Monte-Carlo Path Weighing

Raphael Douady

January 30, 2002

1. Calibration of Monte-Carlo Models

Monte-Carlo models can be long to calibrate to market data. Let $b_1, ..., b_m$ be a series of assets we wish to calibrate the model to. For instance, in an interest rate model, these may contain:

- Discount factors (in order to match the curve itself),
- Caplets,
- European swaptions.

Let N be the number of paths. Each asset b_i is priced by the Monte-Carlo model by averaging pay-off over paths:

$$b_i \simeq \frac{1}{N} \sum_{k=1}^{N} a_{ik}$$

where a_{ik} is the pay-off of asset b_i in path k . However, these m equations are not satisfied with accuracy. it is possible to get an exact match by weighing the paths. Let w_k be the weight of path k . One needs to solve:

$$\begin{aligned} \forall i\,, \quad b_i \; &= \; \sum_{k=1}^{N} w_k \, a_{ik} \\ \forall k\;, \quad w_k \; &\geq \; 0\;, \quad \sum_{k=1}^{N} w_k = 1 \end{aligned}$$

Without the positivity constraint, the linear problem is easy to solve provided the number of paths N exceeds that of assets $m + 1$ for the normalisation (under non degeneracy assumption).

2. Quadratic Criterion

The usual way to remain in the class of linear problems is to impose that some quadratic criterion of the weights be minimised. This doesn't insure positivity of the weights but, if the prior $\frac{1}{N}$ is already close enough, the solution will be close to $\frac{1}{N}$, hence likely to be positive. Consider for instance the following problem:

$$\left\{ \begin{array}{l} \forall i\,,\quad b_i = \displaystyle\sum_{k=1}^{N} w_k\, a_{ik}\,, \qquad \sum_{k=1}^{N} w_k = 1 \\[2ex] \displaystyle\sum_{k=1}^{N} \mu_k\, (w_k - \bar{w}_k)^2 \quad \text{minimal} \end{array} \right. \tag{2.1}$$

Let us consider the normalisation as an extra asset and set $b_0 = 1$, $a_{0k} = 1$. Coefficients μ_k are "penalising coefficients" whereas $\bar{w}_k$ are priors for the w_k. One may start with $\mu_k = 1$ and $\bar{w}_k = \frac{1}{N}$. The solution to this problem is given by linear algebra. Let

$$\bar{b}_i = \sum_{k=1}^{N} \bar{w}_k\, a_{ik}$$

and define:

$$B = \begin{pmatrix} b_0 \\ \vdots \\ b_m \end{pmatrix} \qquad \bar{B} = \begin{pmatrix} \bar{b}_0 \\ \vdots \\ \bar{b}_m \end{pmatrix} \qquad A = \begin{pmatrix} a_{01} & \cdots & a_{0N} \\ \vdots & & \vdots \\ a_{m1} & \cdots & a_{mN} \end{pmatrix}$$

$$A^* = \begin{pmatrix} a_{01} & \cdots & a_{m1} \\ \vdots & & \vdots \\ a_{0N} & \cdots & a_{mN} \end{pmatrix} \qquad M = \begin{pmatrix} \mu_1 & & 0 \\ & \ddots & \\ 0 & & \mu_N \end{pmatrix}$$

$$W = \begin{pmatrix} w_1 \\ \vdots \\ w_N \end{pmatrix} \qquad \bar{W} = \begin{pmatrix} \bar{w}_1 \\ \vdots \\ \bar{w}_N \end{pmatrix} \qquad X = \begin{pmatrix} w_1 - \bar{w}_1 \\ \vdots \\ w_N - \bar{w}_N \end{pmatrix}$$

The problem can be stated as:

$$\left\{ \begin{array}{l} AX = B - \bar{B} \\[2ex] X.MX \quad \text{minimal} \end{array} \right.$$

If X is a solution, then the gradient of the criterion is a combination of the constraints. Its coefficients are the Lagrange multipliers $\lambda_1, ..., \lambda_m$. Let us set:

$$\Lambda = \begin{pmatrix} \lambda_1 \\ \vdots \\ \lambda_m \end{pmatrix}$$

One has:

$$\begin{aligned} \Lambda &= (AM^{-1}A^*)^{-1}(B - \bar{B}) \\ X &= M^{-1}A^*\Lambda \end{aligned}$$

Remark 1. *The matrix to invert $AM^{-1}A^*$ is $(m+1) \times (m+1)$ and not $N \times N$.*

Remark 2. *The solution X does not change if all weights μ_k are multiplied by the same scaling factor.*

A possible method to get positive weights is as follows:

1. Start from the prior $\bar{w}_k = \frac{1}{N}$ and $\mu_k = 1$.
2. Detect indices k such that the solution $w_k < 0$.
3. For these indices, multiply μ_k by a large factor, for instance 10. Keep prior $\bar{w}_k = \frac{1}{N}$.
4. Repeat operations 2 and 3 until all weights are positive.

