Control variates with kernel smoothing : toward faster than root n rates

Benjamin Cohen, Arthur Gourrin, Julien Toucheboeuf, Hugues René-Bazin

23 mars 2022

Table des matières

Introduction

This document is the summary of our work on Control variates with Kernel smoothing. The goal of this project is to see if Kernel smoothing enables us to be more efficient than other control variates in approximating the integral :

$$I = \int gf d\lambda \tag{1}$$

. Where f is a density function on $\mathbb{R}^d$ and g a function on $\mathbb{R}^d \mapsto \mathbb{R}$. As we might not be able to draw random variables directly from f, we consider a sampler q of the density f.

In a first part we create conventional control variates and compute the approximation of I. Then we will be taking the Kernel smoothing estimators as control variates (in order to control the variance [1]). In a third time we will implement the OLS to those control variables in order to select the best q, the ultimate goal being to compare the efficiency of the different control variables.

1 Creation and Analysis of control variables

The estimator we will focus on is :

$$\hat{I}_n(\phi) = \frac{1}{n}\sum_{i=1}^{n} \frac{(g(X_i)f(X_i) - \phi(X_i))}{q(X_i)}$$

With $\phi(x) = \sum_{k=1}^{m} \beta_k \phi_k(x)$ such that $\forall k \int \phi_k = 0$.

1.1 Why do we need $\hat{\phi}_m$?

The main goal is to minimze the MSE of the method.

When we write it with $\hat{I}_n(\phi)$, we get : $MSE = 0 + V(|\hat{I}_n(\phi) - I|^2)$, with the variance. We would like to find the optimal coefficients $\beta^* \in \mathbb{R}^m$ that minimises the Variance.

$$V_q(\hat{I}_n(\phi)) = \frac{1}{n^2} n V_q(\frac{(g(X_1)f(X_1) - \phi(X_1)}{q(X_1)}) = \frac{1}{n} V_q(\frac{g(X_1)f(X_1) - \phi(X_1)}{q(X_1)}) = \frac{\sigma_{as}^2}{n}$$

With σ_{as}^2 the asymptotic variance.

Since the estimator is unbiased, the asymptotic variance can be rewritten as :

$$V_q(\frac{g(X_1)f(X_1) - \beta^T\phi(X_1)}{q(X_1)}) = \mathbb{E}_q[|\frac{g(X_1)f(X_1) - \beta^T\phi(X_1)}{q(X_1)}|^2]$$

So, it can be minimised over β and the minimiser β_{as} can written as :

$$\beta_{as} = \mathbb{E}[(Z^T Z)]^{-1}\mathbb{E}[ZY] \tag{2}$$

With the notations :

1. $Z = \dfrac{\phi(X_1)}{q(X_1)}$ with $\phi(X_1) = (\phi_1(X_1), ..., \phi_n(X_1))^T$
2. $Y = \dfrac{g(X_1)f(X_1)}{q(X_1)}$
3. NB : we choose the $(\phi_k)_k$ linearly independent, so that $\mathbb{E}[Z^T Z] \in GL_n(\mathbb{R})$

This gives ϕ_{as} (because, $\phi_{as} = \beta_{as}^T \phi$) :

$$\phi_{as} = (\mathbb{E}[(Z^T Z)]^{-1}\mathbb{E}[ZY])^T \phi \tag{3}$$

That gives us the minimal asymptotic variance, which is :

$$\begin{aligned}\sigma_{min} &= V_q(\frac{g(X_1)f(X_1) - \beta_{as}^T \phi(X_1)}{q(X_1)}) \\ &= V_q(\frac{f(X_1)g(X_1)}{q(X_1}) - \mathbb{E}[...]\end{aligned}$$

The previous study gives us the optimal variance that we reach with β_{as}, but in fact we can't compute explicitly β^* so we need an estimator $\hat{\beta}$ to approximate it.

1.2 Computation of $\hat{\phi_m}$:

The main concern with control variates is to lower the variance. So we want to find an optimal $\hat{\phi_m}$ to do so.
But $\hat{\phi_m} = \hat{\beta}^T \phi$ with ϕ constant so the quantity that we want to optimise is $\beta \in \mathbb{R}^m$.

