Financial Numerical Recipes in C#

National Taiwan University

Department of Finance

Andy Dong

2022/4/4

使用說明：

- 改寫自 Financial Numerical Recipes in C++, 2014, Bernt Arne Odegaard。
- 部分物件，Matrix、Vector、Utils，取自 QuantLib C#。
- 部分物件，DStat、MersenneTwister、Utils，自行撰寫或各有出處。
- 目標平台.Net Framework 4.5.2。
- 教學目的使用，沒有任何使用品質保證，使用者自行評估風險。
- 撰寫人員：董夢雲，台大財金所教授級專家。

昀騰金融科技股份有限公司

技術長
金融博士、證券分析師

董夢雲 Andy Dong

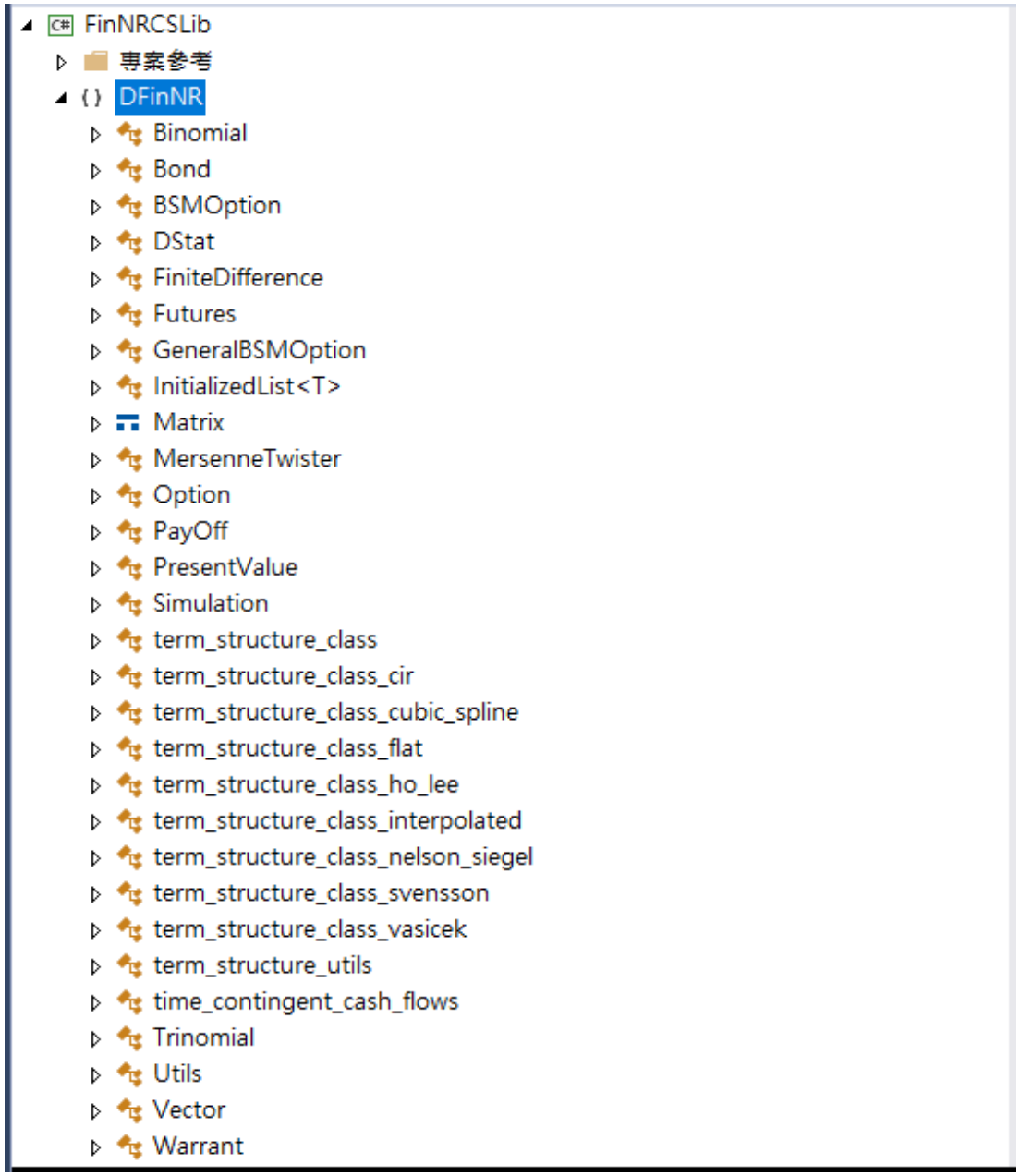
Line/WeChat:andydong3137
E:andydong1209@gmail.com
https://github/andydong1209
M:(T)0988-065-751 (C)1508-919-2872
10647 台北市大安區辛亥路一段 50 號 4 樓

現職：國立台灣大學財務金融研究所兼任教授級專家
國立台灣科技大學財務金融研究所兼任助理教授
台灣金融研訓院 2020 年菁英講座

經歷：中國信託商業銀行交易室研發科主管
凱基證券風險管理部主管兼亞洲區風險管理主管
中華開發金控、工業銀行風險管理處處長
永豐金控、商業銀行風險管理處處長
永豐商業銀行結構商品開發部副總經理

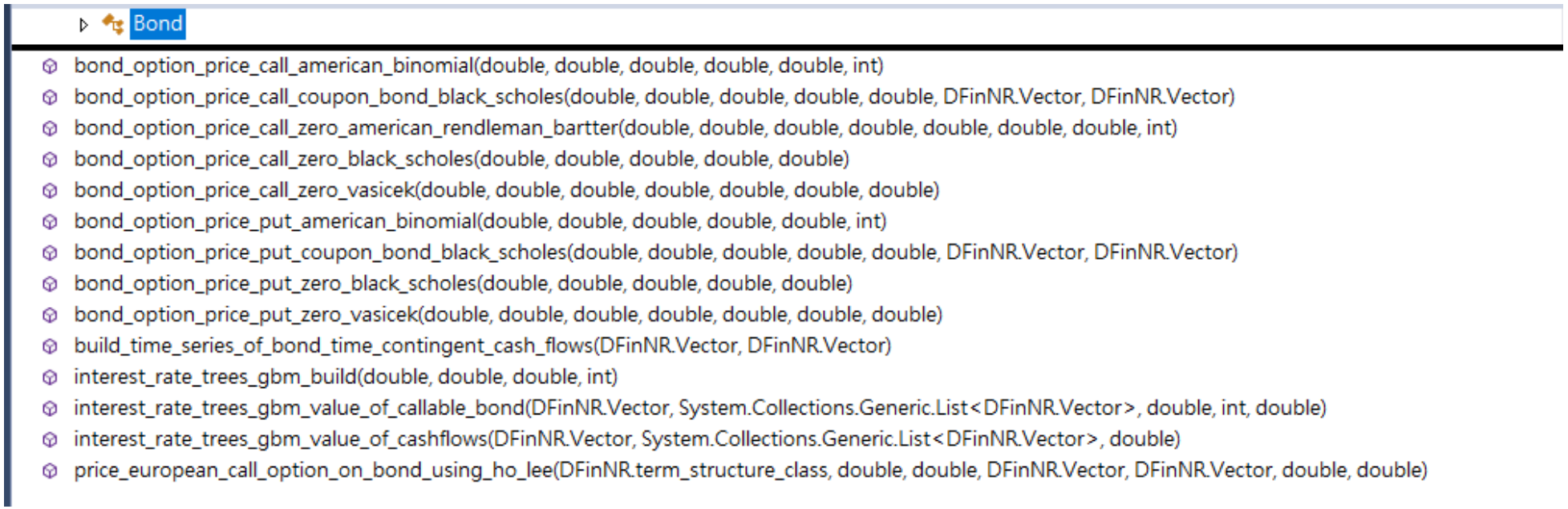
學歷：國立台灣大學電機工程學系學士
國立中央大學財務管理學研究所博士
專業：證券暨投資分析人員合格(1996)

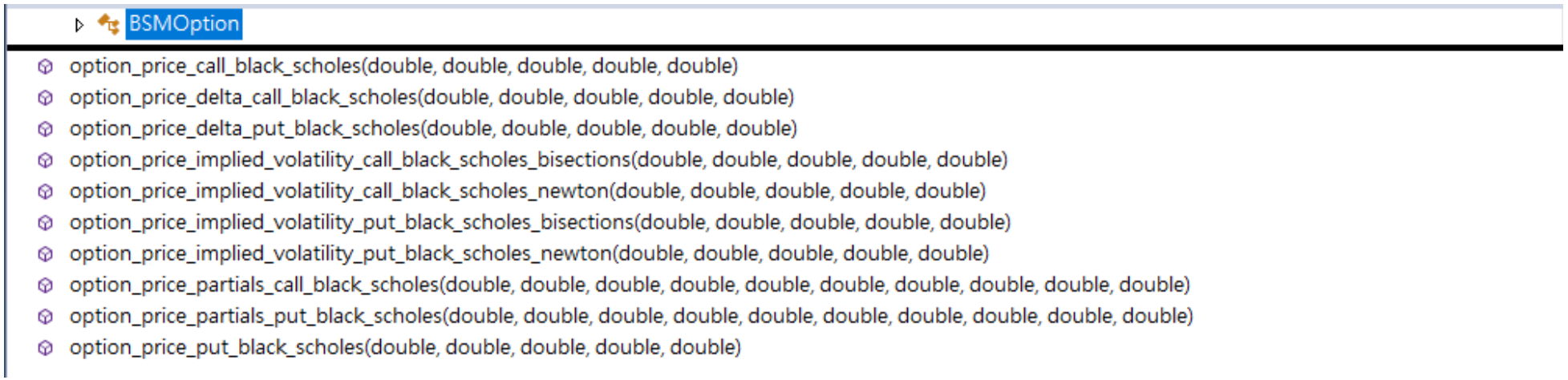
專長：風險管理理論與實務，資本配置與額度規劃、資產負債管理實務
外匯與利率結構商品評價實務，股權與債權及衍生商品評價實務
GPU 平行運算與結構商品系統開發，CUDA、OpenCL
CPU 平行運算與 ALM 系統開發，C#/C++/C、.Net Framework、SQL
人工智慧(Deep Learning)交易策略開發，Python、Keras、TensorFlow

