STAT G8325
Gaussian Processes and Kernel Methods
§12: Probabilistic Integration

John P. Cunningham

Department of Statistics
Columbia University

Outline

Outline

Progress...

§	Dates	Content
10	Nov 23, Dec 2	Kernel statistical tests
11	Dec 7	Speed and Scaling Part 3
12	Dec 9	Probabilistic Integration
	Dec 14, 16	Final project presentations

- Final project presentations Monday Dec 14, 16
  - Present 5-7 minutes of your project results.
  - Build off of project progress report.
  - Send 1-5 pdf slides to me beforehand.
- Monday: Richard, Gamal, Jalaj, Francois, Xu S., Xu R., Tim, Swupnil.
- Wednesday: Kashif, Hal, Ruoxi, Ben, Ryan, Gabriel, Shuawein, Hanxi.
- Soon-to-be-randomly-assigned: Yuanjun, Lichi, Gonzalo, Daniel, Rayleigh.
- Final project writeup then due Friday Dec 18 at noon.
  - 8-16 pages pdf, using the tex template from hw3.
  - Deadline strictly enforced.

Outline

Quadrature

Quadrature

- Quadrature (aka numerical integration) is the problem of calculating

$$Z = \int g(x) dx.$$

Quadrature

- Quadrature (aka numerical integration) is the problem of calculating

$$Z = \int g(x)dx.$$

- We will equivalently consider the familiar expectation problem:

$$Z = E_p(\ell) = \int \ell(x)p(x)dx$$

for $p(x) = \mathcal{N}(x; m_0, s_0)$, which by $\ell(x) = \frac{g(x)}{p(x)}$ is (sort of) wlog.

Quadrature

- Quadrature (aka numerical integration) is the problem of calculating

$$Z = \int g(x)dx.$$

- We will equivalently consider the familiar expectation problem:

$$Z = E_p(\ell) = \int \ell(x)p(x)dx$$

for $p(x) = \mathcal{N}(x; m_0, s_0)$, which by $\ell(x) = \frac{g(x)}{p(x)}$ is (sort of) wlog.
- Our simplest, traditional Monte Carlo estimator is:

$$\hat{Z} = \frac{1}{n}\sum_{i=1}^{n} \ell(x_i) \qquad x_1, ..., x_n \sim_{iid} p(x).$$

Quadrature

- Quadrature (aka numerical integration) is the problem of calculating

$$Z = \int g(x)dx.$$

- We will equivalently consider the familiar expectation problem:

$$Z = E_p(\ell) = \int \ell(x)p(x)dx$$

  for $p(x) = \mathcal{N}(x; m_0, s_0)$, which by $\ell(x) = \frac{g(x)}{p(x)}$ is (sort of) wlog.
- Our simplest, traditional Monte Carlo estimator is:

$$\hat{Z} = \frac{1}{n}\sum_{i=1}^{n} \ell(x_i) \qquad x_1, ..., x_n \sim_{iid} p(x).$$

- Bayesian quadrature (aka probabilistic integration) simply observes that smoothness in $\ell(x)$ should allow us to learn more about the integral from a finite set of samples $x_1, ..., x_n$.

Quadrature

- Quadrature (aka numerical integration) is the problem of calculating

$$Z = \int g(x)dx.$$

- We will equivalently consider the familiar expectation problem:

$$Z = E_p(\ell) = \int \ell(x)p(x)dx$$

for $p(x) = \mathcal{N}(x; m_0, s_0)$, which by $\ell(x) = \frac{g(x)}{p(x)}$ is (sort of) wlog.

- Our simplest, traditional Monte Carlo estimator is:

$$\hat{Z} = \frac{1}{n}\sum_{i=1}^{n} \ell(x_i) \qquad x_1, ..., x_n \sim_{iid} p(x).$$

- Bayesian quadrature (aka probabilistic integration) simply observes that smoothness in $\ell(x)$ should allow us to learn more about the integral from a finite set of samples $x_1, ..., x_n$.
- Example: suppose two draws x_i and x_j are equal (or very close); ignoring this fact leads to double counting.

Bayesian Quadrature (simplest form, as in [O'H91, GR02])

Bayesian Quadrature (simplest form, as in [O'H91, GR02])

- One of the biggest themes in this course has been to replace complicated or unknown functions with gp. Let's do that again.

Bayesian Quadrature (simplest form, as in [O'H91, GR02])

- One of the biggest themes in this course has been to replace complicated or unknown functions with gp. Let's do that again.
- Assume the function $\ell(x) \sim \mathcal{GP}(m, k)$. Then $Z = E_p(\ell)$ is also a random variable, which we can condition on $x_i, \ell(x_i)$ pairs.

Bayesian Quadrature (simplest form, as in [O'H91, GR02])

- One of the biggest themes in this course has been to replace complicated or unknown functions with gp. Let's do that again.
- Assume the function $\ell(x) \sim \mathcal{GP}(m, k)$. Then $Z = E_p(\ell)$ is also a random variable, which we can condition on $x_i, \ell(x_i)$ pairs.
- Repeat:

Bayesian Quadrature (simplest form, as in [O'H91, GR02])

- One of the biggest themes in this course has been to replace complicated or unknown functions with gp. Let's do that again.
- Assume the function $\ell(x) \sim \mathcal{GP}(m, k)$. Then $Z = E_p(\ell)$ is also a random variable, which we can condition on $x_i, \ell(x_i)$ pairs.
- Repeat:
  - Draw $x_i \sim_{iid} p(x)$

Bayesian Quadrature (simplest form, as in [O'H91, GR02])

- One of the biggest themes in this course has been to replace complicated or unknown functions with gp. Let's do that again.

