Stochastic programming
SAA: adaptive sampling

Fabian Bastin

bastin@iro.umontreal.ca

Université de Montréal – CIRRELT – IVADO – Fin-ML

Motivation

Reminder: we consider the stochastic problem

$$\min_{x \in S} g(x) = E_P\left[G(x, \xi)\right],$$

where

- $x \in \mathbb{R}^m$
- S is a compact subset of $\mathbb{R}^m$
- ξ is a real random vector defined on $(\Xi, \mathcal{F}, P)$ and taking values in $(\mathbb{R}^k, \mathcal{B}^k)$ ($\mathcal{B}^k$ is the Borel measure)
- $G : \mathbb{R}^m \times \mathbb{R}^k \to \mathbb{R}$

Sample average approximation:

$$\min_{x \in S} \hat{g}_N(x) = \frac{1}{N} \sum_{i=1}^{N} G(x, \xi_i),$$

Convergence

In addition to our previous consistency results for $N \to \infty$, the central limit theorem tells us that, if the draws are independent and identically distributed (i.i.d.) (and finite $g(x)$),

$$\sqrt{N}[\hat{g}_N(x) - g(x)] \Rightarrow N(0, \sigma^2(x)),$$

where $\sigma^2(x) = \text{var}(G(x, \xi))$, and $\Rightarrow$ denotes convergence in distribution.

This result is only valid for a given x. It is necessary to set stronger conditions in order to have a functional convergence.

Note: under our assumptions, $\hat{g}_N(x)$ is continuous over S, and can thus be considered as a point in the Banach space $C(s)$.

Banach space $C(s)$

$C(S)$ is the space of continuous functions $\psi : S \to \mathbb{R}$, equipped with the sup-norm $\|\psi\| := \sup_{x \in S} |\psi|$

$C(S)$ is a Banach space, i.e. a normed vectorial space, complete under the distance issued from its norm. A metric space M is said complete or complete space if every Cauchy suite in M has a limit in M (i.e. it converges in M).

We will extend the (pointwise) central limit theorem to a functional central limit theorem, assuming as usual that the draws are i.i.d.

Functional central limit theorem

Assume that:

1. For all $x \in S$, the function $G(x, \cdot)$ is measurable (in other words, its expectation exists).
2. $E_P[G(\overline{x}, \xi)^2] < \infty$ for some point $\overline{x} \in S$.
3. (Lipschitz continuity condition) $\exists\, K(\xi) \geq 0$ such that $E[K(\xi)]$ is finite, and $\forall\, x_1, x_2 \in S$ and a.e. ξ,

$$|G(x_1, \xi) - G(x_2, \xi))| \leq K(\xi)\|x_1 - x_2\|,$$

We assume moreover that $E[K^2(\xi)] < \infty$ is finite.

Functional central limit theorem (cont'd)

Under these conditions,

$$N^{1/2}[\hat{g}_n - g] \Rightarrow Y \in C(S).$$

If $x_1, \ldots, x_k$ are drawn i.i.d. from S,

$$(Y(x_1), \ldots, Y(x_k)) \sim N(0, \Sigma),$$

where Σ is the covariance matrix from $(G(x_1, \xi), \ldots, G(x_k, \xi))$.

If $N^{1/2}(\hat{g}_N - g) \Rightarrow Y \in C(S)$ (with $\{\hat{g}_n\}$ and g in $C(S)$), and

$$\hat{v}_N = \min_{x \in S} \hat{g}_N(x) \text{ et } v^* = \min_{x \in S} g(x),$$

then

$$N^{1/2}(\hat{v}_N - v^*) \Rightarrow \min_{x \in S^*} Y(x).$$

Convergence of global solutions

If S^* is a singleton, under the previous assumptions, in the i.i.d. case,

$$N^{1/2}(\hat{v}_n - v^*) \Rightarrow N(0, \sigma^2(x^*)).$$

Under some additional conditions, we also have the convergence of $E[\hat{v}_N]$ to v^*.

But all these results become difficult to extend in the case of local optimization.