1.2.1 OLS problem

To do so, we will choose $\hat{\beta}$ as the argmin of the empirical variance :

$$\hat{\beta} \in \underset{\beta \in \mathbb{R}^m}{\operatorname{argmin}} \frac{1}{n-1} \sum_{i=1}^{n} \left(\frac{f(X_i)g(X_i)}{q(X_i)} - \beta Z_i - \hat{I}_n(\beta) \right)^2$$

With $\forall i \in \{1, ..., n\}, Z_i = (\dfrac{\phi_1(X_i)}{q(X_i)}, ..., \dfrac{\phi_m(X_i)}{q(X_i)})^T$.

1.2.2 Expression of $\hat{\phi}$

If we apply the Hilbert projection Theorem, we get that $\hat{\beta} = (Z_c^T Z_c)^{-1}(Z_c)G$.

Where : $Z_c = (Z_1 - \bar{Z}, Z_2 - \bar{Z}, ..., Z_n - \bar{Z})^T$, $\bar{Z} = \frac{1}{n}\sum_{i=1}^{n} Z_i$ and $G = (\frac{f(X_1)g(X_1)}{q(X_1)}, ..., \frac{f(X_n)g(X_n)}{q(X_n)})^T$.

According to the lecture notes 3.3.1 [3]

This gives us :

$$\hat{\phi}_m = \hat{\beta}\phi \tag{4}$$

The estimator $\hat{I}_n(\hat{\phi}_m)$ is biased but we can do the computations over its asymptotic properties.

1.3 Asymptotic study of $\hat{I}_n(\hat{\phi}_m)$

The asymptotic study moreover gives :

$$\sqrt{n}(\hat{I}_n(\phi) - I) = \sqrt{n}(\hat{I}_n(\phi) - \mathbb{E}[g(X_1)]) \xrightarrow{d} \mathcal{N}(0, \sigma(\phi)) \tag{5}$$

With $\sigma_m^2 = \underset{\beta \in \mathbb{R}^m}{min} V(g(X_1)f(X_1) - \beta^T Z_1)$.

We can write $\hat{I}_n(\hat{\phi}_m)$ as :

$$\hat{I}_n(\hat{\phi}_m) = (1 - \hat{\beta}_n^T)n^{-1}\sum_{i=1}^{n}\begin{pmatrix} g(X_i)f(X_i)/q(X_i) \\ Z_i \end{pmatrix}$$

If we use the law of large number on each part of the expression, we get :

1. $n^{-1}\sum_{i=1}^{n} g(X_i) \mapsto \mathbb{E}[g(X_1)]$
2. $n^{-1}\sum_{i=1}^{n} Z_i \mapsto \mathbb{E}[Z_1] = 0$

1.4 Numerical Results

After the theorical study we did some numerical simulation of the efficiency of the Kernel Smoothing method. At this point of the study we compared 3 Monte Carlo estimators :

- Naïve Monte-Carlo
- Sklearn Monte-Carlo (with Legendre Control Variates)
- Legendre Control Variates (OLS method coded directly)

The first figure highlights the results we got when it came to approximating the I thanks those methods.

In a first time we had some issues because we had not recentered our control variates which gave us a worse result with the Control variates estimation than with the naïve

estimation. But the figure 1 gives the corrected results.
It shows the variates of the log of the MSE depending on n (the number of data points) with steps of 100. To obtain those plots, we consider the function $g : x \mapsto x$ and we take for f the normal Gaussian density (mean 0, variance 1) in dimension 1. For the sampler q, we take the Gaussian density with mean $\theta = 0.1$ and variance 1.

