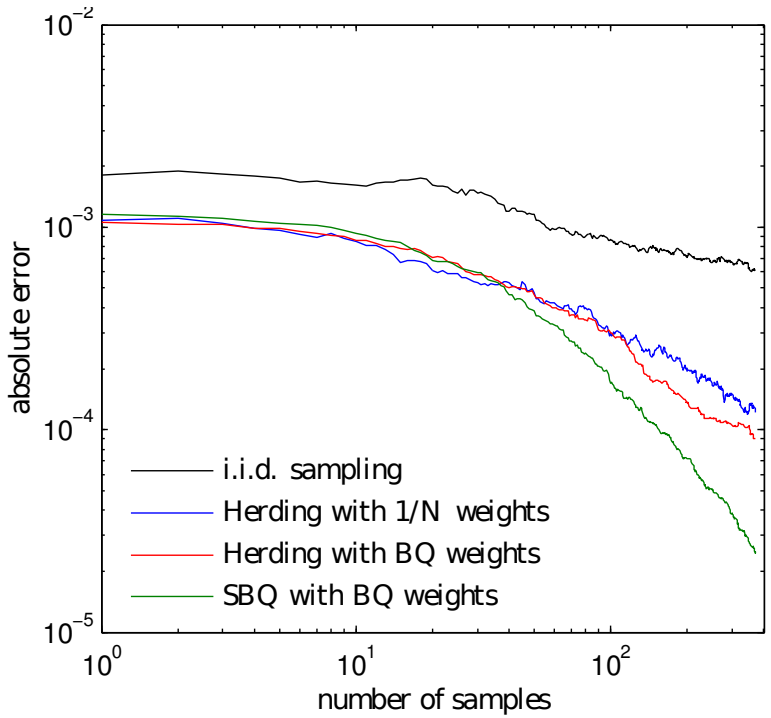
Rates of Convergence

What is rate of convergence of BQ?

Expected Variance / MMD



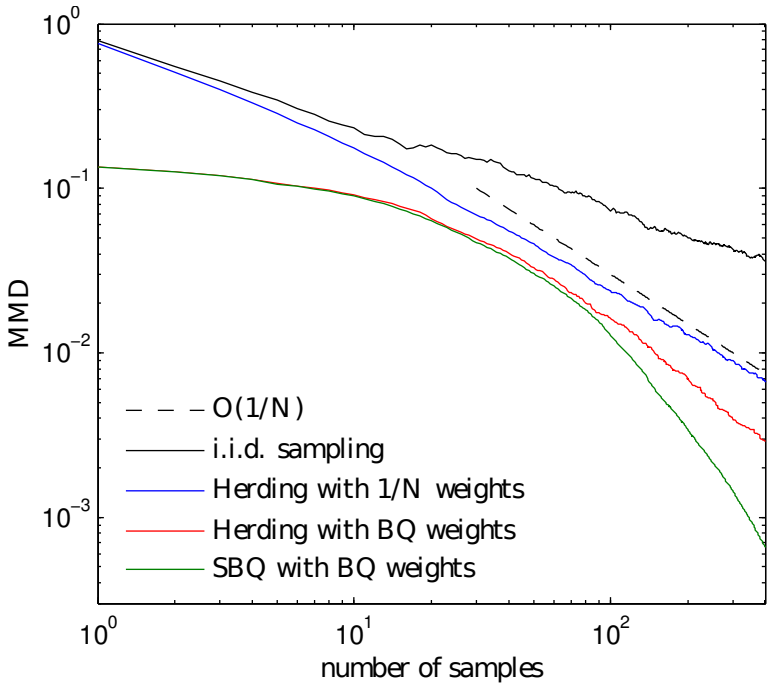
Empirical Rates in RKHS



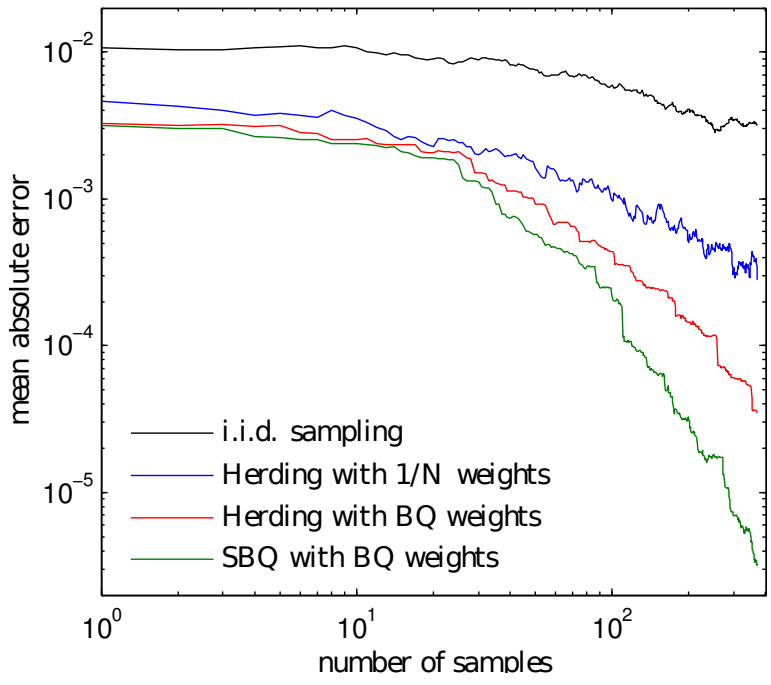
Rates of Convergence

What is rate of convergence of BQ?

