

[Source](#) | [Model](#) | [Option](#)  
[Model\\_Option](#) | [Help on mc methods](#) | [Archived Tests](#)

## mc\_merton

### Input parameters:

- Number of iterations  $N$
- Generator\_Type
- Confidence Value

### Output parameters:

- Price  $P$
- Error Price  $\sigma_P$
- Delta  $\delta$
- Error delta  $\sigma_\delta$
- Price Confidence Interval:  $IC_P = [\text{Inf Price}, \text{Sup Price}]$
- Delta Confidence Interval:  $IC_\delta = [\text{Inf Delta}, \text{Sup Delta}]$

### Description:

Computation for a Call - Put - CallSpread or Digit European Option of its Price and its Delta via the Malliavin method with the [Standard Monte Carlo](#) or [Quasi-Monte Carlo simulation](#). In the case of Monte Carlo simulation, the method also provides an estimation for the integration error and a confidence interval.

The underlying asset price evolves according to the Merton model, that is:

$$\begin{cases} S_{T-t} = s \\ \frac{dS_u}{S_u} = (r - \lambda \mathbb{E}U_1 - d)du + \sigma dB_u + d(\sum_{j=1}^{N_u} U_j), \end{cases} \quad (1)$$

where  $(B_u)_{t \geq 0}$  is a Brownian motion,  $(N_u)_{u \geq 0}$  is a Poisson process with deterministic jump intensity  $\lambda$ ,  $(U_u)_{j \geq 1}$  is a sequence of positive, independent stochastic variables and  $\sigma$  is a constant, such that  $\sigma > 0$ . We suppose that  $r$  is a deterministic risk-free interest rate. Then, we have

$$S_T = s \left( \prod_{j=1}^{N_T} (U_j + 1) \right) e^{(r - \lambda \mathbb{E} U_1 - d - \frac{\sigma^2}{2})T + \sigma B_T}. \quad (2)$$

Where  $S_T$  denotes the spot at maturity  $T$ ,  $s$  is the initial spot,  $t$  is the time to maturity.

In this context we suppose that the jump variables  $U$  are log-normal distributed with constant mean  $\mu$  and variance  $\gamma$ .

The Price of an option at  $T - t$  is:

$$P = E [\exp(-rt) f(K, S_T, R)]$$

where  $f$  denotes the payoff of the option,  $K$  the strike and  $R$  the rebate (for Digit option only).

Using the Malliavin integration by parts formula the Delta is given by:

$$\delta = E \left[ \frac{P B_T}{s \sigma T} \right]$$

Estimators are expressed as:

$$\tilde{P} = \frac{1}{N} \exp(-rt) \sum_{i=1}^N P(i)$$

where  $P(i) = f(S_T(i), K)$

$$\tilde{\delta} = \frac{1}{N} \exp(-rt) \sum_{i=1}^N \delta(i)$$

where

$$\delta(i) = \frac{P(i) B_T(i)}{s \sigma T}.$$

The values for  $P(i)$  are detailed for each option.

- **Put:** The payoff is  $(K - S_T)^+$ . We have:

$$P(i) = (K - S_T(i))^+$$

- **Call:** The payoff is  $(S_T - K)^+$ .

The *Call-Put Parity* relations for price and delta are expressed by:

$$C = P + s \exp(-dt) - K \exp(-rt)$$

$$\delta_C = \delta_P + \exp(-dt)$$

where  $C$  and  $P$  respectively denotes the Call and the Put prices.

They will be used for a Call simulation (which corresponds to a method of control variate and leads generally to a reduced variance for the estimator).

- **CallSpread:** The payoff is  $(S_T - K_1)^+ - (S_T - K_2)^+$ .

We have:

$$P(i) = [(S_T(i) - K_1)^+ - (S_T(i) - K_2)^+]$$

- **Digit:** The payoff is  $R1_{\{S_T - K \geq 0\}}$ .

We have:

$$P(i) = R1_{\{S_T(i) - K \geq 0\}}$$

For digital options we use the Malliavin techniques.

### Algorithm:

/\* Value to construct the confidence interval \*/  
For example if the confidence value is equal to 95% then the value  $z_\alpha$  used to construct the confidence interval is 1.96. This parameter is taken into account only for MC simulation and not for QMC simulation.