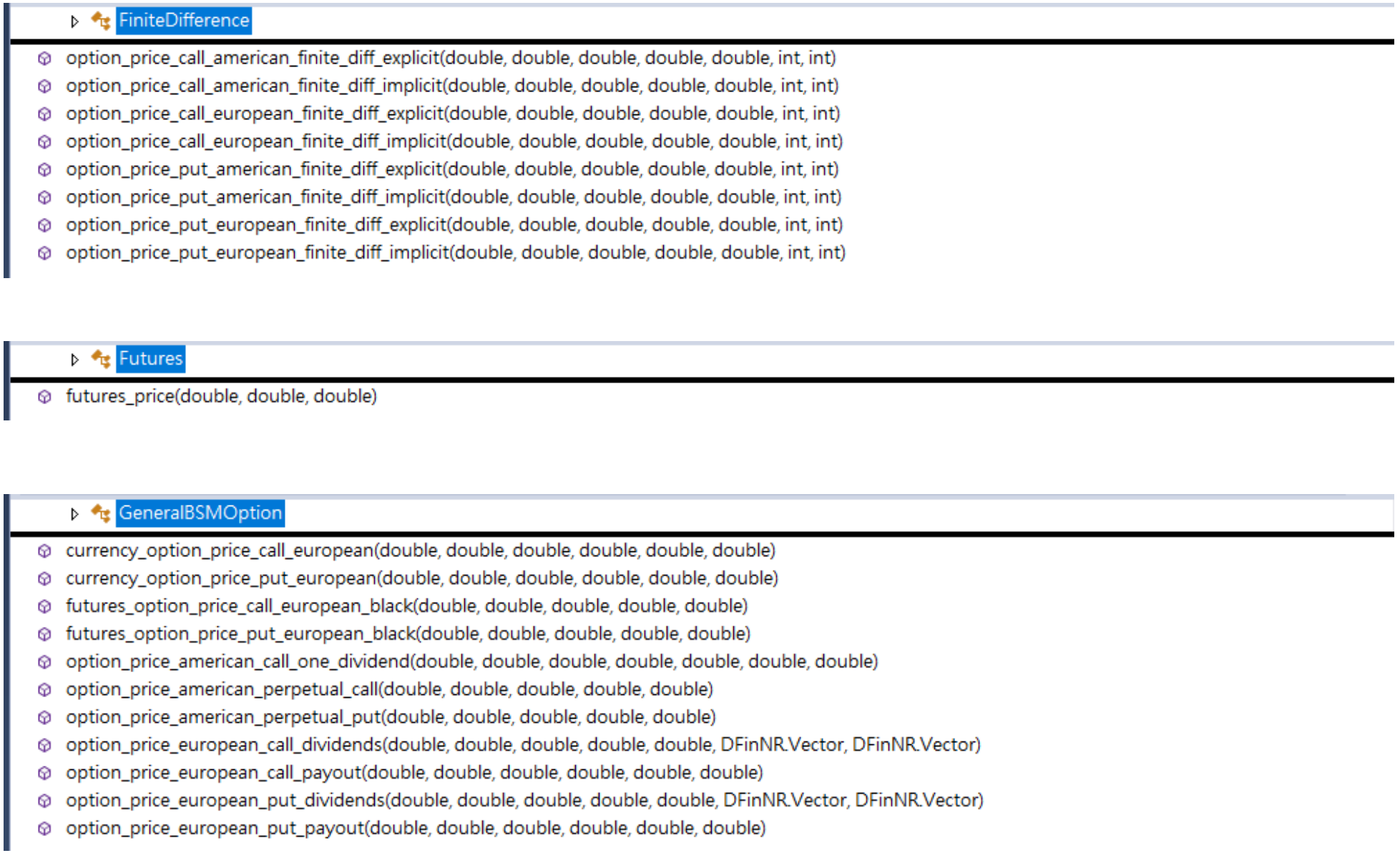

FinNRCSLib
專案參考
DFinNR
Binomial
Bond
BSMOption
DStat
FiniteDifference
Futures
GeneralBSMOption
InitializedList<T>
Matrix
MersenneTwister
Option
PayOff
PresentValue
Simulation
term_structure_class
term_structure_class_cir
term_structure_class_cubic_spline
term_structure_class_flat
term_structure_class_ho_lee
term_structure_class_interpolated
term_structure_class_nelson_siegel
term_structure_class_svensson
term_structure_class_vasicek
term_structure_utils
time_contingent_cash_flows
Trinomial
Utils
Vector
Warrant

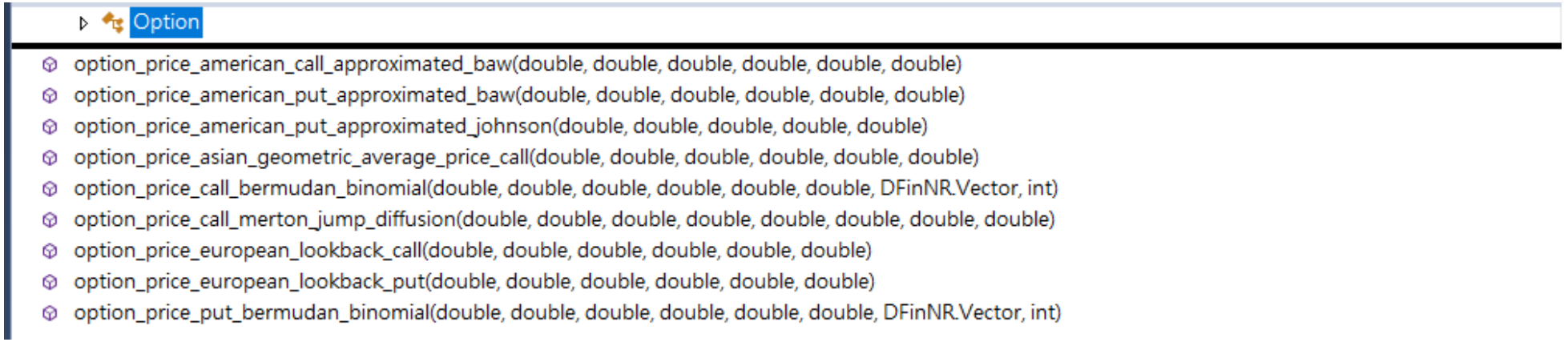
一、財務物件與方法

- Binomial
  - binomial_tree(double, double, double, int)
  - currency_option_price_call_american_binomial(double, double, double, double, double, double, int)
  - currency_option_price_put_american_binomial(double, double, double, double, double, double, int)
  - futures_option_price_call_american_binomial(double, double, double, double, double, int)
  - futures_option_price_put_american_binomial(double, double, double, double, double, int)
  - option_price_call_american_binomial(double, double, double, double, double, double, int)
  - option_price_call_american_binomial(double, double, double, double, double, int)
  - option_price_call_american_discrete_dividends_binomial(double, double, double, double, double, int, DFinNR.Vector, DFinNR.Vector)
  - option_price_call_american_proportional_dividends_binomial(double, double, double, double, double, int, DFinNR.Vector, DFinNR.Vector)
  - option_price_call_european_binomial(double, double, double, double, double, int)
  - option_price_call_european_binomial_multi_period_given_ud(double, double, double, double, double, int)
  - option_price_call_european_binomial_single_period(double, double, double, double, double)
  - option_price_delta_american_call_binomial(double, double, double, double, double, int)
  - option_price_delta_american_put_binomial(double, double, double, double, double, int)
  - option_price_delta_generic_binomial(double, double, System.Func<double, double, double>, double, double, double, int)
  - option_price_generic_binomial(double, double, System.Func<double, double, double>, double, double, double, int)
  - option_price_partials_american_call_binomial(double, double, double, double, double, int, double, double, double, double, double)
  - option_price_partials_american_put_binomial(double, double, double, double, double, int, double, double, double, double, double)
  - option_price_put_american_binomial(double, double, double, double, double, double, int)
  - option_price_put_american_binomial(double, double, double, double, double, int)
  - option_price_put_american_discrete_dividends_binomial(double, double, double, double, double, int, DFinNR.Vector, DFinNR.Vector)
  - option_price_put_american_proportional_dividends_binomial(double, double, double, double, double, int, DFinNR.Vector, DFinNR.Vector)
  - option_price_put_european_binomial(double, double, double, double, double, int)