- Assume the function $\ell(x) \sim \mathcal{GP}(m, k)$. Then $Z = E_p(\ell)$ is also a random variable, which we can condition on $x_i, \ell(x_i)$ pairs.

- Repeat:
  - Draw $x_i \sim_{iid} p(x)$
  - Observe (evaluate) $\ell(x_i)$

Bayesian Quadrature (simplest form, as in [O'H91, GR02])

- One of the biggest themes in this course has been to replace complicated or unknown functions with gp. Let's do that again.

- Assume the function $\ell(x) \sim \mathcal{GP}(m, k)$. Then $Z = E_p(\ell)$ is also a random variable, which we can condition on $x_i, \ell(x_i)$ pairs.

- Repeat:
  - Draw $x_i \sim_{iid} p(x)$
  - Observe (evaluate) $\ell(x_i)$
  - Infer the posterior $Z | x_1, \ell(x_1), ...x_i, \ell(x_i)$.

Bayesian Quadrature (simplest form, as in [O'H91, GR02])

- One of the biggest themes in this course has been to replace complicated or unknown functions with gp. Let's do that again.

- Assume the function $\ell(x) \sim \mathcal{GP}(m, k)$. Then $Z = E_p(\ell)$ is also a random variable, which we can condition on $x_i, \ell(x_i)$ pairs.

- Repeat:
  - Draw $x_i \sim_{iid} p(x)$
  - Observe (evaluate) $\ell(x_i)$
  - Infer the posterior $Z | x_1, \ell(x_1), ... x_i, \ell(x_i)$.

- Posterior mean $E_\ell \left(Z | D \right) = E_\ell \left(E_p(\ell) | D \right)$

Bayesian Quadrature (simplest form, as in [O'H91, GR02])

- One of the biggest themes in this course has been to replace complicated or unknown functions with gp. Let's do that again.
- Assume the function $\ell(x) \sim \mathcal{GP}(m, k)$. Then $Z = E_p(\ell)$ is also a random variable, which we can condition on $x_i, \ell(x_i)$ pairs.
- Repeat:
  - Draw $x_i \sim_{iid} p(x)$
  - Observe (evaluate) $\ell(x_i)$
  - Infer the posterior $Z|x_1, \ell(x_1), ...x_i, \ell(x_i)$.
- Posterior mean $E_\ell\left(Z|D\right) = E_\ell\left(E_p(\ell)|D\right)$

...using the usual data $D \triangleq x_1, \ell(x_1), ..., x_n, \ell(x_n)$.

Bayesian Quadrature (simplest form, as in [O'H91, GR02])

- One of the biggest themes in this course has been to replace complicated or unknown functions with gp. Let's do that again.

- Assume the function $\ell(x) \sim \mathcal{GP}(m, k)$. Then $Z = E_p(\ell)$ is also a random variable, which we can condition on $x_i, \ell(x_i)$ pairs.

- Repeat:
  - Draw $x_i \sim_{iid} p(x)$
  - Observe (evaluate) $\ell(x_i)$
  - Infer the posterior $Z|x_1, \ell(x_1), ...x_i, \ell(x_i)$.

- Posterior mean $E_\ell\left(Z|D\right) = E_\ell\left(E_p(\ell)|D\right)$

  ...using the usual data $D \triangleq x_1, \ell(x_1), ..., x_n, \ell(x_n)$.

- $E\left(Z|D\right)$ is the quantity of interest: expected quadrature value.

Bayesian Quadrature (simplest form, as in [O'H91, GR02])

- One of the biggest themes in this course has been to replace complicated or unknown functions with gp. Let's do that again.

- Assume the function $\ell(x) \sim \mathcal{GP}(m, k)$. Then $Z = E_p(\ell)$ is also a random variable, which we can condition on $x_i, \ell(x_i)$ pairs.

- Repeat:
  - Draw $x_i \sim_{iid} p(x)$
  - Observe (evaluate) $\ell(x_i)$
  - Infer the posterior $Z|x_1, \ell(x_1), ...x_i, \ell(x_i)$.

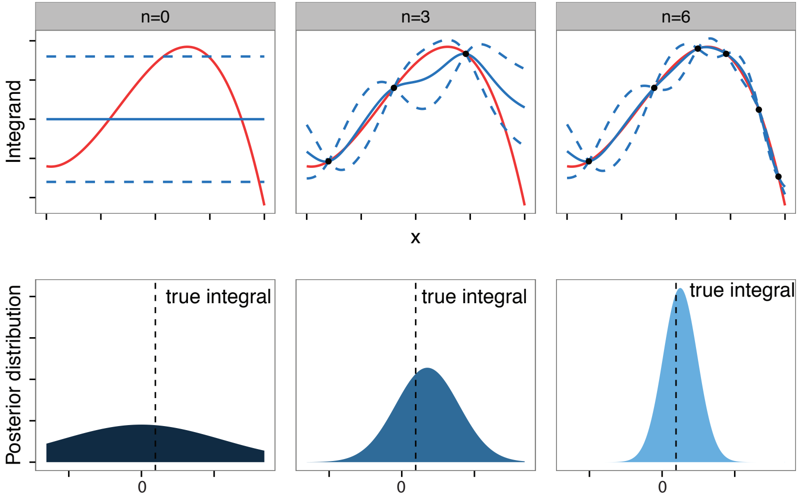
- Posterior mean $E_\ell\left(Z|D\right) = E_\ell\left(E_p(\ell)|D\right)$

  ...using the usual data $D \triangleq x_1, \ell(x_1), \ldots, x_n, \ell(x_n)$.

