Option Valuation in Practice

A **Master Thesis** *presented*
by
Denada Prifti
(3080976)
to
Prof. Dr. Wolfgang Härdle
Institute for Statistics and Econometrics

in partial fulfillment of the requirements
for the degree of

Master of Statistics

Humboldt-Universität zu Berlin
Spandauer Str. 1
D-10178 Berlin

Berlin, August 26, 2002

Declaration of Authorship

I hereby confirm that I have authored this master thesis independently and without use of others than the indicated resources.

All passages, which are literally or in general matter taken out of publications or other resources, are marked as such.

Denada Prifti

Berlin, August 26, 2002

Abstract

This thesis aims to introduce some fundamental concepts underlying option valuation theory involving interactive implementation of computational tools within XploRe. The outlined theory consists of four major parts: analysis of asset price dynamics, valuation models for plain vanilla options in continuous- and discrete-time, sensitivities and implied volatilities. Several numerical examples and plots have integrated both software application and theory.

Thanks to

Wolfgang Härdle (Professor at Institute for Statistics and Econometrics of Humboldt-Universität zu Berlin), Siegbert Klinke, Matthias Fengler, Torsten Kleinow, Oliver Blaskowitz and Karel Komorad (all Institute for Statistics and Econometrics of Humboldt-Universität zu Berlin), Gerhard Stahl and Petra Haferkorn (German Financial Supervisory Authority) for help and advice.

Contents

1 Introduction

Option valuation is a major accomplishment of modern finance. It spurred the development and widespread use of familiar financial options, such as puts and calls in common assets, as well as exotic options. This thesis reflects both option valuation theory and practice. It aims to provide an accessible and interactive approach with which to explore the theory of option valuation, sensitivities and implied volatility, integrated with user-friendly software.

The accessibility to users of various backgrounds is facilitated by organizational design. After a brief summary within each section, the software application is presented. The main focus hereby is a comprehensive coverage of implementation with XploRe procedures, so called quantlets. The related theory is thereafter outlined. Such an approach offers users, who are confident with finance, to bypass the theory and focus directly on computational tools in software application. For didactic purposes, a quantlet is repeated in different subsections, when designed to carry out several tasks. For example, the quantlet `european`() is used either to analytically calculate the price of a European option using the Black-Scholes formula or to calculate its implied volatility, and is therefore presented both in subsections 3.3.1 and 6.1.1.

The basis of any option valuation model involves a description of the stochastic process followed by the underlying asset. Therefore this thesis begins with the analysis of asset price dynamics in section 2. Turbulence in financial markets, which cause increases in volatility, are captured by continuous-time stochastic processes. Shocks, such as market crashes, which cause discontinuity on the observed underlying price, are incorporated within a jump-diffusion process.

Two well-known approaches to option valuation are presented and illustrated through computational examples: the continuous-time Black-Scholes model and the binomial tree model.

The Black-Scholes model has become the basic benchmark for pricing equity and commodity options. It is also used in modified form to price Eurodollar future options, foreign currency, Treasury bond options, caps and floors.

Section 3 firstly describes the classical Black-Scholes model for pricing European options both on dividend and non-dividend underlying stocks. It then extends to the class of quadratic approximation methods for pricing American options within the Black-Scholes world.

Section 4 discusses the binomial tree model, which provides discrete approximation to the continuous process underlying the Black-Scholes model. It basically solves the same equation, using a numerical procedure, as opposed to an analytical approach. In doing so, it offers opportunities along the way to check for early exercise of American options. This is the advantage that the binomial model has compared to the classical Black-Scholes model. The binomial model is very important, because it shows how to get away from a reliance on closed form solutions and is a means of valuing options that relies on simple, fast and accurate numerical methods.

The benefit of option valuation models, is not necessarily to provide the "right" price. The best pricing method adopted until now is the market price - an efficient market will provide the best and the truest price for options. The true benefits of option valuation models are that they provide an accurate "snap shot" of the current market conditions, and more importantly, they break the option market price into each of the factors that comprises it. Thus each factor can be examined separately and its contribution to the determination of the option price assessed. Relying on option valuation models, it is possible to predict how the market price of the option should change, when for example the key factor - volatility -, or any other factor changes. The sensitivities of the option price to these factors are known as Greeks. Section 5 addresses these sensitivities and their importance to option traders.

An additional benefit of option valuation models is that they can be used to extract information from option markets. For example, extracting market expectations from options data is the purpose of calculating implied volatilities from option prices. Implied volatility is the risk perceived by the market today and is built into the time value of the option premium. It is therefore a crucial indicator that traders observe to assist them in assessing the value of an option. However, when the Black Scholes formula is inverted to imply volatilities from option market prices, the volatility estimates vary with both strike and expiration. The convex shape of the implied volatility with respect to differing strike values, is referred to as the smile effect. Section 6 explores implied volatilities, their computation and the smile effect. This section then continues with discussing implied tree theories extending the Black-Scholes theory, making it consistent with the smile. Implied trees are constructed in order for local volatility to vary from node to node, resulting in a flexible tree, where the market price of all standard options can be matched.

2 Asset Price Dynamics

The dynamics of asset prices are reflected by uncertain movements of their values over time. Cuthbertson (1996, ch. 5) and Wilmott et al. (1997, ch. 2) state that the efficient market hypothesis (EMH) is one possible reason for the random behavior of the asset price. In spite of its different forms, the EMH basically states that past history is fully reflected in present prices and markets respond immediately to any new information about the asset. These two assumptions imply that changes in the asset price are a Markov process.

In this context, modelling the asset price is concerned with modelling the arrival of new information, which affects the price. Depending on the appearance of the so called "normal" and "rare" events, there are two basic blocks in modelling the continuous-time asset price. Neftci (2000, ch. 8) states that the main difference between the "normal" and the "rare" behavior concerns the size of the events and their probability to occur. As the interval of observation (h) gets smaller, the size of "normal" events also gets smaller and it becomes unimportant as $h \to 0$. Even in a short time interval, there is always a non-zero probability that some non-noticeable news will arrive. In contrast to a "normal" event, when a "rare" event (or shock) occurs, the value of the random variable can change significantly over a short period of time. An example of a "rare" event is a market crash, such as the one that occurred in 1987. In other words, as $h \to 0$, the probability of a "rare" event moves to zero, but its size may not shrink.

If markets are dominated by "normal" events, then a Brownian motion process can be used. This is a continuous-time stochastic process, where extremes occur only infrequently according to the probabilities in the tails of the normal distribution. The continuous-time diffusion (Brownian motion-based) process is written in form of the following stochastic differential equation for the asset return:

$$\frac{dS_t}{S_t} = \mu\, dt + \sigma\, dW_t \tag{2.1}$$

with

$$\begin{aligned} S_t &= \text{the current price of the underlying asset,} \\ \mu &= \text{the constant trend or drift,} \\ \sigma &= \text{the constant annualized volatility,} \\ W_t &= \text{the standard Wiener process.} \end{aligned}$$

These asset returns follow the so called geometric Brownian motion. Expression (2.1) is called a differential equation, because the asset price S_t is only defined implicitly by describing its changes through time. Since this process has a continuous-time sample path, it does not allow for discontinuities or jumps in its values when "rare" events occur. In this case, the Poisson jump process can be useful. In particular, the time series of the asset price can be modelled as the sum of the continuous-time diffusion process and Poisson jump processes. XploRe uses the Merton jump-diffusion model to simulate and estimate simultaneously the "normal" and "rare" events in the asset price. The stochastic differential equation for S_t is:

$$\frac{dS_t}{S_t} = \mu dt + \sigma dW_t + d\sum_{j=1}^{N_t}(Y_j - 1) \tag{2.2}$$

with the following addition of variables to (2.1):

$$\begin{aligned} Y_j - 1 &= \text{a log-normal distributed random variable representing the jump size,} \\ N_t &= \text{jumps (shocks) in the interval (0,t) governed by a Poisson process with parameter } \lambda t, \\ d &= \text{constant.} \end{aligned}$$

Jump-diffusion models undoubtedly capture a real phenomenon that is missing from the Black-Scholes models. Yet, they are rarely used in practice. There are three main reasons for this, difficulty in parameter estimation, solution and impossibility of perfect hedging (Wilmott, 1999, ch. 26).

2.1 Software Application

XploRe offers the following three quantlets to simulate and estimate the price of the underling asset according to (2.1) and (2.2):

```
stocksim()

stocksim(S,sigma,tau,rate,shocks,jumps,jumpvola)
     simulates a random process for the stock price,
     specifying either interactively or directly the
     input parameters.

stockest(data)
     estimates the parameters of the Brownian motion
     process or the Merton jump-diffusion process for
     the given dataset.

stockestsim(data)
     estimates the parameters of the Brownian motion
     process or the Merton jump-diffusion process for
     the given data and uses simulation to compare both
     models with the real dataset.
```

The quantlet `stocksim` provides three means to simulate the asset price as a random process by using i) the geometric Brownian motion (2.1), ii) a compounded Poisson jump process with the lognormal distribution of the jump height and iii) a mixture of both, namely the Merton jump-diffusion model (2.2).

If a direct approach is preferable, then the following input parameters have to be defined: `S` - the starting value of the underlying asset, `sigma` - the volatility of the asset return for the continuous-time diffusion process, `tau` - time (days) to expiration, `rate` - shocks per days, `jumps` - the expected number of jumps, which corresponds to λ in the Poisson process and `jumpvola` - the volatility of the jump height. The following example simulates the asset price by directly giving the values of the input parameters.

```
library("finance")
 S=200           ; starting value of the underlying asset
 sigma=10        ; annualized volatility in percentage
 tau=200         ; days to expiration
 rate=5          ; increasing rate of return
 shocks=2        ; number of shocks per day
 jumps=20        ; the expected number of jumps
 jumpvola=0.5    ; volatility for the height of jump
 stocksim(S,sigma,tau,rate,shocks,jumps,jumpvola)
```

stocksim.xpl

The quantlet returns a display as output (Figure 2.1), where all three processes are plotted together.

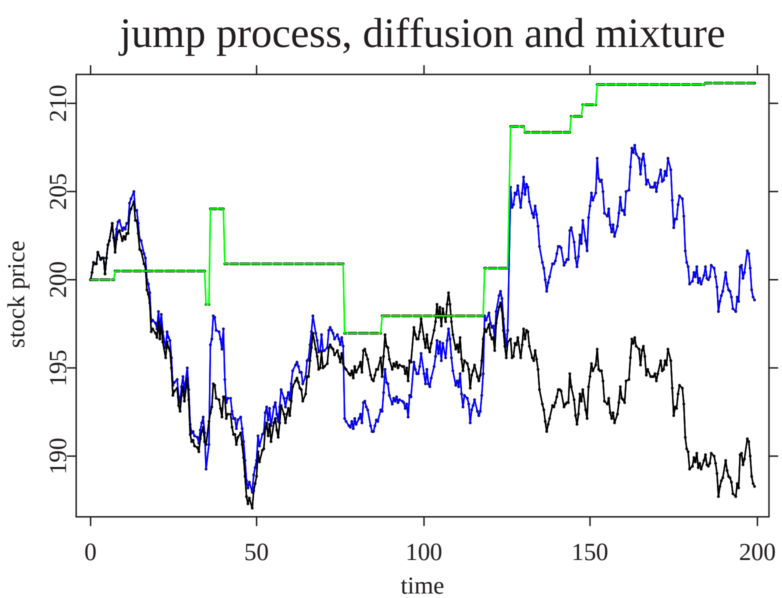

Figure 2.1: Simulation of the stock price:
—— Brownian motion based process
—— Poison process
—— mixed process (Merton jump-diffusion)

In the example above, the stock price is simulated at 400 points. The number of discretization points `n` is calculated as a product of days to expirations `tau` and `jumps` (shocks) per day. The jump consists of two random variables: one for the shock arrival process, which is Poisson distributed with parameter λ and the other one for the jump size, which is log-normal distributed with volatility `jumpvola`. A jump is detected if $\left(\frac{\lambda}{n} \geq y\right)$, where y is a random variable with a standard uniform distribution, i.e. $y \sim U[0,1]$. For the mixed process, the geometric Brownian motion is simulated with an overlaying Poisson process.

For a given dataset, the quantlet `stockest` estimates the parameters of a random process, when the asset returns are assumed to follow either a geometric Brownian motion, as in (2.1), or a mix of the geometric Brownian motion and of a compounded Poisson Jump Process, namely the Merton jump-diffusion process given in (2.2). The following illustrates `stockest` on the data for Motorola stock prices.

```
library("finance")
 data=read("motorola")    ; read the data
 data=data[,2]            ; select the stock prices in the
                          ; second column
 stockest(data)
```

`stockest.xpl`

The output window contains the following information:

```
Contents of _tmp.mue
[1,]   7.0066

Contents of _tmp.sigma
[1,]   44.191

Contents of _tmp.lambda
[1,]        4

Contents of _tmp.mue2
[1,]   3.2302

Contents of _tmp.sigma2
[1,]   38.819

Contents of _tmp.jump
[1,]     10.9
```

When no jumps are assumed, the estimated parameters for μ and σ are 7.0066 and 44.191, respectively. The intensity of the Poisson process is

$\lambda = 4$. When the jump-diffusion process is assumed, then the estimated values for μ, σ and the jump volatility are 3.2302, 38.819 and 10.9, respectively.

The quantlet `stockestsim` is a combination of both `stocksim` and `stockest`. First, the parameters for the given dataset are estimated for both, the Brownian motion based and the compounded Poisson jump process. Then both models are compared with the real dataset by means of a simulation. `stockestsim` is illustrated in the following example. The results are graphically displayed in Figure (2.2).

```
library("finance")
 data=read("motorola")     ; read the data
 data=data[,2]             ; select the stock prices in the
                           ; second column
 stockestsim(data)
```

stockestsim.xpl

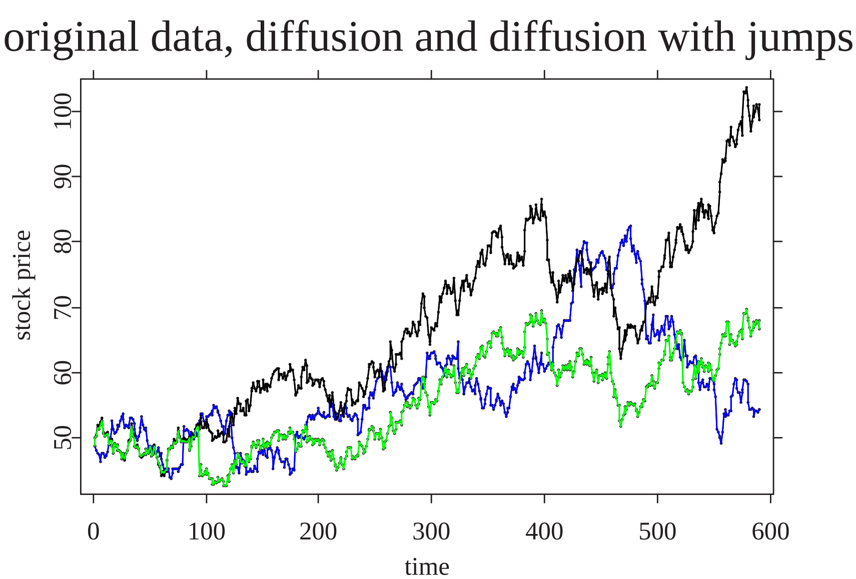

Figure 2.2: Estimation and simulation of the stock price:
— Brownian motion based process
— mixed process (Merton jump-diffusion)
— the original data

2.2 Asset Price as a Stochastic Process

2.2.1 Geometric Brownian Motion

The stochastic process for asset price movement, when only "normal" events are considered, is assumed to be continuous in time, so that analytical tools, such as stochastic calculus can be employed. In general, the asset price is assumed to follow a Markov process, which means that only the present value of the asset price is relevant for the future price movements. This feature is consistent with the weak form of market efficiency, which assumes that the present value of the asset price already involves all information contained in past prices and history is irrelevant.