However, we see that the results should be better with larger N. But increasing N also increase the computation cost of the approximate function, as

$$\hat{g}_N(x) = \frac{1}{N} \sum_{i=1}^{N} G(x, \xi_i).$$

External adaptive method

What does interest us?

$$\min_{x \in S} \hat{g}_N(x) = \frac{1}{N} \sum_{i=1}^{N} G(x, \xi_i).$$

We can start with a small sample and extend if over the iterations: the adaptive sampling procedure can be external to the algorithm, or internal.

An external approach is known as retrospective approximation consists to repeatedly apply the optimization algorithm with samples of increasing sizes.

Retrospective approximation (RA)

Introduced by Healy and Schruben in 1991.

Let k be the iteration index. Components:

1. A procedure for solving a generated sample-path problem to specified tolerance vector ϵ_k, delivering a solution x_k.
2. A sequence $\{N_k\}$ of sample sizes tending to infinity.
3. A sequence $\{\epsilon_k\}$ of error-tolerances tending to zero.
4. A sequence of weights $\{w_{kj},\ j = 1, 2, \ldots, k\}$ for each iteration.

We define

$$\overline{x}_k := \sum_{j=1}^{k} w_{kj} x_j.$$

Retrospective approximation: principle

Step 0. Initialize the retrospective iteration number $k = 1$.

Step 1. Generate a sample-path problem with sample size N_k, with a “warm start”, i.e. starting from $\overline{x}_{k-1}$ to solve the generated problem with the error-tolerance ϵ_k. Denote the obtained retrospective solution by x_k.

Step 2. Compute the solution $\overline{x}_k$.

Step 3. Set $k \leftarrow k + 1$ and go to Step 1.

Retrospective approximation: assumptions

- The true function g has a unique minimizer $x^* \in \Theta$.
- $G(\cdot, \xi)$ is Lipschitz with Lipschitz constant $L(\xi)$ on Θ a.s., and $E[L(\xi)] < \infty$.
- $G(\cdot, \xi)$ is continuously differentiable at any x in a neighborhood of x^* a.s.
- $E[\|\nabla_x G(x, \xi)\|^2] < \infty$, for some $x \in \Theta$.
- The sample function $\hat{g}_N(x)$ has a unique minimum x_N^* a.s.
- When $\hat{g}_N(x)$ attains a unique minimum X_N^*, $\hat{g}_N(x)$ is twice differentiable at x_N^*. Furthermore, the $\nabla^2 \hat{g}_N(x_N^*)$ is positive definite with smallest eigenvalue uniformly bounded away from 0 a.s.
- The solution x_k obtained from the k^{th} iteration of RA satisfies $\|\nabla \hat{g}_{N_k}(x_k)\| \leq \epsilon_k$.

Retrospective approximation: assumptions

- The numerical procedure used to solve the sample-path problems in RA exhibits p^{th} order sublinear convergence or p^{th} order linear convergence with respect to the observed derivatives.
- The sample sizes are increased linearly, i.e., $N_k/N_{k-1} = c > 1$ for all k.
- The error-tolerances are chosen so that $\epsilon_k = O(1/\sqrt{N_k})$.

Retrospective approximation: convergence rate

Under the previous assumptions, the sequence of solutions obtained using the RA procedure satisfies $C_k\|x_k - x^*\|^2 = O_p(1)$ as $k \to \infty$, where C_k is the total amount of computational work done until the kth iteration and is given by $C_k = \sum_{i=1}^{k} Q_i N_i$. Here Q_i is the number of points visited by the numerical procedure during the ith iteration.

We recover the convergence rate of stochastic approximation method.

Internal-external adaptive method

The major issue with this procedure if how to quantify the word “approximative” in Step 1. If no care is taken, the resulting algorithm can in fact be more time-consuming that the direct minimization of $\hat{g}_{N_{\max}}$.

We can also replace the stopping test on $N_{\max}$ by a test of the criticality conditions of optimality.

The internal approach is a non-monotone strategy that depends on the underlying optimization methods. Here, we consider the unconstrained case.

More precisely, we generate a sample before the optimization process, with $N_{\max}$ i.i.d. random draws. At iteration k, we will use a subset of this initial sample, using N_k of the $N_{\max}$ random draws, typically the first ones.