This method is not ensured to converge. The next section describes a more secured algorithm.

3. Entropy Minimisation

This technique comes from statistical physics and has been brought to finance by M. Avellaneda. It consists of replacing the linear problem of minimising a quadratic criterion by a non linear one. Let $\phi : [0,1] \to \mathbb{R}$ be a strongly convex function, i.e. $\phi'' > 0$, with a minimum in the interior of the interval. A standard example of function ϕ is the Boltzman-Gibbs entropy:

$$\phi(w) = w \log w$$

The non linear problem is to find a vector of weights W such that:

$$\begin{cases} AW = B \\ \\ w_k \in [0,1] \quad \forall k \\ \\ \Phi(W) = \sum_{k=1}^{N} \phi(w_k) \quad \text{minimal} \end{cases} \tag{3.1}$$

It is obvious that the function Φ is convex on $[0,1]^N$, hence also on the subspace of solutions to the linear equation $AW = B$.

Remark 3. *Constraints can, and should, be weighted. It suffices to multiply lines of the matrix A and the corresponding co-ordinate of B by the importance one wishes to attach to the constraint. For instance, prices of discount factors should have a strong weight, inversely proportional to the square of maturity in order to match evenly yields, and weights of option prices should be inversely proportional to the square of the Vega, in order to match evenly volatilities.*

Here are two approaches to find the minimum of Φ.

3.1. Legendre Transform

For every vector $\Lambda = (\lambda_0, ..., \lambda_m) \in \mathbb{R}^{m+1}$ of Lagrange multipliers, let:

$$\begin{aligned} \Phi_\Lambda(W) &= \Lambda.(AW - B) - \Phi(W) \\ \\ \Psi(\Lambda) &= \max_{W \in [0,1]^N} \Phi_\Lambda(W) \end{aligned}$$

The function Ψ is called the *Legendre transform* of Φ. As a maximum of linear functions of Λ, it is convex. It is easy to see that it tends to $+\infty$ when $\|\Lambda\|$ does, so it has a minimum $\tilde{\Lambda}$. The solution to our problem is the vector of weights $\tilde{W}$

which minimises the function $\Phi_{\tilde{\Lambda}}$ over the whole hypercube $[0,1]^N$, regardless of the constraints.

Each of the steps of this approach looks unfeasible. In fact almost everything is explicitly computable, thanks to the special shape of Φ. Define $\psi : \mathbb{R} \to \mathbb{R}$ to be the Legendre transform of ϕ :

$$\psi(\lambda) = \max_{w \in [0,1]} (\lambda w - \phi(w)) = \lambda u(\lambda) - \phi \circ u(\lambda)$$

where u is characterised by:

$$u(\lambda) = w \iff \begin{cases} \lambda = \phi'(w) & \text{if } \ \phi'(0) < \lambda < \phi'(1) \\ w = 0 & \text{if } \ \lambda \leq \phi'(0) \\ w = 1 & \text{if } \ \lambda \geq \phi'(1) \end{cases}$$

In the case of the Boltzman-Gibbs entropy $\phi(w) = w \log w$, functions u and ψ are given by:

$$u(\lambda) = \min(e^{\lambda-1}, 1) \qquad \psi(\lambda) = \min(e^{\lambda-1}, \lambda)$$

The function Ψ is given by:

$$\Psi(\Lambda) = \sum_{k=1}^{N} \psi\left(\sum_{i=0}^{m} \lambda_i a_{ik}\right)$$

and the vector $W(\Lambda)$ at which Φ_Λ is maximum is given by:

$$w_k = u\left(\sum_{i=0}^{m} \lambda_i a_{ik}\right)$$

One then uses numerical methods to minimise Ψ and applies the above equation to get the weights w_k . These weights are automatically in the range $[0,1]$, which is that of u. The smaller the gradient of Ψ, the more accurate constraints are satisfied, i.e. calibration is efficient.

Remark 4. *Unequal importance can be attached to paths by introducing weights in the definition of* Φ :

$$\Phi(W) = \sum_{k=1}^{N} \mu_k \phi(w_k) \Longrightarrow \Psi(\Lambda) = \sum_{k=1}^{N} \mu_k \psi\left(\frac{1}{\mu_k} \sum_{i=0}^{m} \lambda_i a_{ik}\right)$$

This allows to adjust the weights of an already weighted Monte-Carlo scheme.

