1. What are evolution strategies (ES)? ➔ Section II + Section III

Concepts: *Generation, parents, offspring, mutation, and fitness*

ES are a family of algorithms that mimic the natural selection procedure in biology. They can be utilized to solve optimization problems especially when the cost function is unknown, or hard to differentiate. There are a few important concepts and each of them has a correspondent concept in optimization language. Each iteration in optimization is called a generation in ES. Within each generation, the result solution from last iteration is called the parent, and some new solutions which are called offspring will be generated based on the parent. The offspring solutions are sampled from a normal distribution centered at the parent solution, which is called mutation. The cost function that we want to minimize is called fitness function in ES. Therefore, ES try to find solutions that have lower and lower fitness values as there are more and more generations.

Notations:

$x\text{: n} \times 1$	The variable that need to be optimized (parent)
$z_k\ (k = 1, \dots, \lambda)\text{: n} \times 1$	k-th offspring
$f(x)$ or $f(z)$: $\mathbb{R}^n \rightarrow\ \mathbb{R}$	Fitness function
$\Sigma = \mathrm{A^T A}$	Mutation matrix (i.e. the covariance matrix used to sample children solutions)
$\theta = (x, \Sigma)$	

The offspring z follows this distribution:

$$\pi(z|\theta) = \frac{1}{(2\pi)^{n/2}|\Sigma|^{1/2}} \exp\left[-\frac{1}{2}(z-x)^{\mathrm{T}}\Sigma^{-1}(z-x)\right] \qquad (1)$$

Other ES algorithms before this paper discard all offspring solutions except the one with smallest fitness value. But the method proposed in this paper is different. It utilizes the information from all sampled offspring. They want to minimize the expected fitness of the next generation, so that every generation's mean fitness is lower than the previous generation.

$$J(\theta) = \mathrm{E_z}[f(z)] = \int \pi(z|\theta) f(z) dz \qquad (2)$$

To optimize $J(\theta)$, we need to compute the derivative

$$\begin{aligned} \nabla_\theta J &= \nabla_\theta \int \pi(z|\theta) f(z) dz \\ &= \int \nabla_\theta \pi(z|\theta) f(z) dz \\ &= \int \frac{\pi(z|\theta)}{\pi(z|\theta)} \nabla_\theta \pi(z|\theta) f(z) dz \\ &= \int \pi(z|\theta) \nabla_\theta (\log \pi(z|\theta)) f(z) dz \\ &= \mathrm{E}[\nabla_\theta (\log \pi(z|\theta)) f(z)] \end{aligned}$$

$$\approx \frac{1}{\lambda}\sum_{k=1}^{\lambda} \nabla_{\theta}(\log \pi(z_k|\theta))f(z_k) \qquad (3)$$

Commented [LJ1]: Monte Carlo approximation

Expansion of $\log \pi(z_k|\theta)$

$$\log \pi(z_k|\theta) = \frac{n}{2}\log 2\pi - \frac{1}{2}\log|\Sigma| - \frac{1}{2}(z-x)^{\mathrm{T}}\Sigma^{-1}(z-x) \qquad (4)$$

Take the derivative w.r.t x and Σ

$$\nabla_x \log \pi(z_k|\theta) = \Sigma(z-x) \qquad (5)$$

$$\nabla_{\Sigma} \log \pi(z|\theta) = \frac{1}{2}\Sigma^{-1}(z-x)(z-x)^{\mathrm{T}}\Sigma^{-1} - \frac{1}{2}\Sigma^{-1} \qquad (6)$$

The covariance matrix is positive semi-definite, hence it can be decomposed: $\Sigma = \mathrm{A}^{\mathrm{T}}\mathrm{A}$. We compute the derivative w.r.t. A (proof is in my hand-written derivation).

$$\nabla_{\mathrm{A}} \log \pi(z|\theta) = \mathrm{A}[\nabla_{\Sigma} \log \pi(z|\theta) + (\nabla_{\Sigma} \log \pi(z|\theta))^{\mathrm{T}}] \qquad (7)$$

2. What is natural gradient, and why to use it? ➔ Section V + Algorithm

---------

In standard gradient descent, the derivatives tell us to which direction that the parameters should be move. Then we update the parameters along that direction by a small distance. The main problem is how to define *a small distance*. In standard gradient descent, the distance is defined by Euclidean distance. However, it has a problem. Let's imagine two pairs of Gaussian distributions: (1) $N(1, 1000)$ and $N(2, 1000)$; (2) $N(1, 1)$ and $N(2, 1)$. In both pairs, the distribution mean is shifted by 1. However, the effect is much larger on the second pair than the first pair. In this case, the Euclidean distance between the parameters of the two distributions within each pair is the same, but has very different influence. We need to find a better way to define the distance between distributions.