- $E\left(Z|D\right)$ is the quantity of interest: expected quadrature value.

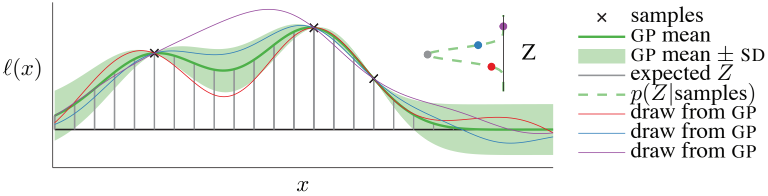
- It can have (surprisingly?) tractable form...

Intuitive picture



modified from http://arxiv.org/abs/1512.00933.

Another intuitive picture



from [OGG+12].

Bayesian Quadrature (simplest form, as in [O'H91, GR02])

Bayesian Quadrature (simplest form, as in [O'H91, GR02])

- The expected quadrature value:

Bayesian Quadrature (simplest form, as in [O'H91, GR02])

- The expected quadrature value:

$$E(Z|D) = E_\ell \left(E_p(\ell) | D \right)$$

Bayesian Quadrature (simplest form, as in [O'H91, GR02])

- The expected quadrature value:

$$
\begin{aligned}
E(Z|D) &= E_\ell\left(E_p(\ell)|D\right) \\
&= \int_\ell \left(\int_x \ell(x)p(x)dx \right) p(\ell|D)d\ell
\end{aligned}
$$

Bayesian Quadrature (simplest form, as in [O'H91, GR02])

- The expected quadrature value:

$$
\begin{aligned}
E(Z|D) &= E_\ell\left(E_p(\ell)|D\right) \\
&= \int_\ell \left(\int_x \ell(x)p(x)dx\right) p(\ell|D)d\ell \\
&= \int_x \left(\int_\ell \ell(x)p(\ell|D)d\ell\right) p(x)dx
\end{aligned}
$$

Bayesian Quadrature (simplest form, as in [O'H91, GR02])

- The expected quadrature value:

$$
\begin{aligned}
E(Z|D) &= E_\ell\left(E_p(\ell)|D\right) \\
&= \int_\ell \left(\int_x \ell(x)p(x)dx\right) p(\ell|D)d\ell \\
&= \int_x \left(\int_\ell \ell(x)p(\ell|D)d\ell\right) p(x)dx \\
&= \int_x \left(m_x + K_{xD}(K_{DD} + \sigma_\epsilon^2 I)^{-1}\left(\ell_D - m_\ell\right)\right) p(x)dx
\end{aligned}
$$

Bayesian Quadrature (simplest form, as in [O'H91, GR02])

- The expected quadrature value:

$$
\begin{aligned}
E(Z|D) &= E_\ell\left(E_p(\ell)|D\right) \\
&= \int_\ell \left(\int_x \ell(x)p(x)dx\right) p(\ell|D)d\ell \\
&= \int_x \left(\int_\ell \ell(x)p(\ell|D)d\ell\right) p(x)dx \\
&= \int_x \left(m_x + K_{xD}(K_{DD} + \sigma_\epsilon^2 I)^{-1}\left(\ell_D - m_\ell\right)\right) p(x)dx \\
&= \int_x m_x p(x)dx + \left(\int_x K_{xD}p(x)dx\right)(K_{DD} + \sigma_\epsilon^2 I)^{-1}\left(\ell_D - m_\ell\right)
\end{aligned}
$$

Bayesian Quadrature (simplest form, as in [O'H91, GR02])

- The expected quadrature value:

$$
\begin{aligned}
E(Z|D) &= E_\ell\left(E_p(\ell)|D\right) \\
&= \int_\ell \left(\int_x \ell(x) p(x) dx \right) p(\ell|D) d\ell \\
&= \int_x \left(\int_\ell \ell(x) p(\ell|D) d\ell \right) p(x) dx \\
&= \int_x \left(m_x + K_{xD}(K_{DD} + \sigma_\epsilon^2 I)^{-1} \left(\ell_D - m_\ell\right)\right) p(x) dx \\
&= \int_x m_x p(x) dx + \left(\int_x K_{xD} p(x) dx \right) (K_{DD} + \sigma_\epsilon^2 I)^{-1} \left(\ell_D - m_\ell\right) \\
&= E_p(m) + {\mu_D^p}^\top (K_{DD} + \sigma_\epsilon^2 I)^{-1} \left(\ell_D - m_\ell\right)
\end{aligned}
$$

where $\mu^p = E_p(k_x)$ is the familiar kernel mean embedding in the rkhs $\mathcal{H}$.

Bayesian Quadrature (simplest form, as in [O'H91, GR02])

- The expected quadrature value:

$$\begin{aligned}
E(Z|D) &= E_\ell\left(E_p(\ell)|D\right) \\
&= \int_\ell \left(\int_x \ell(x)p(x)dx\right) p(\ell|D)d\ell \\
&= \int_x \left(\int_\ell \ell(x)p(\ell|D)d\ell\right) p(x)dx \\
&= \int_x \left(m_x + K_{xD}(K_{DD}+\sigma_\epsilon^2 I)^{-1}\left(\ell_D - m_\ell\right)\right) p(x)dx \\
&= \int_x m_x p(x)dx + \left(\int_x K_{xD}p(x)dx\right)(K_{DD}+\sigma_\epsilon^2 I)^{-1}\left(\ell_D - m_\ell\right) \\
&= E_p(m) + {\mu_D^p}^\top (K_{DD}+\sigma_\epsilon^2 I)^{-1}\left(\ell_D - m_\ell\right)
\end{aligned}$$

  where $\mu^p = E_p(k_x)$ is the familiar kernel mean embedding in the rkhs $\mathcal{H}$.