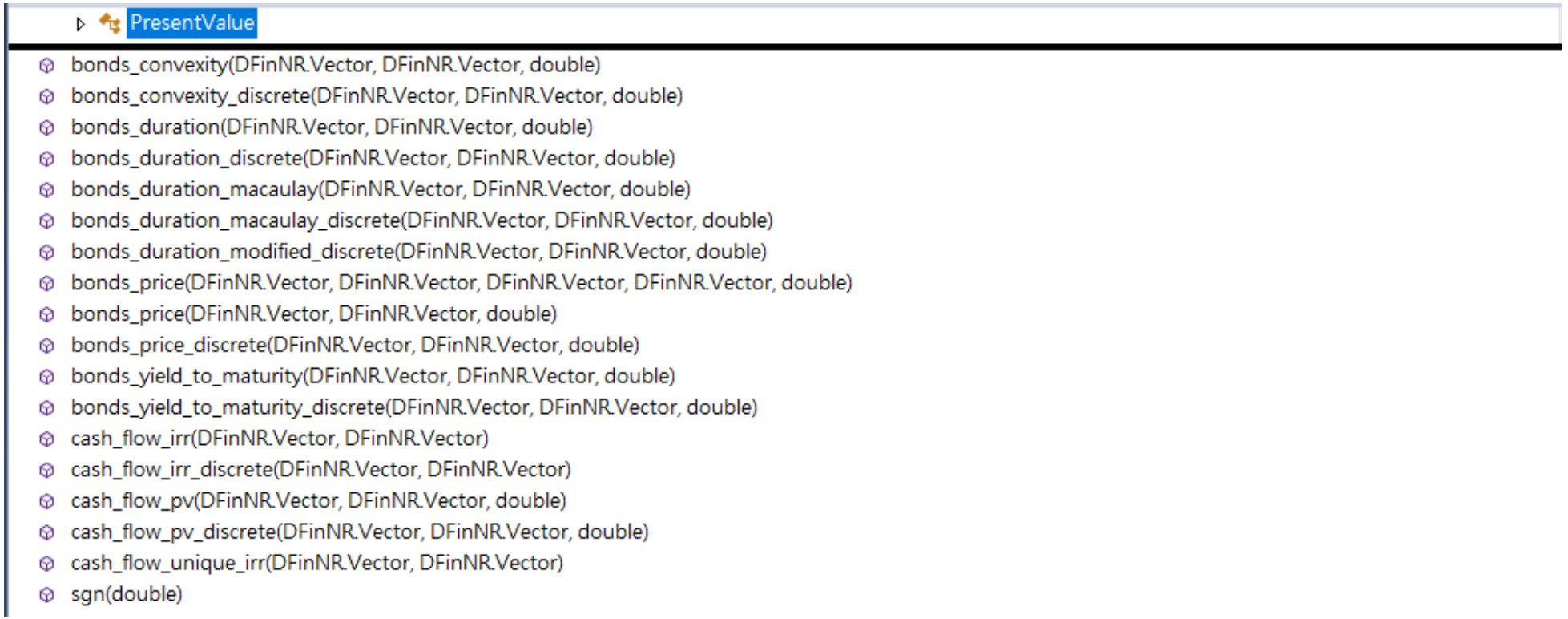

Bond
bond_option_price_call_american_binomial(double, double, double, double, double, int)
bond_option_price_call_coupon_bond_black_scholes(double, double, double, double, double, DFinNR.Vector, DFinNR.Vector)
bond_option_price_call_zero_american_rendleman_bartter(double, double, double, double, double, double, double, int)
bond_option_price_call_zero_black_scholes(double, double, double, double, double)
bond_option_price_call_zero_vasicek(double, double, double, double, double, double, double)
bond_option_price_put_american_binomial(double, double, double, double, double, int)
bond_option_price_put_coupon_bond_black_scholes(double, double, double, double, double, DFinNR.Vector, DFinNR.Vector)
bond_option_price_put_zero_black_scholes(double, double, double, double, double)
bond_option_price_put_zero_vasicek(double, double, double, double, double, double, double)
build_time_series_of_bond_time_contingent_cash_flows(DFinNR.Vector, DFinNR.Vector)
interest_rate_trees_gbm_build(double, double, double, int)
interest_rate_trees_gbm_value_of_callable_bond(DFinNR.Vector, System.Collections.Generic.List<DFinNR.Vector>, double, int, double)
interest_rate_trees_gbm_value_of_cashflows(DFinNR.Vector, System.Collections.Generic.List<DFinNR.Vector>, double)
price_european_call_option_on_bond_using_ho_lee(DFinNR.term_structure_class, double, double, DFinNR.Vector, DFinNR.Vector, double, double)

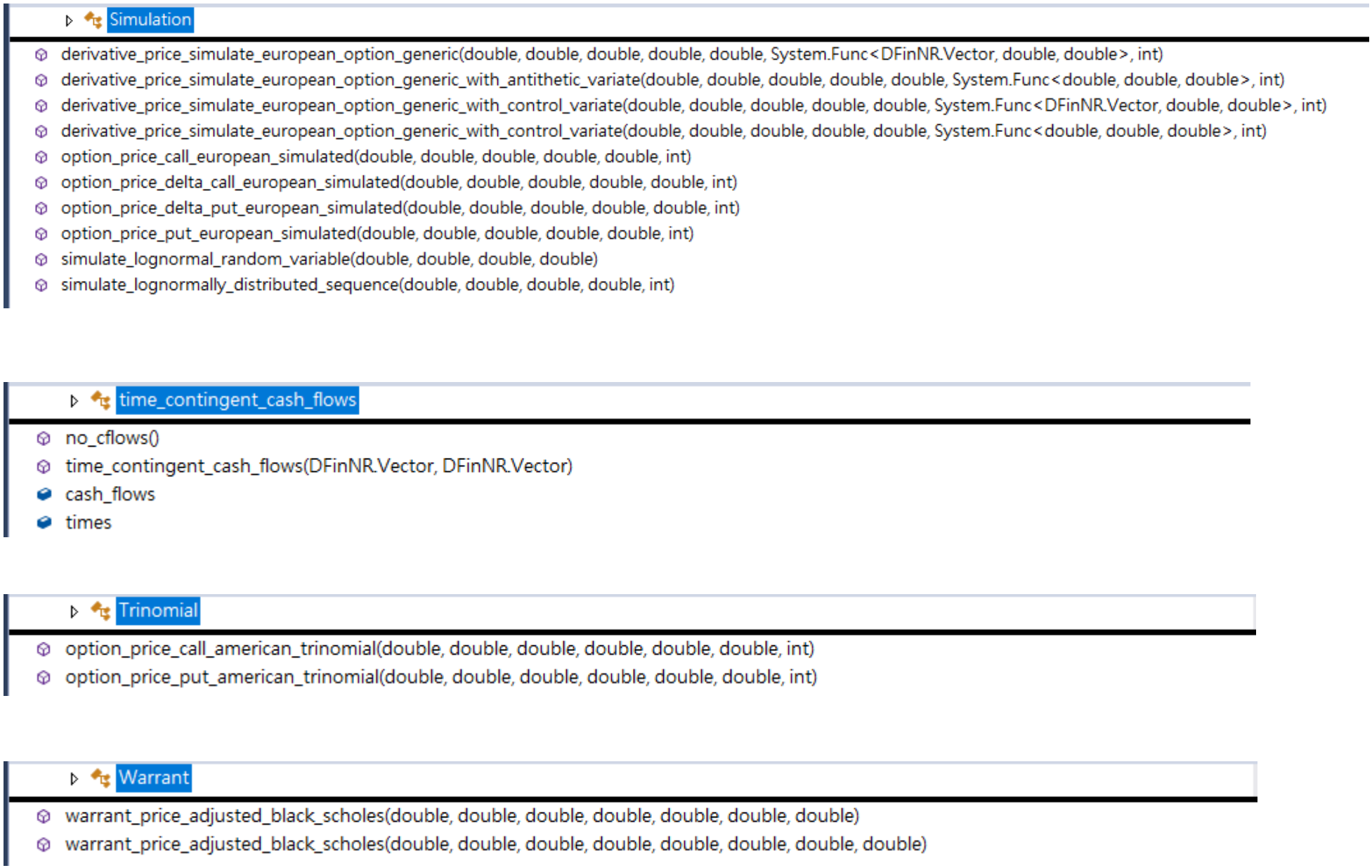

BSMOption
option_price_call_black_scholes(double, double, double, double, double)
option_price_delta_call_black_scholes(double, double, double, double, double)
option_price_delta_put_black_scholes(double, double, double, double, double)
option_price_implied_volatility_call_black_scholes_bisections(double, double, double, double, double)
option_price_implied_volatility_call_black_scholes_newton(double, double, double, double, double)
option_price_implied_volatility_put_black_scholes_bisections(double, double, double, double, double)
option_price_implied_volatility_put_black_scholes_newton(double, double, double, double, double)
option_price_partials_call_black_scholes(double, double, double, double, double, double, double, double, double, double)
option_price_partials_put_black_scholes(double, double, double, double, double, double, double, double, double, double)
option_price_put_black_scholes(double, double, double, double, double)