-----------

A natural way to define the distance between distributions is to use KL-divergence. Then the natural gradient can be formalized as the solution of the following problem:

$$\begin{cases} \max\limits_{\delta\theta} J(\theta + \delta\theta) - J(\theta) = \delta\theta^{\mathrm{T}}\nabla_{\theta} J \\ \qquad s.t. D(\theta + \delta\theta || \theta) = \epsilon \end{cases} \qquad (8)$$

Where $\delta\theta$ is the natural gradient. The intuitive meaning of this problem is that: there is a neighborhood set of θ' around θ; the KL-divergence between all of the θ' and θ is a small constant. Now we want to find within all θ', which one makes J change most.

When $\delta\theta$ is very small, KL divergence can be approximated as

$$D(\theta + \delta\theta || \theta) \approx \delta\theta^{\mathrm{T}} F(\theta)\delta\theta + \text{const.}$$

where $F(\theta)$ is the Fisher information matrix

$$F(\theta) = \int \pi(z|\theta)\nabla_{\theta}(\log \pi(z|\theta))\nabla_{\theta}(\log \pi(z|\theta))^{\mathrm{T}} dz$$
$$= E[\nabla_{\theta}(\log \pi(z|\theta))\nabla_{\theta}(\log \pi(z|\theta))^{\mathrm{T}}] \qquad (9)$$

A not very rigorous proof is in my hand-written derivation.

Then if we solve the problem (8) by introducing a Lagrangian multiplier α, we can get

$$F(\theta)\delta\theta = \frac{1}{\alpha}\nabla_{\theta} J \qquad (10)$$

----------

From Eq. (3) we can know that the smaller $\mathrm{Var}[\nabla_\theta(\log\pi(z|\theta))f(z)]$ is, we can use fewer samples of z_k to accurately estimate $\nabla_\theta J$, hence estimate the natural gradient. To reduce the variance, the authors introduced a fitness baseline

$$\begin{aligned}
\nabla_\theta J &= \int \nabla_\theta\pi(z|\theta)f(z)dz + 0 \\
&= \int \nabla_\theta\pi(z|\theta)f(z)dz + b\nabla_\theta \int \pi(z|\theta)dz \\
&= \int \nabla_\theta\pi(z|\theta)f(z)dz + \int b\nabla_\theta\pi(z|\theta)dz \\
&= \int \nabla_\theta\pi(z|\theta)(f(z)-b)dz \\
&= \int \pi(z|\,\theta)\nabla_\theta\log\pi(z|\theta)\,(f(z)-b)dz \qquad (11)
\end{aligned}$$

Commented [LJ2]: Derivative of 1 is 0

$$\begin{aligned}
&\mathrm{Var}[\nabla_\theta\log\pi(z|\theta)\,(f(z)-b)] = \mathrm{Var}\left[\phi(z)\left(f(z)-\bar{f}\right)\right] + \mathrm{Var}\left[\phi(z)\left(\bar{f}-b\right)\right] \\
&= \mathrm{E}\left[\left\{\phi(z)\left(f(z)-\bar{f}\right)\right\}\left\{\phi(z)\left(f(z)-\bar{f}\right)\right\}^{\mathrm{T}}\right] - \mathrm{E}\left[\phi(z)\left(f(z)-\bar{f}\right)\right]\mathrm{E}\left[\phi(z)\left(f(z)-\bar{f}\right)\right]^{\mathrm{T}} \\
&\qquad\qquad + \mathrm{Var}\left[\phi(z)\left(\bar{f}-b\right)\right] \\
&\le \mathrm{E}\left[\phi(z)\phi(z)^{\mathrm{T}}\left(f(z)-\bar{f}\right)^2\right] + \mathrm{Var}\left[\phi(z)\left(\bar{f}-b\right)\right] \\
&\le \mathrm{E}[\phi(z)\phi(z)^{\mathrm{T}}f(z)^2] + \bar{f}^2\mathrm{E}[\phi(z)\phi(z)^{\mathrm{T}}] - 2\mathrm{E}\left[\phi(z)\phi(z)^{\mathrm{T}}f(z)\bar{f}\right] + \mathrm{Var}\left[\phi(z)\left(\bar{f}-b\right)\right] \quad (12)
\end{aligned}$$

