Stock Price Prediction in Bursa Malaysia

Nurfadhlina bt Abd Halim[A)], Goh Khang Wen[B)]

[A,B)]Jabatan Matematik, Fakulti Sains dan Teknologi, Universiti Malaysia Terengganu (UMT),
21030 Kuala Terengganu, Terengganu Malaysia.
[A)]lina@umt.edu.my, [B)]goh_khangwen@yahoo.com

Abstract

Investment in stock is a highly risky investment, it is because the existence of randomness in the stock price. In lecture, usually we used Binomial model to price the stock. But, in real world, how do we price the stock? Because the stock price is random, the volatility and drift is a crucial items to behold. The main questions is how to calculate this volatility and drift, and the answer to the question is the sample variance and the sample mean. At any time, the stock price will be either up or down from the previous price. This is where we need a method or model to calculate parameters for up-state and down-state for the stock price. And it will cover the volatility and the drift in an embrace. The method we used in this paper is the Hull-White algorithm. Hull-White algorithm is to find the parameters value of u and d for prediction to stock price. Using SPSS, we will run the data to get the sample variance and sample mean. Then, using Maple 10, we calculate the u and d before enter the value of u and d into programming C++.

Keywords: Random, Binomial model, volatility, drift, sample variance, sample mean, Hull-White algorithm, parameter u and d.

1. Introduction

For a long time, investment in stock being an interesting subject to talk on. Investors eager to scrape in profit from investment in stock. But, investment in stock open to a very high risk because of the randomness in the stock itself. So, many method and model built to predict the performance and the price of the stock and one of it is Hull-White algorithm. Hull-White algorithm will be used in this paper.

Usually, to price a stock we will use a Binomial model. But, in the real world, there are two important things to look on when we talk about stock price. That is the volatility and the drift of the stock. But, the main question is how to value this two things. The volatility is value using the sample variance and the drift is value using sample mean. When we obtain the value of sample variance and the value of sample mean, we must find value for u and d because stock price is random and we can't give an absolute price for the stock. All we can give is the expected price for the stock and the probability that price will occur on specified time. The variable u is weight for the up-state and the variable d is weight for down-state in stock price. Once the variable u and d are known, we then run it in C++ programming to obtain the stock price at time t. The data we used is the stock price listed on the main board of Bursa Malaysia and in consumer product sector.

2. Materials and Methods

Mathematical models are intended to help us understand the present and predict the future. If the model succeed they must harmonize with the real world [Stampfli et al., 2000]. To be able to create a model, we must understand the object itself and start with the simplest method. We start with the Binomial model. The stock price have the up-state and down-state, so in binomial model we get,
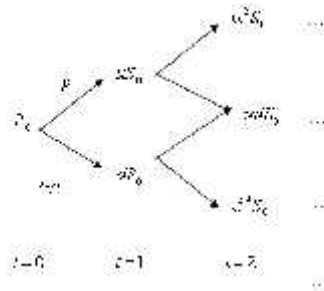
Figure 1. Binomial model of stock price

with p is the probability on the up-state branch and 1-p is the probability on the down-state branch. The value for p and 1-p are same on all branch.

In the real world, what is p, u and d? We need to match this parameters to the important components in the stock's behavior. The two components are drift and volatility. If we regard the stock price after a brief time period, Δt, as a random variable, S, then the relative return is the ratio $\dfrac{S - S_0}{S_0}$ (Stampfli et al., 2000).

The drift parameter, μ, measures the average percentage change in the stock price over time.

$$\mu \Delta t = E\left[\frac{\Delta S}{S_0} \right] \tag{1}$$

The $\mu \Delta t$ is the average relative return, so $1 + \mu \Delta t$ is the mean of the stock ratio.