Expected Variance / MMD



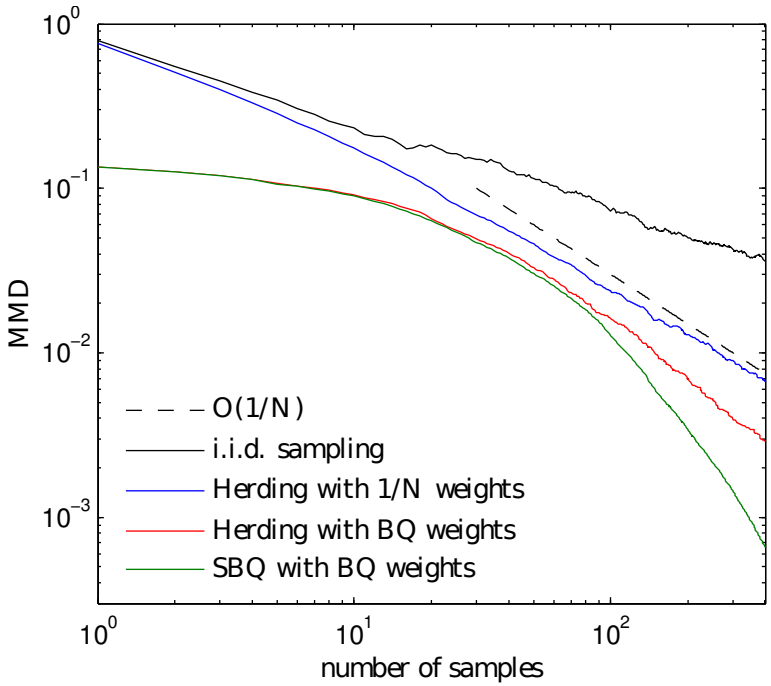
Empirical Rates out of RKHS



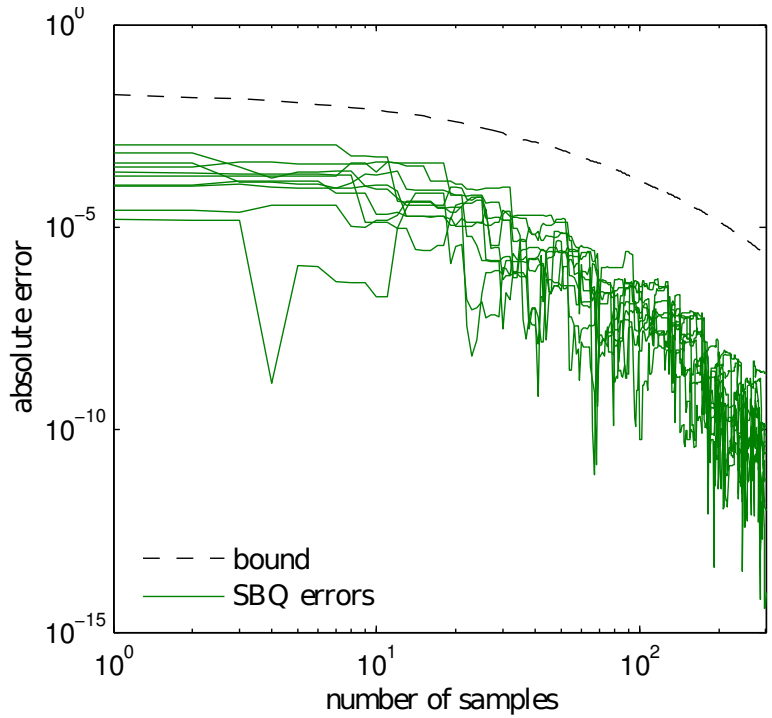
Rates of Convergence

What is rate of convergence of BQ?

Expected Variance / MMD



Bound on Bayesian Error



Summary

- Posterior variance of Z under GP prior is equivalent to Maximum Mean Discrepancy.

Summary

- Posterior variance of Z under GP prior is equivalent to Maximum Mean Discrepancy.
- RKHS assumption gives a tight, closed-form upper bound on Bayesian error.

Summary

- Posterior variance of Z under GP prior is equivalent to Maximum Mean Discrepancy.
- RKHS assumption gives a tight, closed-form upper bound on Bayesian error.
- BQ has very fast, but unknown convergence rate.

Summary

- Posterior variance of Z under GP prior is equivalent to Maximum Mean Discrepancy.
- RKHS assumption gives a tight, closed-form upper bound on Bayesian error.
- BQ has very fast, but unknown convergence rate.
- The optimal weighted herding strategy is Bayesian quadrature.