Commented [LJ3]: $\nabla_\theta\log\pi(z|\theta) =: \phi(z)$

The last term $\mathrm{Var}\left[\phi(z)\left(\bar{f}-b\right)\right] = 0$, iff $b = \bar{f}$.

Plug Eq. (9) and (11) into Eq. (10)

$$\mathrm{E}[\nabla_\theta(\log\pi(z|\theta))\nabla_\theta(\log\pi(z|\theta))^{\mathrm{T}}]\alpha\delta\theta = \mathrm{E}[\nabla_\theta(\log\pi(z|\theta))f(z)] - \mathrm{E}[\nabla_\theta(\log\pi(z|\theta))b] \quad (13)$$

If we take the expectation of Eq. (5) & (6), we can see that $\mathrm{E}[\phi(z)] = 0$.

Therefore

$$\mathrm{E}[\phi(z)]^{\mathrm{T}}\alpha\delta\theta + b = \mathrm{E}[f(z)] \qquad (14)$$

Combine Eq. (11) and (14), we have a linear system

$$\begin{cases} \mathrm{E}[\phi(z)\phi(z)^{\mathrm{T}}]\alpha\delta\theta + b\mathrm{E}[\phi(z)] = \mathrm{E}[\phi(z)f(z)] \\ \qquad\quad \mathrm{E}[\phi(z)]^{\mathrm{T}}\alpha\delta\theta + b = \mathrm{E}[f(z)] \end{cases}$$

Solve this linear system

$$\alpha\delta\theta = \left(\Phi^{\mathrm{T}}\Phi\right)^{-1}\Phi^{\mathrm{T}}\mathrm{R}$$

where

$$\Phi = \begin{bmatrix} \nabla_\theta\log\pi(z_1|\theta) \\ \vdots \\ \nabla_\theta\log\pi(z_\lambda|\theta) \end{bmatrix}$$

$$\mathrm{R} = \begin{bmatrix} f(z_1) \\ \vdots \\ f(z_\lambda) \end{bmatrix}$$

Here I have different results from the paper. I don't know how the vector of 1s is generated. Maybe the authors manually introduce this regressor to demean. Now we get the algorithm shown in the paper. We can see that the whole algorithm does not need to compute the derivatives of fitness function.

3. Experiment (show figures of fitness functions)

In their experiments, they evaluated the fitness by the functions shown in Table 1. They chose 9 functions with only one optimal value, and 4 functions with multiple optimal values.
For the unimodal fitness functions, they used two setups of the dimension of x and z. One is 5; one is 15. When the dimension is 5, the number of offspring in each generation is 50. When the dimension is 15, the number of offspring is set to be 250. This setting is to make sure that the least square approach is not ill-posed.
Figure 2 shows how the cost function decreased along the iterations. The left one is with dimension 5; the right one is with dimension 15. Each curve is one type of fitness function. When the dimension is 5, Rosenbrock function is the hardest to optimize. When dimension is 15, Cigar function is the hardest to optimize.

For the multimodal functions, they compare their algorithm with covariance matrix adaption (CMA) algorithm. The drawback of CMA is that it is sensitive to local optima, and it has many magically set parameters, and the settings of those parameters do not have any theoretical reason. Table 2 shows the percentage of runs where the two algorithms found the global optimum. For each fitness function, they have 3 different settings of the distance between initial guess and the global optimum: 1, 10, and 100. As you can expect, the larger the distance is, the harder to get the true global optimum, i.e. the accuracy becomes lower. In most cases, for example, all settings of Rastrigin function, the proposed algorithm has higher accuracy than CMA. The authors want to show that their algorithm is less sensitive to local optima compared to CMA.