The volatility parameter, σ, measure the randomness of the relative return. The variance is

$$\sigma^2 \Delta t = E\left[\left(\frac{S}{S_0} - 1 - \mu \Delta t \right)^2 \right] \tag{2}$$

Then, take a look at Bernoulli random variable X. For the Bernoulli random variable X, it is

The mean = $E[X] = pa + (1-p)b$ (3)

and

The variance = $E\left[(X - \mu)^2 \right] = p(1-p)(a-b)^2$ (4)

Next apply information in equation (3) and (4) to Figure 1. Then, according to (1) and (2),

$$1 + \mu \Delta t = E\left[\frac{S}{S_0} \right] = pu + (1-p)d$$

and

$$\sigma \sqrt{\Delta t} = \sqrt{ E\left[\left(\left(\frac{S}{S_0} \right) - 1 - \mu \Delta t \right)^2 \right] } = \sqrt{p(1-p)}\,(u - d)$$

Now, we take a look at Hull-White algorithm. We need to set $p = \dfrac{1}{2}$ and determine u and d from;

(i) $\dfrac{u+d}{2} = 1 + \mu \Delta t$

(ii) $u - d = 2\sigma \sqrt{\Delta t}$

Second step is to assume

$$S_1 = X_1 S_0$$

and

$$S_{k+1} = X_{k+1} S_k$$

where X_k are independent Bernoulli random variables with

$$P\left[\frac{S_k}{S_{k-1}} = u\right] = P\left[\frac{S_k}{S_{k-1}} = d\right] = \frac{1}{2}.$$

From (4), the estimators of $\mu\,\Delta t$ and $\sigma^2\,\Delta t$ are

$$\overline{U} = \frac{1}{n}\sum_{k=1}^{n}\left(\frac{S_k}{S_{k-1}} - 1\right)$$

and

$$s^2 = \frac{1}{n-1}\left[\sum\left(\frac{S_k}{S_{k-1}} - 1\right)^2 - n\overline{U}^2\right]$$

$\overline{U}$ and s^2 are the sample mean and sample variance.

So, μ and σ is $\mu \approx \dfrac{\overline{U}}{\Delta t}$ and $\sigma \approx \dfrac{s}{\sqrt{\Delta t}}$. Then, using this estimates, replacing μ and σ into equation (i) and (ii). The solution is

$$\begin{aligned} u &= 1 + \mu\,\Delta t + \sigma\sqrt{\Delta t} \\ d &= 1 + \mu\,\Delta t - \sigma\sqrt{\Delta t} \end{aligned} \qquad (5)$$

$$(6)$$

3. Case Study

This study is done based on the data from Bursa Malaysia. That is the stock price listed on the main board of Bursa Malaysia and in consumer product sector. The data is from 1 March 2007 until 31 March 2007. The data as given in Table 1 below.

Table 1. Closing stock price in Bursa Malaysia

Date	Closing Price (RM/unit)	Ratio
1 March 2007	7.20	0.965278
2 March 2007	6.95	1.000000
5 March 2007	6.95	0.928058
6 March 2007	6.45	0.961240
7 March 2007	6.20	1.024194
8 March 2007	6.35	1.062992
9 March 2007	6.75	0.970370
12 March 2007	6.55	0.984733
13 March 2007	6.45	1.023256
14 March 2007	6.60	0.984848
15 March 2007	6.50	1.000000
16 March 2007	6.50	1.007692
19 March 2007	6.55	1.015267
20 March 2007	6.65	1.030075
21 March 2007	6.85	1.021898
22 March 2007	7.00	1.000000
23 March 2007	7.00	1.000000
26 March 2007	7.00	0.978571
27 March 2007	6.85	1.000000

28 March 2007	6.85	0.912409
29 March 2007	6.25	

Source : Bursa Malaysia 2007.

The first step is to find the sample mean and sample variance for the data. Using SPSS 10, we get the sample mean and the sample variance as shown on Table 2.