A special case of a Markov process is the Brownian motion. The Brownian motion with drift is defined as a stochastic process $\{X_t; t \geq 0\}$ with the following properties:

(i) for $t > 0$ and $s > 0$, every increment $X_{t+s} - X_s$ is normally distributed with mean μt and variance $\sigma^2 t$, where μ and σ are fixed parameters,

(ii) for every $t_1 < t_2 < ... < t_n$, the increments $X_{t_2} - X_{t_1}, ..., X_{t_n} - X_{t_{n-1}}$ are independent random variables with distribution given in (i),

(iii) $X_0 = 0$ and the sample paths of X_t are continuous.

Note that $X_{t+s} - X_s$ is independent of history, i.e. knowing X_τ, $\tau < s$ has no effect on the probability distribution of $X_{t+s} - X_s$. This is exactly the Markovian property for the Brownian motion. For the particular case, when $\mu = 0$ and $\sigma^2 = 1$, the Brownian motion is called standard Brownian motion (or standard Wiener process) denoted as W_t.

Fluctuations in asset price can be explained by using the following stochastic differential equation, also known as Ito's process:

$$dS_t = m(t, S_t)\, dt + \sigma(t, S_t)\, dW_t, \tag{2.3}$$

where

$$\begin{aligned} S_t &= \text{the current underlying asset price,} \\ m(t, S_t) &= \text{the trend or drift,} \\ \sigma(t, S_t) &= \text{the annualized instantaneous volatility of the underlying asset,} \\ W_t &= \text{the standard Wiener process.} \end{aligned}$$

Different assumptions about the form of volatility give rise to different solutions for S_t to this stochastic differential equation. Classical models, such as the Black-Scholes option price model (section 3), assume that $\sigma(t, S_t) = \sigma S_t$ and $\mu(t, S_t) = \mu S_t$. In this case, the asset return (dS_t/S_t) follows a so called geometric Brownian motion, written as:

$$\frac{dS_t}{S_t} = \mu\, dt + \sigma\, dW_t \tag{2.4}$$

The first term μdt represents the deterministic return within a short interval dt, where μ is the average growth rate of the asset price. The second term σdW_t, where σ is assumed to be constant, takes into account the random changes of the asset price to external effects, such as unexpected news.

Brownian motion and normal distribution have been widely used in the Black-Scholes option pricing framework (section 3). However, two puzzles have emerged from several empirical investigations and are subject to current research. These are the leptokurtic distribution of asset returns and evidence on volatility smile (section 6).

A distribution is leptokurtic, if it has a higher peak and heavier tails than those of the normal distribution. One possible reason for fat tails can be the discontinuous path of the asset returns. To reflect this discontinuity, jump-diffusion processes have been widely used to model financial time series. The jump-diffusion model gives higher values than the Black-Scholes model for deep out-of-the-money and in-the-money options, especially when the time to maturity is short. The main reason for this is that a jump-diffusion process results in a distribution with fatter tails than the normal distributed returns (log-normal asset price). For low values of gamma, the jump-diffusion model gives similar results as the Black-Scholes model.

2.2.2 The Jump-Diffusion Model

Merton (1976) was one of the first, who applied Poisson jumps to normal Brownian motion process, in order to approximate the movement of stock prices subject to occasional discontinuous breaks. In this section the concept behind the jump-diffusion model is described following Merton (1976), and the later works of Jiang (1998) and Neftci (2000, ch. 8 and ch. 7.2).

In the jump-diffusion model, changes in the asset price are a mixture of normal events that occur in continuous fashion and of rare events, which are modelled as jumps that occur sporadically. The continuous component of the change in the asset price is a Wiener process. The jump component is a Poisson-driven process. It is assumed that the arrivals of rare events are independently, identically distributed (i.i.d.). The probability of a jump that occurs during a time interval of length h is written as

$$\text{Prob}\{\text{the event occurs in } (t, t+h)\} = \lambda h,$$

$$\text{Prob}\{\text{the event does not occur in } (t, t+h)\} = 1 - \lambda h,$$

where λ is the intensity of the jump process, i.e. the number of jumps per unit of time, which do not depend on the information set available at time t. It can be calculated by dividing the corresponding probability λh by h.

The jump process can be modelled using a Poisson counting process, which has the following properties:

1. as $h \to 0$, at most one event can occur with probability very close to 1,
2. the information up to time t does not help to predict the occurrence (non-occurrence) of the event in the next instant h,
3. the events occur at a constant rate λ.

Note that the time interval between two successive jumps is exponentially distributed with parameter λ. Given that the rare event occurs in the time interval $(t, t+h)$ causing a jump, the asset price S_{t+h} at time $t+h$ will be the random variable $S_{t+h} = S(t)Y$. The discontinuous change in the asset price will be $S_{t+h} - S(t) = S(t)(Y-1)$. The random variable $Y-1$, also called the jump size, gives the percentage change in the stock price, if the Poisson event occurs. In general, $\tilde{Y}$ may be a random variable. It is assumed that the successive jump sizes $(\tilde{Y}-1)$ are i.i.d.

One aspect of the jump should once again be highlighted. The process has two sources of randomness. The occurrence of a jump is a random event. But once the jump occurs, the size of the jump is also random. Moreover, it is assumed that these two sources of randomness are independent of each other. Under this structure, the general parametric jump-diffusion process, as a mixture of both, continuous diffusion path and discontinuous jump path, can be written in general form as (Jiang, 1998):

$$\frac{dS_t}{S_t} = \{\alpha_t(\beta) - \lambda\mu_0\}dt + \sigma_t(\beta)dW_t + dQ_t(\lambda) \tag{2.5}$$

where

$$\begin{aligned}
S_t &= \text{asset price at time t} \\
\alpha_t &= \text{the instantaneous expected return} \\
\mu_0 &= \text{expectation of the relative jump size, i.e. } \mu_0 = E[Y_t - 1], \\
\lambda &= \text{the intensity parameter of the Poisson distribution,} \\
\sigma_t &= \text{the instantaneous volatility of the asset's return} \\
&\quad\ \text{conditional on the Poisson jump event not to occur,} \\
W_t &= \text{a standard Wiener process,} \\
Q_t(\lambda) &= \text{a Poisson process with parameter } \lambda, \\
(\beta, \mu_0, \lambda) \in \Theta &= \text{the parameter space, which parametrizes the} \\
&\quad\ \text{coefficient functions, the jump sizes, as well as the} \\
&\quad\ \text{intensity of the Poisson process.}
\end{aligned}$$

Both, the Wiener process W_t and the Poisson process $dQ_t(\lambda)$ are infinitely divisible in time and appropriated scaled; $dQ_t(\lambda)$ and dW_t are statistically independent. By definition μ_0 is the mean jump size. Thus, the expected change in S_t from the jump component dQ_t over the time interval dt is $\lambda\mu_0 dt$. Therefore, if α_t denotes the total expected return (rate of change) on S_t, $\lambda\mu_0 dt$ needs to be subtracted from the drift term of S_t:

$$\begin{aligned}
E\left(\frac{dS_t}{S_t}\right) &= E\{(\alpha_t - \lambda\mu_0)dt\} + E(\sigma_t dW_t) + E(dQ_t) \\
&= (\alpha_t - \lambda\mu_0)dt + 0 + \lambda\mu_0 dt = \alpha_t dt
\end{aligned} \tag{2.6}$$

Note, that a sample path S_t for a process described by equation (2.5) will be continuous most of the time, but has finite jumps of different signs and sizes at discrete points in time. Assuming α_t and σ_t are constant, so that the continuous component of $S(t)$ is lognormally distributed, and conditional upon there being N_t jumps in the time interval $(0, t)$, the asset price at time t can be written as

$$S_t = S_0\, exp\left(\alpha_t - \frac{1}{2}\sigma_t^2 - \lambda\mu_0\right) t + \sigma W_t \tilde{Y}(N_t), \tag{2.7}$$

where $W_t \sim N(0, t)$ and $\left\{\tilde{Y}_j\right\}_{j=1}^{n}$ is a set of i.i.d. jumps, such that:

$$\begin{cases} \tilde{Y}(N_t) = 0 & \text{for} \quad N_t = 0 \\ \tilde{Y}(N_t) = \prod_{j=1}^{N_t} Y_j & \text{for} \quad N_t \geq 1 \end{cases}$$

Special cases of this model include those by Press (1967) with $\sigma_t(\cdot) = 0$ and $\mu_t(\cdot) = 0$, Merton (1976) with $\sigma_t(\cdot) = \sigma$, $\mu_t(\cdot) = \mu$ and lognormal jumps, Lo

(1998) with $ln(Y_t) = f(s_t)$, i.e. the jump size is determined by the process itself, and Ornstein-Uhlenbeck process with $\mu_t(\cdot) = \mu s_t$ and exponentially decaying jumps (see Gourieroux (2001, pp. 249-253) and Küchler et al. (1997, pp. 27-28) for the explanation of the Ornstein-Uhlenbeck process).

In XploRe the geometric Brownian motion with i.i.d. lognormal jumps is used to simulate and estimate the asset price in a mixed process. This method was proposed by Merton (1976) and is now known as the Merton's jump-diffusion Model. It assumes that in equation (2.6) $\mu_t(\cdot) = \mu$, $\sigma_t(\cdot) = \sigma$, and $lnY_t \sim$ i.i.d. $N(\mu_0, \nu^2)$, where all μ, μ_0, σ, ν are constant. Then the stochastic differential equation for S_t is written as:

$$\frac{dS_t}{S_t} = \mu dt + \sigma dW_t + d\sum_{j=1}^{N_t}(Y_j - 1) \tag{2.8}$$

where d is constant. The term $\sum_{j=1}^{N_t}(Y_j - 1)$ is also known as a compounded Poisson process (Küchler et al., 1997, pp. 9). Equation (2.8) can be equivalently written as:

$$S_t = S_0\, exp\left(\mu dt + \sigma dW_t\right)\prod_{j=1}^{N_t} Y_j \tag{2.9}$$

with $\left(\prod_{j=1}^{N_t} Y_j\right) = 0$ if $N_t = 0$ and $\left(\prod_{j=1}^{N_t} Y_j\right) \neq 0$ if $N_t \geq 1$, where Y_j are i.i.d. and N_t is a Poisson distributed process with parameter λt.

3 The Black-Scholes Model

In 1973, the Chicago Board of Options Exchange begun trading options in exchanges, although previously options had been regularly traded by financial institutions in over the counter markets. In the same year, Black and Scholes (1973), and Merton (1973), published their seminal papers on the theory of option pricing. Since then the growth of the field of derivative securities has been phenomenal. In recognition of their pioneering and fundamental contribution to option valuation, Scholes and Merton received in 1997 the Award of the Nobel Prize in Economics. Unfortunately, Black was unable to receive the award since he had already passed away.

In essence, the Black-Scholes model states that by continuously adjusting the proportions of stocks and options in a portfolio, the investor can create a riskless hedge portfolio, where all market risks are eliminated. The ability to construct such a portfolio relies on the assumptions of continuous trading and continuous sample paths of the asset price. In an efficient market with no riskless arbitrage opportunities, any portfolio with a zero market risk must have an expected rate of return equal to the risk-free interest rate. This approach led to the differential equation, known in physics as the "heat equation". Its solution is the Black-Scholes formula for pricing European options on non-dividend paying stocks:

$$C(S,t) = S\Phi(d_1) - K\,exp(-r\tau)\Phi(d_2), \tag{3.1}$$

$$P(S,t) = K\,exp(-r\tau)\Phi(-d_2) - S\Phi(-d_1), \tag{3.2}$$

where

$$d_1 = \frac{ln(S/K) + (r + \sigma^2/2)\tau}{\sigma\sqrt{\tau}},$$

$$d_2 = \frac{ln(S/K) + (r - \sigma^2/2)\tau}{\sigma\sqrt{\tau}} = d_1 - \sigma\sqrt{\tau}.$$

$C(S, t)$ = price of the European the call option,
$P(S, t)$ = price of the European put option,
S = current underlying asset (stock) price,
K = strike price,
τ = $T - t$ is the current annualized time-to-expiration, where T is the expiration date,
r = the annualized risk-free interest rate,
σ = the annualized standard deviation of underlying asset price,
Φ = the cumulative distribution function for a standardized normal variable.

3.1 Software Application

The option price according to the Black-Scholes formula can be calculated with XploRe. First, the functions in library `finance` must be loaded by typing the command:

```
library("finance")
```

There are mainly two ways for computing the option prices according to (3.1) and (3.2) in XploRe. One way is by giving the requested input parameters through interactive menus and the other way is by directly defining the values of the input parameters within the quantlet.

```
opc = BlackScholes(S,K,r,sigma,tau, task)
     calculates the European option price on
     non-dividend paying underlying asset specifying
     interactively the input parameters.

{opvv,sel,ingred} = bs1(1)

opvv = bs1(S,K,r,sigma,tau,opt,1)
     calculates European option prices on non-dividend
     paying underlying assets, specifying either
     interactively or directly the input parameters.
```

`BlackScholes` only calculates the European option price for a non-dividend paying underlying asset, whereas `bs1` calculates the European option price for different kinds of dividends and is therefore more general. The input parameter of the interactive form of `bs1` has a value of 1. This means that the underlying asset pays no dividends. Other values of this input parameter, in the case of dividend paying underlying asset, are explained in detail in "Software Application" in subsection 3.3. When the direct specification of the input parameters is selected, `bs1(S,K,r,sigma,tau,opt,1)`, the first five follow the usual notation: `S` for the current level of the underlying asset, `K` for the strike price, `r` for the continuously compounded risk-free interest rate, `sigma` for the instantaneous standard deviation of underlying asset and `tau` for time to expiration. `task` is a scalar, which specifies the type of option. For `task` = `1`, a European call is selected and for `task` = `0`, a European put. The computation result is assigned to the variable `opc` when `BlackScholes` is used and `opvv` when `bs1` is used. The other two output variables `sel` and `ingred` contain input parameters (see "Software Application" in subsection 3.3, pp. 28).

For example, the command `BlackScholes(100,120,0.05,0.25,0.5,1)` yields the call price of $C(S,t) = 1.952$, and the command

```
BlackScholes(100,120,0.05,0.25,0.5,0)
```

yields the put price of $P(S,t) = 18.898$.

bs1.xpl

BlackScholes.xpl

3.2 Derivation of the Black-Scholes Formula

In the following, the basic assumptions and methodology originally employed to derive the option price will briefly be described. The Black-Scholes assumes the following:

- The asset price follows a geometric Brownian motion (see also subsection 2.2.1)

$$\frac{dS_t}{S_t} = \mu\, dt + \sigma\, dW_t. \tag{3.3}$$

- Trading can take place continuously without any transaction costs or taxes.

- Short selling is permitted and the assets are perfectly divisible. Therefore assets can be sold that are not own and any number (not necessarily an integer) of the underlying assets can be bought and sold.
- The continuously compounded risk-free interest rate is constant.
- Investors can borrow or lend at the same risk-free rate of interest.
- There are no riskless arbitrage opportunities. It follows that all risk-free portfolios must earn the same return.