FiniteDifference
option_price_call_american_finite_diff_explicit(double, double, double, double, double, int, int)
option_price_call_american_finite_diff_implicit(double, double, double, double, double, int, int)
option_price_call_european_finite_diff_explicit(double, double, double, double, double, int, int)
option_price_call_european_finite_diff_implicit(double, double, double, double, double, int, int)
option_price_put_american_finite_diff_explicit(double, double, double, double, double, int, int)
option_price_put_american_finite_diff_implicit(double, double, double, double, double, int, int)
option_price_put_european_finite_diff_explicit(double, double, double, double, double, int, int)
option_price_put_european_finite_diff_implicit(double, double, double, double, double, int, int)
Futures
futures_price(double, double, double)
GeneralBSMOption
currency_option_price_call_european(double, double, double, double, double, double)
currency_option_price_put_european(double, double, double, double, double, double)
futures_option_price_call_european_black(double, double, double, double, double)
futures_option_price_put_european_black(double, double, double, double, double)
option_price_american_call_one_dividend(double, double, double, double, double, double, double)
option_price_american_perpetual_call(double, double, double, double, double)
option_price_american_perpetual_put(double, double, double, double, double)
option_price_european_call_dividends(double, double, double, double, double, DFinNR.Vector, DFinNR.Vector)
option_price_european_call_payout(double, double, double, double, double, double)
option_price_european_put_dividends(double, double, double, double, double, DFinNR.Vector, DFinNR.Vector)
option_price_european_put_payout(double, double, double, double, double, double)

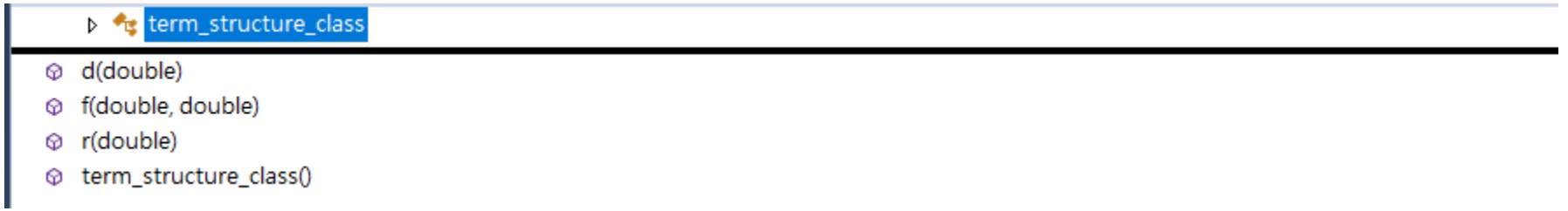

Option
option_price_american_call_approximated_baw(double, double, double, double, double, double)
option_price_american_put_approximated_baw(double, double, double, double, double, double)
option_price_american_put_approximated_johnson(double, double, double, double, double)
option_price_asian_geometric_average_price_call(double, double, double, double, double, double)
option_price_call_bermudan_binomial(double, double, double, double, double, double, DFinNR.Vector, int)
option_price_call_merton_jump_diffusion(double, double, double, double, double, double, double, double)
option_price_european_lookback_call(double, double, double, double, double, double)
option_price_european_lookback_put(double, double, double, double, double, double)
option_price_put_bermudan_binomial(double, double, double, double, double, double, DFinNR.Vector, int)

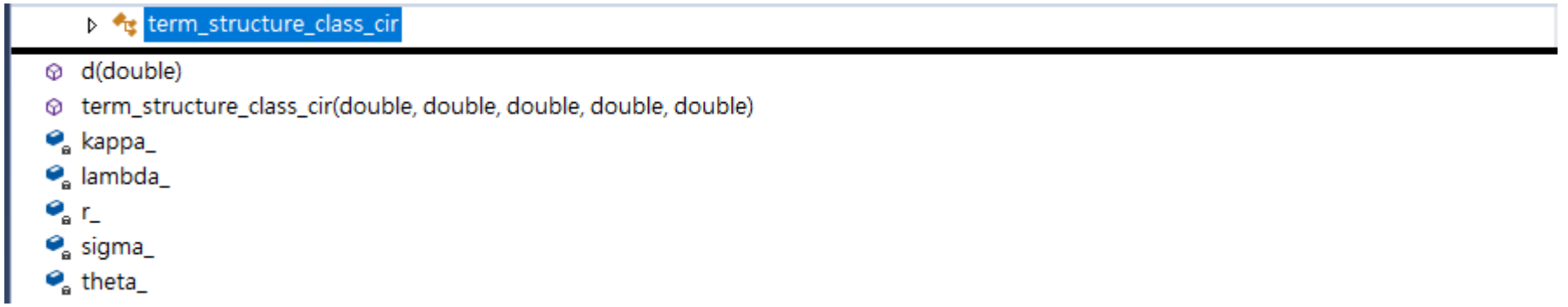

PayOff
payoff_arithmetric_average_call(DFinNR.Vector, double)
payoff_asset_or_nothing_call(double, double)
payoff_binary_call(double, double)
payoff_binary_put(double, double)
payoff_call(double, double)
payoff_cash_or_nothing_call(double, double)
payoff_geometric_average_call(DFinNR.Vector, double)
payoff_lookback_call(DFinNR.Vector, double)
payoff_lookback_put(DFinNR.Vector, double)
payoff_put(double, double)

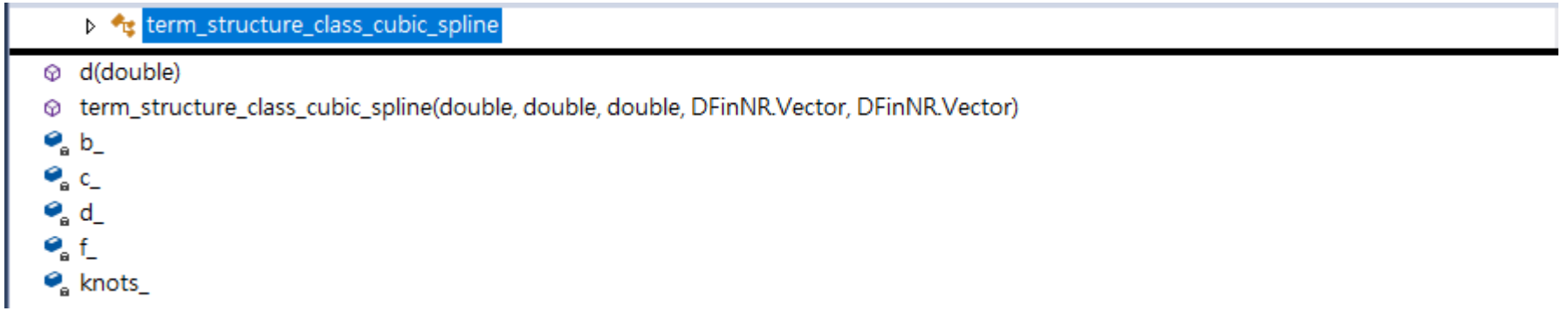

PresentValue
bonds_convexity(DFinNR.Vector, DFinNR.Vector, double)
bonds_convexity_discrete(DFinNR.Vector, DFinNR.Vector, double)
bonds_duration(DFinNR.Vector, DFinNR.Vector, double)
bonds_duration_discrete(DFinNR.Vector, DFinNR.Vector, double)
bonds_duration_macaulay(DFinNR.Vector, DFinNR.Vector, double)
bonds_duration_macaulay_discrete(DFinNR.Vector, DFinNR.Vector, double)
bonds_duration_modified_discrete(DFinNR.Vector, DFinNR.Vector, double)
bonds_price(DFinNR.Vector, DFinNR.Vector, DFinNR.Vector, DFinNR.Vector, double)
bonds_price(DFinNR.Vector, DFinNR.Vector, double)
bonds_price_discrete(DFinNR.Vector, DFinNR.Vector, double)
bonds_yield_to_maturity(DFinNR.Vector, DFinNR.Vector, double)
bonds_yield_to_maturity_discrete(DFinNR.Vector, DFinNR.Vector, double)
cash_flow_irr(DFinNR.Vector, DFinNR.Vector)
cash_flow_irr_discrete(DFinNR.Vector, DFinNR.Vector)
cash_flow_pv(DFinNR.Vector, DFinNR.Vector, double)
cash_flow_pv_discrete(DFinNR.Vector, DFinNR.Vector, double)
cash_flow_unique_irr(DFinNR.Vector, DFinNR.Vector)
sgn(double)