- Recall that when $k(x,x') = \sigma_\ell^2 \mathcal{N}(x; x', w_\ell)$ (an SE or RBF kernel), we have the further simplification:

$$\mu^p(x_i) = \int k(x_i, x)p(x)dx = \sigma_\ell^2 \mathcal{N}(x_i; m_0, s_0 + w_\ell).$$

Bayesian Quadrature (simplest form, as in [O'H91, GR02])

- The expected quadrature value:

$$
\begin{aligned}
E(Z|D) &= E_\ell\left(E_p(\ell)|D\right) \\
&= \int_\ell \left(\int_x \ell(x)p(x)dx\right) p(\ell|D)d\ell \\
&= \int_x \left(\int_\ell \ell(x)p(\ell|D)d\ell\right) p(x)dx \\
&= \int_x \left(m_x + K_{xD}(K_{DD} + \sigma_\epsilon^2 I)^{-1}\left(\ell_D - m_\ell\right)\right) p(x)dx \\
&= \int_x m_x p(x)dx + \left(\int_x K_{xD}p(x)dx\right)(K_{DD} + \sigma_\epsilon^2 I)^{-1}\left(\ell_D - m_\ell\right) \\
&= E_p(m) + {\mu_D^p}^\top (K_{DD} + \sigma_\epsilon^2 I)^{-1}\left(\ell_D - m_\ell\right)
\end{aligned}
$$

  where $\mu^p = E_p(k_x)$ is the familiar kernel mean embedding in the rkhs $\mathcal{H}$.

- Recall that when $k(x, x') = \sigma_\ell^2 \mathcal{N}(x; x', w_\ell)$ (an SE or RBF kernel), we have the further simplification:

$$\mu^p(x_i) = \int k(x_i, x)p(x)dx = \sigma_\ell^2 \mathcal{N}(x_i; m_0, s_0 + w_\ell).$$

- Often $\sigma_\ell = 0$ when the integrand can be evaluated precisely.

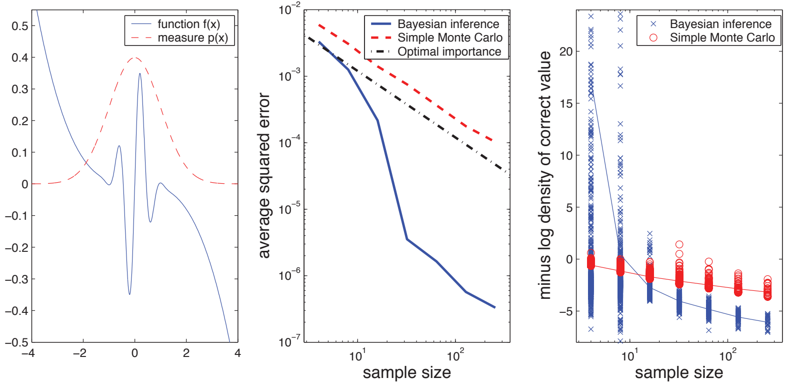
Empirical result from [GR02]

Empirical result from [GR02]

- A toy example:

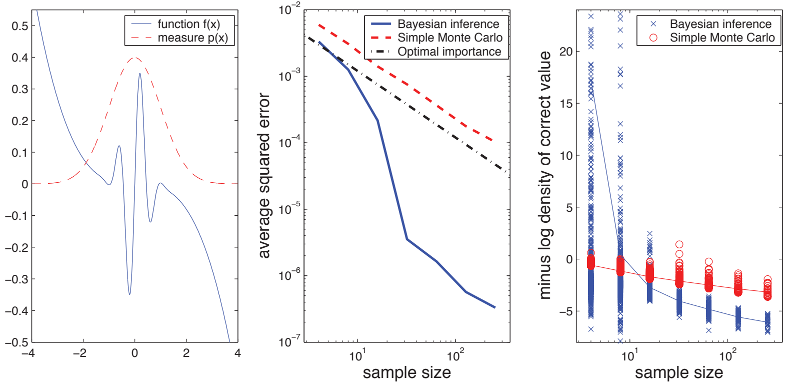
Empirical result from [GR02]

- A toy example:



Empirical result from [GR02]

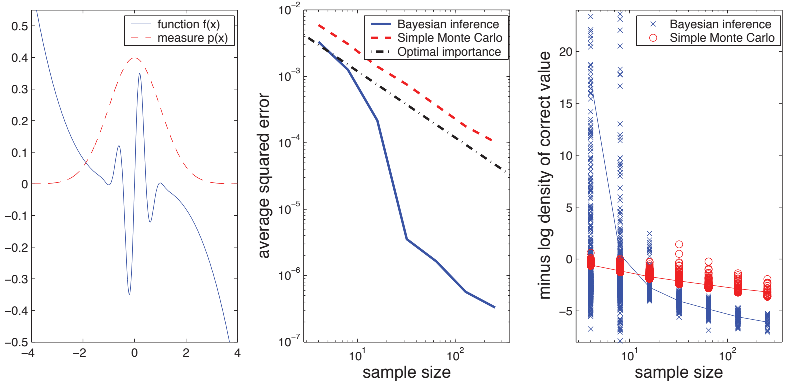
- A toy example:



- BQ uses larger sample sizes more effectively.

Empirical result from [GR02]

- A toy example:



- BQ uses larger sample sizes more effectively.
- BQ has higher variance with small sample sizes. Why?