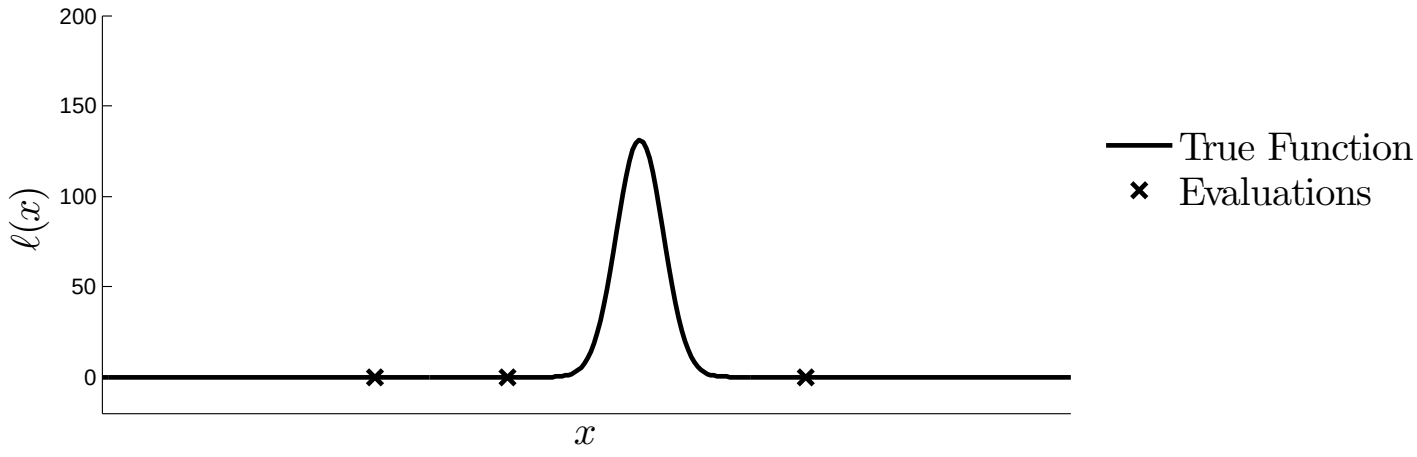
Summary

- Posterior variance of Z under GP prior is equivalent to Maximum Mean Discrepancy.
- RKHS assumption gives a tight, closed-form upper bound on Bayesian error.
- BQ has very fast, but unknown convergence rate.
- The optimal weighted herding strategy is Bayesian quadrature.
- Joint work with Ferenc Huzsar

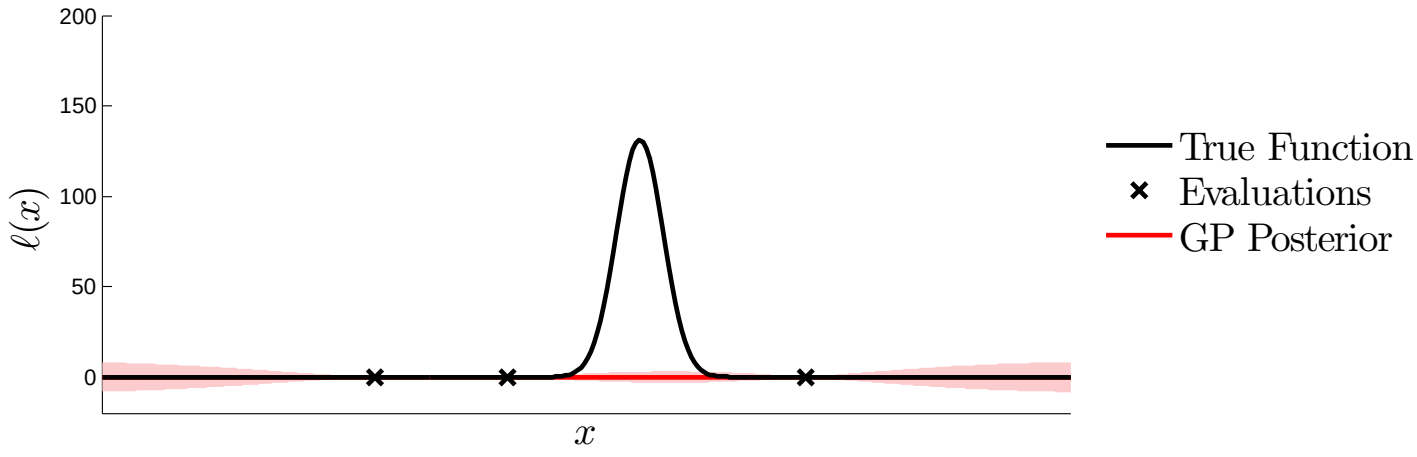
Outline

- Optimally-weighted Herding is Bayesian Quadrature
  - Kernel Herding
  - Bayesian Quadrature
  - Unifying Results
  - Demos
- Frequentist Methods, Bayesian Takeaways
  - Kernel Herding
  - Mean Embeddings
  - Kernel Two-sample Test
  - Hilbert-Schmidt Independence Criterion
  - Determinantal Point Processes