For notational convenience, the following uses S for the stock price at time t, i.e. $S_t = S$. Suppose that the value of the option V depends on the underlying asset S and time t. The idea is to construct a portfolio, which involves short selling of one unit of the European option and holding of Δ units of underlying stock S. The value of this portfolio Π is given as:

$$\Pi = -V + \Delta\, S. \tag{3.4}$$

The change in the value of this portfolio in one time-step is

$$d\Pi = -dV + \Delta\, dS, \tag{3.5}$$

where Δ is held fixed during the time-step. Since both V and Π are random variables, Ito's lemma is applied to compute their stochastic differentials. The stochastic differential for the option is written as

$$dV = \sigma S \frac{\partial V}{\partial S}\, dW + \left(\mu S \frac{\partial V}{\partial S} + \frac{1}{2}\sigma^2\, S^2 \frac{\partial^2 V}{\partial S^2} + \frac{\partial V}{\partial t} \right) dt, \tag{3.6}$$

where it is required that $\frac{\partial V}{\partial t}$, $\frac{\partial V}{\partial S}$ and $\frac{\partial^2 V}{\partial S^2}$ exist. This expression provides the random walk followed by V. By placing (3.3) and (3.6) in (3.5) together, it follows

$$d\Pi = \sigma S \left(-\frac{\partial V}{\partial S} + \Delta \right) dW + \left\{ -\frac{\partial V}{\partial t} - \frac{\sigma^2}{2} S^2 \frac{\partial^2 V}{\partial S^2} + \left(-\frac{\partial V}{\partial S} + \Delta \right) \mu S \right\} dt.$$

If Δ is chosen equal to $\frac{\partial V}{\partial S}$, then the portfolio becomes a riskless hedge, since the stochastic term dW in the portfolio disappears. In an efficient market with no riskless arbitrage opportunities, any portfolio with a zero market risk, also a perfectly hedged portfolio, must earn the risk-free interest rate. The return on an amount Π invested in riskless assets would face a growth of $r\Pi\, dt$ in the intervall dt. Hence, it follows

$$d\Pi = -\left(\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2\, S^2 \frac{\partial^2 V}{\partial S^2} \right) dt = r\Pi\, dt = r \left(-V + S \frac{\partial V}{\partial S} \right) dt \tag{3.7}$$

and after rearranging terms,

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + rS\frac{\partial V}{\partial S} - rV = 0 \tag{3.8}$$

is obtained. This is the Black-Scholes partial differential equation. The solution of this equation with different auxiliary conditions, such as boundary and final conditions, provides pricing formulae for different types of derivative securities. For example the call option final condition is:

$$C(S,T) = max(S-K,0)$$

and the boundary conditions are:

$$C(0,t) = 0, \qquad \lim_{S\to\infty} C(S,t) = S$$

The Black-Scholes formulae for pricing a European call $C(S,t)$ and a European put $P(S,t)$ on non-dividend paying stocks are (3.1) and (3.2). Technical details on how to solve equation (3.8) with the auxiliary conditions can be found in Hull (2000, ch. 11) or Wilmott et al. (1997, pp. 75-80). The price of an American call on a non-dividend paying underlying asset is equivalent to its European counterpart, since such an American call will not be optimally exercised prior to expiration (Hull, 2000). Hence the Black-Scholes pricing formula (3.1) is also valid for pricing American calls.
Note, that the price of a European put option on a non-dividend paying asset (3.2) is derived by combining the call option price formula (3.1) and the **put-call parity** under the continuous-time assumption:

$$P(S,t) = C(S,t) - S + Kexp(-r\tau) \tag{3.9}$$

Kwock (1998, pp. 52) provides a probabilistic interpretation of the option pricing formulae. For example, under the assumption of risk-neutrality the call option price formula (3.1), $\Phi(d_2)$ is seen as the probability of the call option being in-the-money at expiration. Hence, $K\Phi(d_2)$ is the risk neutral expectation of the payment made by the holder of the call option at expiration by exercising the option. On the other hand, $Sexp(r\tau)\Phi(d_1)$ is the risk neutral expectation of the asset price at expiration, conditional on the call being in-the-money. It follows that the expectation of the call value at expiration is $Sexp(-r\tau)\Phi(d_1) - K\Phi(d_2)$. This is (in the risk neutral world) discounted by the factor $exp(-r\tau)$ in order to obtain the present value of the call price.

3.3 Options on Dividend Paying Assets

The dividend received by holding an asset may be stochastic or deterministic. The modelling of stochastic dividends is complicated, since there is another random variable in addition to the underlying asset. The Black-Scholes formula, however, requires only some slight modification to remain valid under the crucial assumption that the dividend yields are deterministic. This means that over the remaining time to expiration, the option dividends are at most a known function of time and/or of the underlying asset. This assumption is not unrealistic given the short-term life of trade options (generally less than one year) and given the stable dividend policy that most companies tend to follow over a short horizon. The prices of European call and put options on continuously dividend paying underlying asset, noted as q, are:

$$C(S,t) = Sexp(-q\tau)\Phi(d_1) - K\,exp(-r\tau)\Phi(d_2), \tag{3.10}$$

$$P(S,t) = K\,exp(-r\tau)\Phi(-d_2) - Sexp(-q\tau)\Phi(-d_1), \tag{3.11}$$

where

$$d_1 = \frac{ln(S/K) + (r - q + \sigma^2/2)\tau}{\sigma\sqrt{\tau}},$$

$$d_2 = d_1 - \sigma\sqrt{\tau}.$$

The Black-Scholes pricing formulae show that a European option, on an underlying asset S paying continuous dividend yields at rate q, has the same value as the corresponding European option on an underlying asset with the price $Sexp(-q\tau)$ that pays no dividend.

3.3.1 Software Application

The quantlets `bs1`, `european` and `optstart` in XploRe can be used to price European options within the Black-Scholes framework, when dividends exist according to (3.10) and (3.11) either directly or through interactive menus.

```
{opvv,sel,ingred} = bs1(typeofdiv)

{opvv} = bs1(S,K,r,sigma,tau,opt,typeofdiv,div)
     calculates European option prices, specifying
     either interactively or directly the input
     parameters.

european()
     calculates the prices of European options, or their
     implied volatilities, specifying interactively the
     input parameters.

optstart()
     calculates the prices of either European or
     American options, or their implied volatilities,
     specifying interactively the input parameters.  For
     American options the McMillan formula or binomial
     trees can be used.
```

`european` calculates either the option price by using the quantlet `bs1`, or the implied volatility by using the quantlet `volatility`. The quantlet `optstart` uses several interactive menus to compute either i) the price of an American option through the McMillan formula (subsection 3.4.2) or through binomial trees (subsection 4.1), or ii) the price of a European option using the the Black-Scholes formula, or iii) the implied volatilities (section 6).

The input parameter `typeofdiv` in `bs1` is an integer that specifies the type of the dividend payment: for `typeofdiv`=1 no dividend, for `typeofdiv`=2 continuous dividends and for `typeofdiv`=3 a fixed dividend at the end of T is assumed. Finally, if `typeofdiv`=4, then an exchange rate is assumed as underlying. The call and put formulae for the foreign currency options are analog to (3.10) and (3.11), except that the dividend yield q is replaced by the foreign interest rate. In this case, S is replaced by the exchange rate - the domestic currency price of a unit foreign currency -

that is assumed to follow the lognormal diffusion process. In addition, both domestic and foreign interest rates are assumed to be constant. In bs1(S, K, r, sigma, tau, opt, typeofdiv, div), the first five parameters follow the usual notation. The parameter opt specifies the option type. It has the value 1 for a call and 0 for a put. The parameter div refers to the dividend amount. It must be given the value zero, if no dividend is assumed.

When the quantlet bs1 is used interactively, the output variable dat is a scalar containing the computed option price. The option type is denoted through a (2x1) dimensional vector sel: sel=1|0 for a European call and sel=0|1 for a European put. The third variable ingred is a (6x1) dimensional vector that contains six input parameters: the price of the underlying asset, the strike price, the time to expiration, the volatility of the underlying asset, the domestic risk-free interest rate and the dividend payment(s).

bs1(S, K, r, sigma, tau, opt, typeofdiv, div) offers the possibility to compute simultaneously the price of more than one European call or put option, with or without dividends. The output variable opvv is a (nx9) dimensional matrix, where the first columns contain the input parameters and the last one contains the computed option prices.

In order to use (3.10) and (3.11), the quantlet bs1 converts the fixed dividend D to a constant continuous dividend yield q through

$$q = \frac{S(1+r)}{S(1+r)-D} - 1.$$

A simple example in which the underlying asset pays continuous dividend, i.e. typeofdiv=2, is given as in the following

```
library("finance")
 bs1(2)                ; continuous dividend payment
```

bs1.xpl

with default values (230, 210, 5, 25, 0.5) for (S, K, r, σ,τ) and with $q = 15\%$, yields a call price of 20.024. The same result is achieved by computing a call option price with a fixed dividend payment of $D = 31.5$, and default values (230, 210, 5, 25, 0.5) for (S, K, r, σ,τ), by typing the following commands:

```
 library("finance")
 bs1(3)                 ; a fixed amount of dividend payments
```

bs1.xpl

The following XploRe code reads the input parameters and calculates directly the prices of two calls and three puts:

```
library("finance")
 S=aseq(230,5,10)         ; additive sequence of five underlying
 K=aseq(210,5,15)         ; additive sequence of the strike prices
 r=5                      ; the annualized risk-free interest rate
                          ; in %
 sigma=aseq(25,5,-5)      ; additive sequence of volatility
 tau=0.5                  ; annualized time to expiration
 opt=#(1,1,0,0,0)         ; options type: two calls and three puts
 typeofdiv=#(0,1,2,3,2)   ; type of dividend payments
 dividend=#(0,10,35,8,45); the value of the dividend payments
 dat= bs1(S,K,r,sigma,tau,opt,typeofdiv,dividend)
 dat.opvv                 ; displays the results
```

bs1.xpl

yielding the following output matrix `dat.opvv`:

```
Contents of opvv

[1,]   230   210   0.04879   0.25   0.5   1   0   0          30.986
[2,]   240   225   0.04879   0.20   0.5   1   1   0.09531    17.8
[3,]   250   240   0.04879   0.15   0.5   0   2   0.1431     10.632
[4,]   260   255   0.04879   0.10   0.5   0   3   0.076961    6.392
[5,]   270   270   0.04879   0.05   0.5   0   2   0.17284    15.991
```

Starting from the left, each column contains for each option the underlying price, the strike price, the continuous risk-free interest rate, the volatility, the time to expiration, the option type, the type of dividend, the dividend payment and the computed option price.

3.3.2 Derivation of the Formula

The concept of pricing European options on dividend paying underlying assets will now be briefly outlined.

The constant continuous dividend yield is represented by $q = q(S,t)$. In other words, it is the dividend payment per unit of time, which always represents the same fraction q of the stock price. The holder then receives dividend payment(s) equal to $qSdt$ within the interval dt. As the price of the underlying asset falls by the amount of the dividend, the asset price dynamics based on the geometric Brownian motion model becomes

$$\frac{dS(t)}{S} = (\mu - q)\, dt + \sigma\, dW(t). \tag{3.12}$$

For every asset held $qSdt$ is received. The holder of the portfolio, who holds Δ assets, earns an amount equal to $\Delta\Pi$ and dividend payment(s) equal to $qS\Delta dt$ in the interval dt. The change in value of the portfolio Π is given by

$$\begin{aligned} d\Pi &= -dV + \Delta dS + q\Delta S dt \\ &= -\left(\frac{\partial V}{\partial t} - \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + qS\frac{\partial V}{\partial S}dt\right) dt, \end{aligned} \tag{3.13}$$

where $\Delta = \frac{\partial V}{\partial S}$. The last term $qS\Delta dt$ denotes the wealth added to the portfolio due to the dividend yields. By applying the no arbitrage argument, the hedged portfolio should earn the risk-free interest rate, so that

$$-\left(\frac{\partial V}{\partial t} - \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + qS\frac{\partial V}{\partial S}dt\right) dt = r\left(-V + S\frac{\partial V}{\partial S}\right) dt. \tag{3.14}$$

This leads to the following modified form of the Black-Scholes equation:

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + (r - q)\, S\frac{\partial V}{\partial S} - rV = 0 \tag{3.15}$$

For a call option, the only change to the boundary conditions is

$$\lim_{S\to\infty} C(S,t) = Sexp(-q\tau).$$

The final condition is still $C(S,T) = max(S - K, 0)$. Without working through a similar integration procedure, the price of a European option can be obtained by a simple modification of the Black-Scholes price formula. The European option on an underlying asset with price S paying continuous dividend yields at rate q has the same value as the corresponding European option on an underlying asset with the price $Sexp(-q\tau)$ that pays no dividend. This yields the option valuation formulae (3.10) and (3.11).

Note, that if the underlying asset is a commodity, like grain or livestock, there may be additional costs in holding the asset, such as storage or insurance. In simple terms, these additional costs, denoted for example with `u`, can be considered as negative dividends paid by the underlying asset. In this case, the option price is equivalent to (3.10) and (3.11), where the continuous compound dividend `q` is simply replaced by $-$`u`. The term $r + u$ is interpreted as the cost of carry, denoted with b. It consists of two parts, the costs of funds tied up in the asset that require interest for borrowing and additional costs due to storage, insurance etc. When the underlying asset pays a continuous dividend q, then the cost of carry b equals $r - q$, and the Black-Scholes differential equation (3.8) is written as

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + bS\frac{\partial V}{\partial S} - rV = 0. \qquad (3.16)$$

3.4 Valuation of American Options

An American option confers all the rights of its European counterpart plus the privilege of early exercise. Since this additional privilege should not be worthless, it has potentially a higher value than its European counterpart. The extra cost is usually called early *exercise premium*. An exception to this is an American call option on non-dividend paying underlying asset, whose price always equals its European counterpart. For other American options it may be optimal to exercise them prior to expiration. For an American call or put, this only happens when the price of the underlying asset at a given time to expiration τ, rises above (falls below) a critical asset value $S^*(\tau)$, known as *the optimal exercise price*. The collection of these critical values for all times constitutes a curve, which is known as *the optimal exercise boundary*. It is not a known priori where to apply the boundary condition, so that the optimal exercise boundary has to be determined in the solution process.

The early exercise premium can be expressed in terms of the exercise boundary in a stochastic integral. A detailed explanation is given in Kwock (1998, ch. 4) and Wilmott et al. (1997, ch. 7). The direct solution of the stochastic integral equation is in many cases unmanageable, so that several analytic approximation methods for the valuation of American options and the associated optimal exercise boundaries have been developed.

In `XploRe` the price of the American option is computed by using the popular quadratic approximation method, which was first proposed by MacMillan (1986) for non-dividend paying stock options and later extended to commodity options by Barone-Adessi and Whaley (1987). This class of approximation methods involves the reduction of the Black-Scholes partial differential equation to an ordinary one. The idea is to transform the Black-Scholes differential equation (3.8), so that the temporal derivative term can be considered as a quadratic small term and then dropped as an approximation (Kwock, 1998, pp. 166). Then applying the boundary conditions, the prices of American call and put options follow.

The price of an American call for $b \geq r$ equals the price of its European counterpart (see subsection 3.4.1 for an example). The price of an American call for $b < r$ is:

$$C_{am}(S,t) = \begin{cases} C(S,t) + A_2 \left(\frac{S}{S^*}\right)^{\gamma_2} & when \quad S < S^* \\ S - K & when \quad S \geq S^* \end{cases} \tag{3.17}$$

$C_{am}(S,t)$ = price of the American call option,
$C(S,t)$ = price of the European counterpart,
S = current underlying stock price,
S^* = critical price of the underlying stock, above which the call should be exercised.

The price of an American put option is given in (3.18) and holds for all values of b.

$$P_{am}(S,t) = \begin{cases} P(S) + A_1 \left(\frac{S}{S^{**}}\right)^{\gamma_2} & when \quad S > S^{**} \\ K - S & when \quad S \leq S^{**} \end{cases} \tag{3.18}$$

$P_{am}(S,t)$ = price of the American put option,
$P(S,t)$ = price of the European counterpart,
S = current underlying stock price,
S^{**} = critical price of the underlying stock, below which the put should be exercised.