Outline

Closed-form kernel mean embeddings

Closed-form kernel mean embeddings

- Things got simpler when we had $k(x, x') = \sigma_\ell^2 \mathcal{N}(x; x', w_\ell)$ (SE kernel):

$$\mu^p(x_i) = \int k(x_i, x)p(x)dx = \sigma_\ell^2 \mathcal{N}(x_i; m_0, s_0 + w_\ell).$$

Closed-form kernel mean embeddings

- Things got simpler when we had $k(x, x') = \sigma_\ell^2 \mathcal{N}(x; x', w_\ell)$ (SE kernel):

$$\mu^p(x_i) = \int k(x_i, x)p(x)dx = \sigma_\ell^2 \mathcal{N}(x_i; m_0, s_0 + w_\ell).$$

- That is, the kernel mean embedding is closed form.

Closed-form kernel mean embeddings

- Things got simpler when we had $k(x, x') = \sigma_\ell^2 \mathcal{N}(x; x', w_\ell)$ (SE kernel):

$$\mu^p(x_i) = \int k(x_i, x)p(x)dx = \sigma_\ell^2 \mathcal{N}(x_i; m_0, s_0 + w_\ell).$$

- That is, the kernel mean embedding is closed form.
- Here are some other $p, k, \mathcal{X}$ triplets such that μ^p is closed form:

Closed-form kernel mean embeddings

- Things got simpler when we had $k(x, x') = \sigma_\ell^2 \mathcal{N}(x; x', w_\ell)$ (SE kernel):

$$\mu^p(x_i) = \int k(x_i, x)p(x)dx = \sigma_\ell^2 \mathcal{N}(x_i; m_0, s_0 + w_\ell).$$

- That is, the kernel mean embedding is closed form.
- Here are some other $p, k, \mathcal{X}$ triplets such that μ^p is closed form:

$\mathcal{X}$	p	k	Reference
$[0,1]^d$	Unif($\mathcal{X}$)	Wendland TP	Oates and Girolami (2015)
$[0,1]^d$	Unif($\mathcal{X}$)	Matérn Weighted TP	Sec. 5.2.3
$[0,1]^d$	Unif($\mathcal{X}$)	Korobov TP	Appendix D
$[0,1]^d$	Unif($\mathcal{X}$)	Exponentiated quadratic	Appendix J
$\mathbb{R}^d$	Mixt. of Gaussians	Exponentiated quadratic	O'Hagan (1991)
$\mathbb{S}^d$	Unif($\mathcal{X}$)	Gegenbauer	Sec. 5.2.1
Arbitrary	Unif($\mathcal{X}$) / Mixt. of Gauss.	trigonometric	Integration by parts
Arbitrary	Unif($\mathcal{X}$)	Splines	Minka (2000)
Arbitrary	Known moments	Polynomial TP	Briol et al. (2015)
Arbitrary	Known $\partial \log \pi(\boldsymbol{x})$	Control functional	Sec. 4.3

again from `http://arxiv.org/abs/1512.00933`.

Closed-form kernel mean embeddings

- Things got simpler when we had $k(x, x') = \sigma_\ell^2 \mathcal{N}(x; x', w_\ell)$ (SE kernel):

$$\mu^p(x_i) = \int k(x_i, x)p(x)dx = \sigma_\ell^2 \mathcal{N}(x_i; m_0, s_0 + w_\ell).$$

- That is, the kernel mean embedding is closed form.
- Here are some other $p, k, \mathcal{X}$ triplets such that μ^p is closed form:

$\mathcal{X}$	p	k	Reference
$[0,1]^d$	Unif($\mathcal{X}$)	Wendland TP	Oates and Girolami (2015)
$[0,1]^d$	Unif($\mathcal{X}$)	Matérn Weighted TP	Sec. 5.2.3
$[0,1]^d$	Unif($\mathcal{X}$)	Korobov TP	Appendix D
$[0,1]^d$	Unif($\mathcal{X}$)	Exponentiated quadratic	Appendix J
$\mathbb{R}^d$	Mixt. of Gaussians	Exponentiated quadratic	O'Hagan (1991)
$\mathbb{S}^d$	Unif($\mathcal{X}$)	Gegenbauer	Sec. 5.2.1
Arbitrary	Unif($\mathcal{X}$) / Mixt. of Gauss.	trigonometric	Integration by parts
Arbitrary	Unif($\mathcal{X}$)	Splines	Minka (2000)
Arbitrary	Known moments	Polynomial TP	Briol et al. (2015)
Arbitrary	Known $\partial \log \pi(\boldsymbol{x})$	Control functional	Sec. 4.3

again from `http://arxiv.org/abs/1512.00933`.

- Here TP means tensor product.

Outline

Possible extensions

Possible extensions

- BQ uses gp to share information about input points $x_1, ..., x_n$ via a kernel.

Possible extensions

- BQ uses gp to share information about input points $x_1, ..., x_n$ via a kernel.
- But also, throughout the semester we have seen other possible moves...