Accuracy estimation

This implies that $\hat{g}_N$ is a smooth function, well defined for each choice of N.
In order to determine a sample size, we have to measure the approximation accuracy. Let α_δ be the quantile of a $\mathcal{N}(0, 1)$ associated to some significance level δ, i.e.
$P_{\boldsymbol{\xi}}[-\alpha_\delta \leq Y \leq \alpha_\delta] = \delta$, where $Y \sim \mathcal{N}(0, 1)$.

We will use the central limit theorem

$$g(x) - \hat{g}_N(x) \Rightarrow \mathcal{N}\left(0, \frac{\sigma^2(x)}{N}\right),$$

where $\sigma^2(x)$ is the variance of g, taken at the point x, in order to build a confidence interval for $g(x)$ around $\hat{g}_N(x)$, as

$$[\hat{g}_N(x) - \epsilon_N^\delta(x),\ \hat{g}_N(x) + \epsilon_N^\delta(x)],$$

Accuracy estimation (cont'd)

$\epsilon_N^{\delta}(x)$ is given by

$$\epsilon_{\delta}^{N}(x) = \alpha_{\delta}\frac{\sigma(x)}{\sqrt{N}}.$$

Typically, we will choose $\alpha_{0.9} \approx 1.64$ or $\alpha_{0.95} \approx 1.96$.

In practice, we do not know $\sigma^2(x)$, but we can use its estimator

$$\hat{\sigma}_N^2(x) = \frac{1}{N-1}\sum_{i=1}^{N}(G(x,\xi_i) - \hat{g}_N(x))^2.$$

We will exploit this error estimation in the context of trust-region methods.

Basic principles

The basic idea is that if the model approximates the objective function well enough with respect to the accuracy of the objective function (which depends on the sample size), we presume that we could work with a less accurate approximation, and therefore reduce the sample size.

On the other hand, if the adequation of the model with respect to the accuracy of the objective function is poor, we can increase the sample size in an attempt to correct this deficiency.

We assume the assumptions developed for the consistency analysis hold.

A formal algorithm description follows.

Algorithm BTRDA

(Basic) trust-region algorithm with dynamic accuracy.

Step 0. Initialization. initial point: x_0, initial trust-region radius: Δ_0. Set η_1 and η_2 such that $0 < \eta_1 \leq \eta_2 < 1$ (for instance, $\eta_1 = 0.01$ and $\eta_2 = 0.75$), $N_{\text{min}} = N^0_{\text{min}}$ and N_0 satisfying $\|\nabla \hat{g}_{N_0}(x_0)\| \neq 0$ if $\epsilon_\delta^{N_0}(x_{k+1}) \neq 0$, except if $N_0 = N_{\text{max}}$. Compute $\hat{g}_{N_0}(x_0)$ and set $k = 0$, $t = 0$.

Step 1. Stopping test. Stop if $\|\nabla \hat{g}_{N_k}(x_k)\| = 0$ and either $N_k = N_{\text{max}}$, either $\epsilon_\delta^{N_k}(x_k) = 0$. Otherwise, go to Step 2.

Step 2. Model definition Define a model $m_k^{N_k}$ of $\hat{g}_{N_k}(x)$ in $\mathcal{B}_k$. Compute a new adequate sample size N^+, and set $N^- = N_k$.

Algorithm BTRDA (cont'd)

Step 3. Step computation Compute a step s_k that sufficiently reduces $m_k^{N_k}$ and s.t. $x_k + s_k \in \mathcal{B}_k$. Set

$$\Delta m_k^{N_k} = m_k^{N_k}(x_k) - m_k^{N_k}(x_k + s_k).$$

Step 4. Comparaison of decreases Compute $\hat{g}_{N^+}(x_k + s_k)$ and

$$\rho_k = \frac{\hat{g}_{N_k}(x_k) - \hat{g}_{N^+}(x_k + s_k)}{\Delta m_k^{N_k}}.$$

Step 5. Sample size update If $\rho_k < \eta_1$ and $N_k \neq N^+$, modify N^- or the candidate sample size N^+ in order to take account of variance differences. Update ρ_k.