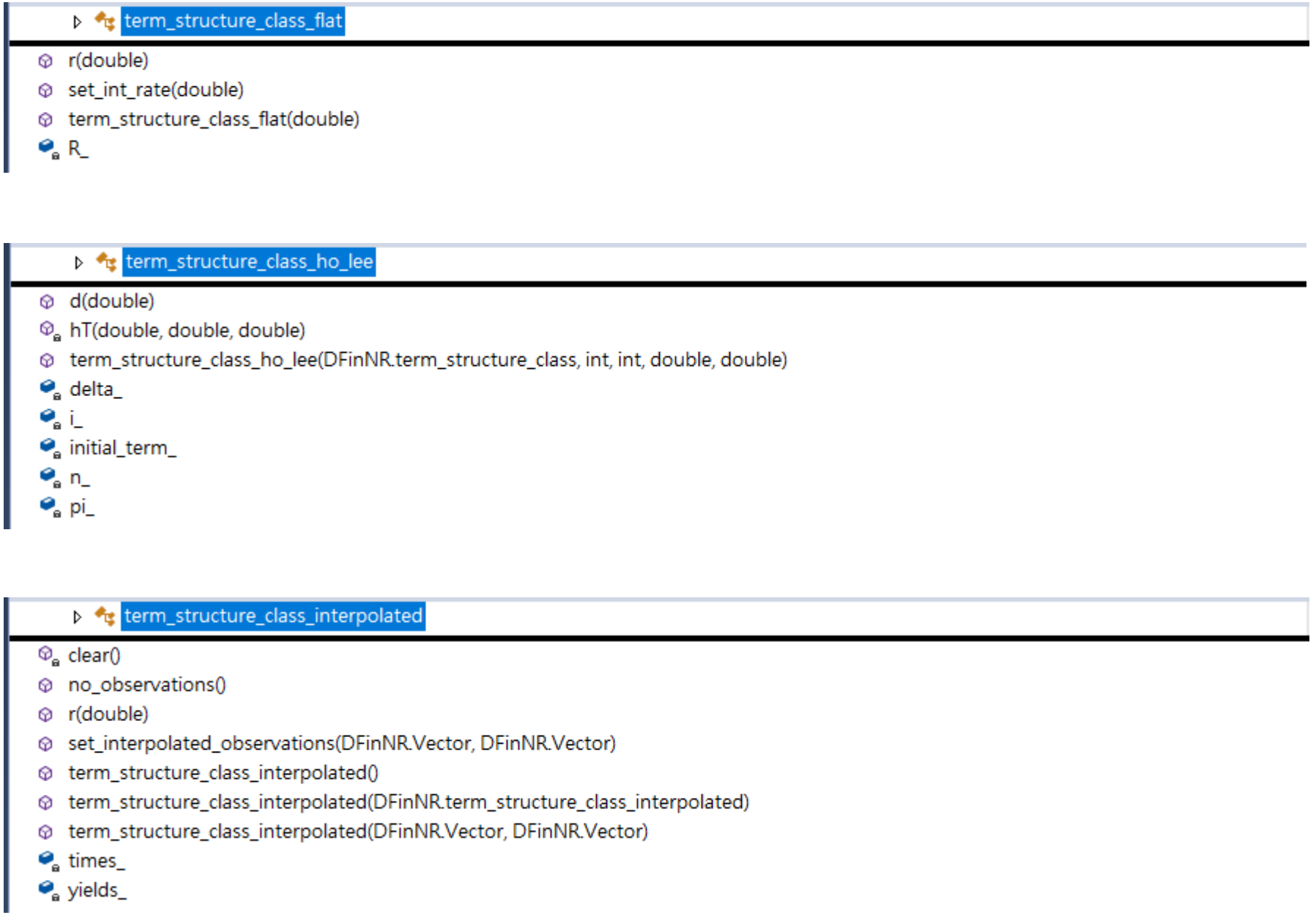

Simulation
derivative_price_simulate_european_option_generic(double, double, double, double, double, System.Func<DFinNR.Vector, double, double>, int)
derivative_price_simulate_european_option_generic_with_antithetic_variate(double, double, double, double, double, System.Func<double, double, double>, int)
derivative_price_simulate_european_option_generic_with_control_variate(double, double, double, double, double, System.Func<DFinNR.Vector, double, double>, int)
derivative_price_simulate_european_option_generic_with_control_variate(double, double, double, double, double, System.Func<double, double, double>, int)
option_price_call_european_simulated(double, double, double, double, double, int)
option_price_delta_call_european_simulated(double, double, double, double, double, int)
option_price_delta_put_european_simulated(double, double, double, double, double, int)
option_price_put_european_simulated(double, double, double, double, double, int)
simulate_lognormal_random_variable(double, double, double, double)
simulate_lognormally_distributed_sequence(double, double, double, double, int)
time_contingent_cash_flows
no_cflows()
time_contingent_cash_flows(DFinNR.Vector, DFinNR.Vector)
cash_flows
times
Trinomial
option_price_call_american_trinomial(double, double, double, double, double, double, int)
option_price_put_american_trinomial(double, double, double, double, double, double, int)
Warrant
warrant_price_adjusted_black_scholes(double, double, double, double, double, double, double)
warrant_price_adjusted_black_scholes(double, double, double, double, double, double, double, double)

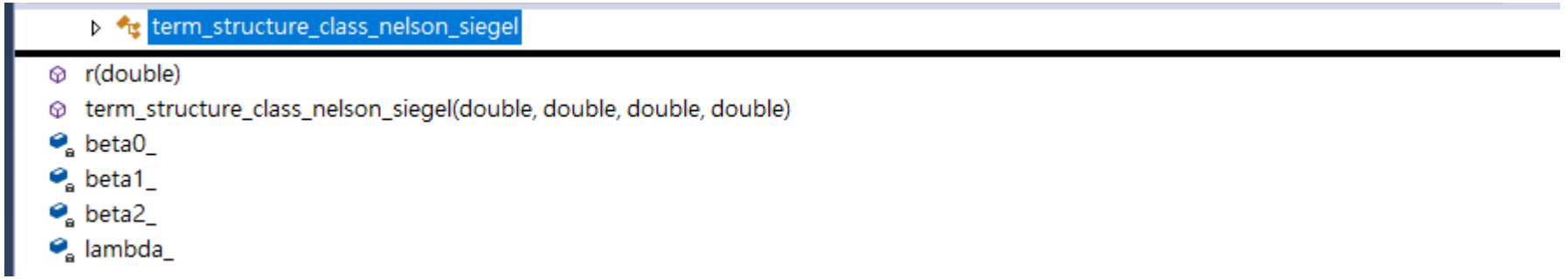
二、期限結構物件與方法

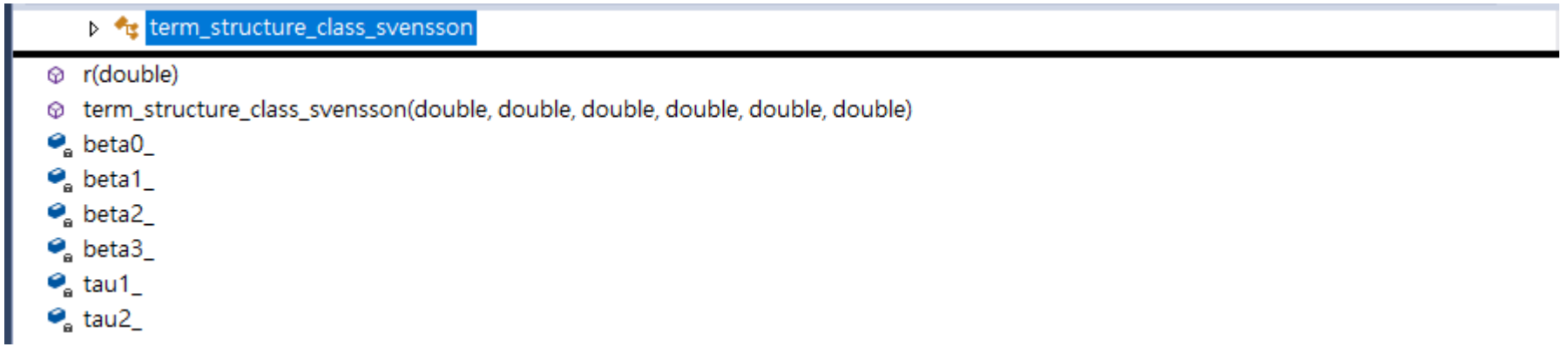

term_structure_class
d(double)
f(double, double)
r(double)
term_structure_class()

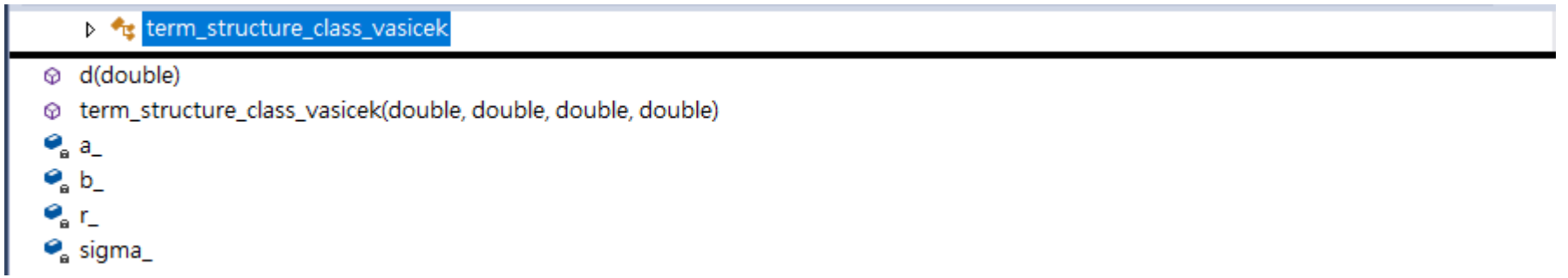

term_structure_class_cir
d(double)
term_structure_class_cir(double, double, double, double, double)
kappa_
lambda_
r_
sigma_
theta_