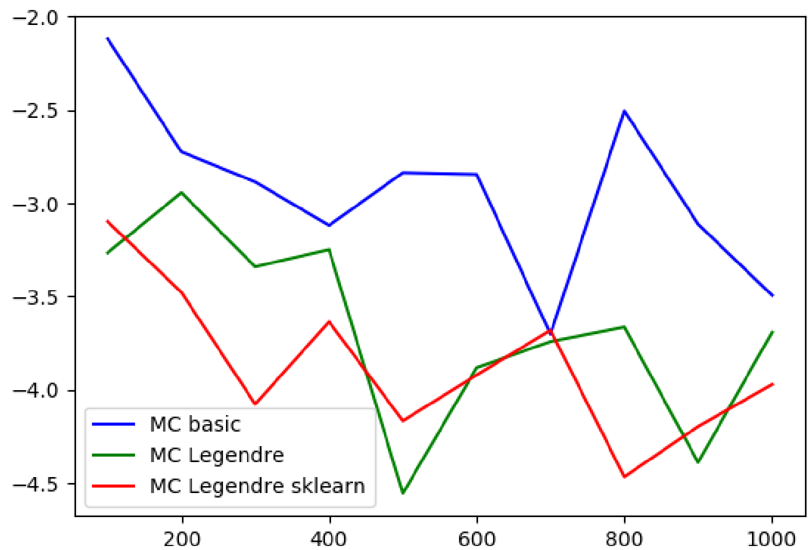

FIGURE 1 – Legendre CV vs MC Basic

2 Kernel smoothing estimators

Kernel smoothing is a non parametric method that enables us to estimate a given function thanks to Kernels.
The principle of Kernel estimation (or histogram estimation) is very simple : we take n iid samples $(X_i)_i$ that follows the unknown distribution to estimate and thanks to a Kernel K we compute $K(\dfrac{x - X_j}{h})$ (where h is the called the bandwidth) with which we will approximate the function near the point x.

The link between this estimator and control variates is that :
$\mathbb{E}_q[\frac{K_h}{q}] = 1 \leftrightarrow \mathbb{E}_q[K_h - 1] = 0$, so $K_h - 1$ is a control variate.
Kernel Smoothing is not only a density estimator but we use it here as a control variate.

So we can deduce a new control variate estimator for I that can be written as :

$$h(X) = \sum_{k=1}^{m} \alpha_k h_x(X_j) \tag{6}$$

Where, we denote : $h_x(X) = \frac{K_h(x-X)}{q(X)} - 1$.

In order to make the MSE go to 0 we need to pick the right weights for our coefficients $(\alpha_i)_i$. using the same method as before the OLS minimisation of the Variance.

2.0.1 Minimisation of the MSE

The goal of this part is to find conditions under which $MSE \to 0$.

We get almost the same optimisation problem as before : we want to optimize the empirical variance so we get, the problem :

$$\hat{\alpha_i} \in \underset{\alpha \in \mathbb{R}^m}{\operatorname{argmin}} \sum_i \left(\frac{f(X_i)g(X_i)}{q(X_i)} - \sum_{j=1}^{m} \alpha_j \left(\frac{K_h(X_i - X_j)}{q(X_i)} - 1 \right) \right)^2 \tag{7}$$

2.1 Numerical Study of the new estimator

Now that we have defined new control variate estimators for our Monte Carlo method we can study them numerically to see if they are more efficient than the previous ones.

To do so, we have decided to choose the weight of each control variate with OLS algorithm on the one hand and with LASSO on the other hand. With OLS, we try to find $\beta \in \mathbb{R}^m$ minimizing $||Y - \beta^T Z||$ (m is the number of control variates, Z the matrix of those control variates evaluated on the (X_i) and Y is such that $Y_i = \frac{g(X_i)f(X_i)}{q(X_i)}$). With LASSO, we penalize the $L1$ norm of β, so we try to solve the OLS problem but with a β as small as possible in the $L1$ norm sense. We try to minimize $||Y - \beta^T Z|| + \lambda ||\beta||_1$.

To obtain the test plots, we consider the function $g : x \mapsto x$ and we take for f the sum of two Gaussian density in dimension $d = 10$: $f = \frac{1}{2}f_1 + \frac{1}{2}f_2$ with f_1 centered on $(-1, -1, ..., -1)$ and f_2 centered on $(1, 1, ..., 1)$. For the sampler q, we also take the sum of two Gaussian density : $q = \frac{1}{2}q_1 + \frac{1}{2}q_2$, with q_1 centered on $(-1-\theta, -1-\theta, ..., -1-\theta)$ and q_2 centered on $(1+\theta, 1+\theta, ..., 1+\theta)$. We always take Id for the variance matrix.