Possible extensions

- BQ uses gp to share information about input points $x_1, ..., x_n$ via a kernel.
- But also, throughout the semester we have seen other possible moves...
  - Exploiting structure in ℓ:

Possible extensions

- BQ uses gp to share information about input points $x_1, ..., x_n$ via a kernel.
- But also, throughout the semester we have seen other possible moves...
  - Exploiting structure in ℓ:
    - Often $\ell(x)$ is likelihood or other density, hence nonnegative:

$$p(D) = \int p(D|x)p(x)dx \triangleq \int \ell(x)p(x)dx.$$

Possible extensions

- BQ uses gp to share information about input points $x_1, ..., x_n$ via a kernel.
- But also, throughout the semester we have seen other possible moves...
  - Exploiting structure in ℓ:
    - Often $\ell(x)$ is likelihood or other density, hence nonnegative:

$$p(D) = \int p(D|x)p(x)dx \triangleq \int \ell(x)p(x)dx.$$

    - Maybe a ratio of integrals with common terms:

$$p(f|D) = \frac{\int p(f|D,\theta)p(D|\theta)p(\theta)d\theta}{\int p(D|\theta)p(\theta)d\theta}.$$

Possible extensions

- BQ uses gp to share information about input points $x_1, ..., x_n$ via a kernel.
- But also, throughout the semester we have seen other possible moves...
  - Exploiting structure in ℓ:
    - Often $\ell(x)$ is likelihood or other density, hence nonnegative:

$$p(D) = \int p(D|x)p(x)dx \triangleq \int \ell(x)p(x)dx.$$

    - Maybe a ratio of integrals with common terms:

$$p(f|D) = \frac{\int p(f|D,\theta)p(D|\theta)p(\theta)d\theta}{\int p(D|\theta)p(\theta)d\theta}.$$

  - Active learning: choose point x_{i+1} based on observations $\ell(x_1), ..., \ell(x_i)$.

Possible extensions

- BQ uses gp to share information about input points $x_1, ..., x_n$ via a kernel.
- But also, throughout the semester we have seen other possible moves...
  - Exploiting structure in ℓ:
    - Often $\ell(x)$ is likelihood or other density, hence nonnegative:

$$p(D) = \int p(D|x)p(x)dx \triangleq \int \ell(x)p(x)dx.$$

    - Maybe a ratio of integrals with common terms:

$$p(f|D) = \frac{\int p(f|D,\theta)p(D|\theta)p(\theta)d\theta}{\int p(D|\theta)p(\theta)d\theta}.$$

  - Active learning: choose point x_{i+1} based on observations $\ell(x_1), ..., \ell(x_i)$.
  - Model selection: in the simplest version we would do something like optimization of hyperparameters, but properly marginalizing over hyperparameters should improve accuracy.

Possible extensions

- BQ uses gp to share information about input points $x_1, ..., x_n$ via a kernel.
- But also, throughout the semester we have seen other possible moves...
  - Exploiting structure in ℓ:
    - Often $\ell(x)$ is likelihood or other density, hence nonnegative:

$$p(D) = \int p(D|x)p(x)dx \triangleq \int \ell(x)p(x)dx.$$

    - Maybe a ratio of integrals with common terms:

$$p(f|D) = \frac{\int p(f|D,\theta)p(D|\theta)p(\theta)d\theta}{\int p(D|\theta)p(\theta)d\theta}.$$

  - Active learning: choose point x_{i+1} based on observations $\ell(x_1), ..., \ell(x_i)$.
  - Model selection: in the simplest version we would do something like optimization of hyperparameters, but properly marginalizing over hyperparameters should improve accuracy.
- These extensions in BQ: [OGR+12, OGG+12, GOH14, GOG+14, HOG15].

Possible extensions

- BQ uses gp to share information about input points $x_1, ..., x_n$ via a kernel.
- But also, throughout the semester we have seen other possible moves...
  - Exploiting structure in ℓ:
    - Often $\ell(x)$ is likelihood or other density, hence nonnegative:

$$p(D) = \int p(D|x)p(x)dx \triangleq \int \ell(x)p(x)dx.$$

    - Maybe a ratio of integrals with common terms:

$$p(f|D) = \frac{\int p(f|D,\theta)p(D|\theta)p(\theta)d\theta}{\int p(D|\theta)p(\theta)d\theta}.$$

  - Active learning: choose point x_{i+1} based on observations $\ell(x_1), ..., \ell(x_i)$.
  - Model selection: in the simplest version we would do something like optimization of hyperparameters, but properly marginalizing over hyperparameters should improve accuracy.
- These extensions in BQ: [OGR+12, OGG+12, GOH14, GOG+14, HOG15].
- Theory for BQ is just starting; see `http://arxiv.org/abs/1512.00933`.

Exploiting structure in ℓ

Exploiting structure in ℓ

- If we know $\ell(x) \geq 0$ everywhere, a gp prior on $\ell(x)$ is a bad model.

Exploiting structure in ℓ

- If we know $\ell(x) \geq 0$ everywhere, a gp prior on $\ell(x)$ is a bad model.
- [OGG+12] uses a log transform, namely:

$$E(Z|D) = E_{\log \ell}(E_p(\ell)|D)$$

Exploiting structure in ℓ

- If we know $\ell(x) \geq 0$ everywhere, a gp prior on $\ell(x)$ is a bad model.
- [OGG+12] uses a log transform, namely:

$$\begin{aligned} E(Z|D) &= E_{\log \ell}\left(E_p(\ell)|D\right) \\ &= \int_{\log \ell} \left(\int_x \exp\left\{\log \ell(x)\right\} p(x) dx \right) p(\log \ell | D) d \log \ell \\ \text{where} \quad & \log \ell = \hat{\ell} \sim \mathcal{GP}(0, k), \end{aligned}$$

Exploiting structure in ℓ

- If we know $\ell(x) \geq 0$ everywhere, a gp prior on $\ell(x)$ is a bad model.
- [OGG$^+$12] uses a log transform, namely:

$$\begin{aligned} E(Z|D) &= E_{\log \ell}\left(E_p(\ell)|D\right) \\ &= \int_{\log \ell} \left(\int_x \exp\left\{\log \ell(x)\right\} p(x) dx \right) p(\log \ell | D) d \log \ell \\ \text{where} \quad & \log \ell = \hat{\ell} \sim \mathcal{GP}(0, k), \end{aligned}$$

which for tractability subsequently linearizes the integrand as:

$$\exp\left\{\log \ell(x)\right\} \approx \exp\left\{\log \ell_0(x)\right\} + \exp\left\{\log \ell(x)\right\} \left(\log \ell(x) - \log \ell_0(x)\right).$$