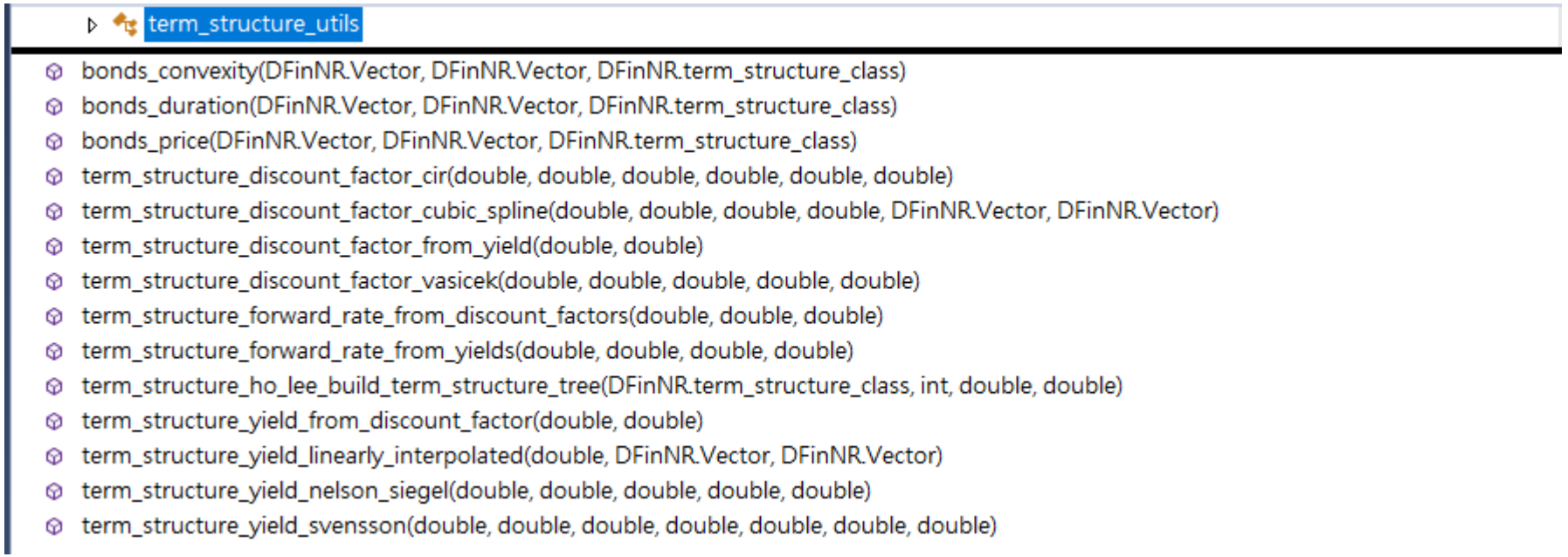

term_structure_class_cubic_spline
d(double)
term_structure_class_cubic_spline(double, double, double, DFinNR.Vector, DFinNR.Vector)
b_
c_
d_
f_
knots_


term_structure_class_flat
r(double)
set_int_rate(double)
term_structure_class_flat(double)
R_
term_structure_class_ho_lee
d(double)
hT(double, double, double)
term_structure_class_ho_lee(DFinNR.term_structure_class, int, int, double, double)
delta_
i_
initial_term_
n_
pi_
term_structure_class_interpolated
clear()
no_observations()
r(double)
set_interpolated_observations(DFinNR.Vector, DFinNR.Vector)
term_structure_class_interpolated()
term_structure_class_interpolated(DFinNR.term_structure_class_interpolated)
term_structure_class_interpolated(DFinNR.Vector, DFinNR.Vector)
times_
yields_

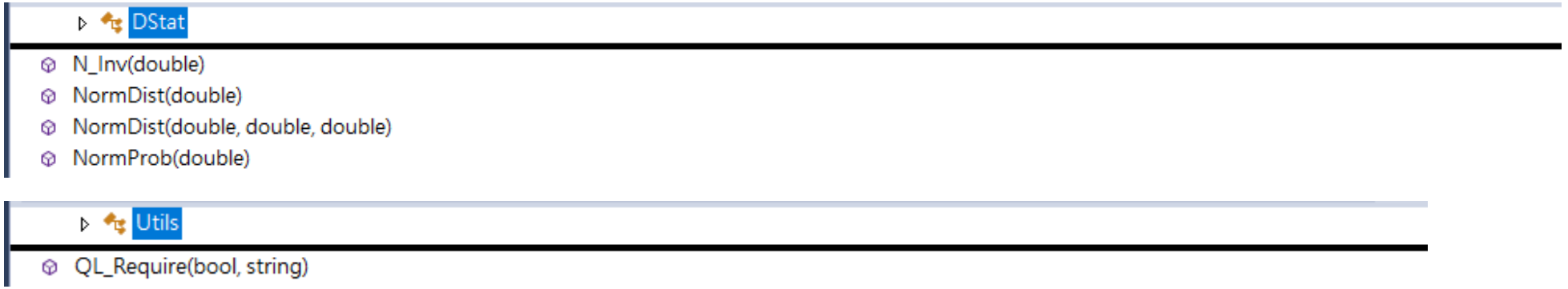

term_structure_class_nelson_siegel
r(double)
term_structure_class_nelson_siegel(double, double, double, double)
beta0_
beta1_
beta2_
lambda_


term_structure_class_svensson
r(double)
term_structure_class_svensson(double, double, double, double, double, double)
beta0_
beta1_
beta2_
beta3_
tau1_
tau2_

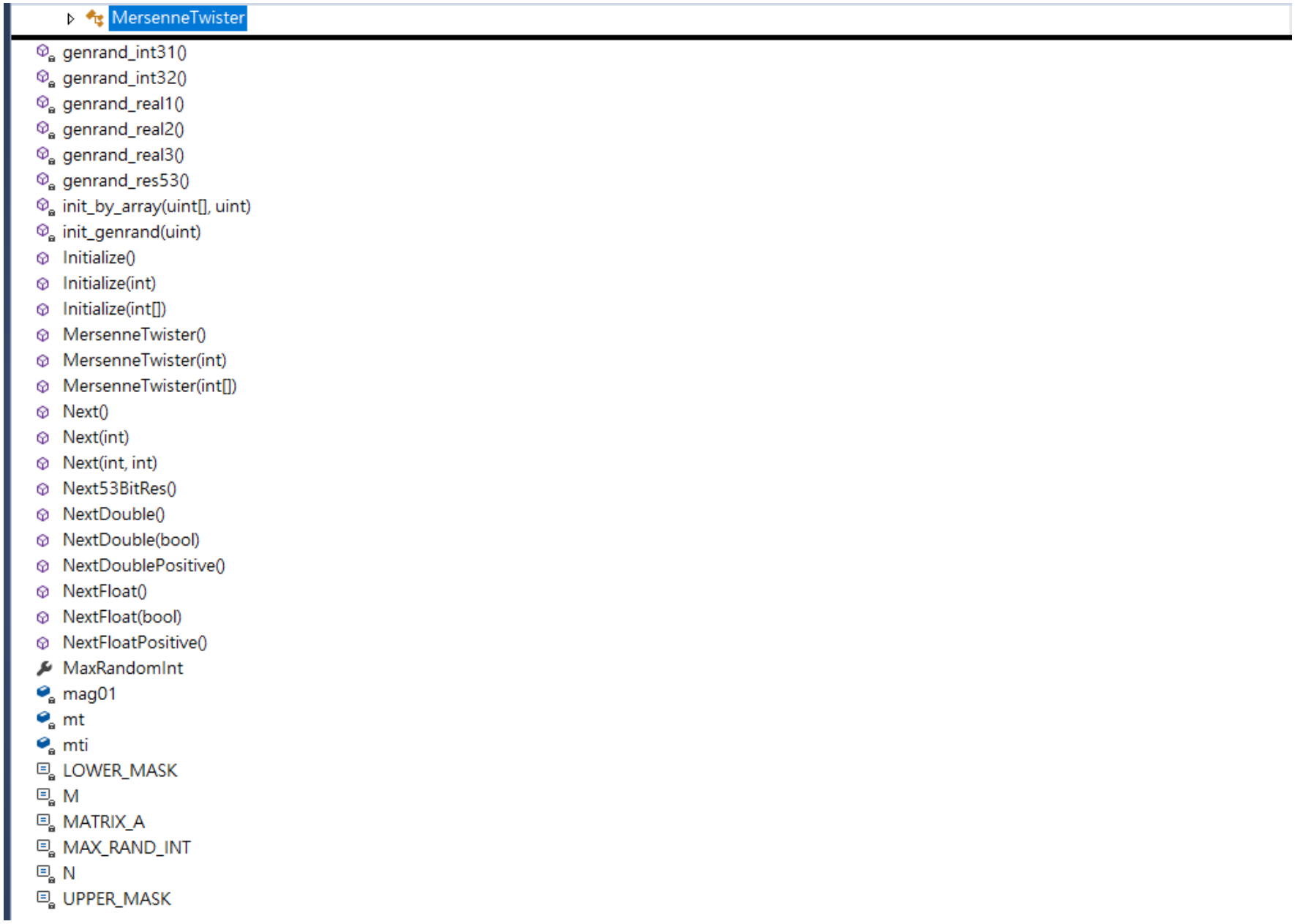

term_structure_class_vasicek
d(double)
term_structure_class_vasicek(double, double, double, double)
a_
b_
r_
sigma_