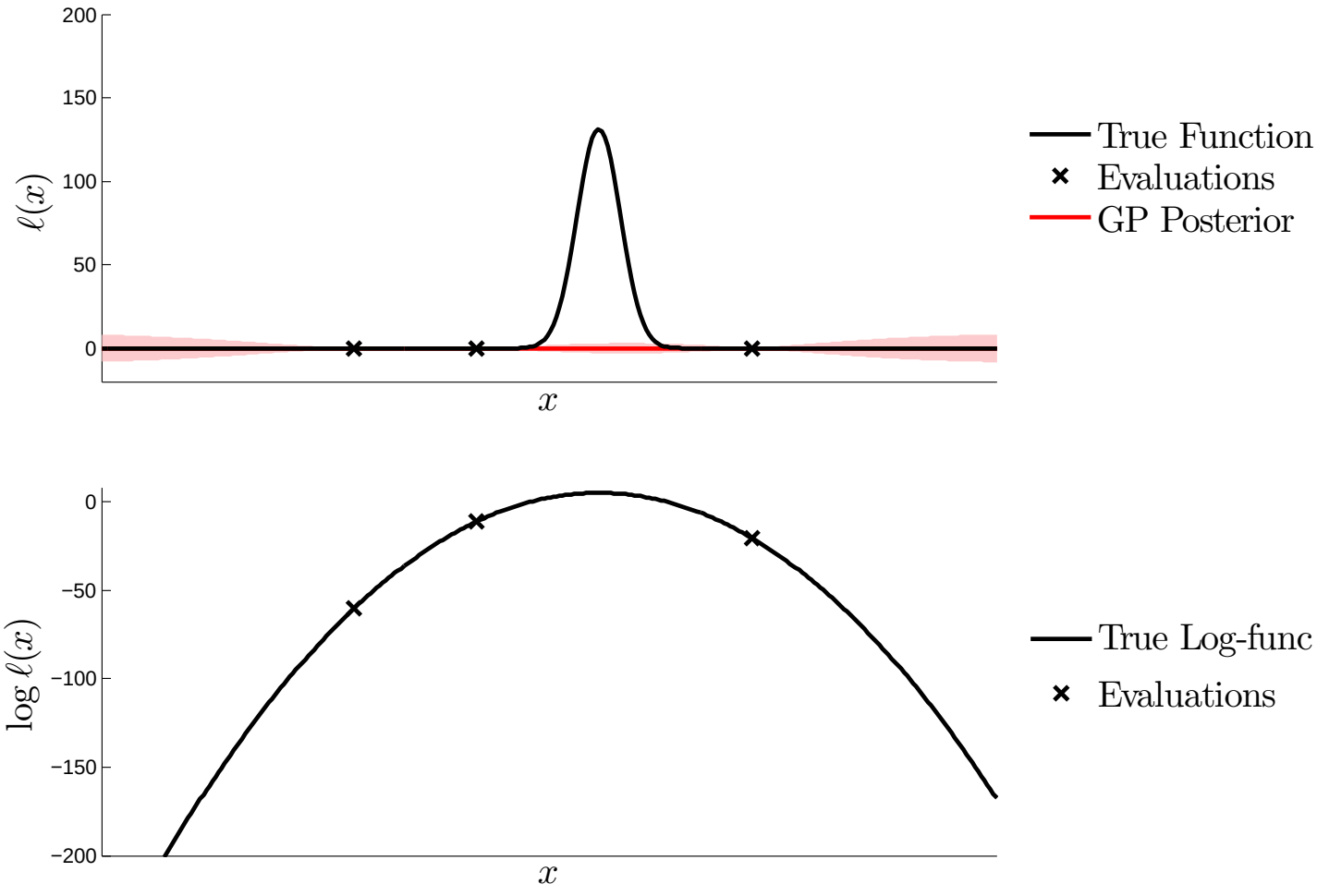
GPs vs Log-GPs for Inference



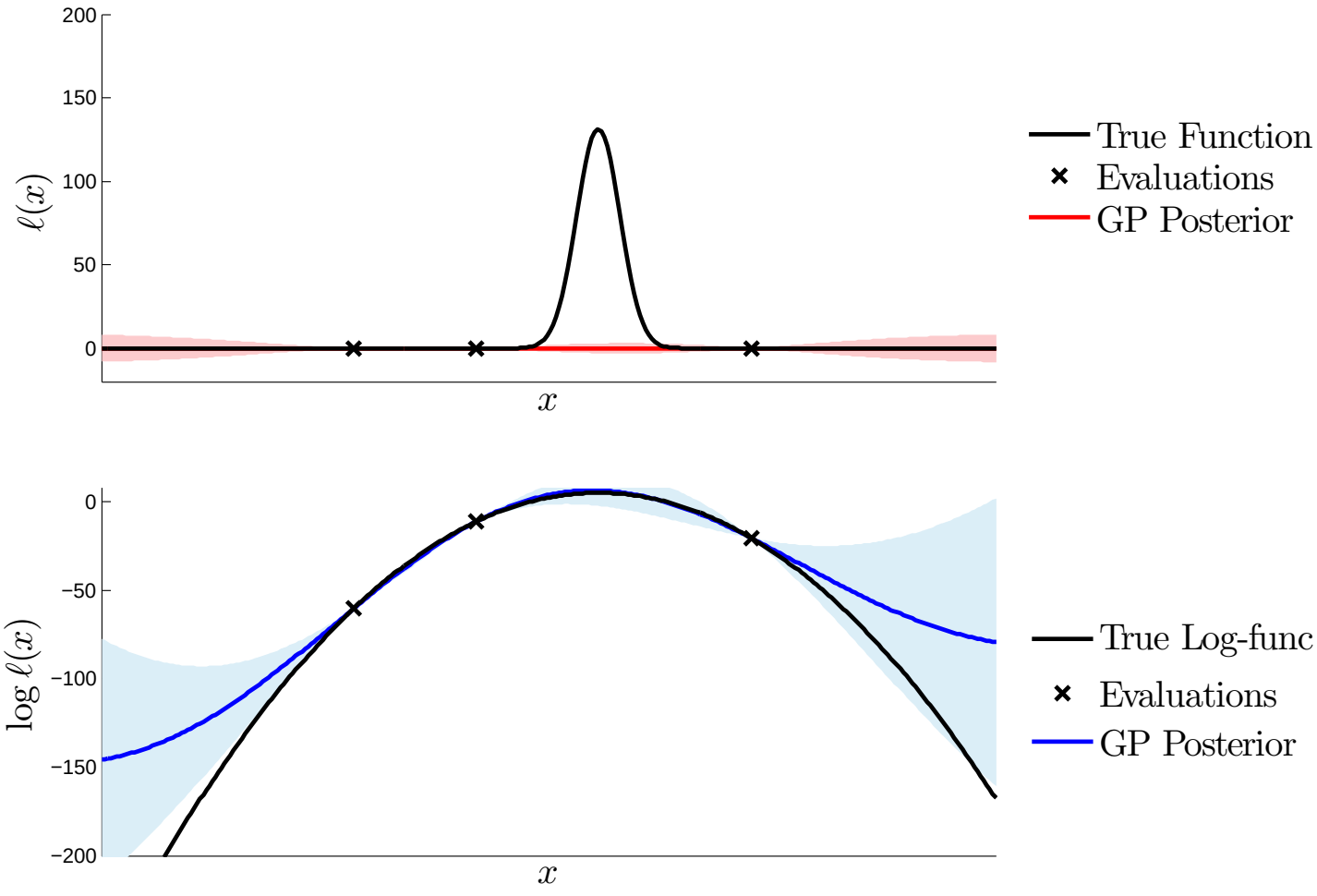
GPs vs Log-GPs for Inference



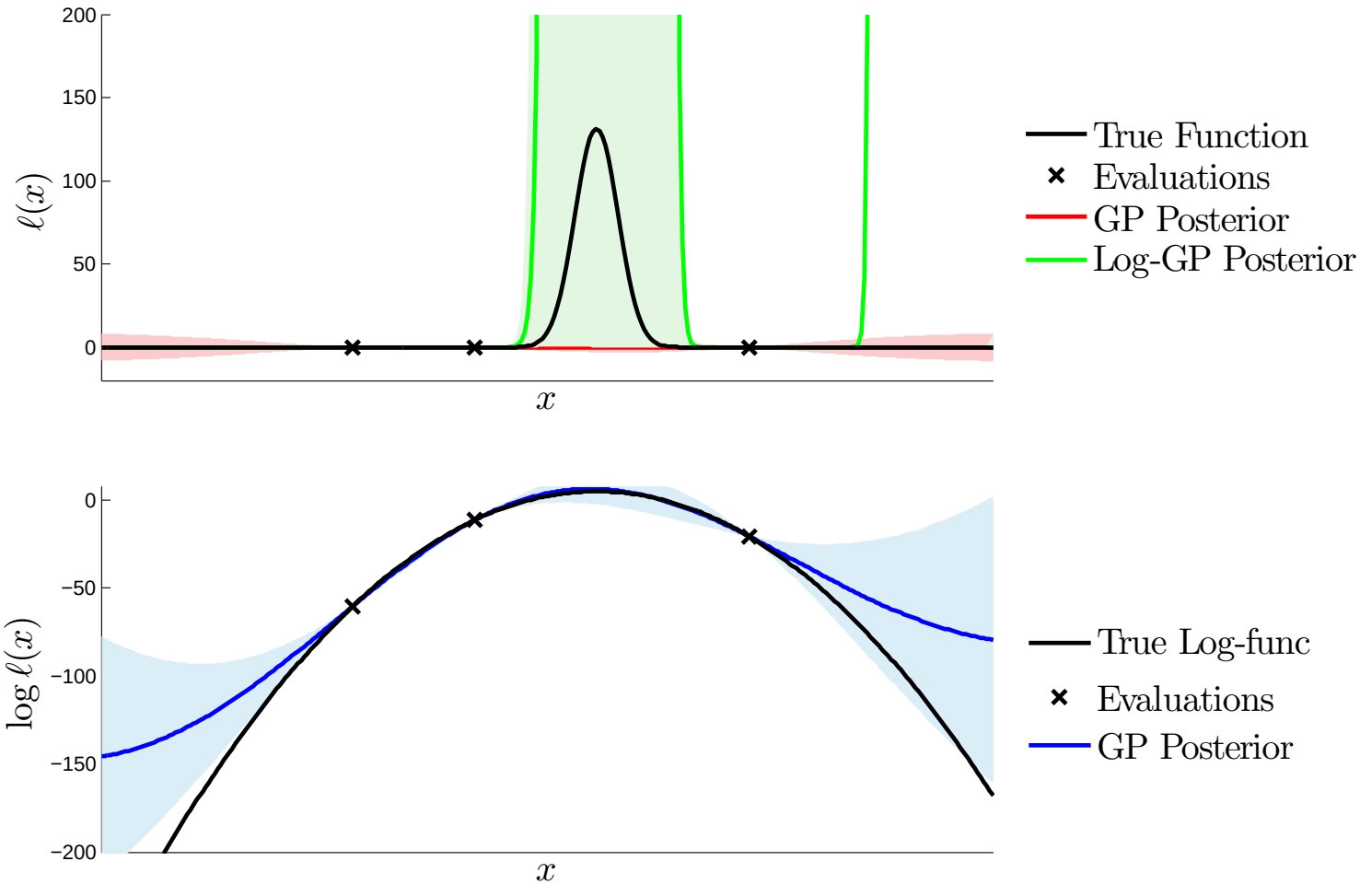
GPs vs Log-GPs for Inference



GPs vs Log-GPs for Inference

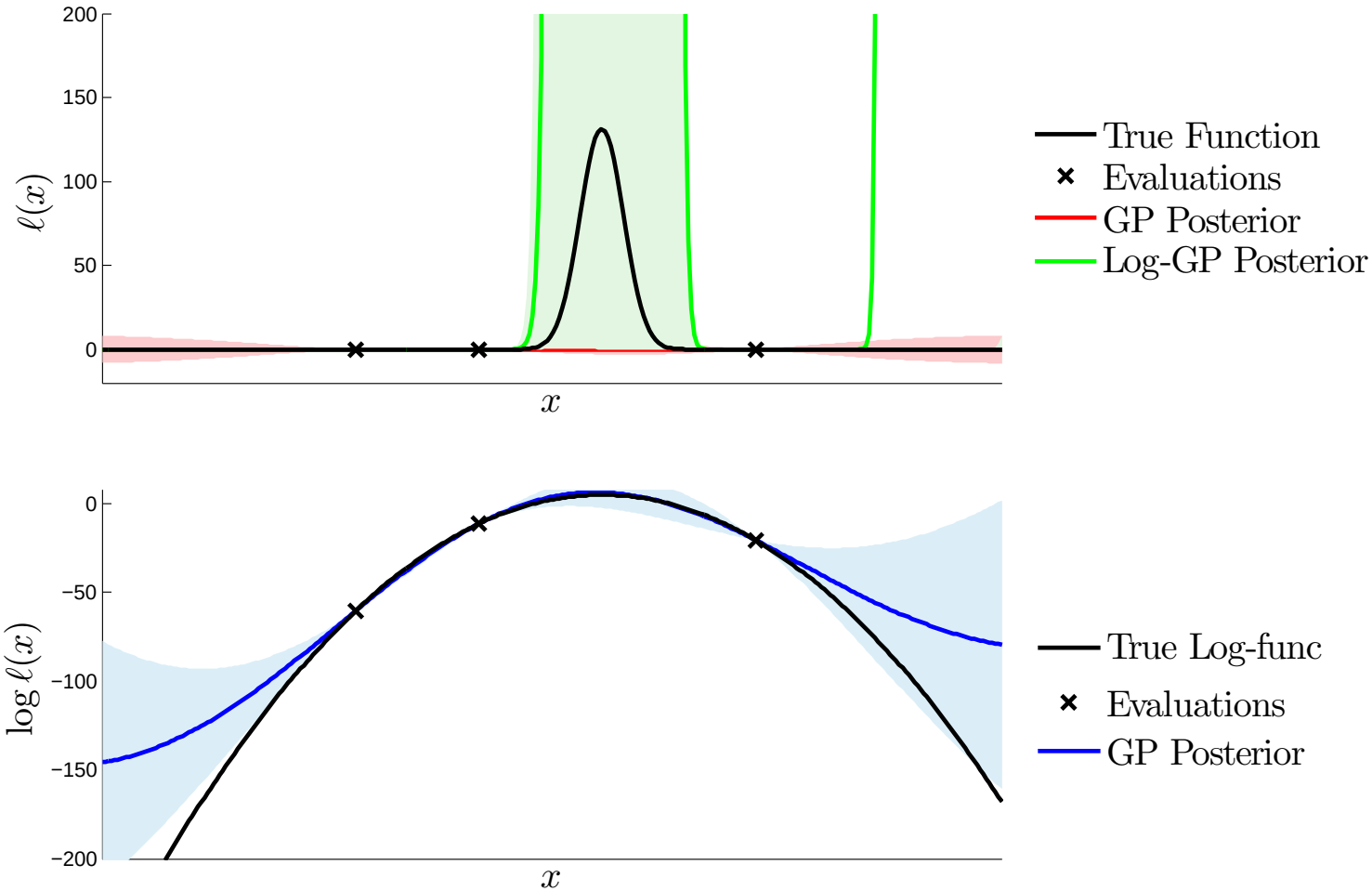


GPs vs Log-GPs for Inference



GPs vs Log-GPs for Inference

Takeaway: Herding assumptions are innapropriate for inference



Mean Embedding Interpretation

- [Muandet & Ghahramani, 2012] showed that if $f \sim \mathrm{GP}$,

Mean Embedding Interpretation

- [Muandet & Ghahramani, 2012] showed that if $f \sim \mathrm{GP}$,

$$\begin{aligned}
\mu_{p(x)} &= \int \phi(x) p(x) dx \\
&= \int k(x, \cdot) p(x) dx \\