Algorithm BTRDA (cont'd)

Step 6. Candidate iterate acceptance If $\rho_k < \eta_1$, define $x_{k+1} = x_k$, $N_{k+1} = N^-$. Otherwise, define $x_{k+1} = x_k + s_k$ and set $N_{k+1} = N^+$; increment t.

If $N_{k+1} \neq N^{\max}$, $\|\nabla \hat{g}_{N_{k+1}}(x_{k+1})\| = 0$, and $\epsilon_\delta^{N_{k+1}}(x_{k+1}) \neq 0$, increase N_{k+1} to some size less or equal to $N_{\max}$ such that $\|\nabla \hat{g}_{N_{k+1}}(x_{k+1})\| \neq 0$ if $N_{k+1} \neq N_{\max}$, and compute $\hat{g}_{N_{k+1}}(x_{k+1})$.

If $N_k = N_{k+1}$ or if a sufficient decrease has been observed since the last evaluation of $\hat{g}_{N_{k+1}}$, set $N_{\min}^{k+1} = N_{\min}^{k}$. Otherwise, set $N_{\min}^{k+1} > N_{\min}^{k}$.

Algorithme: BTRDA (cont'd)

Step 7. Trust region radius update

$$\Delta_{k+1} \in \begin{cases} [\Delta_k, \infty) & \text{if } \rho_k \geq \eta_2, \\ [\gamma_2 \Delta_k, \Delta_k] & \text{if } \rho_k \in [\eta_1, \eta_2), \\ [\gamma_1 \Delta_k, \gamma_2 \Delta_k] & \text{if } \rho_k < \eta_1, \end{cases}$$

In this algorithm the variable t is used to count the number of successful iterations.

Remark also that the algorithms BTR and BTRDA coincide if we fixe N_k to $N_{\max}$ for all $k \geq 0$.

Variable sample size strategy

Before the optimization, the user chooses a maximal sample size $N_{\max}$. A minimum sample size $N_{\min}^0$ is defined in order to allow the estimation of the accuracy.

We also define $N_0 = \max\{N_{\min}^0, 0.1 N_{\max}\}$ if $\|\nabla \hat{g}_{N_0}(x_0)\| \neq 0$ and $\epsilon_\delta^{N_0}(x_0) \neq 0$, $N_0 = N_{\max}$ otherwise.

The choice of N^+ in Step 3 of the BTRDA algorithm is described below

Define constants ν_1 and χ_1 such that $\nu_1, \chi_1 \in (0, 1)$. Use $\epsilon_\delta^{N_k}(x)$ to estimate the sample size required to obtain an accuracy equal to the model decrease, i.e.

$$N^s = \max\left\{ N_{\min}^k, \left\lceil \frac{\alpha_\delta^2 \hat{\sigma}_N^2(x)}{(\Delta m_k^{N_k})^2} \right\rceil \right\}.$$

Variable sample size strategy (cont'd)

Compute the ratio between the model improvement and the estimated accuracy:

$$\tau_1^k = \frac{\Delta m_k^{N_k}}{\epsilon_\delta^{N_k}(x_k)},$$

and the ratio between the curent sample size and the sample size suggested for the next iteration:

$$\tau_2^k = \frac{N_k}{\min\{N_{\max}, N^s\}}.$$

Define

$$N' = \begin{cases} \min\left\{\lceil \chi_1 N_{\max} \rceil, \lceil N^s \rceil\right\} & \text{if } \tau_1^k \geq 1, \\ \min\left\{\lceil \chi_1 N_{\max} \rceil, \lceil \tau_1^k N^s \rceil\right\} & \text{if } \tau_1^k < 1 \text{ and } \tau_1^k \geq \tau_2^k, \\ \lceil \chi_1 N_{\max} \rceil & \text{if } \nu_1 \leq \tau_1^k < 1 \text{ and } \tau_1^k < \tau_2^k, \\ N_{\max} & \text{if } \tau_1^k < \nu_1 \text{ and } \tau_1^k < \tau_2^k. \end{cases}$$

Set $N^+ = \max\{N', N_{\min}^k\}$.

Variable sample size strategy (cont'd)

A possible value for χ_1 is 0.5.