☆ Proof of " $\nabla_A \log \pi(z_k) = A\left[\nabla_\Sigma \log \pi(z_k) + \nabla_\Sigma \log \pi(z_k)^T\right]$

First, by definition: $\nabla_A \log(\pi(z_k)) = \begin{bmatrix} \frac{\partial \log(\cdot)}{\partial a_{11}} & \cdots & \frac{\partial \log(\cdot)}{\partial a_{1n}} \\ \vdots & & \vdots \\ \frac{\partial \log(\cdot)}{\partial a_{n1}} & \cdots & \frac{\partial \log(\cdot)}{\partial a_{nn}} \end{bmatrix}$

where $a_{ij} \triangleq (A)_{ij}$, $\log(\cdot)$ represents $\log \pi(z_k)$

Therefore, the goal is to prove:

$$\frac{\partial \log(\cdot)}{\partial a_{ij}} = \left(A\left[\nabla_\Sigma \log \pi(z_k) + \cancel{\Sigma}\, \nabla_\Sigma \log \pi(z_k)^T\right]\right)_{ij}$$

$$= \left(A\, \nabla_\Sigma \log \pi(z_k)\right)_{ij} + \left(A \nabla_\Sigma \log \pi(z_k)^T\right)_{ij} \quad (1)$$

Chain rule: $$\frac{\partial \log \pi(z_k)}{\partial a_{ij}} = \sum_{p=1}^{n} \sum_{q=1}^{n} \frac{\partial \log \pi(z_k)}{\partial \sigma_{pq}} \cdot \frac{\partial \sigma_{pq}}{\partial a_{ij}} \quad (\sigma_{pq} \triangleq (\Sigma)_{pq}) \rightarrow (2)$$

Now let's compute $\frac{\partial \sigma_{pq}}{\partial a_{ij}}$.

We know that $\sigma_{pq} = \sum_{s=1}^{n} a_{sk}\, a_{sl}$, because $\sigma_{pq} = \vec{a}_p^{\,T} \cdot \vec{a}_q$ $\left(\vec{a}_p \triangleq \begin{bmatrix} a_{1p} \\ \vdots \\ a_{np} \end{bmatrix} \right)$

$$\Rightarrow \frac{\partial \sigma_{pq}}{\partial a_{ij}} = \begin{cases} a_{iq}, & p = j \\ a_{ip}, & q = j \\ 0, & \text{others} \end{cases} \quad (3)$$

Plug into (2),

$$\frac{\partial \log \pi(z_k)}{\partial a_{ij}} = \sum_{p=1}^{n} \frac{\partial \log \pi(z_k)}{\partial \sigma_{pj}} \cdot a_{ip} + \sum_{q=1}^{n} \frac{\partial \log \pi(z_k)}{\partial \sigma_{jq}} \cdot a_{iq} \quad (4)$$

Now let's look at the r.h.s of the original equation (1)

$$A\, \nabla_\Sigma \log \pi(z_k) = \begin{bmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & & \vdots \\ a_{n1} & \cdots & a_{nn} \end{bmatrix} \begin{bmatrix} \frac{\partial \log(\cdot)}{\partial \sigma_{11}} & \cdots & \frac{\partial \log(\cdot)}{\partial \sigma_{1n}} \\ \vdots & & \\ \frac{\partial \log(\cdot)}{\partial \sigma_{n1}} & \cdots & \frac{\partial \log(\cdot)}{\partial \sigma_{nn}} \end{bmatrix}$$

$$\Rightarrow (A\, \nabla_\Sigma \log \pi(z_k))_{ij} = \sum_{p=1}^{n} a_{ip} \frac{\partial \log(\pi(z_k))}{\partial \sigma_{pj}} \quad (5)$$

similarly, $(A\, \nabla_\Sigma \log \pi(z_k)^T)_{ij} = \sum_{p=1}^{n} a_{ip} \frac{\partial \log(\pi(z_k))}{\partial \sigma_{jp}}$ (6)