&= \mathbb{E}_{f \sim \mathrm{GP}} \left[\int f(x) f(\cdot) p(x) dx \right] \\
&= \mathbb{E}_{f \sim \mathrm{GP}} \left[f(\cdot) \int f(x) p(x) dx \right] \\
&= \mathrm{cov}_{f \sim \mathrm{GP}} \left(f(\cdot), Z_p \right)
\end{aligned}$$

Mean Embedding Interpretation

- [Muandet & Ghahramani, 2012] showed that if $f \sim \mathrm{GP}$,

$$\begin{aligned}
\mu_{p(x)} &= \int \phi(x) p(x) dx \\
&= \int k(x, \cdot) p(x) dx \\
&= \mathbb{E}_{f \sim \mathrm{GP}} \left[\int f(x) f(\cdot) p(x) dx \right] \\
&= \mathbb{E}_{f \sim \mathrm{GP}} \left[f(\cdot) \int f(x) p(x) dx \right] \\
&= \mathrm{cov}_{f \sim \mathrm{GP}} \left(f(\cdot), Z_p \right)
\end{aligned}$$

- $\mu_{p(x)}$ is the covariance the function with its integral with respect to $p(x)$.

Kernel two-sample test

- How to test whether two distributions $p(x)$ and $q(x)$ are the same?

Kernel two-sample test

- How to test whether two distributions $p(x)$ and $q(x)$ are the same?
- New test statistic: $\mathrm{MMD}(p, q)$ [Gretton et. al, 2005]

Kernel two-sample test

- How to test whether two distributions $p(x)$ and $q(x)$ are the same?
- New test statistic: $\mathrm{MMD}(p, q)$ [Gretton et. al, 2005]
- Equivalent Bayesian intepretation:

$$\mathrm{MMD}_k^2(p, q) = \mathbb{V}_{f \sim \mathrm{GP}_k} \left[\int f(x)p(x)dx - \int f(x)q(x)dx \right]$$

p and q are similar if integrals of functions drawn from a GP prior have similar integrals.