Exploiting structure in ℓ

- If we know $\ell(x) \geq 0$ everywhere, a gp prior on $\ell(x)$ is a bad model.
- [OGG+12] uses a log transform, namely:

$$
\begin{aligned}
E(Z|D) &= E_{\log \ell}\left(E_p(\ell)|D\right) \\
&= \int_{\log \ell} \left(\int_x \exp\left\{\log \ell(x)\right\} p(x) dx \right) p(\log \ell | D) d \log \ell \\
\text{where} \quad & \log \ell = \hat{\ell} \sim \mathcal{GP}(0, k),
\end{aligned}
$$

which for tractability subsequently linearizes the integrand as:

$$\exp\left\{\log \ell(x)\right\} \approx \exp\left\{\log \ell_0(x)\right\} + \exp\left\{\log \ell(x)\right\} \left(\log \ell(x) - \log \ell_0(x)\right).$$

- [GOG+14] uses a square-root transformation:

$$\hat{\ell} = \sqrt{2(\ell - \alpha)} \sim \mathcal{GP}(0, k), \qquad \text{such that } \hat{\ell}(x) = \alpha + \frac{1}{2}\hat{\ell}^2(x).$$

Exploiting structure in ℓ

- If we know $\ell(x) \geq 0$ everywhere, a gp prior on $\ell(x)$ is a bad model.
- [OGG+12] uses a log transform, namely:

$$\begin{aligned} E(Z|D) &= E_{\log \ell}\left(E_p(\ell)|D\right) \\ &= \int_{\log \ell} \left(\int_x \exp\left\{\log \ell(x)\right\} p(x) dx \right) p(\log \ell | D) d \log \ell \\ \text{where} \quad & \log \ell = \hat{\ell} \sim \mathcal{GP}(0, k), \end{aligned}$$

  which for tractability subsequently linearizes the integrand as:

$$\exp\left\{\log \ell(x)\right\} \approx \exp\left\{\log \ell_0(x)\right\} + \exp\left\{\log \ell(x)\right\} \left(\log \ell(x) - \log \ell_0(x)\right).$$

- [GOG+14] uses a square-root transformation:

$$\hat{\ell} = \sqrt{2(\ell - \alpha)} \sim \mathcal{GP}(0, k), \qquad \text{such that } \hat{\ell}(x) = \alpha + \frac{1}{2}\hat{\ell}^2(x).$$

- As you might expect these choices induce some technical details but improve estimation in the right settings.

Model selection

Model selection

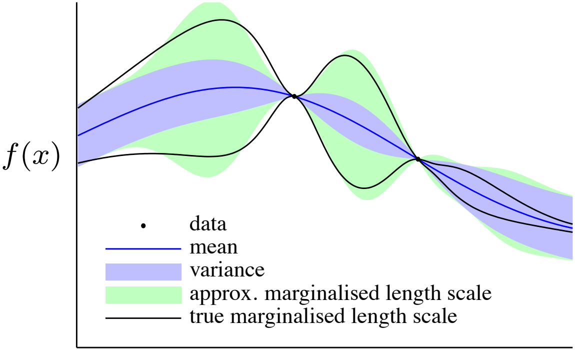
- In §02 we considered approximate integration of hyperparameters [GOH14].

Model selection

- In §02 we considered approximate integration of hyperparameters [GOH14].
- Accurate uncertainty estimates on ℓ seem valuable:

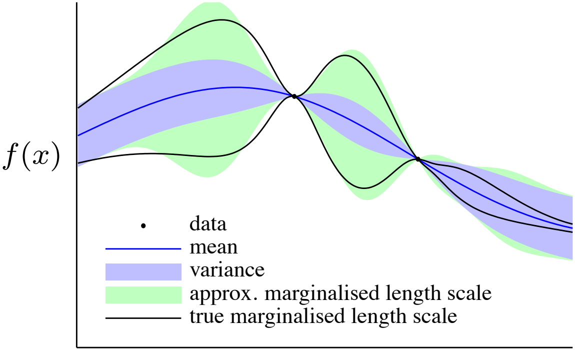
Model selection

- In §02 we considered approximate integration of hyperparameters [GOH14].
- Accurate uncertainty estimates on ℓ seem valuable:



Model selection

- In §02 we considered approximate integration of hyperparameters [GOH14].
- Accurate uncertainty estimates on ℓ seem valuable:



- [OGG+12, GOG+14, HOG15] use model selection to good effect.

Actively choosing quadrature points $x_1, ..., x_n$

Actively choosing quadrature points $x_1, ..., x_n$

- In §07 we considered bayesian active learning via [GSW$^+$15].

Actively choosing quadrature points $x_1, ..., x_n$

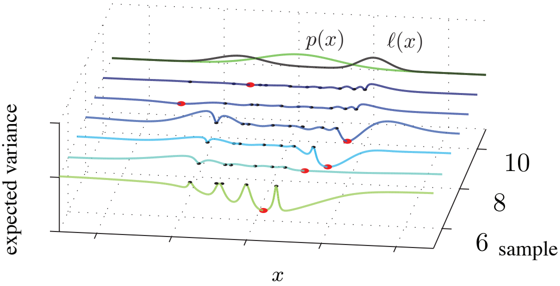
- In §07 we considered bayesian active learning via [GSW$^+$15].
- A sensible BQ acquisition function is to minimize the variance of the estimate $E(Z|D)$.