Table 2   Descriptive Statistics

	N	Minimum	Maximum	Mean	Std. Deviation
Stock price ratio	20	0.912409	1.062992	0.99354405	0.035013130
N	20				

From Table 2, we find the value for $\overline{U}$ and s. Using equation (5) and (6), we find the value for variable u and d. We get as follows;

$$\overline{U} = 0.99354405 - 1 = -0.00645595 \quad \text{and} \quad s = 0.18711795$$

with $\Delta t = 1$ and $S_0 = \text{RM}6.25$

$$u = 1 + \overline{U}\Delta t + s\,\Delta t$$
$$d = 1 + \overline{U}\Delta t - s\,\Delta t$$

So,

$$u = 1 + (-0.00645595)(1) + (0.18711795)(1) = 1.180662$$
$$d = 1 + (-0.00645595)(1) - (0.18711795)(1) = 0.8064261$$

The next step is to enter the u, d and S_0 value into C++ programming. Using C++, we manage to calculate the prediction price for the stock and the probability it will occur without illustrate the stock price tree.

The C++ programming is as shown below;

```
#include <iostream>
#include <string>
#include <cstdlib>
#include <cmath>
#include "lib.h"
using namespace std ;

void main()
{
    int n;
    double S, u, d, p;

    cout << "Nilai S awal, S = " ;
    cin >> S ;
    cout << endl ;

    cout << "Kadar Kenaikan, u = " ;
    cin >> u ;
    cout << endl ;

    cout << "Kadar Penurunan, d = " ;
    cin >> d ;
    cout << endl ;
```

```
        cout << "Kebarangkalian Kenaikan, P = " ;
        cin >> p;
        cout << endl ;

        cout << "Jumlah Peringkat, t = " ;
        cin >> n ;
        cout << endl ;

        algo(S,u,d,p,n);
```

and as follows;

```
// deklarasi.h

#include <iostream>
#include <iomanip>

using namespace std ;

double* vektor(int n)

{
        double* B = NULL ;
        int i ;

   B = new double [n] ;
        for (i=0 ; i<n ; i++)
            B[i] = 0.0 ;
        return B ;
}

void algo(double S, double u, double d, double p, int n)
{
        int i;
        double* A;
        double* kebarangkalian;
        A = vektor(n+1);
        A[1] = S;
        kebarangkalian = vektor(n+1);
        kebarangkalian[1] = 1.0;

        cout                                                                    <<
"======================================================================="
<< endl;
        cout << "t = 0" << endl;
        cout << "---------" << endl;
        cout << "X[0] = " << A[1] << endl;

        for(i=1;i<=n;i++)
        {
                int j,k;
                double* B;
                double* semen;
                B = vektor(i+1);
                semen = vektor(i+1);

                cout                                                            <<
"======================================================================"    <<
endl;
```

```
cout << "t = " << i << endl;
cout << "---------" << endl;
for(j=1;j<=i;j++)
{
        if(j==1)
        {
                B[j] = A[j]*u;
                B[j+1] = A[j]*d;
                semen[j] = kebarangkalian[j]*p;
                semen[j+1] = kebarangkalian[j]*(1-p);
        }
        else
        {
                B[j+1] = A[j]*d;
                semen[j+1] = kebarangkalian[j]*(1-p);
        }
}
A = B;
kebarangkalian = semen;
for(k=1;k<=i+1;k++)
{
        cout <<"X[" << k << "]= " << A[k];
        cout << "                    ";
        cout << "Kebarangkalian = " << kebarangkalian[k] << endl;
}
    }
}
```

The result show the stock price on 2 April 2007 is RM8.71227, RM5.95073 or RM4.06452 with the probability it will occur is 0.25.

4. Discussion

The Hull-White algorithm provides a simple method to predict the stock price and the result is expectable base on the result we get from the calculation and the programming. The nature of shares itself is random and unpredictable, it is quite difficult to predict the stock price and make an investment on it. But, using this method, investors hopefully is enable to gain more profit and be able to manage their investment well.

5. Reference

Oksendal, B., 2003. Stochastic Differential Equations : An Introduction with Applications, New York.

Ladde, G.S., Sambandham, M., 2004. Stochastic Versus Deterministic Systems of Differential Equations, New York.

Stampfli, J., Goodman, V., 2000. The Mathematics of Finance : Modeling and Hedging, United States.

http://biz.thestar.com.my/marketwatch (Access on 1 March 2007)