The next figure show the comparison between those estimators. This graph illustrates the log MSE of the different methods that were used here.

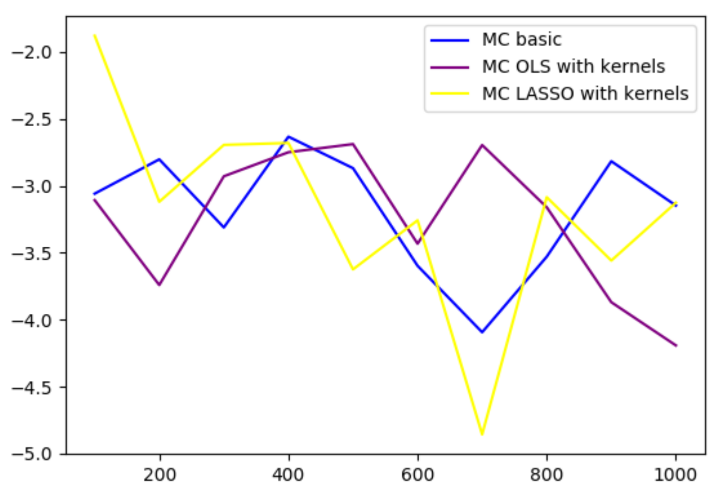

FIGURE 2 – Kernel vs Basic Monte Carlo

3 New estimator based on Kernels

In this section we study :

$$\tilde{\phi}_m(x) = \frac{1}{m}\sum_{i=1}^{m}\frac{g(X'_j)f(X'_j)}{q(X'_j)}K_h(x - X'_j)$$

This is an estimator of gf , we can give its bias and variance :

3.1 Bias & Variance of the estimator

3.1.1 Bias

$$\begin{aligned}
\mathbb{E}_q[\tilde{\phi}_m(x)] &= \int (g(y)f(y))/q(y) \times K_h(x-y)dy \\
&= gf * K_h(x), \text{ produit de convolution} \\
&= (gf * K_h)(x) - gf(x) + gf(x) \\
&\overset{u=\frac{x-y}{h}}{=} \; gf(x) + \int (gf(x-hu) - gf(x)) \times K(u)du
\end{aligned}$$

3.1.2 Variance

$$\begin{aligned}
\sigma_K^2(x) &= V_q(\tilde{\phi}_m(x)) \\
&= \frac{1}{m^2} \times m \times V_q(\frac{g(X'_1)f(X'_1)}{q(X'_1)}K_h(x - X'_1)) \\
&= \frac{1}{m}V_q(\frac{g(X'_1)f(X'_1)}{q(X'_1)}K_h(x - X'_1))
\end{aligned}$$

3.2 From the estimator to control variates

Now that we have an estimator of gf, what we want to have is a control variate (meaning a null mean).

The mean of $\tilde{\phi}$ is for X a rv is :

$$\mathbb{E}[\frac{\tilde{\phi}_m}{q}(X)] = \frac{1}{m}\sum_{j=1}^{m}\frac{g(X'_j)f(X'_j)}{q(X'_j)}K_h(X - X'_j) = \hat{\mu}_m$$

So, to get a null mean, we take the control variates :

$$\forall i \in \{1, ..., n\}, Z_{m,i} = \frac{\tilde{\phi}_m}{q}(X_i) - \hat{\mu}_m \tag{8}$$

3.3 Monte Carlo with the CV

We now use the control variates that we have computed before in the Monte Carlo estimator, we get :

$$\hat{I}_n^{cv}(\tilde{\phi}_m) = \frac{1}{n}\sum_{i=1}^{n}\frac{g(X_i)f(X_i) - Z_{m,i}}{q(X_i)}$$

Now we have to estimate the bias, the variance and the asymptotical properties of this quantity.