Kernel two-sample test

- How to test whether two distributions $p(x)$ and $q(x)$ are the same?
- New test statistic: $\mathrm{MMD}(p, q)$ [Gretton et. al, 2005]
- Equivalent Bayesian intepretation:

$$\mathrm{MMD}_k^2(p, q) = \mathbb{V}_{f \sim \mathrm{GP}_k} \left[\int f(x)p(x)dx - \int f(x)q(x)dx \right]$$

p and q are similar if integrals of functions drawn from a GP prior have similar integrals.

Possible takeaways: Decision-theoretic choice of kernel, sampling-based methods for computing MMD

Hilbert-Schmidt Independence Criterion

- Given samples $\{X, Y\} \sim p(x, y)$, how to test whether $p(x, y) = p(x)p(y)$?

Hilbert-Schmidt Independence Criterion

- Given samples $\{X, Y\} \sim p(x, y)$, how to test whether $p(x, y) = p(x)p(y)$?
- New test statistic based on infinite-dimensional Frobenius norm of cross-covariance matrix of features of x and y: [Gretton et. al, 2005]

Hilbert-Schmidt Independence Criterion

- Given samples $\{X, Y\} \sim p(x, y)$, how to test whether $p(x, y) = p(x)p(y)$?
- New test statistic based on infinite-dimensional Frobenius norm of cross-covariance matrix of features of x and y: [Gretton et. al, 2005]

$$\begin{aligned}
&\mathrm{HSIC}(p(x, y), k_x, k_y) = ||C_{xy}||^2_{HS} \\
&= \mathbb{E}_{x,x',y,y'}\left[k_x(x, x')k_y(y, y)\right] + \mathbb{E}_{x,x'}\left[k_x(x, x')\right]\mathbb{E}_{y,y'}\left[k_y(y, y')\right] \\
&\quad - 2\mathbb{E}_{x,y}\left[\mathbb{E}_{x'}\left[k_x(x, x')\right]\mathbb{E}_{y'}\left[k_y(y, y')\right]\right]
\end{aligned}$$

Hilbert-Schmidt Independence Criterion

- Given samples $\{X, Y\} \sim p(x, y)$, how to test whether $p(x, y) = p(x)p(y)$?
- New result: Assuming $k(x, y, x', y') = k_x(x, x')k_y(y, y')$

$$\mathrm{HSIC}(p(x, y), k_x, k_y) =$$

$$= \mathbb{V}_{f \sim \mathrm{GP}_k} \left[\int f(x, y)p(x, y)dxdy - \int f(x, y)p(x)p(y)dxdy \right]$$

Determinantal Point Processes

- Probability of a set $P(\mathcal{X}) = |K(\mathcal{X}, \mathcal{X})|$

Determinantal Point Processes

- Probability of a set $P(\mathcal{X}) = |K(\mathcal{X}, \mathcal{X})|$
- Greedy MAP maximizes $P(\mathcal{X} \cup x_i)$

Determinantal Point Processes

- Probability of a set $P(\mathcal{X}) = |K(\mathcal{X}, \mathcal{X})|$
- Greedy MAP maximizes $P(\mathcal{X} \cup x_i)$

$$
\begin{aligned}
& P(\mathcal{X} \cup x_i) \\
&= |K(\mathcal{X} \cup x_i, \mathcal{X} \cup x_i| \\
&= |K(\mathcal{X}, \mathcal{X})| \left[k(x_i, x_i) - k(x_i, \mathcal{X}) K(\mathcal{X}, \mathcal{X})^{-1} k(\mathcal{X}, x_i)\right] \\
&\propto k(x_i, x_i) - k(x_i, \mathcal{X}) K(\mathcal{X}, \mathcal{X})^{-1} k(\mathcal{X}, x_i) \\
&= \mathbb{V}_{f \sim \mathrm{GP}_k} \left[f(x_i) | \mathcal{X}\right]
\end{aligned}
$$

Determinantal Point Processes

- Probability of a set $P(\mathcal{X}) = |K(\mathcal{X}, \mathcal{X})|$
- Greedy MAP maximizes $P(\mathcal{X} \cup x_i)$

$$
\begin{aligned}
& P(\mathcal{X} \cup x_i) \\
&= |K(\mathcal{X} \cup x_i, \mathcal{X} \cup x_i| \\
&= |K(\mathcal{X}, \mathcal{X})| \left[k(x_i, x_i) - k(x_i, \mathcal{X}) K(\mathcal{X}, \mathcal{X})^{-1} k(\mathcal{X}, x_i)\right] \\
&\propto k(x_i, x_i) - k(x_i, \mathcal{X}) K(\mathcal{X}, \mathcal{X})^{-1} k(\mathcal{X}, x_i) \\
&= \mathbb{V}_{f \sim \mathrm{GP}_k} \left[f(x_i) | \mathcal{X}\right]
\end{aligned}
$$

- New DPP Point added at location with highest marginal variance in GP posterior, conditioned on the other points

Determinantal Point Processes

- Probability of a set $P(\mathcal{X}) = |K(\mathcal{X}, \mathcal{X})|$
- Greedy MAP maximizes $P(\mathcal{X} \cup x_i)$

$$
\begin{aligned}
& P(\mathcal{X} \cup x_i) \\
&= |K(\mathcal{X} \cup x_i, \mathcal{X} \cup x_i| \\
&= |K(\mathcal{X}, \mathcal{X})| \left[k(x_i, x_i) - k(x_i, \mathcal{X}) K(\mathcal{X}, \mathcal{X})^{-1} k(\mathcal{X}, x_i)\right] \\
&\propto k(x_i, x_i) - k(x_i, \mathcal{X}) K(\mathcal{X}, \mathcal{X})^{-1} k(\mathcal{X}, x_i) \\
&= \mathbb{V}_{f \sim \mathrm{GP}_k} \left[f(x_i) | \mathcal{X}\right]
\end{aligned}
$$