Compare (4) and (5) + (6), equation (1) is proved. #

✯ Proof of $D_{KL}(\theta+\delta\theta\|\theta) \approx \delta\theta^T F(\theta)\delta\theta + \text{const.}$

(where $F(\theta) = \int \pi(z|\theta) \nabla_\theta (\log \pi(z|\theta)) \nabla_\theta (\log \pi(z|\theta))^T dz.$)

$$D_{KL}(\theta+\delta\theta\|\theta) = \int \pi(z|\theta+\delta\theta) \log \frac{\pi(z|\theta+\delta\theta)}{\pi(z|\theta)} dz$$

Let's say $\delta \overset{\pi}{\cancel{p}}(z|\theta) = \pi(z|\theta+\delta\theta) - \pi(z|\theta)$

$$= \int \pi(z|\theta)\left(1+\frac{\delta\pi(z|\theta)}{\pi(z|\theta)}\right) \log\left(1+\frac{\delta\pi(z|\theta)}{\pi(z|\theta)}\right) dz$$

Taylor expansion $\log(1+x) = x + o(x^2)$

$$\approx \int \pi(z|\theta)\left(1+\frac{\delta\pi(z|\theta)}{\pi(z|\theta)}\right)\left(\frac{\delta\pi(z|\theta)}{\pi(z|\theta)}\right) dz$$

$$= \int \delta\pi(z|\theta) dz + \int \pi(z|\theta)\left(\frac{\delta\pi(z|\theta)}{\pi(z|\theta)}\right)^2 dz$$

$$\delta\pi(z|\theta) = \sum_{i=1}^{p} \frac{\partial \pi(z|\theta)}{\partial\theta^i}(\delta\theta)^i \Rightarrow \int \delta\pi(z|\theta) dz = \int \sum_{i=1}^{p} \frac{\partial\pi(z|\theta)}{\partial\theta^i}(\delta\theta)^i dz$$

θ has p dimensions

$$= \sum_{i=1}^{p} (\delta\theta)^i \int \frac{\partial\pi(z|\theta)}{\partial\theta^i} dz$$

$$= \sum_{i=1}^{p} (\delta\theta)^i \frac{\partial \int \pi(z|\theta) dz}{\partial\theta^i}$$

$$= \sum_{i=1}^{p} (\delta\theta)^i \cdot 0 = 0.$$

$$D_{KL}(\theta+\delta\theta\|\theta) \approx \int \pi(z|\theta) \left(\frac{\sum_{i=1}^{p} \frac{\partial\pi(z|\theta)}{\partial\theta^i}(\delta\theta)^i}{\pi(z|\theta)}\right)^2 dz.$$

$$= \int \pi(z|\theta) \left(\sum_{i=1}^{p} \frac{\partial \log(\pi(z|\theta))}{\partial\theta^i}(\delta\theta)^i\right)^2 dz$$

$$= \int \pi(z|\theta) \left(\sum_{i=1}^{p} \frac{\partial\log(\pi(z|\theta))}{\partial\theta^i}(\delta\theta)^i\right)\left(\sum_{j=1}^{p} \frac{\partial\log(\pi(z|\theta))}{\partial\theta^j}(\delta\theta)^j\right) dz$$

$$= \int \pi(z|\theta) \sum_{i=1}^{p}\sum_{j=1}^{p} \left(\frac{\partial\log(\pi(z|\theta))}{\partial\theta^i}(\delta\theta)^i \frac{\partial\log(\pi(z|\theta))}{\partial\theta^j}(\delta\theta)^j\right) dz$$

$$= \sum_{i=1}^{p}\sum_{j=1}^{p} \left(\int \pi(z|\theta) \frac{\partial\log(\pi(z|\theta))}{\partial\theta^i} \frac{\partial\log(\pi(z|\theta))}{\partial\theta^j} dz\right)(\delta\theta)^i(\delta\theta)^j$$

$$= \sum_{i=1}^{p}\sum_{j=1}^{p} F(\theta)_{ij} \cdot \delta\theta^i \delta\theta^j$$

$$= \delta\theta^T F(\theta) \delta\theta$$