Project Report

Pavlov Sergey

March 2019

1 Problem statement

In this work given two gaussian random variables $Y \sim \mathcal{N}(0,1), Z \sim \mathcal{N}(0,1)$ we shall calculate two following quantities:

- $p_0 := \mathbb{P}(Y > 3)$;
- $p_1 := \mathbb{P}(Y > 3, Y + Z < -1)$;

These quantities are probabilities of two rare events, moreover one can observe that $1 >> p_0 >> p_1$. In this document we compare:

- the **classical** Monte-Carlo(MC) approach;
- a method of **importance sampling**;
- a method of **particle filter**(méthode particulaire);

We note that the true value of p_0 can be expressed as $CDF(-3)$ for the gaussian random variable and equals to **0.0013498980**.

2 Classical approach

We approach p_0 by

$$\hat{p}_0^N = \sum_{i=1}^{N} 1_{\{Y_i > 3\}},$$

and p_1 by

$$\hat{p}_1^N = \sum_{i=1}^{N} 1_{\{Y_i > 3, Y_i + Z_i < -1\}}$$

for $Y_1, \ldots, Y_N, Z_1, \ldots, Z_N \sim \mathcal{N}(0,1)$ - i.i.d.

By the strong law of large numbers we have:

$$\hat{p}_0^N \xrightarrow{a.s} p_0,$$

$$\hat{p}_1^N \xrightarrow{a.s} p_1, N \to \infty,$$

and by the central limit theorem:

$$\sqrt{N}(\hat{p}_0^N - p_0) \xrightarrow{d} \mathcal{N}(0, p_0(1-p_0)),$$

$$\sqrt{N}(\hat{p}_1^N - p_1) \xrightarrow{d} \mathcal{N}(0, p_1(1-p_1)), N \to \infty$$

.

We can of course approach the variance by $\bar{\sigma}_i^2 := \hat{p}_i(1-\hat{p}_i)$ and we write a confidence interval for $\alpha = 95\%$:

$$I_i := [\hat{p}_i^N - \frac{1.96\bar{\sigma}_i}{\sqrt{N}}, \hat{p}_i^N + \frac{1.96\bar{\sigma}_i}{\sqrt{N}}],$$

$$\mathbb{P}(p_i \in I_i) \approx 0.95, i = 0, 1$$

2.1 Results

$N = 10^8$	value	CI($\pm$)
p_0	0.001352	7.2e-06
p_1	1e-08	1.95e-08

$N = 10^9$	value	CI($\pm$)
p_0	0.001350	2.3e-06
p_1	1.9e-08	8.5e-09

Conclusion: for p_0 the classical method works well enough. The order of p_1 is too small and we need to reduce variance of our estimator.

3 Importance sampling

We first present the mean shift trick. For any given function ϕ (such that all integrals below exist), $a > 0, X \sim^{\mathbb{P}} \mathcal{N}(0,1)$, we write:

$$\mathbb{E}^{\mathbb{P}} e^{-\frac{a^2}{2}+aX}\phi(X) = \frac{1}{\sqrt{2\pi}}\int e^{-\frac{a^2}{2}} e^{ax-\frac{x^2}{2}}\phi(x)dx = \frac{1}{\sqrt{2\pi}}\int e^{\frac{-(x-a)^2}{2}}\phi(x)dx =$$

$$= \frac{1}{\sqrt{2\pi}}\int e^{\frac{-z^2}{2}}\phi(z+a)dx = \mathbb{E}^{\mathbb{P}}\phi(X+a) = \mathbb{E}^{\mathbb{Q}^a}\phi(X),$$

for a change of measure defined by:

$$Z^a(X) = \frac{d\mathbb{Q}^a}{d\mathbb{P}} := e^{-\frac{a^2}{2}+aX}$$

Let $\phi(x) := \mathbb{1}(x > 3)$ and $\tilde{\phi}(x) := \mathbb{1}(x > 0)$, then we have($a = -3$):

$$p_0 = \mathbb{E}^{\mathbb{P}}\phi(Y) = \mathbb{E}^{\mathbb{P}}\tilde{\phi}(Y-3) = \mathbb{E}^{\mathbb{Q}^a}\tilde{\phi}(Y) = \mathbb{E}^{\mathbb{P}} Z^a(Y)\tilde{\phi}(Y)$$

We observe that now $\tilde{\phi}$ is not zero with a probability 0.5 and value of p_0 is controlled by the term $Z^a(Y)$ which is supposed to be essentially small.
Now using the independence between Y and Z in the definition of p_1 and using what we've just derived here above(for two changes of measure for $a_1 = -3$ and $a_2 = 4$) we obtain the following formula for p_1:

$$p_1 = \mathbb{E}^{\mathbb{P}} Z^{a_1}(Y) Z^{a_2}(Z) \hat{\phi}(Y, Z),$$

where $\hat{\phi}(x, y) := 1_{\{x>0, x+y<0\}}$. We then use again the approximations and apply SLLN and CLT as in the previous chapter. The only difference is that in this case the variance of terms under expectations is not easily calculable, so we calculate an **approximation of variance** in both cases($\bar{\sigma_0}^2$ and $\bar{\sigma_1}^2$ respectively). We write again:

$$I_i := [\hat{p}_i^N - \frac{1.96\bar{\sigma}_i}{\sqrt{N}}, \hat{p}_i^N + \frac{1.96\bar{\sigma}_i}{\sqrt{N}}],$$

$$\mathbb{P}(p_i \in I_i) \approx 0.95, i = 0, 1$$

3.1 Results

$N = 10^8$	value	CI($\pm$)
p$_0$	0.0013499	4.8e-07
p$_1$	1.875e-08	2.7e-11

Conclusion: this method behaves well and shows good performance for both p_0 and p_1. One of the possible **disadvantages** (in general case, of course) could be the fact that we don't know the structure of (indicator) functions and to find good values a, a_1 and a_2 could be generally difficult.

4 Particle filter method

We shall apply this method and calculate p_1(p_0 doesn't involve the interaction between variables).
We'll here briefly explain how we use the formulas from the chapter II of the book in terms of notations from the book:

$$X_{0:1} = (X_0, X_1),$$
$$g_0(X_{0:0}) = g_0(X_0),$$
$$\mathbb{E} f_1(X_0, X_1) = \eta_1(\tilde{f}_1)\eta_0(g_0),$$
$$\tilde{f}_1(x_0, x_1) = \frac{f_1(x_0, x_1)}{g_0(x_0)},$$
$$f(x_0, x_1) = \mathbb{1}_{\{x_0>3, x_1<-1\}},$$

$$(X_1|X_0 = x) \sim \mathcal{N}(x, 1)$$

To implement the algorithm we:

- simulate $(X_0^i)_{i=1}^N \sim \mathcal{N}(0, 1)$;
- we re-sample $(X_0^i{}')$ from (X_0^i) with respect to the weights $(g(X_0^i))_i$;
- we sample $X_1^i \sim \mathcal{N}(X_0^i{}', 1)$;
- we calculate the approximation of p_1 using $(X_0^i{}')$ and (X_1^i) using the formula for the expectation above;

4.1 Confidence interval calculation

Here we explain how we calculate the confidence interval for a product of two approximations. Let in general case we have a function $g(t_1, t_2)$ of two random variables X and Y: $g(X, Y)$. Applying **Delta method** principle here, we obtain:

$$Var(g(X,Y)) \approx g_x'^2(\theta)Var(X) + g_y'^2(\theta)Var(Y) + 2g_y'(\theta)g_x'(\theta)Cov(X,Y),$$

where

$$\theta = \mathbb{E}(X, Y),$$

$$\mathbb{E}g(X, Y) \approx g(\theta)$$

In our case $g(x, y) = xy$, so we get:

$$Var(\hat{X}\hat{Y}) \approx \hat{Y}^2 Var(\hat{X}) + \hat{X}^2 Var(\hat{Y}) + 2\hat{X}\hat{Y}Cov(\hat{X}, \hat{Y}),$$

$$Var(\hat{X}) = \frac{Var(X)}{N} \approx \frac{\hat{Var}(X)}{N}$$

$$Var(\hat{Y}) = \frac{Var(Y)}{N} \approx \frac{\hat{Var}(Y)}{N}$$

$$Cov(\hat{X}, \hat{Y}) = \frac{Cov(X,Y)}{N} \approx \frac{\hat{Cov}(X,Y)}{N}$$

To obtain a confidence interval we use:

$$\hat{X}\hat{Y} \pm 1.96\sqrt{Var(\hat{X}\hat{Y})}$$

4.2 Subsampling

We implement two different ways of subsampling:

- stratified sampling (via the binary search($\log N$ implementation));
- multinomial sampling with a Tensorflow pluged-in function;

We report that the second option has worked much faster and we will provide results for it.

4.3 Results

We try out two families of possible functions g_0: **exponential** and **polynomial**. In both cases we clip values by 10^{-15} from below. Here are results for the polynomial family($g_0(x) = (x^k - 10^{-15})_+ + 10^{-15}$):

$N = 10^8$	k	value	CI($\pm$)
	1	2.8e-08	1.2e-08
	2	2.6e-08	1.0e-08
	3	1.7e-08	4.1e-09
	4	1.8e-08	4.7e-09
	5	1.86e-08	2.8e-09
	7	1.72e-08	2.1e-09
	9	2.0e-08	2.0e-09

Here are results for the polynomial family($g_0(x) = (e^{\alpha x} - 10^{-15})_+ + 10^{-15}$):

$N = 10^8$	α	value	CI($\pm$)
	0.1	1.5e-08	2.1e-08
	0.2	1.7e-08	1.9e-08
	0.5	1.9e-08	1.3e-08
	0.7	1.75e-08	9.9e-09
	1.0	2.05e-08	7.6e-09
	1.5	1.9e-08	4.6e-09
	2.0	2.1e-08	3.4e-09
	3.0	1.87e-08	2.4e-09
	4.0	1.85e-08	3.1e-09
	5.0	1.96e-08	4.1e-09

Conclusion: if we choose the function g_0 wisely we can shrink the confidence interval up to 4 times compared to the classical method. Though, the performance of the importance sampling method is far to be reachable for this algorithm.