S^* and S^{**} are estimated by solving the following equations (3.19) and

(3.20) iteratively

$$S^* - K = C(S^*, t) + [1 - exp\{(b-r)\tau\}\ \Phi\{d_1(S^*)\}]\ \frac{S^*}{\gamma_2} \tag{3.19}$$

$$K - S^{**} = P(S^{**}, t) - [1 - exp\{(b-r)\tau\}\ \Phi\{-d_1(S^{**})\}]\ \frac{S^{**}}{\gamma_1} \tag{3.20}$$

The other variables used in (3.19) and (3.20) are

$$\gamma_1 = \frac{1}{2}\left\{-(\beta-1) - \sqrt{(\beta-1)^2 + \frac{4\alpha}{h}}\right\} \tag{3.21}$$

$$\gamma_2 = \frac{1}{2}\left\{-(\beta-1) + \sqrt{(\beta-1)^2 + \frac{4\alpha}{h}}\right\} \tag{3.22}$$

$$A_1 = -\frac{S^{**}}{\gamma_1}\,[1 - exp\{(b-r)\tau\}\ \Phi\{-d_1(S^{**})\}] \tag{3.23}$$

$$A_2 = \frac{S^*}{\gamma_2}\,[1 - exp\{(b-r)\tau\}\ \Phi\{d_1(S^*)\}] \tag{3.24}$$

$$d_1(S) = \frac{ln(S/K) + (b + \sigma^2/2)\tau}{\sigma\sqrt{\tau}}, \tag{3.25}$$

where $b = r - q$ is the cost of carry and the variables α, β are defined as in (3.27). This method is considered to be quite efficient and to have a reasonably good level of accuracy for valuing American options, particulary for valuing options with a short time to expiration. Other approximation methods, such as the interpolation between bounds, or the approximation of the optimal exercise boundary by a known analytical (e.g. exponential) curve (see Kwock ch.4), or direct numerical approaches by using finite difference algorithms, or Monte Carlo simulations (see Hull ch. 14 and Wilmott ch. 8 to 10), are not discussed.

3.4.1 Software Application

XploRe offers the following quantlets to calculate the price of american options using the MacMillan approximation:

```
mcmillan(eopv,sel,task,ingred)
     calculates the American option price, specifying
     directly the input parameters.

american()
     calculates the American option price, specifying
     interactively the input parameters.

optstart()
     calculates the prices of either European or
     American options, or their implied volatilities,
     specifying interactively the input parameters.  For
     American options the McMillan formula or binomial
     trees can be used.
```

The price of American options can be calculated directly with the quantlet `mcmillan` or through interactive menus by using `american`. The third quantlet `optstart` is more general. It uses several interactive menus to compute the price of American or European options, or their implied volatility. However, in spite of the different comfortability that quantlet `american` or `optstart` offers, both use the quantlet `mcmillan` to calculate the price of American options.

The parameter `eopv` specifies the price of the European option. The option type is specified through the (2x1) dimensional vector `sel`, with `sel`=1|0 for American call and `sel`=0|1 for put. The third variable `ingred` is a (6x1) dimensional vector that contains six input parameters: the price of the underlying asset, the strike price, the time to expiration, the annualized volatility of the underlying asset, the annualized risk-free interest rate and the dividend payment(s). The following example computes the price of an American option on non-dividend paying underlying asset.

```
library("finance")
 eopv=12.70          ; price of the European call
 sel=1|0             ; specifying an American call
 task=1              ; no dividend payments
```

```
ingred=230|231|0.3|0.25|0.05|0.00
mcmillan(eopv,sel,task,ingred)
```

mcmillan.xpl

american.xpl

optstart.xpl

The XploRe output shows the expected result, that an American call on a non-dividend paying underlying asset will have the same price as its European counterpart:

```
Contents of aus
 [1,] " "
 [2,]"--------------------------------------"
 [3,] " The Price of Your American Call-Option "
 [4,] " on Given Stock is "
 [5,]"12.7000"
 [6,]"--------------------------------------"
 [7,] " "
```

A second example computes the price of an American option when the underlying is a commodity that involves continuous annualized costs of around 5% of the commodity. The annualized risk-free interest rate is $r = 5\%$, so that the cost of carry is $b = r - q = 10\%$.

```
library("finance")
eopv=12.70       ; price of the European call
sel=1|0          ; specifying an American call
task=2           ; continuous costs, e.g. storage or insurance
ingred=230|231|0.3|0.25|0.05|-0.05
mcmillan(eopv,sel,task,ingred)
```

mcmillan.xpl

When $b > r$, the American call will never be exercised, since the present value of all dividend payments until options expiration date is less than the present value of the interest rate that can be earned on the strike price of the call during its remaining time to expiration. In this case, the value of the American call simply equals its European counterpart, as is confirmed in the XploRe output.

```
Contents of aus
 [1,] " "
 [2,]"--------------------------------------"
 [3,] " The Price of Your American Call-Option "
 [4,] " on Given Commodity with cont. Costs is"
 [5,]"12.7000"
 [6,]"--------------------------------------"
 [7,] " "
```

When $b < r$ the value of the American call is given in (3.17). It is always higher than the price of its European counterpart, as illustrated in a third example.

```
library("finance")
 eopv=12.70          ; price of the European call
 sel=1|0             ; specifying an American call
 task=2              ; continuous dividend payments
 ingred=230|231|0.3|0.25|0.05|0.05
 mcmillan(eopv,sel,task,ingred)
```

mcmillan.xpl

The Xplore output is:

```
Contents of aus
 [1,] " "
 [2,]"--------------------------------------"
 [3,] " The Price of Your American Call-Option "
 [4,] " on Given Stock with cont. Dividends is "
 [5,]"12.7373"
 [6,]"--------------------------------------"
 [7,] " "
```

3.4.2 Derivation of the Formula

The quadratic approximation method will be briefly outlined following Hull (2000, pp. 432-434) and Kwock (1998, pp. 174-177). Consider an American option on a stock, paying continuous dividends at rate q. The early exercise premium, defined by $v(S,t)$, is

$$v(S,t) = C_{am}(S,t) - C(S,t),$$

where $C_{am}(S,t)$ is the value of an American call option and $C(S,t)$ is its European counterpart. Within the continuation region both $C_{am}(S,t)$ and

$C(S,t)$ satisfy the Black-Scholes differential equation (3.16). It follows that $v(S,t)$ also satisfies and can therefore be written as

$$\frac{\partial v}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 v}{\partial S^2} + b\, S \frac{\partial v}{\partial S} = rv, \tag{3.26}$$

where $b = r - q$ is the cost of carry. By writing

$$\begin{aligned} \alpha &= \frac{2r}{\sigma^2}, \\ \beta &= \frac{2b}{\sigma^2}, \\ \tau &= T - t, \\ h(\tau) &= 1 - exp(-r\tau) \end{aligned} \tag{3.27}$$

and defining

$$v(S,\tau) = h(\tau) f(S,h),$$

equation (3.26) can be transformed into the following form

$$S^2 \frac{\partial^2 f}{\partial S^2} + \beta S \frac{\partial f}{\partial S} - \frac{\alpha}{h}\left\{ f - (1-h)h\frac{\partial f}{\partial h} \right\} = 0. \tag{3.28}$$

The approximation used involves assuming that the last term $(1-h)h\frac{\partial f}{\partial h}$ equals zero, so that (3.28) is reduced to an ordinary differential equation with the error being controlled by the quadratic term $(1-h)h$. This last term, which is ignored, is fairly small. When τ is large, it moves to zero and when τ is small, $\frac{\partial f}{\partial h}$ will be nearly zero. Then the approximative equation will be reduced to

$$S^2 \frac{\partial^2 f}{\partial S^2} + \beta S \frac{\partial f}{\partial S} - \frac{\alpha}{h} f = 0, \tag{3.29}$$

where h is assumed to be non-zero. When h is treated as a parameter, equation (3.29) becomes a non-homogeneous second order differential equation and can be solved with the standard techniques. After applying boundary conditions, the valuation formulae for the American call and put options (3.17) and (3.18) follow respectively.

4 The Binomial Pricing Model

The binomial pricing model arises from discrete random walk models of the underlying asset. This method is only a reasonable approximation of the evolution of the stock prices when the number of trading intervals is large and the time between trades is small (Jarrow and Turnbull, 1996, pp. 213). It is particularly useful for pricing American options numerically, since it can deal with the possibility of early option exercise. An exact analytical solution with the Black-Scholes model for the American options is not possible, because of the complexity of the boundary conditions (see subsection 3.4).

The binomial model breaks down the time to expiration of an option into potentially very large number of time intervals, or steps. A tree of stock prices is initially produced, moving forward from the present to expiration. At each interval, the asset price S can branch upwards to the value Su (Figure 4.1) or downwards to the value Sd, by an amount calculated using the volatility and time to expiration. A binomial distribution of prices, for the underlying asset is thus produced. The tree represents all the possible paths that the stock price can take during the life of the option.

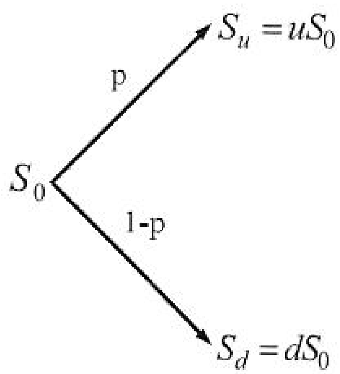

Figure 4.1: A single step of a binomial tree at time t_0 with respective probabilities p and q, where $p + q = 1$.

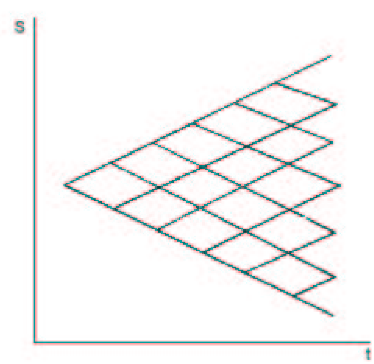

Figure 4.2: CCR binomial tree. (Derman, Kani and Chriss, 1996)

At the end of the tree, i.e. at expiration of the option, the option values for each possible stock price are known, as they are equal to their intrinsic values. Assuming that the payoff function of the option is determined only by the value of the underlying asset at expiration, the model then works backwards through each time interval, calculating the option value at each step. The final step is at current time and stock price, where the theoretical fair value of the option is calculated. This recursive pricing procedure is based on the assumption of risk neutrality. In a risk neutral world all individuals require no compensation for risk, so that the option can be priced as though the underlying asset's expected return is risk-free.

The most popular binomial tree is that from Cox, Ross and Rubinstein (1979), also known as the Cox-Ross-Rubinstein (CRR) binomial tree. In this approach, the underlying asset evolves along a risk-neutral binomial tree with constant logarithmic price spacing, corresponding to constant volatility, as illustrated in Figure (4.2).

The CRR binomial tree is a discrete version of the Black-Scholes constant volatility process. Any higher multinomial tree, for example a trinomial tree, can used as a discrete development of the geometric Brownian motion. However, all of them converge, as the time interval tends to zero, to the same continuous constant volatility process. The CRR tree is discussed in subsection (4.2) and illustrated in the following computational examples in XploRe.

4.1 Software Application

XploRe offers the folllowing quantlets to calculate European and American option prices with the Cox-Ross-Rubinstein binomial tree:

```
asset(vers)
     uses the quantlet bitree to calculate the option
     price and the price process of the underlying
     asset, specifying interactively the input
     parameters.

{s,ow,op} = bitree(vers,task)

{s,ow,op} = bitree(S,K,r,sigma,tau,n,vers,opt,typeofdiv,div)

     calculates the option price and the price process
     of the underlying asset using the CRR binomial
     tree.  The input parameters are specified either
     interactively or directly.

IBTcrr(S,K,r,sigma,level,delta,task)
     calculates the European option price on
     non-dividend paying underlying asset, specifying
     directly the input parameters.

optstart()
     calculates the prices of either European or
     American options, or their implied volatilities,
     specifying interactively the input parameters.  For
     American options the McMillan formula or binomial
     trees can be used.
```

`asset` opens different interactive menus for input parameters. It uses the quantlet `bitree` to price European and American options.

The input parameter `vers` in the quantlets `asset` and `bitree` specifies the type of option. It has the value 1 for a call and 0 for a put. `task` is a scalar that specifies the type of dividend payment(s): for `task`=1 no dividend, for `task`=2 a continuously paid dividend, for `task`=3 a dividend as a percentage of the value of the underlying asset and for `task`=4 a fixed dividend at the end of T is assumed. If `task`=5, then an exchange rate is assumed as underlying. In this case, S is replaced by the exchange rate, i.e. the domestic currency price of a unit foreign currency.

When the quantlet bitree is used interactively, the first five input parameters follow the usual notation: S for the price of the underlying asset, K for the strike price, r for the annualized risk-free interest rate in %, sigma for the annualized volatility in % and tau for time to expiration. The other parameters are: n for the number of intervals in the tree, opt for the type of option, which has the value 1 for a call (default) or 0 for a put. The input parameter typeofdiv specifies the type of dividend payments. It has the values 0 to 4 for the same cases as in task, with typeofdividend = 0 as default. If typeofdividend ≠ 0, then the value(s) of dividend(s) must be specified in div. For more than one dividend payment, div is a (m x 2) dimensional matrix, where the first column contains the time points when dividends should be paid and the second, the corresponding dividend values.

Both quantlets, asset and bitree output the tree of possible prices of the underlying asset, which is contained in a $(n+1)\times(n+1)$ dimensional matrix s, the tree of option prices, which is contained in a $(n+1)\times(n+1)$ dimensional matrix ow and the price of the option op.

In the following example, the European put price on a dividend paying underlying asset S is computed through quantlet betree:

```
library ("finance")
 S=100                ; spot price
 K=100                ; strike price
 r=10.517             ; annualized risk-free interest rate in %
 sigma=30             ; annualized volatility in %
 tau=1                ; time to expiration in years
 n=5                  ; number of intervals
 vers=0               ; European option
 opt=0                ; put
 typeofdiv=3          ; fixed dividend
 t=0.25|0.5|0.75|1    ; dividends paid quarterly
 f=10|10|10|15        ; dividend amounts
 div=t~f
 bitree(S,K,r,sigma,tau,n,vers,opt,typeofdiv,div)
```

bitree.xpl

The output contains the tree of possible stock prices, the tree of option prices and the computed option price:

```
Contents of _tmp
 [1,] " tree of possible stock prices "

Contents of s
 [1,]       100   93.583   77.945   62.968   48.548   29.596
 [2,]       100   109.16   91.568    74.88   58.965   38.705
 [3,]       100   100.85   109.38   90.459   72.588   50.617
 [4,]       100   100.85   91.568   110.83   90.403   66.196
 [5,]       100   100.85   91.568   82.148    113.7   86.569
 [6,]       100   100.85   91.568   82.148   72.588   113.21

Contents of _tmp
 [1,] " tree of option prices "

Contents of ow
 [1,]    33.197   41.725   49.935   57.386    64.18   70.404
 [2,]         0   27.201   36.318   45.477   53.765   61.295
 [3,]         0        0   20.481   29.903   40.144   49.383
 [4,]         0        0        0   13.252   22.331   33.804
 [5,]         0        0        0        0   6.0428   13.431
 [6,]         0        0        0        0        0        0

Contents of _tmp
 [1,] " the option price "

Contents of op
 [1,]   33.197
```

When binomial trees are used in practice, the life of the option is typically divided into 30 or more time steps, of length Δt. This computation can be easily carried out with XploRe. With 30 time steps, 31 possible stock prices and 2^{30}, or about one billion, possible stock prices are considered. The asset returns in one step of the tree, u and d, are chosen to match the stock price volatility. A popular way of doing this is by setting

$$u = exp(\sigma\sqrt{\Delta t}) \quad \text{and} \quad d = \frac{1}{u}, \tag{4.1}$$

as explained in subsection (4.2).

`IBTcrr` calculates the price of a European option on a non-dividend paying underlying asset. `level` specifies the number of intervals in the tree and `deltat` is the length of the discrete time interval. `task` is a scalar, which has the value 1 for call and 0 for put. The other parameters follow the usual notation. The output window shows the calculated European option price.

The same price results when the quantlet `betree` is used. The last quantlet is recommended for computation, since it yields not only the option price as in `IBTcrr`, but also the whole tree.