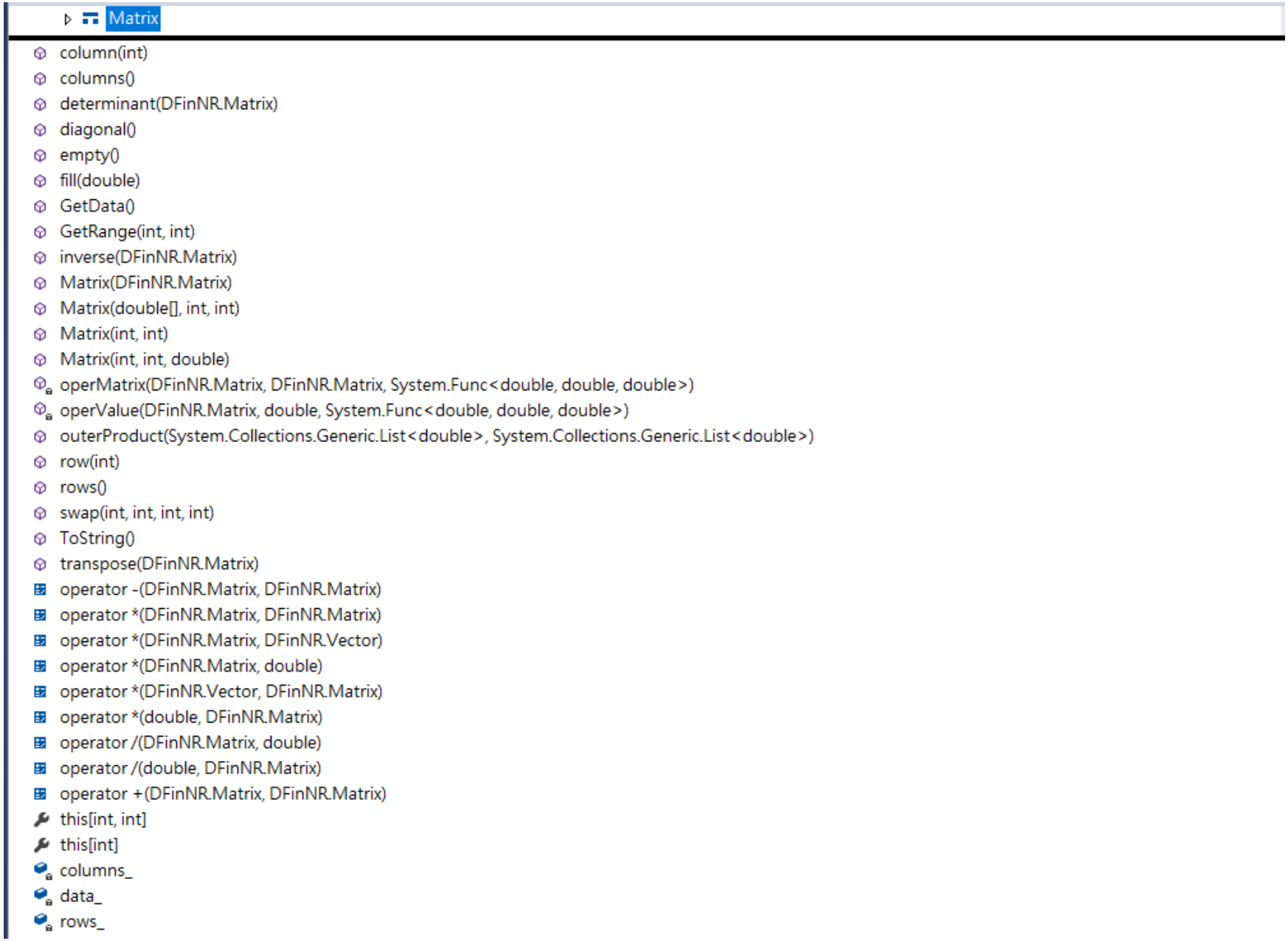

term_structure_utils
bonds_convexity(DFinNR.Vector, DFinNR.Vector, DFinNR.term_structure_class)
bonds_duration(DFinNR.Vector, DFinNR.Vector, DFinNR.term_structure_class)
bonds_price(DFinNR.Vector, DFinNR.Vector, DFinNR.term_structure_class)
term_structure_discount_factor_cir(double, double, double, double, double, double)
term_structure_discount_factor_cubic_spline(double, double, double, double, DFinNR.Vector, DFinNR.Vector)
term_structure_discount_factor_from_yield(double, double)
term_structure_discount_factor_vasicek(double, double, double, double, double)
term_structure_forward_rate_from_discount_factors(double, double, double)
term_structure_forward_rate_from_yields(double, double, double, double)
term_structure_ho_lee_build_term_structure_tree(DFinNR.term_structure_class, int, double, double)
term_structure_yield_from_discount_factor(double, double)
term_structure_yield_linearly_interpolated(double, DFinNR.Vector, DFinNR.Vector)
term_structure_yield_nelson_siegel(double, double, double, double, double)
term_structure_yield_svensson(double, double, double, double, double, double, double)

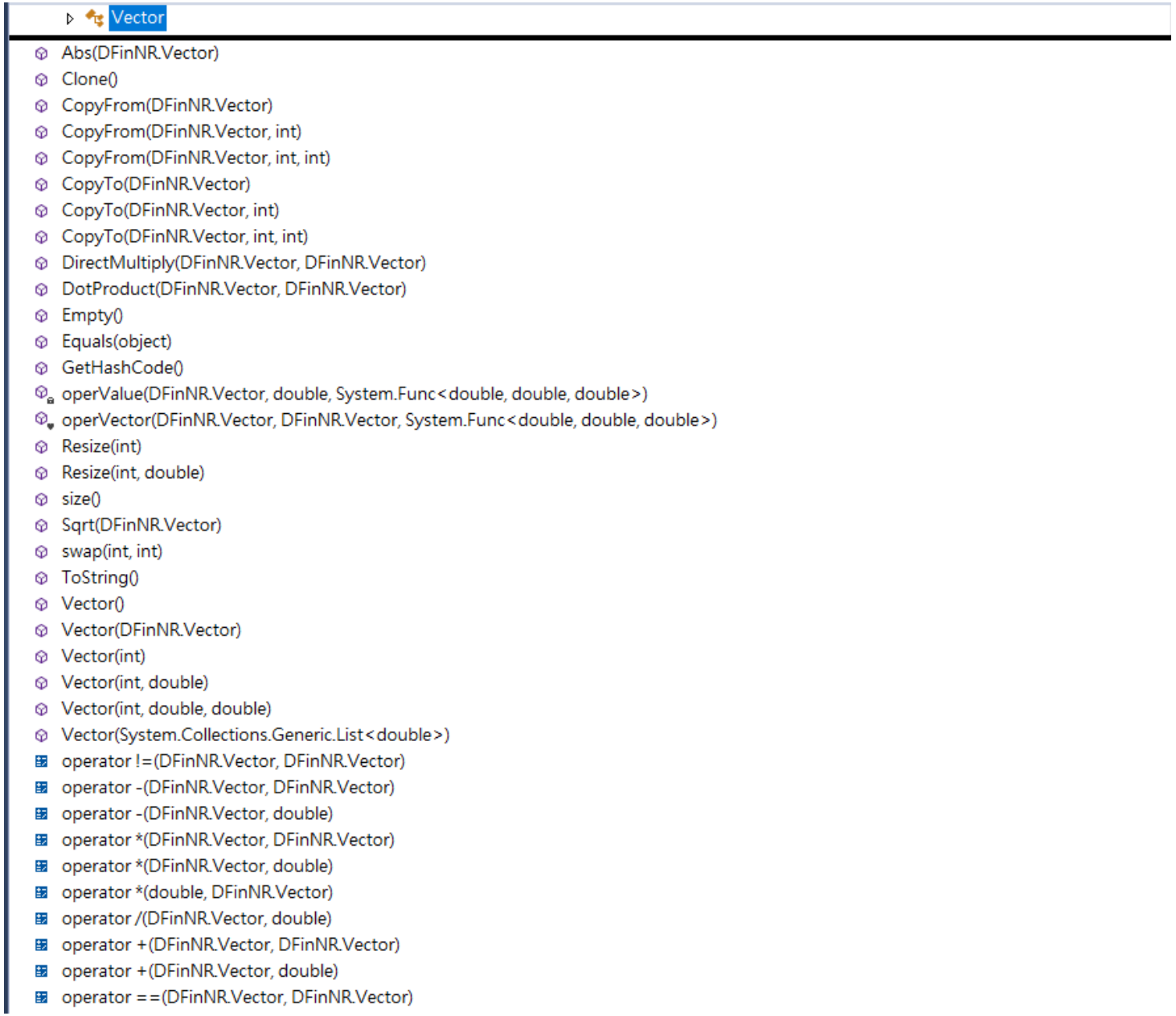
三、輔助物件與方法


DStat
N_Inv(double)
NormDist(double)
NormDist(double, double, double)
NormProb(double)