If $\tau_1^k \geq 1$, the model decrease if greater of equal to the estimated accuracy, and we reduce the sample size to $\min\{N^s, \lceil \chi_1 N_{\max} \rceil\}$.

If $\tau_1^k < 1$, the improvement is smaller than the accuracy. However, since the sample has been generated before the optimization process, a sufficient improvement during several consecutive iterations can lead to a significant improvement in comparison to the approximation accuracy, while keeping the computational cost lower than if $N_{\max}$ draws were used.

Variable sample size strategy (cont'd)

We then consider two cases.

1. If $\tau_1^k \geq \tau_2^k$, the ratio between the current sample size and the potential next one is smaller than the ratio between the model decrease and the estimated error. If the sample size increases, the error decreases for a similar $\Delta m_j^{N_j}$ ($j \geq k$), and therefore τ_1^k increases.

We capitalize on τ_1^k by computing a sample size smaller than N^s, such that an improvement of the order $\epsilon_\delta^{N_k}(z_k)$ would be reached in approximatively $\lceil \tau_1^k \rceil$ iterations if τ_1^j is similar to τ_1^k for j close to k.

We therefore propose to use the minimum between $\lceil \chi_1 N_{\max} \rceil$ and $\lceil \tau_1^k N^s \rceil$ as a new sample size.

Variable sample size strategy (cont'd)

2. If $\tau_1^k < \tau_2^k$, it can nevertheless be cheaper to continue to work with a smaller sample size, defined again as $\lceil \chi_1 N_{\text{max}} \rceil$.

We therefore choose to use this smaller sample size as long as τ_1^k is greater to some threshold ν_1 (for instance 0.2). Below this threshold, we consider that the decrease is too small compared to the accuracy, and we possiblty increase the sample size.

Accuracy differences

If N^+ is not equal to N_k, the computation of

$$\hat{g}_{N_k}(x_k) - \hat{g}_{N^+}(x_k + s_k)$$

is affected by the change in approximation variance. This can lead to a small ratio, or even a negative ratio ρ_k, and this even if the model $m_k^{N_k}$ gives a good predicition for the sample size N^k.

In particular, $\hat{g}_{N^+}(x)$ can be greater than $\hat{g}_{N_k}(x_k)$ for all x in a neighborhood of x_k. It is therefore important to avoid such cases, motivating the new definition of ρ_k, as described hereafter.

Sample size update

Assume that $N_k \neq N^+$. If $\rho_k < \eta_1$, compare N^+ and N_k. If $N^+ > N_k$, compute $\hat{g}_{N^+}(x_k)$, $\Delta m_k^{N^+}$ and $\epsilon_\delta^{N^+}(x_k)$, otherwise if $N^+ < N_k$ compute $\hat{g}_{N_k}(x_k + s_k)$. Set N^- to $\max\{N_k, N^+\}$, and

$$\rho_k = \frac{\hat{g}_{N^-}(x_k + s_k) - \hat{g}_{N^-}(x_k)}{\Delta m_k^{N^-}}.$$

While we expect to take advantage of smaller sample sizes when we are far from the solution, we should be sure to use a sample size equal to $N_{\max}$ during the final iteration, in order to work with the desired accuracy.

To this end, we increase the minimum sample size when the adaptive strategy does not deliver sufficient numerical gains.

Minimum sample size update

We first define two vectors v and ℓ, of dimension $N_{\max}$, and, at iteration $k = 0$, set $v(N_0) = \hat{g}_{N_0}(x_0)$, $\ell(N_0) = 0$, and for $i = 1, \ldots, N_{\max}$, $i \neq N_0$, set $v(i) = +\infty$, $\ell(i) = -1$.

At the beginning of iteration k, $v(i) = \hat{g}_i(x_{h(i)})$, where $h(i)$ corresponds to the index of the last iteration for which $N_{h(i)} = i$, and $N_{h(i)-1} \neq N_{h(i)}$ if $h(i) > 0$, or $+\infty$ if the size i has not yet been used. $\ell(i)$ contains the number of successful iterations up to the iteration $h(i)$, or -1 if the size i has not yet been used.

Recall that t contains the total number of successful iterations encountered until iteration k (included).