A second example illustrates how to price a European call with `IBTcrr`:

```
library ("finance")
 S=100          ; spot price
 K=100          ; strike price
 r=0.1057       ; annualized risk-free interest rate
 sigma=0.3      ; annualized volatility
 level=5        ; number of intervals
 deltat=0.20    ; length of time interval
 task=1         ; European call
 C=IBTcrr(S, K, r, sigma, level, deltat,task)
 C              ; call price
```

IBTcrr.xpl

Then the output window shows the European call price:

```
Contents of C
 [1,]   17.583
```

`optstart` asks the user to specify the model, which will compute the option price. It offers the Black-Scholes and the MacMillan formulae as an analytical approach and the binomial tree model as a numerical method. In the latter, the quantlet `bitree` is used for building the tree and pricing the option.

4.2 CRR Binomial Tree

The Cox, Ross and Rubinstein (CRR) binomial tree can be interpreted as a numerical procedure to solve the Black-Scholes equation. There are two main ideas underlying the tree. First, a continuous random walk (3.3) may be modelled by a discrete random walk with the following properties:

- The price of the underlying asset S changes only at discrete times $t_0 = 0$, $t_1 = \Delta t$, $t_2 = 2\Delta t$, ..., $t_n = n\Delta t$, ..., $t_N = N\Delta t = T$, where T is the expiration date of the option and $\Delta t = \frac{T}{N}$ denotes the one time step.

- If the price of the underlying asset is $S_{n,i}$ at state i and time t_n, then at t_{n+1} it may take only one of two possible values, $S_{n+1,i+1} = uS_{n,i} > S_{n,i}$ or $S_{n+1,i} = dS_{n,i} < S_{n,i}$ (Figure 4.3). This is equivalent to assuming that there are only two returns possible at each time step, $u - 1 > 0$ and $d - 1 < 0$. These two returns are the same for all time steps.
- The probability, p of $S_{n,i}$ moving up to $S_{n+1,i+1} = uS_{n,i}$ is known. The same results for the probability q of $S_{n,i}$ moving down to $dS_{n+1,i}$, since $p + q = 1$.

The second assumption underlying a binomial tree is that of a risk-neutral world, i.e. the investor risk preferences are irrelevant to option valuation. This has two implications. First, the expected return from all traded securities is the risk-free interest rate. This means that the drift term μ in the stochastic differential equation for the asset return (2.4) is replaced by the risk-free interest rate r whenever it appears

$$\frac{dS_t}{S_t} = r\,dt + \sigma\,dW_t. \tag{4.2}$$

Second, the option value V_n, at $t_n = n\Delta t$, is its expected value at $t_{n+1} = (n+1)\Delta t$, discounted by the risk-free interest rate r

$$V_n = E[exp(-r\Delta t)V_{n+1}]. \tag{4.3}$$

Within this framework, the probabilities p, q and the returns u, d should reflect the important statistical properties of the continuous random walk (4.2), which means that they have to insure that for $\Delta t \to 0$ the underlying asset S follows the Brownian motion. In other words, the parameters p, q, u, d should give the correct values for the mean and the variance of the underlying asset, i.e. $lnS_{n+1} \sim N\left(lnS_n + (b - \frac{\sigma^2}{2}\Delta t), \sigma^2\Delta t\right)$, during a time interval Δt. Consequently, these parameters must solve the following equations:

$$\begin{aligned} p + q &= 1 & (4.4)\\ E = pln(u\,S_n) + qln(d\,S_n) &= ln(S_n) + \left(b - \frac{\sigma^2}{2}\right)\Delta t & (4.5)\\ p\,(ln(u\,S_n) - E)^2 + q\,(ln(d\,S_n) - E)^2 &= \sigma^2\Delta t & (4.6)\end{aligned}$$

Substituting $q = p - 1$ in (4.5) and (4.6), there are three unknown parameters and two non-linear equations to solve. To obtain a unique solution, a supplementary restriction for the parameters is needed. Cox, Ross and Rubinstein (1979) chose the restriction $u\,d = 1$, since it drastically simplifies the tree. At time point t_n there are only $i = 1, ..., n+1$ possible nodes and

$$S_{n,i} = u^n d^{n-i} S_0, \tag{4.7}$$

where S_0 is the asset price in t_0 (Figure 4.3).

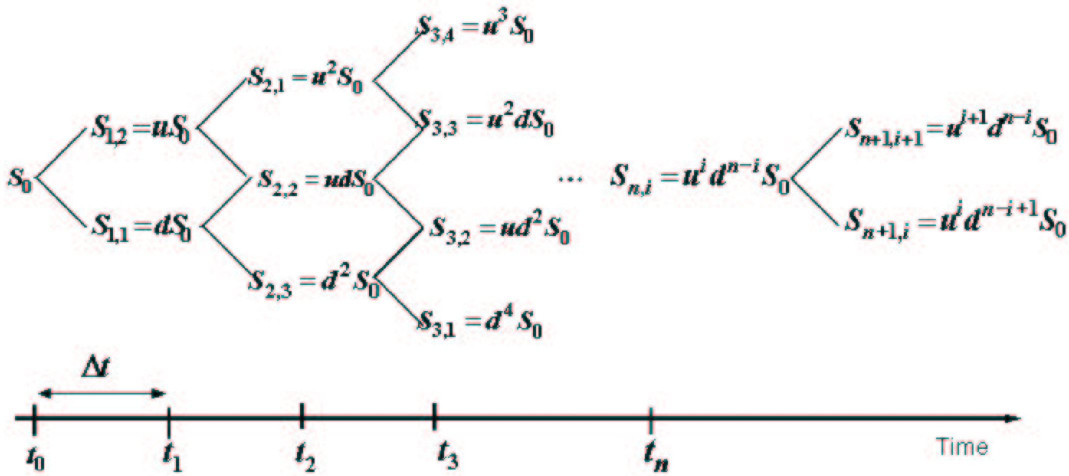

Figure 4.3: CRR binomial tree for S, with the restriction $ud = 1$

Solving the equations (4.4), (4.5) and (4.6) for p, u, and d and neglecting the terms smaller then Δt results in:

$$p = \frac{1}{2} + \frac{1}{2}\left(b - \frac{1}{2}\sigma^2\right)\frac{\sqrt{\Delta t}}{\sigma}, \quad u = exp(\sigma\sqrt{\Delta t}), \quad d = \frac{1}{u} \tag{4.8}$$

The time steps are of equal length, so that the risk-neutral probability p as calculated by 4.8 is the same at each node. The option price $V_{n,i} = V(S_{n,i}, t_n)$, at node i and time t_n, is the expected payoff at t_{n+1} discounted at the risk-free interest rate:

$$V_{n,i} = exp(-r\Delta t)\left[pV_{n+1,i+1} + (1-p)V_{n,i+1}\right] \tag{4.9}$$

At the end of the tree the option price is known. It equals the option value at expiration. For a call option, it is:

$$V_{N,i} = max\left\{0, S_{N,i} - K\right\}, \quad k = 0, ..., n \tag{4.10}$$

The option values at each node, $V_{n,i}$, $i = 0, ..., n$ and $n = N-1, ..., 0$ will be then determined recursively by working backwards through the tree.

When the underlying asset is a stock, which pays dividend(s), then the reduction of the stock prices by the dividend(s) amount must be considered. Details on the use of binomial trees for fixed or percentage dividend(s) are given in Franke et al. (2001, pp. 87).

In the case of an American put, or a European call on dividend paying underlying asset, the option price will be checked at each node to decide whether or not the early exercise would be optimal. If the option is held until expiration, its value at the final node is the same as for the European option. This is the case for an American call, since there is always the chance that until expiration the underlying price increases. Hence, the price of an American call equals the price of its European counterpart.

5 Greeks

The Black-Scholes formula for non-dividend paying underlying assets (3.1) show that there are essentially five parameters, which determine the option price: the current level of the underlying asset S_t, the strike price K, the continuously compounded risk-free interest rate r, the time to expiration τ and the instantaneous standard deviation σ of the underlying. The influence of these parameters on the option price can be investigated by using the quantlet `influence` from the library `finance` in XploRe:

```
dat = influence()

dat = influence(S,K,r,sigma,tau,carry,opt,pder,v1,ub1,v2,ub2)

     displays graphically the influence of the
     parameters, which enter the Black-Scholes formula,
     on the option price.  The input parameters are
     specified either interactively or directly.
```

This quantlet plots the option price against one or two input parameter(s), maintaining all other parameters constant. The sensitivity of the option price related to this parameter(s) gives additional information. This sensitivity can be represented in terms of a number, or indicator, generally referred to as Greek. By knowing numerical values for Greeks, it allows hedge positions using options to be set up. Greeks can be computed and plotted through the quantlet `greeks`.

```
dat = greeks()

dat = greeks(S,K,r,sigma,tau,carry,opt,pder,v1,ub1,v2,ub2)

     calculates and plots different sensitivities of the
     option price.  The input parameters are specified
     either interactively or directly.
```

When the direct specification of the input parameters is selected, then the first five parameters follow the usual notation: `S` for the current level of the underlying asset, `K` for the strike price, `r` for the continuously compounded risk-free interest rate, `sigma` for the instantaneous standard deviation of underlying asset and `tau` for time to expiration. The parameter `carry` denotes the annualized additional costs as a proportion of the price of the underlying asset. The parameter `opt` is a scalar defining the type of option. It has the value 1 for a call and 0 for a put. The parameter concerned is specified in `v1`. The type of sensitivity computed through the quantlet `greeks` is specified in `pder`. It has the value 1 for delta, 2 for gamma, 3 for eta, 4 for delta-k, 5 for vega, 6 for theta and 7 for rho.

The output is a two dimensional plot, which shows the dependence of the option price (when the quantlet `influence` is used) or of its sensitivity (when the quantlet `greeks` is used) on the specified parameter. If an additional parameter is specified in `v2`, a three dimensional plot with both parameters as explanatory variables is produced.

For graphical representation, the option price is computed within 30 discrete intervals of the explanatory variable(s). This is done mainly in two steps. Firstly, the quantlet `asset` is used to create a discrete grid of 31 points. To achieve this, the lowest and highest bounds for the parameter(s) are requested. The highest bound must be inputed into `ub1` (and into `ub2` in the case of two exploratory variables). The specified input value of the exploratory variable(s) is considered as the lowest bound. When both quantlets `influence` and `greeks` are used interactively, the user can freely decide which bound values to apply. Secondly, the option price is computed for each of the 31 different grid points using the Black-Scholes Formula. The results are presented in a two dimensional, or for two exploratory variables, in a three dimensional plot.

In the following, the sensitivity of the option price with respect to changes in one of the five parameters is analyzed: `S`, `K`, `r`, `tau` and `sigma`. Details on these sensitivities can be found in different financial sources, e.g. in the e-book Statistics of Financial Markets, ch. 7.3..

The following theoretical descriptions are based on Franke et al. (2001), Gibson (1991), Hull (2000), Kwock (1998) and Tompkinks (1994). To demonstrate how the option price and its sensitivity relates to the changes in the parameters above, the quantlets `influence` and `greeks` are used.

5.1 Delta

The delta (Δ) of a derivative security is defined as the rate of change of its price with respect to the price of the underlying asset. It is the slope of the curve that relates the derivative security price V to the price of the underlying S:

$$\Delta = \frac{\partial V}{\partial S}$$

Delta plays a crucial role in portfolio hedging. In the derivation of the Black-Scholes equation a covered call position is maintained by creating a risk-free portfolio, where the writer of a call sells one unit of the call and buys Δ units of the underlying.

The delta of a call (Δ_C) is always positive, as an increase in the asset price will increase the probability of a positive payoff at expiration resulting in a higher value. On the other hand, there is a negative relationship between the put price and the underlying asset price, as an increase in the asset price, will reduce the put's current exercise value $\{K exp(-r\tau) - S\}$ and therefore the put's price will decrease. This explains a negative Δ_P as given in (5.2).

When the price of the underlying asset changes, put and call option values move in opposite directions, since $(0 \leq \Delta_C \leq 1)$ and $(-1 \leq \Delta_P \leq 0)$. However, the absolute changes in their prices will never exceed those of the underlying asset.

Δ_C of a European call on a non-dividend paying underlying asset can be easily derived from the Black-Scholes formula (3.1):

$$\begin{aligned}
\Delta_C &= \Phi(d_1) + S\frac{1}{\sqrt{2\pi}}exp\left(-\frac{d_1^2}{2}\right)\frac{\partial d_1}{\partial S} - K\ exp(-r\tau)\frac{1}{\sqrt{2\pi}}exp\left(-\frac{d_2^2}{2}\right)\frac{\partial d_2}{\partial S} \\
&= \Phi(d_1) + \frac{1}{\sigma\sqrt{2\pi\tau}}\left[exp\left(-\frac{d_1^2}{2}\right) - exp\left\{-\left(r\tau + ln\frac{S}{K}\right)\right\}exp\left(-\frac{d_2^2}{2}\right)\right] \\
&= \Phi(d_1) > 0 \qquad (5.1)
\end{aligned}$$

The delta of a European put option is then derived from the put-call parity relation:

$$\Delta_P = \frac{\partial P}{\partial S} = \Delta_C - 1 = \Phi(d_1) - 1 = -\Phi(-d_1) < 0 \qquad (5.2)$$

In the following, it is shown through examples how the option price and its delta is calculated and plotted as a function of the underlying asset.

```
library("finance")
S=230               ; (spot) price of the underlying
K=210               ; exercise price
r=5                 ; the annualized risk-free interest rate in %
sigma=25            ; annualized volatility in %
tau=0.5             ; annualized time to expiration
carry=5             ; cost of carry
opt=1               ; call
v1=1                ; spot price as an explanatory variable
ub1=400             ; highest bound of the spot price
influence(S,K,r,sigma,tau,carry,opt,v1,ub1)

pdr=1               ; delta of the call
greeks(S,K,r,sigma,tau,carry,opt,pder,v1,ub1)
```

influence.xpl

greeks.xpl

The computation yields a (31x2) dimensional matrix with the prices of the underlying asset in the first column. The second column contains the respective call prices (for the quantlet `influence`), or the values of Δ_C (for the quantlet `greeks`).

The two dimensional plot (Figure 5.1) displays a positive relationship between the call price and the underlying, which supports the theoretical results from (5.1). Note, for explanation purposes the underlying ranges from 100 to 400. This is achieved through running the quantlets `influence` and `greeks` once again and specifing interactively the parameters with the same values, as in the example above.

Figure 5.2 shows that Δ_C is an increasing function of S. This result is not surprising, since $\frac{\partial \Phi(d_1)}{\partial S}$ is always positive. It follows that the call price is an increasing convex function of the underlying price (see subsection 5.3 for further details on convexity).

The fact that the delta changes as the underlying price changes, means that the delta provides only instantaneous information. To remain perfectly risk-free, a hedged position in options may have to be revised continuously. The delta-hedge frequency, depends on the derivative of the delta with respect to the price of the underlying, commonly referred to as the gamma. For detailed explanations and examples on gamma see subsection 5.3.

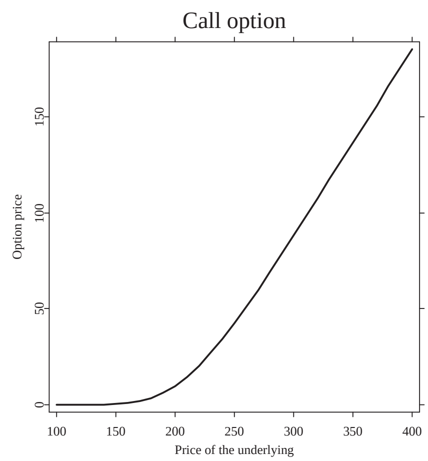

Figure 5.1: The price-stock relationship for a European call.