Utils
QL_Require(bool, string)

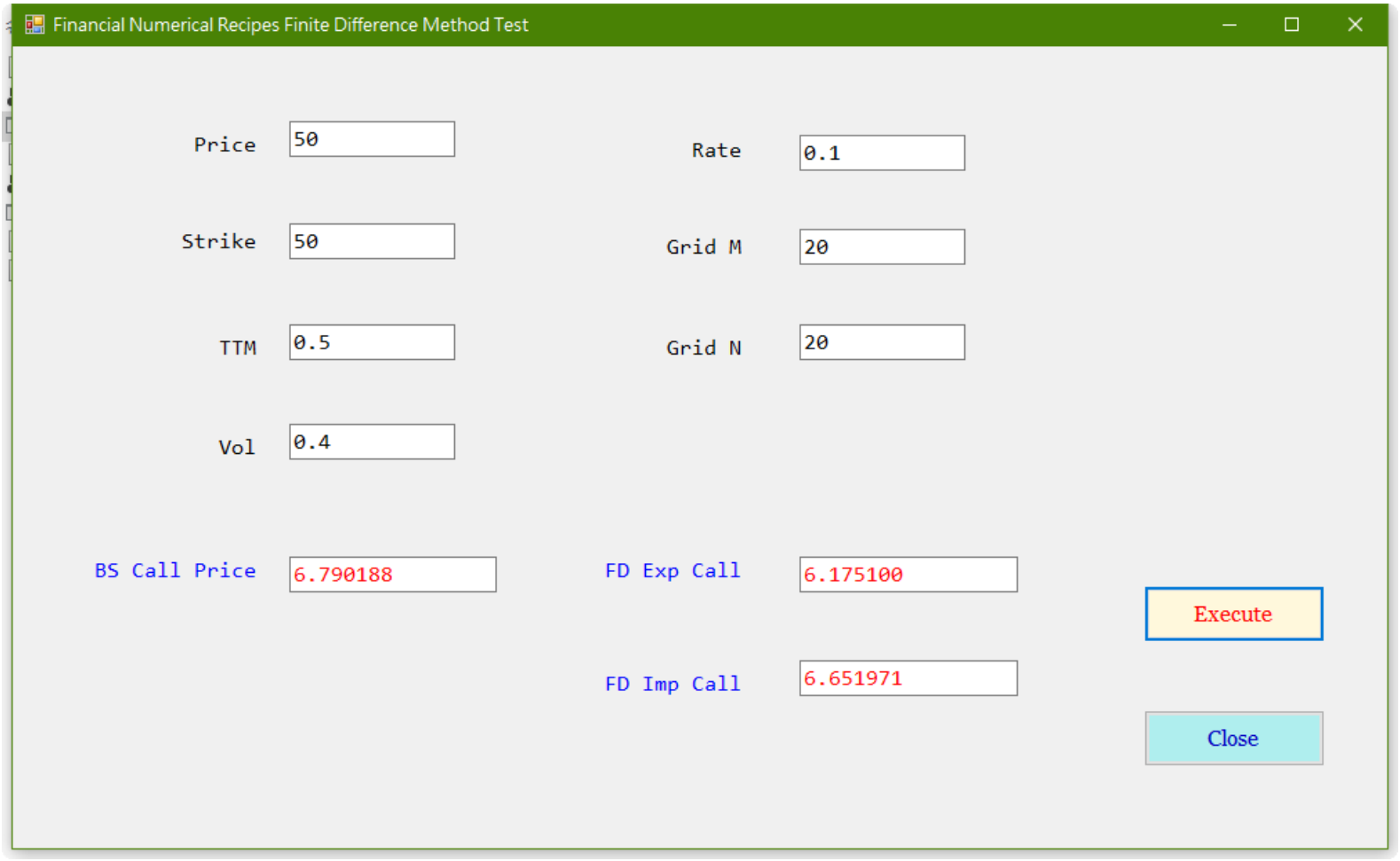

MersenneTwister
genrand_int31()
genrand_int32()
genrand_real1()
genrand_real2()
genrand_real3()
genrand_res53()
init_by_array(uint[], uint)
init_genrand(uint)
Initialize()
Initialize(int)
Initialize(int[])
MersenneTwister()
MersenneTwister(int)
MersenneTwister(int[])
Next()
Next(int)
Next(int, int)
Next53BitRes()
NextDouble()
NextDouble(bool)
NextDoublePositive()
NextFloat()
NextFloat(bool)
NextFloatPositive()
MaxRandomInt
mag01
mt
mti
LOWER_MASK
M
MATRIX_A
MAX_RAND_INT
N
UPPER_MASK


Matrix
column(int)
columns()
determinant(DFinNR.Matrix)
diagonal()
empty()
fill(double)
GetData()
GetRange(int, int)
inverse(DFinNR.Matrix)
Matrix(DFinNR.Matrix)
Matrix(double[], int, int)
Matrix(int, int)
Matrix(int, int, double)
operMatrix(DFinNR.Matrix, DFinNR.Matrix, System.Func<double, double, double>)
operValue(DFinNR.Matrix, double, System.Func<double, double, double>)
outerProduct(System.Collections.Generic.List<double>, System.Collections.Generic.List<double>)
row(int)
rows()
swap(int, int, int, int)
ToString()
transpose(DFinNR.Matrix)
operator -(DFinNR.Matrix, DFinNR.Matrix)
operator *(DFinNR.Matrix, DFinNR.Matrix)
operator *(DFinNR.Matrix, DFinNR.Vector)
operator *(DFinNR.Matrix, double)
operator *(DFinNR.Vector, DFinNR.Matrix)
operator *(double, DFinNR.Matrix)
operator /(DFinNR.Matrix, double)
operator /(double, DFinNR.Matrix)
operator +(DFinNR.Matrix, DFinNR.Matrix)
this[int, int]
this[int]
columns_
data_
rows_


Vector
Abs(DFinNR.Vector)
Clone()
CopyFrom(DFinNR.Vector)
CopyFrom(DFinNR.Vector, int)
CopyFrom(DFinNR.Vector, int, int)
CopyTo(DFinNR.Vector)
CopyTo(DFinNR.Vector, int)
CopyTo(DFinNR.Vector, int, int)
DirectMultiply(DFinNR.Vector, DFinNR.Vector)
DotProduct(DFinNR.Vector, DFinNR.Vector)
Empty()
Equals(object)
GetHashCode()
operValue(DFinNR.Vector, double, System.Func<double, double, double>)
operVector(DFinNR.Vector, DFinNR.Vector, System.Func<double, double, double>)
Resize(int)
Resize(int, double)
size()
Sqrt(DFinNR.Vector)
swap(int, int)
ToString()
Vector()
Vector(DFinNR.Vector)
Vector(int)
Vector(int, double)
Vector(int, double, double)
Vector(System.Collections.Generic.List<double>)
operator !=(DFinNR.Vector, DFinNR.Vector)
operator -(DFinNR.Vector, DFinNR.Vector)
operator -(DFinNR.Vector, double)
operator *(DFinNR.Vector, DFinNR.Vector)
operator *(DFinNR.Vector, double)
operator *(double, DFinNR.Vector)
operator /(DFinNR.Vector, double)
operator +(DFinNR.Vector, DFinNR.Vector)
operator +(DFinNR.Vector, double)
operator ==(DFinNR.Vector, DFinNR.Vector)

四、使用範例

◆ 範例畫面


Financial Numerical Recipes Finite Difference Method Test
Price
50
Rate
0.1
Strike
50
Grid M
20
TTM
0.5
Grid N
20
Vol
0.4
BS Call Price
6.790188
FD Exp Call
6.175100
Execute
FD Imp Call
6.651971
Close

◆ 程式

```
using System;
using System.Windows.Forms;

using DFinNR;

namespace TestFinNR
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {
            Application.Exit();
        }
```

```
        private void button2_Click(object sender, EventArgs e)
        {
            double S = double.Parse(textBox1.Text);
            double K = double.Parse(textBox2.Text);
            double time = double.Parse(textBox3.Text);
            double sigma = double.Parse(textBox4.Text);
            double r = double.Parse(textBox5.Text);

            int no_S_steps = int.Parse(textBox6.Text);
            int no_t_steps = int.Parse(textBox7.Text);

            double bs_price = BSMOption.option_price_call_black_scholes(S, K, r, sigma, time);
            textBox8.Text = bs_price.ToString("F6");

            double fd_exp_price = FiniteDifference.option_price_call_european_finite_diff_explicit(
                    S, K, r, sigma, time, no_S_steps, no_t_steps);
            textBox9.Text = fd_exp_price.ToString("F6");

            double fd_imp_price = FiniteDifference.option_price_call_european_finite_diff_implicit(
                    S, K, r, sigma, time, no_S_steps, no_t_steps);
            textBox10.Text = fd_imp_price.ToString("F6");
        }
    }
}
```