3.2. Iterated Quadratic Method

This algorithm is nothing but the Newton method applied to the minimisation of Φ in the subspace of vectors W satisfying the constraint $AW = B$. Convergence is ensured by the strong convexity of Φ. Let $W = (w_1, ..., w_N) \in [0,1]^N$ and perform a second order Taylor expansion of Φ near W :

$$\Phi(W + X) = \sum_{k=1}^{N} \phi(w_k) + \phi'(w_k)\, x_\kappa + \frac{1}{2}\, \phi''(w_k)\, x_k^2 + o(x_k^2)$$

Ignoring the remainders $o(x_k)$, one finds the vector $X = (x_1, ..., x_N)$ that minimises $\Phi(W + X)$ up to the second order by solving a quadratic problem of the type mentioned in sect.2, where weights μ_k and priors $\bar{w}_k$ are given by:

$$\mu_k = \phi''(w_k) \qquad \bar{w}_k = w_k - \frac{\phi'(w_k)}{\phi''(w_k)} \tag{3.2}$$

The algorithm goes as follows:

1. Define $\bar{W}^{(0)} = \left(\frac{1}{N}, ..., \frac{1}{N}\right)$ and $\mu^{(0)} = (1, ..., 1)$.
2. Solve the problem (2.1). Let $W^{(1)}$ be the solution, which satisfies $AW^{(1)} = B$.
3. Compute $\mu^{(1)}$ and $\bar{W}^{(1)}$ from $W^{(1)}$ by equation (3.2) above.
4. Repeat operations 2 and 3 then define sequences $W^{(n)}$, $\bar{W}^{(n)}$ and $\mu^{(n)}$ by induction. One always has:
   $$AW^{(n)} = B$$
5. Stop when $\left\|W^{(n)} - \bar{W}^{(n-1)}\right\|$ is sufficiently small.

The only difficulty is that there is no reason why the solution of step 2 would satisfy $w_k^{(n)} \in [0,1]$. Outside this interval, ϕ may not be defined, thus $\mu_k^{(n)}$ and $\bar{w}_k^{(n)}$ may not be computable. Rather than imposing weights in the prescribed interval, which may end into constraint violation, one will extend ϕ to the whole real axis. We now must ensure that the solution has positive weights. Let $\phi^{(n)}$ be a sequence of extensions of ϕ to $\mathbb{R}$ such that:

- $\forall x \in \mathbb{R}, \quad \phi^{(n)}(x) \leq \phi^{(n+1)}(x)$,

- $\forall x \in [\varepsilon_n, 1-\varepsilon_n]\,, \quad \phi^{(n)}(x) = \phi(x)$ and $\varepsilon_n \to 0\,,$
- $\phi^{(n)}(0) \to \phi(0)\,, \quad \phi^{(n)}(1) \to \phi(1)\,,$
- $\forall n\,, \quad \phi^{(n)}$ is of class C^2 and $\phi^{(n)\prime\prime} > 0\,,$
- $\forall x < 0\,, \quad \phi^{(n)}(x) \to +\infty\,.$

Under these conditions, it is possible to show that the modified sequence $\tilde{W}^{(n)}$ which solves, at each step n, the problem (2.1), where ϕ is replaced by $\phi^{(n)}$, converges to a vector W which satisfies all the constraints and solves the entropy problem (3.1).

Remark 5. *As above, weights can easily be assigned to constraints and/or to paths.*

3.3. Examples of Entropies

3.3.1. Boltzman-Gibbs

The standard Boltzman-Gibbs entropy has been introduced:

$$\phi(w) = w \log w \Longrightarrow \psi(\lambda) = \min(e^{\lambda - 1}, 1)$$

In this case, $\phi(w)$ is well defined for $w > 1$ but not for $w < 0$. Here is a suggestion for extensions $\phi^{(n)}$. Let:

- $r_n \to +\infty\,,\ r_n > 0$
- $\varepsilon_n \to 0\,,\ 0 < \varepsilon_n < e^{-1} = u(0)$
- α_n and β_n defined by:

$$\alpha_n \exp(-r_n \varepsilon_n) + \beta_n = \phi(\varepsilon_n) \qquad\qquad -\alpha_n r_n \exp(-r_n \varepsilon_n) = \phi'(\varepsilon_n)$$

The function $\phi^{(n)}$ is defined by:

$$\phi^{(n)}(w) = \begin{cases} \phi(w) & \text{if } \ w \geq \varepsilon_n \\ \\ \alpha_n \exp(-r_n w) + \beta_n & \text{if } \ w \leq \varepsilon_n \end{cases}$$