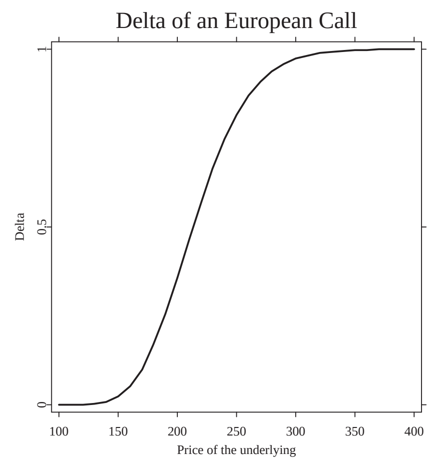

Figure 5.2: Δ_C as a function of the stock price.

In a second example the same procedure is repeated for a put option:

```
library("finance")
 influence(230,210,5,25,0.5,5,1,1,400) ; put option
 greeks(230,210,5,25,0.5,5,0,1,1,400)  ; delta of the put
```

influence.xpl

greeks.xpl

The outputs are presented in Figure (5.3) and Figure (5.4) respectively. Figure (5.4) shows that Δ_P is an increasing function of the asset price S, i.e. the put's price decreases at an increasing rate, when the price of the underlying asset increases. In other words, the put's price is a decreasing convex function of the price of the underlying asset.

Both call and put deltas are functions of S and τ. It can be shown that

$$\lim_{\tau \to \infty} \frac{\partial C}{\partial S} = 1 \qquad \text{for all values of } S.$$

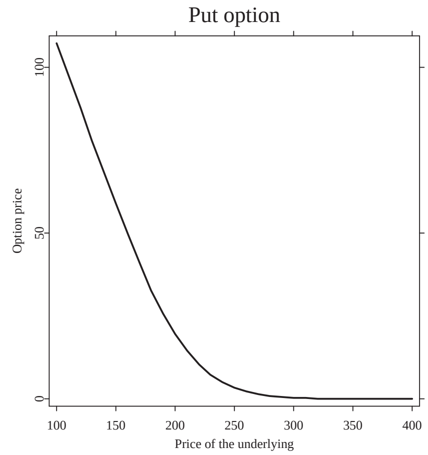

Figure 5.3: The price-stock relationship for a European put.

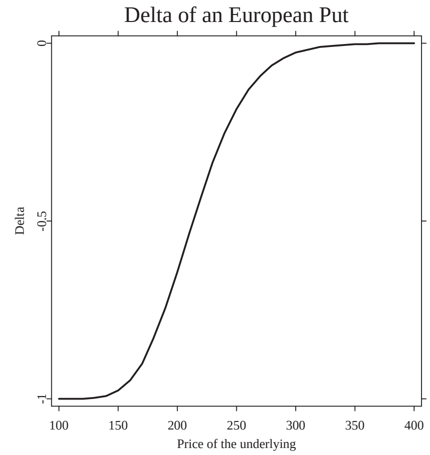

Figure 5.4: Δ_P as a function of the stock price.

This means that Δ_C always tends to one, whereas the time to expiration tends to infinity, since there is a higher probability of an increase in the asset price. In addition, the following holds (Kwock, 1998, pp.57):

$$\lim_{\tau \to 0^+} \frac{\partial C}{\partial S} = \begin{cases} 1 & if \quad S > K \\ \frac{1}{2} & if \quad S = K \\ 0 & if \quad S < K \end{cases}$$

At expiration, delta has different asymptotic limits depending on whether the option is in-the-money ($S > K$), at-the-money ($S = K$), or out-of-the-money ($S < K$). For options deep in-the-money, Δ_C converges to one. In other words, since the option will be exercised at expiration, the writer of the call should hold the asset to hedge the risk. For deep out-of-the-money, the call will not be exercised and the writer no longer needs to hold the asset. Consequently, Δ_C will then converge to zero. Hence at expiration, the option will have either a slope of zero (if out-of-the-money) or one (if in-the-money).

It can be surmised that the at-the-money option, which lies in the middle between these extremes, might have a slope of 0.5. Therefore any time before expiration an out-of-the-money option will have a delta between 0 and 0.5, and in-the money option will have a delta between 0.5 and 1. For example, this can be seen for a call option with six months prior to expiration in Figure 5.2. The "S" shaped curve indicates how the exposure of the call option relative to the underlying asset has a limit loss when the price of the underlying asset falls, and assumes full exposure when the underlying price rises.

Alternative ways to think about delta, as a measure of relative risk of the option to the underlying market, or as the probability of exercise, are explained in Tompkinks (1994, pp. 57-65).

`greeks`() computes Δ_C and displays it as a function of S and τ (Figure 5.5). The input parameters are given interactively. The plot supports the relationship mentioned above.

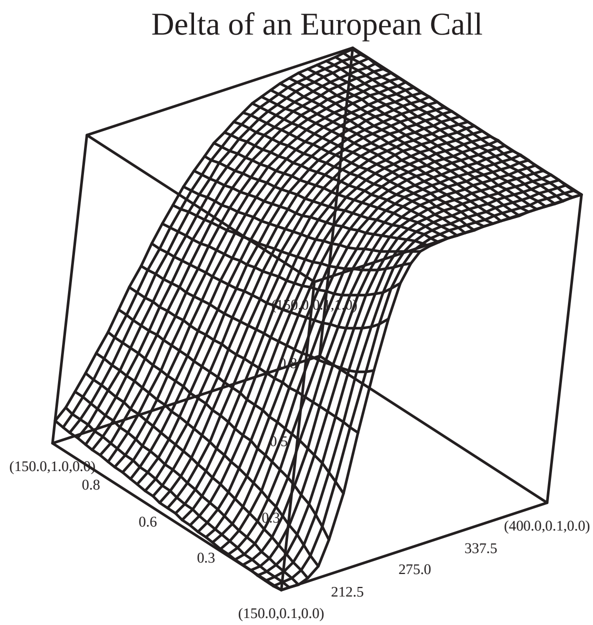

Figure 5.5: The delta of a European call as a function of the stock price S and of time to expiration τ.

`greeks.xpl`

5.2 Delta of the Strike

For most options the strike price is fixed, but some option-like securities, such as convertible bonds, can have a variable "strike" price. In this case the price change of the derivative security V with respect to the strike price K

$$\Delta_K = \frac{\partial V}{\partial K},$$

may be appropriate. The higher the strike price, the less valuable a call option is, since the strike price represents a higher cost of exercising the call and thereby purchasing the stock. In contrast, the higher the exercise price of a put, the higher its price will be. The Black-Scholes formula clearly confirms these relationships:

$$\frac{\partial C}{\partial K} = -exp(-r\tau)\Phi(d_2) < 0, \tag{5.3}$$

$$\frac{\partial P}{\partial K} = exp(-r\tau)\Phi(-d_2) > 0. \tag{5.4}$$

In the following example, the quantlet `influence` displays the relationship between the option price and the strike price for a European call (Figure 5.6) and a European put (Figure 5.8). The quantlet `greeks` is used to plot the respective deltas of the strike (Figure 5.7 and 5.9).

```
library("finance")
 influence(230,210,5,25,0.5,5,1,2,400) ; call
 greeks(230,200,5,25,0.5,5,1,4,1,400)
 influence(230,210,5,25,0.5,5,0,2,400) ; put
 greeks(230,200,5,25,0.5,5,0,4,1,400)
```

`influence.xpl`

`greeks.xpl`

5.3 Gamma

The option gamma (Γ) is defined as the derivative of delta (Δ_V) with respect to the underlying asset price S:

$$\Gamma = \frac{\partial \Delta_V}{\partial S} = \frac{\partial^2 V}{\partial S^2}$$

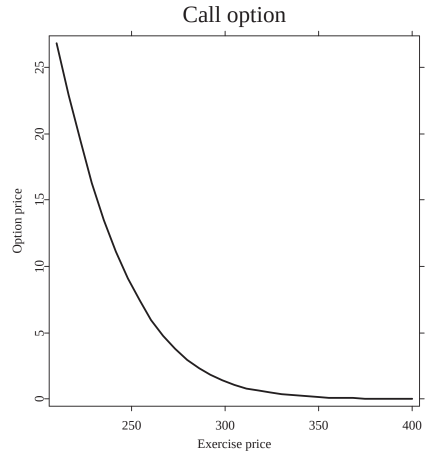

Figure 5.6: The price-stock relationship for a European call.

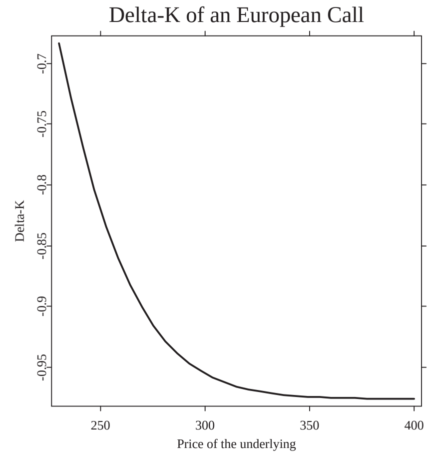

Figure 5.7: Δ_K of a European call as a function of the strike price.

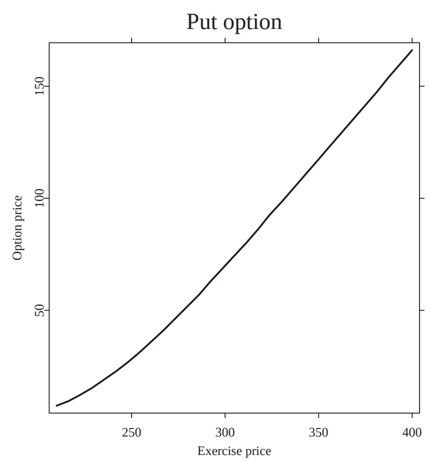

Figure 5.8: The price-stock relationship for a European call.

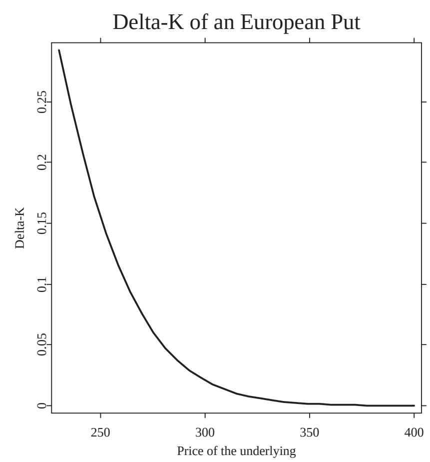

Figure 5.9: Δ_K of a European put as a function of the strike price.

It represents the change in the curvature of the option at different values of S and is therefore also known as convexity. The increments of gamma are often referred to as the number of deltas that will change when the underlying asset price changes by one tick. By definition, the higher the gamma value is, the more the delta will change when the underlying market price changes.

The gamma of a long European call and a long European put on a non-dividend paying underlying asset is

$$\Gamma_C = \Gamma_P = \frac{\partial \Delta_C}{\Delta S} = \frac{\partial^2 C}{\partial S^2} = \frac{1}{S\sigma\sqrt{\tau}}\Phi'(d_1) > 0, \tag{5.5}$$

whereas, for a dividend paying asset it is

$$\Gamma_C = \Gamma_P = \frac{\partial \Delta_C}{\Delta S} = \frac{\partial^2 C}{\partial S^2} = \frac{exp\left(-\frac{d_1^2}{2}\right)}{S\sigma\sqrt{\tau}}\Phi'(d_1) > 0, \tag{5.6}$$

with d_1 defined as in equation (3.1) and $\Phi'(d_1)$ defined as

$$\Phi'(d_1) = \frac{1}{2\sqrt{\pi}} exp\left(-\frac{d_1^2}{2}\right). \tag{5.7}$$

A positive gamma as in (5.5) and (5.6) means that changes in the amount of delta have the same direction as changes in the underlying market. This explains why for any European vanilla call or put option, the curves of the option price functions are convex with respect to the asset price (Figure 5.2 and 5.4).

With gamma being positive, the buyers of the options gain from movements in the price of the underlying assets. For this reason, holders of the options are often referred to as being "long gamma". For sellers of the options, the gamma exposure is exactly opposite to that of buyers of options. Those who sells options can be hurt when gamma is high and the underlying market price moves. Therefore they are often referred to as "short gamma".

When the curvature of the option value is small, its gamma has a low value. A low value of gamma implies that delta changes slowly with the asset price and so adjustments required to keep a portfolio delta neutral can be made less frequently. When gamma is high, in absolute terms, delta is highly sensitive to the price of the underlying asset. It is then quite risky to leave a delta-neutral portfolio unchanged for any remaining length of time.

Intuitively, gamma jointly measures how close the current market is to the option strike price and how close the option is to expiration (Tompkinks, 1994, pp. 67). The closer the market price is to the strike price and the closer the option is to expiration, the higher the gamma will be. This is illustrated in Figure (5.10) for a European call option with `S` = `230`, `K` = `200`, `r` = `5`%, σ = `25`%, τ = `0.5` and `cost of carry` = `5`%. The computation is done interactively by using the quantlet `greeks`().

When the option is at-the-money ($S = K = 230$) with one minute remaining until expiration, it has the highest possible gamma value of 1.0. The reason

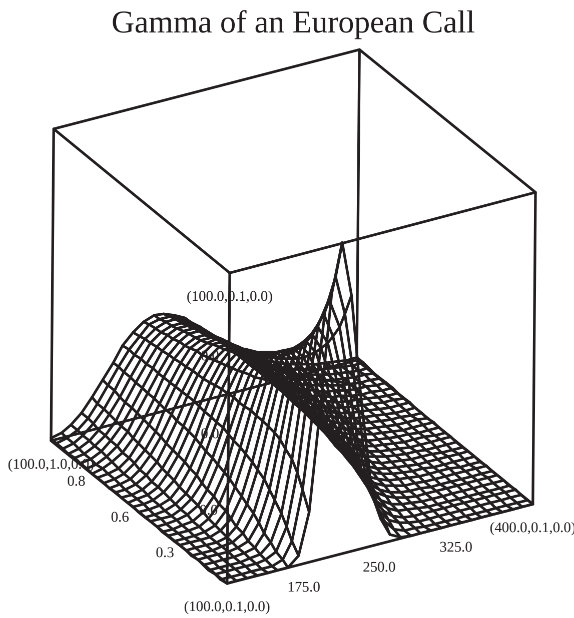

Figure 5.10: Gamma of a long European call (put) as a function of the stock price S and time to expiration τ.

`greeks.xpl`

for this, is that if the underlying asset price moves the tiniest increment up, the option will then be in-the-money with a delta of 1.0 (Figure 5.10). If on the other hand, the underlying asset price falls by an infinitesimally small amount, the option will then be out-of-the-money with a delta of zero.

More moving away from this extreme situation, where the option is at-the-money at expiration, then the lower the gearing effect of the option will be, hence the lower the gamma. The larger the difference between the current underlying asset price and the options's strike price, the less is the time value of the option and the lower the gamma. Additionally, an option with more time remaining until expiration will have a lower gamma (Figure 5.10).

So far the speed (delta) and acceleration (gamma) features of options over the underlying market price have been examined. In the following the other factors are discussed, the most important being volatility.

5.4 Vega

The change in the option price with respect to the change in implied volatility (section 6) is called vega. It is a measure of the option exposure to changes in implied volatility within the option market (Tompkinks, 1994, pp. 69). The vega of a European vanilla call Λ_C and put Λ_C on non-dividend paying underlying asset can be derived from the Black-Scholes formula:

$$\begin{aligned}\Lambda_C &= \frac{\partial C}{\partial \sigma} = S\Phi'(d_1)\frac{\partial d_1}{\partial \sigma} - Kexp(-r\tau)\Phi'(d_2)\frac{\partial d_2}{\partial \sigma} \\ &= S\sqrt{\tau}\Phi'(d_1) > 0 \qquad (5.8)\end{aligned}$$

$$\Lambda_P = \frac{\partial P}{\partial \sigma} = \frac{\partial C}{\partial \sigma} + \frac{\partial}{\partial \sigma}\{Kexp(-r\tau) - S\} = \Lambda_C > 0 \qquad (5.9)$$

with d_1 and $\Phi'(d_1)$ given in 3.1 and 5.7 respectively.