3.3.1 Bias & Variance

Bias

$$\mathbb{E}[\hat{I}_n^{cv}(\tilde{\phi}_m)] = \mathbb{E}[\frac{g(X_1)f(X_1) - Z_{m,1}}{q(X_1)}] = I \tag{9}$$

So we have an unbiaised estimator

Variance

$$V(\hat{I}_n^{cv}(\tilde{\phi}_m)) = \frac{V(\frac{g(X_1)f(X_1)}{q(X_1)}) + V(\frac{\phi(X_1)}{q(X_1)})}{n} - 2Cov\Big(\frac{g(X_1)f(X_1)}{q(X_1)}, \frac{\phi(X_1)}{q(X_1)^2}\Big) - I^2$$

4 Summary of the numerical estimations

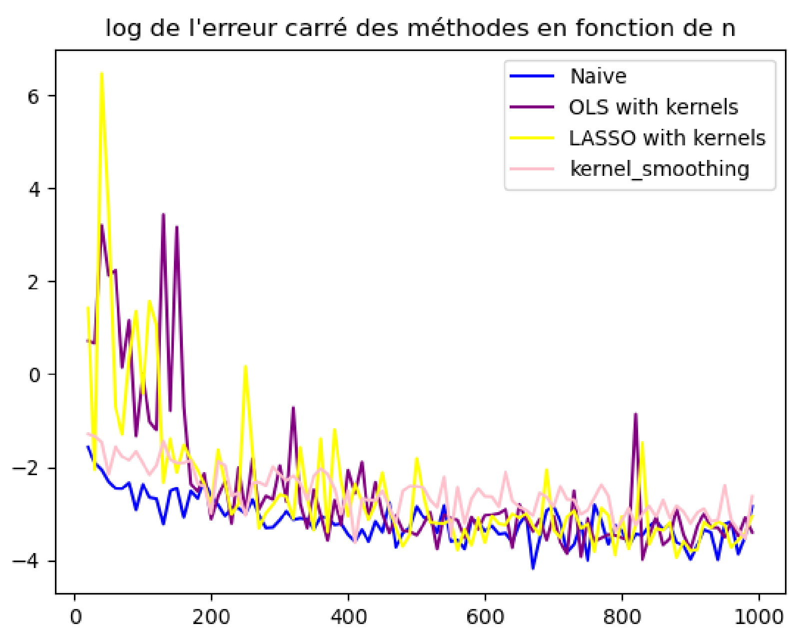

FIGURE 3 – Summary of all the methods used threw this project

So far, we tested two main methods to compute a given integral with stochastic algorithms. In our first attempt, we decided to focus on a simple control variate method as described in the previous parts. To do this, we obviously needed a set of given control variables which integral was known.
Randomly, we chose Legendre polynomials as, on their support, their integral is always 1. The results were not those expected since we didn't know that it was necessary to re-centre the control variables.
Thereafter, once this change achieved, the results were those expected with the results stemming from the control variate method being more accurate than those of the simple Monte-Carlo algorithm. Then, we decided to build a new algorithm of control variate but with Gaussian kernels as new control variables. The results are coherent since it is still more efficient than a simple MC but there are not truly different from those with Legendre polynomials.
Moreover, as it will be shown in the figure 3, we decided to use LASSO algorithm for solving the optimisation problem in some algorithms whose aim is to avoid computational issues when the number of control variables or the number of samples become too important as it is explained in the article of Portier, Leluc and Segers [2].
Yet, regarding the results from OLS, the LASSO results are not significantly different.
Last but not least, we tried to use a density estimator based on kernels to compute the aimed integral.
The results are significantly the same for the control variate method even if with a smaller sample of points, kernel-based estimator performs well enough to be considered.

Références

[1] Johan Segers François Portier. Monte carlo integration with a growing number of control variates, arxiv :1801.01797. October 2019.

[2] Johan Segers François Portier, Rémi Leluc. Control variate selection for monte carlo integration, arxiv :1906.10920v5 [math.st]. 1 Apr 2021.

[3] François Portier. *Monte Carlo Lecture Notes.*