3.3.2. Boltzman-Gibbs + Penalisation

One of the drawbacks of this entropy is that it does not prevent weights from being high, for they are not penalised enough. This induces big losses in accuracy. In practice, weights should never exceed a few times $\frac{1}{N}$, otherwise the order of magnitude of price standard deviations is not anymore $\frac{1}{\sqrt{N}}$. To overcome this difficulty, one may include a penalising term in Boltzman-Gibbs entropy:

$$\phi_a(w) = w \log w + a \exp(Nw)$$

The penalisation parameter a must be of course positive, for instance, $a = 1$. The sequence $\phi_a^{(n)}$ can be defined as above. However, the sequence ε_n must satisfy $\varepsilon_n < u(0)$, i.e. $\phi_a'(\varepsilon_n) < 0$, that is:

$$1 + \log \varepsilon_n + aN \exp(N\varepsilon_n) < 0$$

The order of magnitude of N is at least several thousands, so, if $a \sim 1$, one must have $\varepsilon_n < \exp(-1 - aN)$, which is a numerical 0, as well as $N\varepsilon_n$. In practice, one can set:

$$\begin{aligned} \varepsilon_n &= \exp(-n(1 + aN)) \simeq 0 \\ \phi_a(\varepsilon_n) &\simeq a \qquad \phi_a'(\varepsilon_n) \simeq -(n-1)(1+aN) \end{aligned}$$

and, if we set $r_n = nN$:

$$\alpha_n = \frac{(n-1)(1+aN)}{nN} \qquad \beta_n = a - \alpha_n$$

3.3.3. Hellinger Distance

Hellinger distance between two equivalent probabilities P and Q is defined as follows. Let $h = dQ/dP$ and define the "correlation" of P and Q by:

$$\mathrm{Corr}(P,Q) = \int \sqrt{h} dP = \int \sqrt{\frac{1}{h}} dQ = \int \sqrt{dPdQ}$$

The last expression shows the symmetry in P and Q. It is easy to prove that $0 < \mathrm{Corr}(P,Q) \leq 1$ with equality only if $P = Q$. The Hellinger distance between P and Q is defined by:

$$\delta(P,Q) = \sqrt{1 - \mathrm{Corr}(P,Q)}$$

Minimising the Hellinger distance between probabilities is the same as maximising their correlation. In the present setting, the problem is to find a vector of weights $W = (w_1, ..., w_N)$ that matches the market price of calibrating assets, with a minimum Hellinger distance to the original weights $\left(\frac{1}{N}, ..., \frac{1}{N}\right)$, which is equivalent to a maximum correlation with original weights. Therefore, this problem is of the "entropy" type with:

$$\phi(w) = -\sqrt{w} \quad , \quad w \in [0,1]$$

In this case, one has:

$$u(\lambda) = \min\left(\frac{1}{4\lambda^2}\ , \ 1\right) \qquad \psi(\lambda) = \begin{cases} -\dfrac{1}{4\lambda} & \text{if } \lambda \le -\frac{1}{2} \\ \lambda + 1 & \text{if } \lambda \ge -\frac{1}{2} \end{cases}$$

The difference between Boltzman-Gibbs entropy and Hellinger distance minimisation relies on the behaviour of ϕ near 0. The square root "falls" down much steeper than the entropy, hence weights will stay away from 0. This can also be seen from the shape of function ψ : for the same order of magnitude of λ, the entropy applies an exponential function while the square root only tales the inverse.

3.4. Overflow Control

In order to avoid overflows in the computation of weights, remember that the quadratic problem is invariant under a scaling of coefficients μ_k. One can replace $\mu_k^{(n)} = \phi^{(n)\prime\prime}(w_k^{(n)})$ by any $\tilde{\mu}_k^{(n)} = \gamma^{(n)}\phi^{(n)\prime\prime}(w_k^{(n)})$. In the case of the entropy ϕ_a above, one has:

$$\phi_a''(w) = \frac{1}{w} + aN^2 \exp(Nw)$$

Let $w_{\max}^{(n)} = \max\limits_k w_k^{(n)}$ and $w_{\min}^{(n)} = \min\limits_k w_k^{(n)}$ (possibly negative). The weights $\mu_k^{(n)}$ should be replaced by:

$$\tilde{\mu}_k^{(n)} = \frac{\phi_a^{(n)\prime\prime}(w_k^{(n)})}{\max\left(\phi_a^{(n)\prime\prime}(w_{\min}^{(n)}), \phi_a^{(n)\prime\prime}(w_{\max}^{(n)})\right)}$$

Similarly, the ratio $\phi_a^{(n)\prime}(w_k^{(n)}) \,/\, \phi_a^{(n)\prime\prime}(w_k^{(n)})$ must be computed in a secure way. We here take the example of the penalised Boltzman-Gibbs entropy. If $w > \varepsilon_n$ then:

$$\frac{\phi_a^{(n)\prime}(w)}{\phi_a^{(n)\prime\prime}(w)} = \frac{aNw + (w + w\log w)\exp(-Nw)}{aN^2 w + \exp(-Nw)}$$

(the denominator of the expression is never small) and, if $w < \varepsilon_n$:

$$\frac{\phi_a^{(n)\prime}(w)}{\phi_a^{(n)\prime\prime}(w)} = -\frac{1}{nN}$$