/\*Initialization\*/

/\*Call-Spread\*/

Strike  $K_1$  and  $K_2$  used for a Call-Spread option.

/\*Median forward stock and delta values\*/

Computation of intermediate values we use several times in the program.

/\* Change a Call into a Put to apply the Call-Put parity \*/

In case of Call, we modify parameters of the option; they will be reinitialized at the end of the simulation program. Simulation will be done as for a put.

- /\*MC sampling\*/

Initialization of the simulation: generator type, dimension, size  $N$  of the sample.

/\* Test after initialization for the generator \*/

Test if the dimension of the simulation is compatible with the selected generator. (See remarks on QMC simulation, especially on dimension of low-discrepancy sequences).

Definition of a parameter which exprimes if we realize a MC or QMC simulation. Some differences then appear in the algorithm for simulation of a gaussian variable and in results in the simulation.

/\* Begin N iterations \*/

- /\* Simulation of a gaussian variable according to the generator type, that is Monte Carlo or Quasi Monte Carlo. \*/

Call to the appropriate function to generate a standard gaussian variable. See the part about simulation of random variables for explanations on this point. We just recall that for a MC simulation, we use the Gauss-Abramovitz algorithm, and for a QMC simulation we use an inverse method.

- /\* Simulation of a poisson variable  $N_T$  with parameter  $\lambda T$  \*/

- /\*Price\*/

At the iteration  $i$ , we obtain

$$S_T(i) = s \exp \left[ \left( r - \lambda \mathbb{E}U_1 - d - \frac{\sigma^2}{2} \right) T \right] \exp(\sigma B_T(i)) \left( \prod_{j=1}^{N_T} \exp(\mu + \gamma g_j) \right)$$

$$P(i) = \text{Payoff}(S_T(i), K)$$

from a simulation of  $B_t(i)$  with the selected generator as  $\sqrt{t}g_i$  where  $g_i$  is a standard Gaussian variable.

Payoff functions are given for each option in the previous section.

- /\*Delta\*/

Calculation of Delta  $\delta_i$  with formula previously detailed for each option.

/\*Digit\*/

/\*CallSpread\*/

/\*Call-Put\*/

/\*Sum\*/

Computation of the sums  $\sum P_i$  and  $\sum \delta_i$  for the mean price and the mean delta.

/\*Sum of squares\*/

Computation of the sums  $\sum P_i^2$  and  $\sum \delta_i^2$  necessary for the variance price and the variance delta estimations. (finally only used for MC estimation)

/\* End N iterations \*/

• /\*Price\*/

The price estimator is:

$$P = \frac{1}{N} \exp(-rt) \sum_{i=1}^N P(i)$$

The error estimator is  $\sigma_P$  with :

$$\sigma_P^2 = \frac{1}{N-1} \left( \frac{1}{N} \exp(-2rt) \sum_{i=1}^N P(i)^2 - P^2 \right)$$

• /\*Delta\*/

$$\delta = \frac{1}{N} \exp(-rt) \sum_{i=1}^N \delta(i)$$

The error estimator is  $\sigma_\delta$  with:

$$\sigma_\delta^2 = \frac{1}{N-1} \left( \frac{1}{N} \exp(-2rt) \sum_{i=1}^N \delta(i)^2 - \delta^2 \right)$$

• /\* Call Price and Delta with the Call Put Parity \*/

We now compute the price and the delta in case of a call, because call was considered as a put until now.

Parameters of the option are reinitialized. This step is necessary: if you want to begin an other call simulation just after a first one with the standard method, the parameters must have been modified to specify that we really consider a Call and not a Put.

• /\* Price Confidence Interval \*/

The confidence interval is given as:

$$IC_P = [P - z_\alpha \sigma_P; P + z_\alpha \sigma_P]$$

with  $z_\alpha$  computed from the confidence value.

• /\* Delta Confidence Interval \*/

The confidence interval is given as:

$$IC_\delta = [\delta - z_\alpha \sigma_\delta; \delta + z_\alpha \sigma_\delta]$$

with  $z_\alpha$  computed from the confidence value.

Confidence intervals are always computed, but for a QMC simulation they don't work, thus they don't appear in the results.

## References