- New DPP Point added at location with highest marginal variance in GP posterior, conditioned on the other points
- Related to sensor placement work by Andreas Krause

Determinantal Point Processes

- Probability of a set $P(\mathcal{X}) = |K(\mathcal{X}, \mathcal{X})|$
- Greedy MAP maximizes $P(\mathcal{X} \cup x_i)$

$$
\begin{aligned}
& P(\mathcal{X} \cup x_i) \\
&= |K(\mathcal{X} \cup x_i, \mathcal{X} \cup x_i| \\
&= |K(\mathcal{X}, \mathcal{X})| \left[k(x_i, x_i) - k(x_i, \mathcal{X}) K(\mathcal{X}, \mathcal{X})^{-1} k(\mathcal{X}, x_i)\right] \\
&\propto k(x_i, x_i) - k(x_i, \mathcal{X}) K(\mathcal{X}, \mathcal{X})^{-1} k(\mathcal{X}, x_i) \\
&= \mathbb{V}_{f \sim \mathrm{GP}_k} \left[f(x_i) | \mathcal{X}\right]
\end{aligned}
$$

- New DPP Point added at location with highest marginal variance in GP posterior, conditioned on the other points
- Related to sensor placement work by Andreas Krause
- Thanks to Le Song and Roman Garnett

Summary of Connections

Frequentist Method	**Bayesian Method**
Kernel herding	Bayesian Quadrature
Mean Embedding	Covariance with integral
Kernel Two-sample test	Variance of difference of integrals
HSIC	Variance of difference of integrals
DPP MAP	Marginal uncertainty in GP posterior

Summary of Connections

Frequentist Method	**Bayesian Method**
Kernel herding	Bayesian Quadrature
Mean Embedding	Covariance with integral
Kernel Two-sample test	Variance of difference of integrals
HSIC	Variance of difference of integrals
DPP MAP	Marginal uncertainty in GP posterior

Some possible extensions:

Summary of Connections

Frequentist Method	**Bayesian Method**
Kernel herding	Bayesian Quadrature
Mean Embedding	Covariance with integral
Kernel Two-sample test	Variance of difference of integrals
HSIC	Variance of difference of integrals
DPP MAP	Marginal uncertainty in GP posterior

Some possible extensions:

- Log-kernel herding for inference

Summary of Connections

Frequentist Method	**Bayesian Method**
Kernel herding	Bayesian Quadrature
Mean Embedding	Covariance with integral
Kernel Two-sample test	Variance of difference of integrals
HSIC	Variance of difference of integrals
DPP MAP	Marginal uncertainty in GP posterior

Some possible extensions:

- Log-kernel herding for inference
- Interpretation of conditional mean embeddings

Summary of Connections

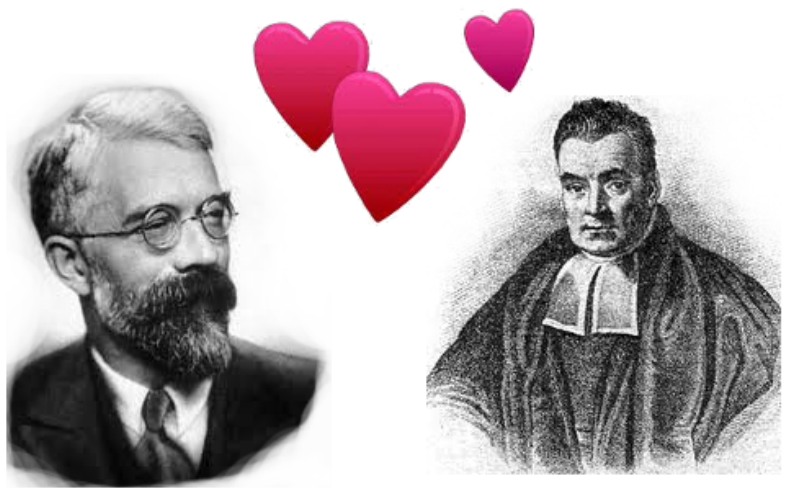
Frequentist Method	**Bayesian Method**
Kernel herding	Bayesian Quadrature
Mean Embedding	Covariance with integral
Kernel Two-sample test	Variance of difference of integrals
HSIC	Variance of difference of integrals
DPP MAP	Marginal uncertainty in GP posterior

Some possible extensions:

- Log-kernel herding for inference
- Interpretation of conditional mean embeddings
- Different low-rank approximations based on sparse GPs

Frequentist Method	**Bayesian Method**
Kernel Regression	GP Regression
Functional ANOVA, Hierchical Kernel Learning	Additive Gaussian Processes
Kernel Bayes' Rule	Bayesian Quadrature for Ratios
RKHS Embeddings of Conditional Distributions	GP Dynamics Models
Kernel Message Passing	???

Frequentist Method	**Bayesian Method**
Kernel Regression	GP Regression
Functional ANOVA, Hierchical Kernel Learning	Additive Gaussian Processes
Kernel Bayes' Rule	Bayesian Quadrature for Ratios
RKHS Embeddings of Conditional Distributions	GP Dynamics Models
Kernel Message Passing	???



Thanks!