Actively choosing quadrature points $x_1, ..., x_n$

- In §07 we considered bayesian active learning via [GSW+15].
- A sensible BQ acquisition function is to minimize the variance of the estimate $E(Z|D)$.
- [OGG+12, GOG+14, HOG15] use this active learning to good effect:

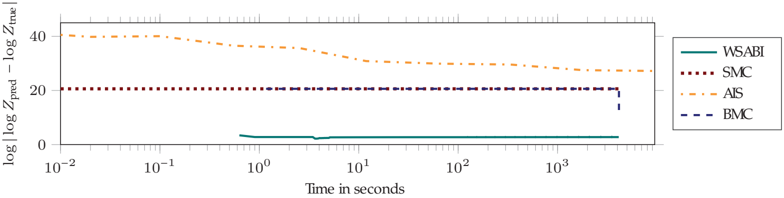
Actively choosing quadrature points $x_1, \ldots, x_n$

- In §07 we considered bayesian active learning via [GSW⁺15].
- A sensible BQ acquisition function is to minimize the variance of the estimate $E(Z|D)$.
- [OGG⁺12, GOG⁺14, HOG15] use this active learning to good effect:



Performance against other competitive sampling methods

Performance against other competitive sampling methods



- AIS: annealed importance sampling (from §02).
- SMC: simple Monte Carlo
- BMC: Bayesian Monte Carlo (what we called BQ [GR02]).
- WSABI: warped sequential active bayesian integration [GOG+14], which uses the tricks we just laid out (plus a bit).

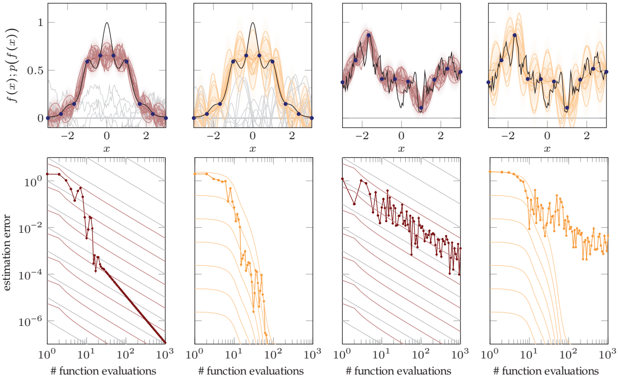
Convergence diagnostic in BQ? [HOG15]

Convergence diagnostic in BQ? [HOG15]

- Is $Var(Z|D) = Var(E_p(\ell)|D)$ a suitable BQ convergence diagnostic?

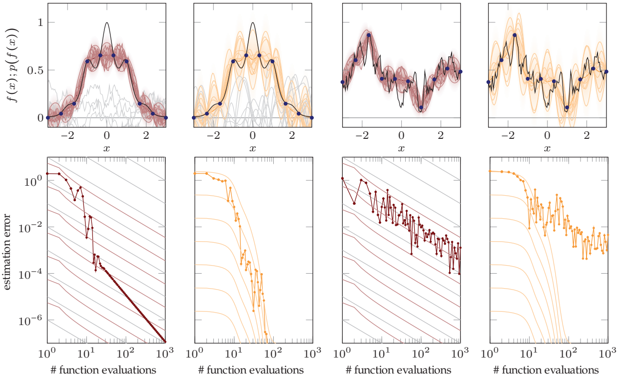
Convergence diagnostic in BQ? [HOG15]

- Is $Var(Z|D) = Var(E_p(\ell)|D)$ a suitable BQ convergence diagnostic?



Convergence diagnostic in BQ? [HOG15]

- Is $Var(Z|D) = Var(E_p(\ell)|D)$ a suitable BQ convergence diagnostic?



- Short answer: really only when the kernel is matched to the function itself.
- Bottom: error and posterior variance estimates thereof, showing the issue.

Outline

References

[GOG+14] Tom Gunter, Michael A Osborne, Roman Garnett, Philipp Hennig, and Stephen J Roberts.
Sampling for inference in probabilistic models with fast bayesian quadrature.
In *Advances in Neural Information Processing Systems*, pages 2789–2797, 2014.

[GOH14] Roman Garnett, Michael A Osborne, and Philipp Hennig.
Active learning of linear embeddings for gaussian processes.
UAI, 2014.

[GR02] Zoubin Ghahramani and Carl E Rasmussen.
Bayesian monte carlo.
In *Advances in neural information processing systems*, pages 489–496, 2002.

[GSW+15] Jacob R Gardner, Xinyu Song, Kilian Q Weinberger, Dennis Barbour, and John P Cunningham.
Psychophysical detection testing with bayesian active learning.
UAI, 2015.

[HOG15] Philipp Hennig, Michael A Osborne, and Mark Girolami.
Probabilistic numerics and uncertainty in computations.
In *Proc. R. Soc. A*, volume 471, page 20150142. The Royal Society, 2015.

[OGG+12] Michael Osborne, Roman Garnett, Zoubin Ghahramani, David K Duvenaud, Stephen J Roberts, and Carl E Rasmussen.
Active learning of model evidence using bayesian quadrature.
In *Advances in Neural Information Processing Systems*, pages 46–54, 2012.

[OGR+12] Michael A Osborne, Roman Garnett, Stephen J Roberts, Christopher Hart, Suzanne Aigrain, and Neale Gibson.
Bayesian quadrature for ratios.
In *International Conference on Artificial Intelligence and Statistics*, pages 832–840, 2012.

[O'H91] Anthony O'Hagan.
Bayes–hermite quadrature.
Journal of statistical planning and inference, 29(3):245–260, 1991.