When the volatility rises, the option's set of favorable outcomes will also rise. As a result, the chances are higher for the option to be either deeper in-the-money or deeper out-of-the-money at expiration. Since the option bears no downside risk, there is no penalty when the option expires deeper out-of-the-money, but a higher payoff when it expires deeper in-the-money. Due to this antisymmetric payoff structure, the vegas for long options are positive, i.e. for the option buyer, the exposure to changes in implied volatility is positive and consequently the vega is positive. By symmetry, the option writer benefits from a decrease in implied volatility and therefore has vega negative exposure.

Using interactive menus in the quantlet `greeks`(), the vega of a European call option is presented as a function of the underlying asset and of time to expiration (Figure 5.11). For at-the-money options, the longer the time to expiration, the higher the sensitivity of the option to the changes in volatility, and hence the higher the vega. In other words, whereas gamma of an at-the-money option increases as the expiration date approaches (see Figure 5.10), the reverse is true for vega.

5.5 Eta

Eta (η) of a derivative security defines the elasticity of its price V with respect to the underlying price S:

$$\eta = \left(\frac{\partial V}{\partial S}\right)\left(\frac{S}{V}\right)$$

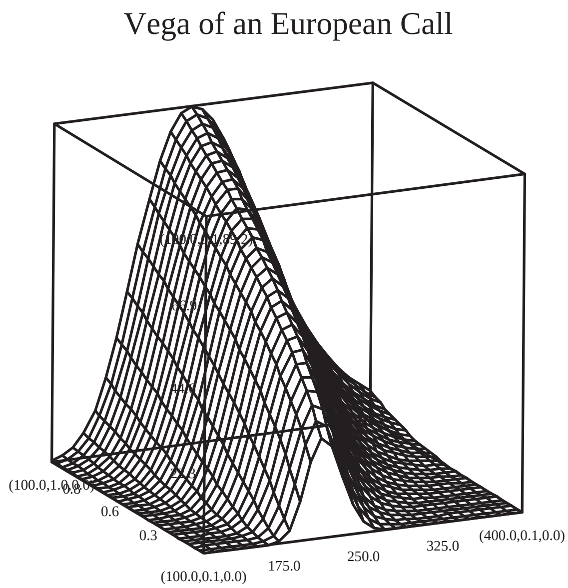

Figure 5.11: The vega of a European long call (put) as a function of the stock price S and time to expiration τ.

`greeks.xpl`

This elasticity parameter measures the percentage change in security price for a unit percentage change in the asset price. The elasticity of a European vanilla call (η_C) on non-dividend paying underlying asset is found to be:

$$\eta_C = \left(\frac{\partial C}{\partial S}\right)\left(\frac{S}{C}\right) = \frac{S\Phi(d_1)}{S\Phi(d_1) - Kexp(-r\tau)\Phi(d_2)} > 1 \qquad (5.10)$$

Equation (5.10) implies that a call option is riskier than the underlying asset in terms of change in percentage. It can be shown that the elasticity is high when the asset price is low (out-of-the-money), and it decreases monotonously with the price of the underlying asset (Kwock, 1998, pp. 57). For sufficiently large values of S, η_C converges to one, as C approaches S when S tends to infinity. This relationship can be seen in Figure (5.12), which plots the elasticity of a European call from the following example:

```
library("finance")
 greeks(230,210,5,25,0.5,5,1,3,2,400) ; call
```

greeks.xpl

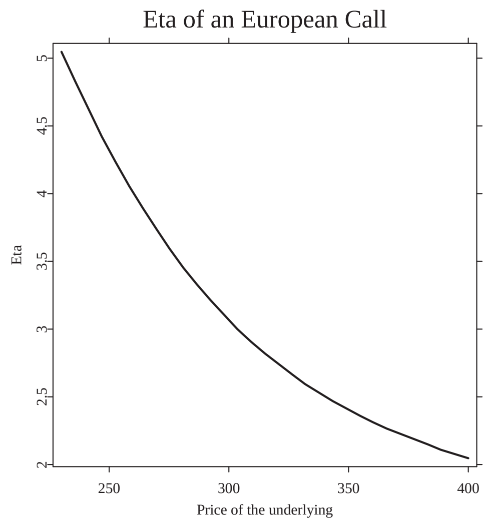

Figure 5.12: η_C as a function of the stock price.

The elasticity of a European put price (η_C) on non-dividend paying underlying asset is:

$$\eta_P = \left(\frac{\partial P}{\partial S}\right)\left(\frac{S}{C}\right) = \frac{-S\Phi(-d_1)}{Kexp(-r\tau)\Phi(-d_2) - S\Phi(-d_1)} \tag{5.11}$$

Equation (5.11) shows that the absolute value of the put elasticity can be less or greater than one. Therefore, a European put option may or may not be riskier than the underlying asset in terms of change in percentage.

For both put and call options, their elasticities increase in absolute value when the corresponding options become more out-of-the-money and move closer to expiration (Kwock, 1998, pp. 57).

5.6 Theta

Theta (Θ) of a derivative security is defined as the rate of change of its price V with respect to time t with all other factors remaining constant:

$$\Theta = \frac{\partial V}{\partial t} = -\frac{\partial V}{\partial \tau}$$

It is sometimes referred to as the 'time decay' of security. The derivative of the call price as determined by the Black-Scholes model in (3.1) with respect to time shows that

$$\Theta_C = \frac{\partial C}{\partial t} = -\frac{\partial C}{\partial \tau} = -\frac{S\sigma}{2\sqrt{\tau}}\Phi'(d_1) - rKexp(-r\tau)\Phi(d_2) < 0. \tag{5.12}$$

Using then the put-call parity relation 3.9, Θ_P is derived as

$$\Theta_P = \frac{\partial P}{\partial t} = -\frac{S\sigma}{2\sqrt{\tau}}\Phi'(d_1) + rKexp(-r\tau)\Phi(-d_2). \tag{5.13}$$

The longer the time to expiration of a European call option, the higher its price ($\frac{\partial C}{\partial \tau} > 0$). The reason for this, is that by prolonging the time interval until the option's expiration, simultaneously increases the chances of the option either ending in-the-money at expiration and making a profit, or ending out-of-the-money and expiring worthless. This also holds for an American call on a non-dividend paying underlying asset, which will not be exercised before expiration.

$\frac{\partial C}{\partial \tau}$ for a European call and a European put is computed in XploRe, by specifying the input parameters interactively. The plots are shown in Figure (5.13) and (5.14). The underlying price S ranges from 150 to 400. The values of the other input parameters are `K` = `210`, `r` = `5`% `p.a.`, σ = `25`% `p.a.`, τ = `0.5 years`, `cost of carry` = `5`%. The positive $\frac{\partial C}{\partial \tau}$ for a European call, is consistent with the result in (5.12), as `greeks` considers $\frac{\partial C}{\partial \tau}$ as theta, instead of $\frac{\partial C}{\partial t}$.

The theta of a European call (Figure 5.13) has its greatest absolute value when the call option is at-the-money, as it may become in-the-money or out-of-the-money soon thereafter. It has a small absolute value when the option is sufficiently out-of-the-money, as it will be highly unlikely for the option to become in-the-money later on. It tends asymptotically to $-rKexp(-r\tau)$ when the asset price is sufficiently high.

The theta of a European put (Figure 5.14) can be any sign, depending on the relative magnitudes of the two terms, which have opposite signs in (5.13). When the European put is deep in-the-money, S assumes a small value, so that $\Phi(-d_2)$ tends to one. The second term $rKexp(-r\tau)\Phi(-d_2)$ is then greater than the first $\frac{S\sigma}{2\sqrt{\tau}}\Phi'(d_1)$. In this case, $\frac{\partial C}{\partial \tau}$ is negative and the theta as defined in (5.13) is positive. When the option is at-the-money or out-of-the-money, $\frac{\partial C}{\partial \tau}$ is typically positive and hence the theta of a European put is negative, as the longer the time to expiration, the higher the chances of positive outcomes.

Note, the ambiguous relationship between a put's price and time to expiration does not hold for American options. An American call or put will

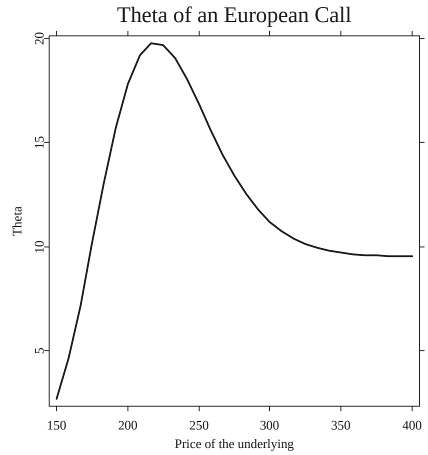

Figure 5.13: $\frac{\partial C}{\partial \tau}$ for a European call as a function of the stock price

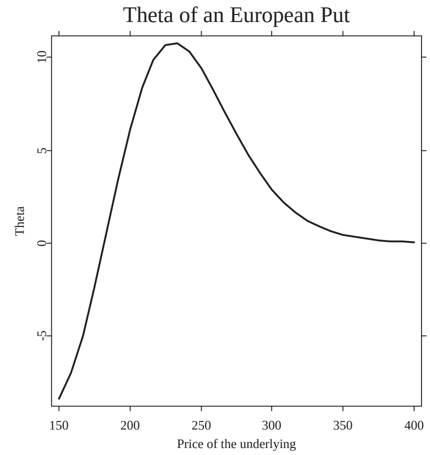

Figure 5.14: $\frac{\partial P}{\partial \tau}$ for a European put as a function of the stock price

greeks.xpl

always show a positive relationship between its price and time to expiration, which corresponds to a negative theta. When the time to expiration is prolonged, an American option has therefore the additional right to be exercised in the prolonged time interval and consequently has a higher value.

5.7 Rho

The Rho (ρ) of a derivative security is defined as the rate of change of derivative price V with respect to the interest rate r:

$$\rho = \frac{\partial V}{\partial r}$$

The rho of a European call ρ_C and the rho of a European put ρ_P for non-dividend paying underlying asset is derived from the Black-Scholes formulae (3.1) and (3.2), respectively

$$\rho_C = \tau K exp(-r\tau)\Phi(d_2) > 0 \quad \text{for } r > 0 \text{ and } K > 0, \tag{5.14}$$

and from the put-call parity relation

$$\rho_p = -\tau K exp(-r\tau)\Phi(-d_2) < 0 \quad \text{for } r > 0 \text{ and } K > 0. \tag{5.15}$$

A higher interest rate decreases the present value of the cost of exercising the European call at expiration (which is a similar effect to the strike price decreasing) and so increases the call price. The reverse effect holds for the put price. Signs of the rho for call and put prices in (5.14) and (5.15) confirm this effect.

The following example calculates ρ_C and ρ_P and plots them against the underlying asset price S (Figure 5.15 and 5.16).

```
library("finance")
 greeks(230,210,5,25,0.5,5,1,7,1,400) ; call
 greeks(230,210,5,25,0.5,5,0,7,1,400) ; put
```

greeks.xpl

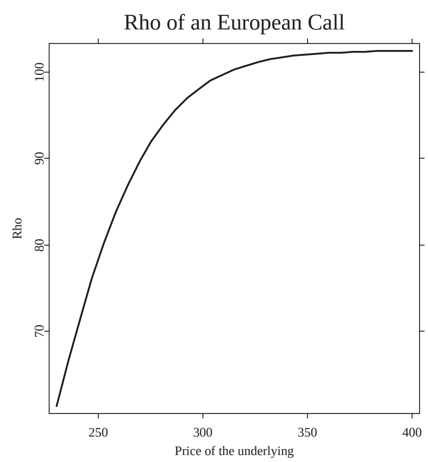

Figure 5.15: ρ_C as a function of the price of the underlying asset S.

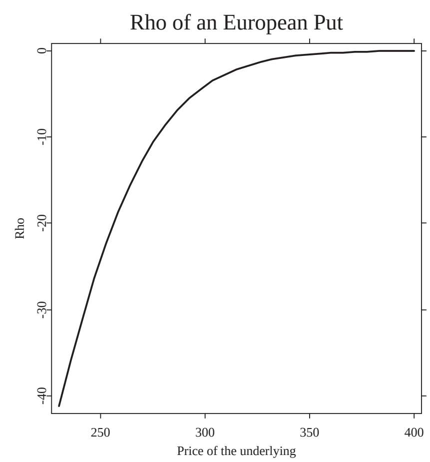

Figure 5.16: ρ_P as a function of the price of the underlying asset S.

6 Implied Volatility

The volatility of an asset is a measure of variability of its returns. Traditionally, volatility is measured on past prices of the underlying asset, known as historical volatility. However for investors, who have a high regard for the 'wisdom' of the market, the best estimate of volatility comes from the market itself.

If the market price of the option is taken to be the correct price, then the volatility implied by the market price reflects the market's opinion of what the volatility should be. The value of the volatility of the underlying asset that would equate the option price to its fair value, is called implied volatility. In other words, implied volatility is the volatility, which is implicitly contained in the option price (Alexander, 1996, pp. 14). It is a timely measure - it reflects the market's perceptions today - and it should therefore provide the market's best estimate of future volatility (Jorion, 2001). This is therefore one reason to believe that option-based forecasts can be superior to historical estimates. Supporting evidence on this point is for example provided in Jorion (1995) and Campa and Chang (1998).

Implied volatilities are a useful tool in monitoring the market's opinion regarding the volatility of a particular stock. Besides this, options are often traded on volatility with the implied volatility becoming the effective price of the option. Implied volatility also has important implications for risk management. If volatility increases, so will the value at risk (VaR). Investors may want to adjust their portfolio in order to reduce their exposure to those instruments, whose volatility is predicted to increase. Hence, in a delta hedged portfolio the vega risk (see subsection 5.4) can become the most significant risk factor within the portfolio.

When an explicit analytic option pricing formula is available, as for instance the Black-Scholes formula (3.1), the quoted price of the option along with known variables, such as the price of the underlying asset `S`, the exercise price `K`, time to expiration τ and the interest rate `r` can be used in an implicit formula to calculate the so called implied volatility. The Black-Scholes implied volatility refers to the market price of the option equal to the price given by the Black-Scholes formula (3.1). For a call option, it can be written as

$$C(S_t, K, \tau, r, \sigma_I(K, \tau)) = C_t^*(K, \tau), \tag{6.1}$$

where $C(S_t, K, \tau, r, \sigma_I(K, \tau))$ is the Black-Scholes call price, $\sigma_I(K, \tau)$ is the implied volatility and $C_t^*(K, \tau)$ is the market price of the call at time instant `t`. The implied volatility of a European put with the same strike and maturity can be derived from the put-call parity (3.9). The existence of the uniqueness of the implied volatility in (6.1), is due to the fact that the value of a call option as a function of volatility is a monotonic mapping from $[0, \infty[$ to $]0,\ S_t - Kexp(-r\tau)[$.

The Black-Scholes model assumes that the underlying asset follows a Brownian motion with constant volatility. If this model is correct, then the distribution of the underlying asset at any option expiration is lognormal, and all options on the underlying asset must have the same implied volatility. Since the market crash in 1987, the market implied volatilities for index options have shown that at-the-money options yield lower volatilities than in-the-money or out-of-the-money options. The convex shape of the implied volatility with respect to the moneyness (K/S) is referred to as the smile effect. The smile effect occurs as at-the-money options are more sensitive to volatility, so that a smaller volatility spread is required for them to achieve the same profit or risk premium as out-of-the-money options.