Minimum sample size update (cont'd)

Assume that $N_k \neq N_{k+1}$. Let γ_3 be a constant in $(0, 1]$. If

$$v(N_{k+1}) - \hat{g}_{N_{k+1}}(x_{k+1}) \geq \gamma_3 \nu_1 (t - \ell(N_{k+1}))\epsilon_\delta^{N_{k+1}}(x_{k+1}),$$

set $N_{\min}^{k+1} = N_{\min}^k$. Otherwise, increase the minimum sample size: set

$$N_{\min}^{k+1} \in \{N_{k+1} + 1, \ldots, N_{\max}\}.$$

Set $\ell(N_{k+1}) = t$ and $v(N_{k+1}) = \hat{g}_{N_{k+1}}(x_{k+1})$.

A practical value for γ_3 is 0.5. Note that $N_{\min}^{k+1} > N^k$ if the above test is not satisfied.

Moreover, we have that if $N_k \neq N_{k+1}$, $t - \ell(N_{k+1}) \geq 1$. This is clearly satisfied if $\ell(N_{k+1}) = -1$, so without losss of generality, we assume that $\ell(N_{k+1}) \geq 0$. At the beginning of iteration k, we have $\ell(N_i) \leq t$, $i = 1, \ldots, N_{\max}$.

Minimum sample size update (cont'd)

If $\rho_k \geq \eta_1$, t is incremented by 1 during Step 6, so $\ell(N_{k+1}) < t$.

If $\rho_k < \eta_1$, we have $N_k < N_{k+1}$, and from the algorithm sample size update, as reducing the sample size can only happen during successful iterations, we have $\ell(N_{k+1}) < \ell(N_k) \leq t$.

Finally, note that if $N_k \neq N_{\max}$, we cannot exclude the pathological case in which z_k is a critical first-order point for $\hat{g}_{N_k}$. If $\epsilon_\delta^{N_k}(x_k) \neq 0$, the algorithm does not stop, but since the model is quadratic, no decrease is achieved if H_k is positive definite. In order to avoid this situation, we force N_{k+1} to increase when it occurs.

Additional safeguards

In practice, the gradient norm usually changes slowly in the neighborhood of such a critical point, and a small gradient typically leads to a small decrease of the model, implying the increase of the sample size, and $N_{\max}$ is reached before this safeguard is deployed.

Convergence: main ideas

Théorème

Under some regularity assumptions, if

$$\exists \kappa > 0 \text{ such that } \epsilon_{\delta}^{N_k}(x_k) \geq \kappa,$$

for all k large enough, then, almost surely, the algorithm converges in a finite number of iterations with a final sample size equal to $N_{\max}$, or the number of iterations is infinite and there exists some j such that for all the iterations i, $i \geq j$, N_i is equal to $N_{\max}$.

Proof: see Bastin, Cirillo et Toint, *An adaptive Monte Carlo algorithm for computing mixed logit estimators*, Computational Management Science 3(1), pp. 55–79, 2006.

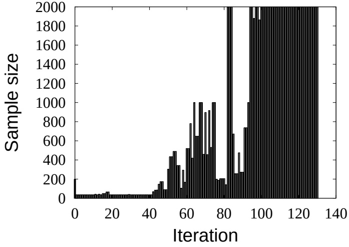
We can then prove the first-order and second-order convergence, for the SAA with $N_{\max}$ draws.

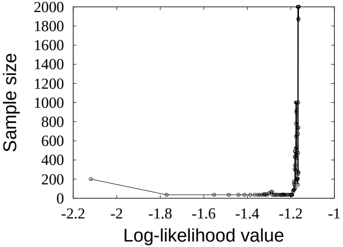
Example

Mixed logit model: stochastic maximum likelihood

$$\max_\theta LL(\theta) = \max_\theta \frac{1}{N} \sum_{n=1}^{N} \ln E[P_{ij_i}(x, \theta, \xi)].$$

Mode choice model: Mobidrive data (Axhausen et al., 2002)
$N = 5799$ observations, $R_{\max} = 2000$ draws per individual, 14 parameters (integration dimension: 3 normal variables).





↓ ↓