Bayesian Quadrature

贝叶斯求积法

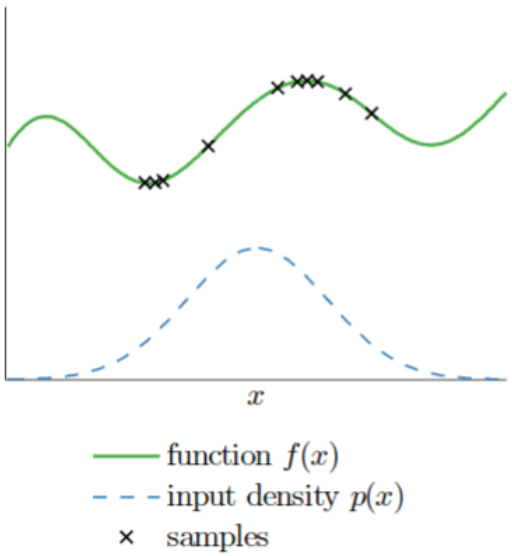
举个栗子

假设我们想要求定积分$\int_0^1 \exp\left(-\frac{(x-0.35)^2}{2(0.1)^2}\right)+\frac{\sin(10x)}{3}\mathrm{d}x$ 的值，直接求这个式子的解析解过于复杂，那么我们有什么比较好的办法求它的近似解呢?

一般时候我们会采用*Monte Carlo* 估计的方法去估计这个积分值，我们在$[0,1]$ 之间撒点，sample出来$\{x_i\}_{i=1}^N$ 个点，然后直接对定积分进行估计：

$$\int_0^1 \exp\left(-\frac{(x-0.35)^2}{2(0.1)^2}\right)+\frac{\sin(10x)}{3}\mathrm{d}x \approx \sum_{i=1}^N [\exp\left(-\frac{(x_i-0.35)^2}{2(0.1)^2}\right)+\frac{\sin(10x_i)}{3}]$$

然而蒙特卡洛的方法有些时候无法得到最优解：一是因为样本可能会有部分聚集在一团，影响了估计结果；二是比较接近的函数可能得到的结果是一样的。在文献[1]中，Hagan指出
MC estimation is fundamentally unsound, as it violates the likelihood principle, and moreover, does not make full use of the data at hand。因此蒙特卡洛方法在一定程度上仍然不够好。



这个时候，本篇的主人公Bayesian Quadrature就登场了。

Bayesian Quadrature

首先将问题重新定义一下：我们想要求定积分$I_p[f]=\int f(x)p(x)\mathrm{d}x$ 。

从原问题出发，我们本来是想要求解一个特别难解的定积分问题，那么我们是否可以将这个复杂的积分转化成可解的简单一些的积分呢？即我们先假设数据具有某个分布的形式，然后我们去根据样本数据找它的后验分布。最后通过求后验分布的积分，来解决最开始积分的问题。这就是从贝叶斯的角度看待这个问题。

我们可以将定积分的值当做是一个随机变量Z，即 $Z=\int_0^1 f(x)\mathrm{d}x$，然后我们选择为$Z$ 选择适当的先验分布，并根据Bayes' rule通过观测的data去找到其后验分布。在实际问题中，找f 的先验比找Z 的先验更加容易一些，并且在这里选择高斯过程(Gaussian process)作为先验很方便，因此我们能够得到：

$$p(f)=\mathcal{GP}(f;\mu,K)$$

为什么说GP在这里比较方便用呢？就是因为高斯过程在仿射变换$L:f\mapsto L[f]$ 中是封闭的，即 $p(L[f])=\mathcal{GP}\left(L[f];L[\mu],L^2[K]\right)$，而定积分实际上是一种线性操作。即我们可以得到：

$$p\left(\int_0^1 f(x)\mathrm{d}x\right)=\mathcal{N}\left(Z;\int_0^1\mu(x)\mathrm{d}x,\int_0^1\int_0^1 K\left(x,x'\right)\mathrm{d}x\mathrm{d}x'\right)$$

实际上Bayesian Quadrature就是将求解比较难的积分问题转化为了先求被积函数f 上的回归问题再求回归模型的积分。

BQ的好处

通过选择GP做先验，一方面让原问题变的更加容易，另一方面我们也可以通过方差来衡量我们对积分结果的不确定性。

此外，BQ的收敛速度比较快，适合少量数据时候使用。在实践中，可能我们需要尽可能少的点去尽量近似原积分，因此我们应当尽可能选择比较好的数据，这些数据也就可以根据acquisition function来计算得到[2]。

> Why is this useful? The main advantages to this approach are that we may explicitly model the structure of f via the covariance function K, and that the posterior variance of the integral may be used to derive an active sampling scheme, revealing the most-informative points to evaluate the function so as to estimate the integral with the highest precision. Note that the posterior variance of the integral only depends on where we sample the function, and not the actual values we observe. This property can be exploited to design optimal quadrature rules.

其实BQ的核心到这里就结束了，但是我们到底应该怎么使用这种方法呢？

流程

Given the surrogate GP on f , the acquisition function a , and a function handle to $f(x)$, BQ essentially iterates the following three steps until the budget of N evaluations is used up:

1. fit the GP $p(f|D_n)$ on the currently available dataset $D_n\{(f\left(x_i\right),x_i)\}_{i=1}^n$
2. condition the acquisition function $a_n(x)$ on $p(f|D_n)$ and find the maximizer, i.e., $x_{n+1}=\arg\max_{x\in\mathbb{X}}a_n(x)$.
3. evaluate the objective function at x_{n+1} to obtain $f(x_{n+1})$, and add the new observation to the dataset $D_{n+1}\leftarrow D_n\cup\{x_{n+1},f\left(x_{n+1}\right)\}$
4. compute the integrate

详细例子见[2]。

在机器学习中的应用

- 求期望的时候可以用这种方法：$\mathbb{E}_p[f] = \int f(x)p(x)\mathrm{d}x$
- 模型选择时候的model evidence：$Z = p(y|X, \mathcal{M}) = \int p(y|X, \theta, \mathcal{M})p(\theta|\mathcal{M})\mathrm{d}\theta$
- 预测 $p\left(y^*|x^*, \mathcal{D}\right) = \int p\left(y^*|x^*, \mathcal{D}, \theta\right) p(\theta|\mathcal{D})\mathrm{d}\theta$

总结

- GP作为先验有时候并不一定完全适合所有情况，但是其他的方法更难解。
- BQ这种方法更偏向适用于函数比较平稳变化，没有过大突变的情况。
- BQ只适合于比较低的维度(<10)，并且只适用于计算$f(x)$比较昂贵的情况。
- How to extend to high dimensions? Gradient observations are helpful, but a D-dimensional gradient is D separate observations.

参考文献

[1]: A. O'Hagan. Monte-Carlo is fundamentally unsound. The Statistician, 36:247{249, 1987.

[2]: tutorial https://nbviewer.jupyter.org/github/amzn/emukit/blob/master/notebooks/Emukit-tutorial-Bayesian-quadrature-introduction.ipynb

[3]: talk http://probabilistic-numerics.org/assets/pdf/nips2015_probint/roman_talk.pdf

[4]: David Duvenaud. Bayesian Quadrature:Model-based Approximate Integration

[5]: lecture_note https://www.cse.wustl.edu/~garnett/cse515t/spring_2017/files/lecture_notes/11.pdf