Jarrow and Rudd (1982) argued that this smile effect can be partially explained by departures from lognormality in the underlying asset price, particulary for out-of-the-money options. The smile is particularly noticeable in the Black-Scholes implied volatility - possibly because of the inappropriate assumptions underlying the Black-Scholes model - and tends to increase as the option approaches expiration (Hull and White, 1987). Hence, the value of the implied volatility depends on time to expiration τ and strike `K`. The function

$$\sigma_I : \quad (K, \tau) \longrightarrow \sigma_I(K, \tau) \tag{6.2}$$

is called the implied volatility surface at date `t`, i.e. it is the plot of implied volatility across strike and time to maturity. Using the moneyness of the option, $m = K/S_t$, the implied volatility surface can be represented as a function of moneyness and of time to expiration. This graphical representation is convenient, because there is usually a range for moneyness around $m = 1$, where options are liquid and therefore empirical data is available (Cont and da Fonseca, 2002). The quantlet `volsurf` in XploRe offers the choice to plot the implied volatiliy surface either as a function of (K, τ) or of (m, τ).

The dependence of implied volatility on strike and maturity is analyzed by various authors for different markets. It is empirically found that the implied volatility surface exhibits a non-flat profile with respect to both strike and term structure, which contradicts the flat profile provided by the Black-Scholes model. Evidence of this is given for example in Dumas et al. (1996), Fengler et al. (2001), Franks and Schwarz (1991), Heynen (1993), Hodges (1996), and Rebonato (1999). The dynamic properties of the implied volatility time series is mainly analyzed using the Principal Component Analysis (PCA). In this context, a cross-section of the implied volatility surface in one direction is considered. If the cross-section is made on different points of the moneyness axis, then a series of term structure-curves is obtained. Analogously, if this is done on the time to expiration axis, a series of smile-curves is obtained. Then the PCA is applied. Examples of the term structure of at-the-money implied volatilities using the PCA can be found in Härdle and Schmidt (2000), Heynen et al. (1995), Zhu and Avellaneda (1997).

There are however some shortcomings with implied volatilities. There is considerable evidence that these volatilities are themselves stochastic. Typically the shape of the distribution (and hence the smile) is unstable because of 'volatility of volatility'. Another problem is that the asset returns and volatility may be correlated, but often non-linearly, usually reflected in fat-tailed and skewed distributions of the underlying asset. Implied volatilities can also be biased, especially if they are based upon options that are thinly traded.

6.1 Software Application

XploRe offers different algorithms to calculate implied volatilities. Further on, the volatility surfaces can be constructed through parametric or non-parametric approaches and plotted. Fengler et al. (2001) analyze implied volatilities using XploRe as a computational tool, see e-book
Applied Quantitative Finance, ch. 7.

6.1.1 Computing Implied Volatility

In practice several implied volatilities are obtained simultaneously from different options on the same stock and a composite implied volatility for the stock is then calculated by taking a suitably weighted average of the individual implied volatilities. Note, that XploRe computes only the implied volatilities from each option. If a composite implied volatility is required, the user then has to decide about the weighting scheme. It is however important that the weights reflect the sensitivity of the option prices to volatility, such as the price of the at-the-money option is far more sensitive to volatility than the price of the deep out-of-the-money option. Different weighting schemes are discussed in Latene and Rendelman (1976), by Chiras and Manaster (1978), and in Whaley (1982).

```
european()
     calculates the prices of European options, or their
     implied volatilities, specifying interactively the
     input parameters.

ImplVola(x,IVmethod)
     calculates implied volatilities assuming the
     Black-Scholes model for European options by
     using either the Newton-Raphson or bisections
     method.  The input parameters are specified either
     interactively or directly.

volatility(task)

volatility(S,K,r,tau,opt,optprice,tyoeofdiv,div)
     calculates implied volatilities of European
     options, specifying either interactively or
     directly the input parameters.
```

`european` uses the quantlet `volatility` to compute implied volatilites. Note, that the quantlet `optstart` is not explicitly mentioned as it performs no calculation. It calls either the quantlet `european`, or `american` to compute implied volatilities for european and american options respectively. XploRe does not recommend the calculation of implied volatilities for American options valuated with the MacMillan approximation method (see subsection 3.4).

`ImplVola` offers two different algorithms to calculate implied volatilities, the bisection and the Newton-Raphson. The most widely used technique for the estimation of the implied volatility is the Newton-Raphson iterative algorithm. It involves making an initial guess as to the implied volatility of the option. It then uses the Greek derivative of the option price relative to changes in volatility (the vega) to make a new guess if the initial guess is off the mark. Tompkinks (1994, pp. 143) writes the algorithm as the following:

$$\begin{aligned} \sigma_{i+1} &= \sigma_i - \frac{Y_i - P}{\Lambda_i} \\ \textit{until} \quad & \mid Y_i - P \mid \leq \epsilon \end{aligned}$$

where

$$\begin{aligned} P &= \text{traded option price,} \\ \sigma_i &= \text{volatility estimate,} \\ Y_i &= \text{option theoretical value with } \sigma_i \text{ volatility,} \\ \Lambda_i &= \text{options vega at theoretical price } Y_i, \\ \epsilon &= \text{desired degree of accuracy.} \end{aligned}$$

The convergence to the correct answer is often achieved in only two or three iterations, if the option price relationship to time is continuous and relatively linear. This is the case for European vanilla options, where the price-volatility relationship is a smooth, relatively linear curve. For other kinds of options including American options, where a significant probability of early exercise exists, or for complex options, which have a kinked rather than a smooth price-volatility relationship, this technique may not work. For these types of options, the bisection method is then preferred. The bisection algorithm can be described as the following:

Step 1. Pick σ_0 and σ_1 so that

$$\begin{aligned} \sigma_0 < \sigma \qquad & \text{i.e.} \quad C(\sigma_0) < C_{observed} \\ \sigma_1 > \sigma \qquad & \text{i.e.} \quad C(\sigma_1) > C_{observed} \end{aligned}$$

Step 2. Choose $\sigma_2 = \frac{\sigma_0+\sigma_1}{2}$

If $C(\sigma_2) > C_{observed}$ then $\sigma_3 = \frac{\sigma_0+\sigma_2}{2}$, else $\sigma_3 = \frac{\sigma_1+\sigma_2}{2}$

Step 2 is repeated until a sufficiently good approximation for σ is obtained.

In the following example `ImplVola` is used to calculate implied volatilities for four different options (two calls and two puts) on different underlying assets with the Newton-Raphson algorithm:

```
library ("finance")
 assetprice = #(5290.36,5290.36,5290.36,5290.36); input data - S
 strike = #(5350,5500,3700,3800)                ; input data - K
 irate = #(0.03294,0.03294,0.03294,0.03294)     ; input data - r
 maturity = #(0.13425,0.13425,0.13425,0.13425)  ; input data - tau
 optionprice = #(221.6,154.2,4.9,6.4)        ; input data - C
 type = #(1,1,0,0)                           ; 2 calls, 2 puts
 x=assetprice~strike~irate~maturity~optionprice~type ; data matrix
 ivola=ImplVola(x)                           ; compute ImplVola
 ivola                                       ; display ImplVola
```

ImplVola.xpl

The XploRe output shows the calculated impled volatilities:

```
Contents of ivola

 [1,]  0.30842
 [2,]  0.2993
 [3,]  0.47033
 [4,]  0.45812
```

Implied volatility for the first European call option is 0.30842. For the same option, implied volatility is 0.3087 when calculated with the quantlet `volatility`. The examples show that it makes no significant difference if implied volatility is computed using `ImplVola` or `volatility`. The results differ only by 10^{-4}.

`volatility` computes the implied volatility of each european option as a result of an optimization process of the option price along the volatility σ. It uses the function `nelmin`, which searches for a minimum of the squared option price function. In each iteration step the function is evaluated at a simplex of (p+1) points. The iteration stops when the variance is less than a predetermined value or when a given iteration number is reached. Technical details are given in Nelder and Mead (1965).

The input parameter `task` in `volatility` is a scalar that specifies the type of the dividend payment: for `task`=1 no dividend, for `task`=2 a continuously paid dividend and for `task`=3 a fixed dividend at the end of T is assumed. Finally, if `task`=4, then an exchange rate is assumed as underlying.

The following example calculates the implied volatility for only one European call, when no dividends are assumed:

```
library("finance")
 volatility(1)              ; no dividends
```

volatility.xpl

with the input values (30, 230, 210, 5, 0.5) for (C,S, K, r, τ). Using the quantlet `european` yields the same result for implied volatility.

european.xpl

The output window yields:

```
Contents of aus
 [1,] " "
 [2,]"--------------------------------------"
 [3,] " The Implied Volatility of Your Option "
 [4,] " on Given Stock is "
 [5,] "0.2292"
 [6,]"--------------------------------------"
 [7,] " "
```

It is possible to calculate simultaneously implied volatility for one or more options by specifying input parameters directly in `volatility`(`S`, `K`, `r`, `tau`, `opt`, `optprice`, `tyoeofdiv`, `div`).

6.1.2 Construction of Smooth Volatility Surfaces

The usual practice to construct implied volatility surfaces for arbitrary strikes `K` and maturities τ is to smooth the discrete data. This can be done in a parametric or non-parametric way. For example, it is common practice in many banks, to use (piecewise) polynomial functions to fit the implied volatility smile (Dumas et al., 1996). XploRe offers the following two quantlets to construct and plot volatility surfaces:

```
volsurf(x,stepwidth,firstXF,lastXF,firstMat,lastMat,metric,
     bandwidth,p,IVmethod)
     calculates the implied volatility surface using a
     Kernel smoothing procedure, specifying directly the
     input parameters.

volsurfplot(IVsurf,IVpoints,AdjustToSurface)
     plots the implied volatility surface computed by
     the quantlet volsurf with original options shown
     as red points, specifying directly the input
     parameters.
```

`volsurf` computes the implied volatility surface using a kernel smoothing procedure. Either a Nadaraya-Watson estimator or a local polynomial regression is employed.

The local polynomial method is used to estimate an unknown function m, which expresses a functional dependence between an explanatory variable $(X_1, X_2) = (K, \tau)$ and the dependent variable $\sigma(X_1, X_2) = m(X_1, X_2)$. In contrast to the parametric regression, there are no restrictions on the form of $m(\cdot)$, i.e. theory does not state whether $m(\cdot)$ is linear, quadratic or increasing in (X_1, X_2) (Härdle et al., 2001). The local polynomial method is based on the idea that under suitable conditions, the function m can be locally, i.e. at an observation point $(x_{10}x_{20})$, approximated through a Taylor expansion. The local polynomial can then be fitted by a weighted least squared regression problem. Only the observations, which are close enough to $(x_{10}x_{20})$ have to be considered in the minimization process. The neighborhood is realized by including kernel weights into this process. In contrast to the parametric least squares, the estimator varies with the observations (x_{1i}, x_{2j}) for $i, j = 0, 1...n$. The whole surface is obtained by running the above local polynomial regression for each observation (x_{1i}, x_{2j}).

Alternatively, `volsurf` uses the filtered data set to construct a smooth estimator of the implied volatility surface, defined on a fixed grid, using the non-parametric Nadaraya-Watson estimator. Given the exploratory variables $(X_1, X_2) = (K, \tau)$, the two dimensional Nadaraya-Watson kernel estimator is

$$\hat{\sigma}(x_1, x_2) = \frac{\sum_{i=1}^{n} K_1\left(\frac{x_1 - x_{1i}}{h_1}\right) K_2\left(\frac{x_2 - x_{2i}}{h_2}\right) \hat{\sigma}_i}{\sum_{i=1}^{n} K_1\left(\frac{x_1 - x_{1i}}{h_1}\right) K_2\left(\frac{x_2 - x_{2i}}{h_2}\right)}, \tag{6.3}$$

where $\hat{\sigma_i}$ is the implied volatility from the observed option price, K_1 and K_2 are univariate kernel functions, and h_1 and h_2 are the bandwidths.

`volsurf` uses a quartic Kernel for both, the local polynomial and the Nadaraya-Watson estimator. The order 2 quartic kernel is given by

$$K_i(u) = \frac{15}{16}(1 - u^2)^2 I(|u| \leq 1). \tag{6.4}$$

The choice of another kernel, for instant a Gaussian kernel as in Cont and da Fonseca (2002), instead of quartic Kernel does not influence the results very much. The important parameters are the bandwidth parameters h_1 and h_2 which determine the degree of smoothing. Too small values will lead to a bumpy surface, too large ones will smooth away important details. Härdle (1994, ch. 5) and Härdle et al. (2001, ch. 4.3) discuss different ways how to calculate the bandwidth, as for instance using a cross-validation criterion, or an adaptive bandwidth estimator in order to obtain an 'optimal' bandwidth.

The first input parameter `x` in `volsurf` is a (n x 6) dimensional data matrix. The columns one to six contain: underlying asset prices `S`, strike prices `K`, interest rates `r`, time to expiration τ, option prices and types of option (1 for call and 0 for put). The next five input parameters are concerned with the construction of the volatility surface. `stepwidth` is a (2 x 1) dimensional vector, where the first element refers to the strike dimension and the second to time to expiration. `firstXF` (`lastXF`) and `firstMat` (`lastMat`) are scalar constants giving the lowest (highest) limit of the strike dimension and of time to expiration in the volatility surface, respectively. The metric in `volsurf` is either moneyness K/F (`metric` = `0`), where F is the (implied) forward price of the underlying asset computed as $F_t = S_t exp(r\tau)$, or is the original strike price K (`metric` = `1`). The parameter `bandwidth` is a (2 x 1) dimensional vector determining the width of the bins for the kernel estimator. The parameter `p` is a scalar, which indicates whether the Nadaraya-Watson estimator (`p` = `0`) or the local polynomial regression (`p` $\neq$ `0`) is used. The last parameter, `IVmethod`, is optional. As in quantlet `ImplVola`, if `IVmethod` = "`bisect`" then the bisection method is used to compute implied volatilities. The default method is the Newton-Raphson algorithm (see subsection 6.1.1).

The output of the quantlet `volsurf` consists of two variables. The first one, `IVsurf`, contains the co-ordinates of the points computed for the volatility surface. It is a (N x 3) dimensional matrix, where N is the number of grid points. The second one, `IVpoints`, is a (M x 3) dimensional matrix, which contains the co-ordinates of the M options used to estimate the surface. In both variables, the columns one to three contain the values of strike dimension, of time to expiration and of estimated implied volatility, respectively.

`volsurfplot` is a graphical tool used to display the volatility surface constructed with the quantlet `volsurf`. Therefore the input parameters are the co-ordinates of the volatility surface `IVsurf` and of the original option values contained in `IVpoints`, which were used in `volsurf` to construct the surface. The third input parameter `AdjustToSurface` is optional. It determines, whether the graph-limits are based on the original option observations stored in `IVpoints`, or based on coordinates of estimated surface `IVsurf`. By default `AdjustToSurface` = `1`, the graph is adjusted according to the estimated surface.

To illustrate, two examples as given in the description part of `volsurf` are used. The first example constructs the volatility surface in moneyness metric (`metric` = `0`), using the Nadaraya-Watson estimator (`p` = `0`). The implied volatilities are computed with the bisection method (`IVmethod` = ”`bisect`”).

```
library ("finance")
 data=read("volsurfdata2.dat"); reads data
 IVmethod="bisect"             ; computes implied volatilities
 sw=0.02|(1/52)                ; stepwidth
 bw=0.1|0.4                    ; bandwidth
 fXF=0.8                       ; firstXF
 lXF=1.2                       ; lastXF
 fMat=0                        ; firstMat
 lMat=1                        ; lastMat
 metric=0                      ; computes in moneyness dimension
 AdjustToSurface=1
 IVSurface,IVpoints=volsurf(data,sw,fXF,lXF,fMat,lMat,metric,
                            bw,0,IVmethod)
 volsurfplot(IVSurface,IVpoints,AdjustToSurface)
```

volsurf.xpl

volsurfplot.xpl

`volsurfplot` displays the implied volatility surface as a function of moneyness and time to expiration in years (Figure 6.1). The original options are marked red. The graph shows a decreasing profile in moneyness ('skew') and changes in the volatility term structure. The 'skew' is the degree of asymmetry on upper and lower sides of the underlying distribution.

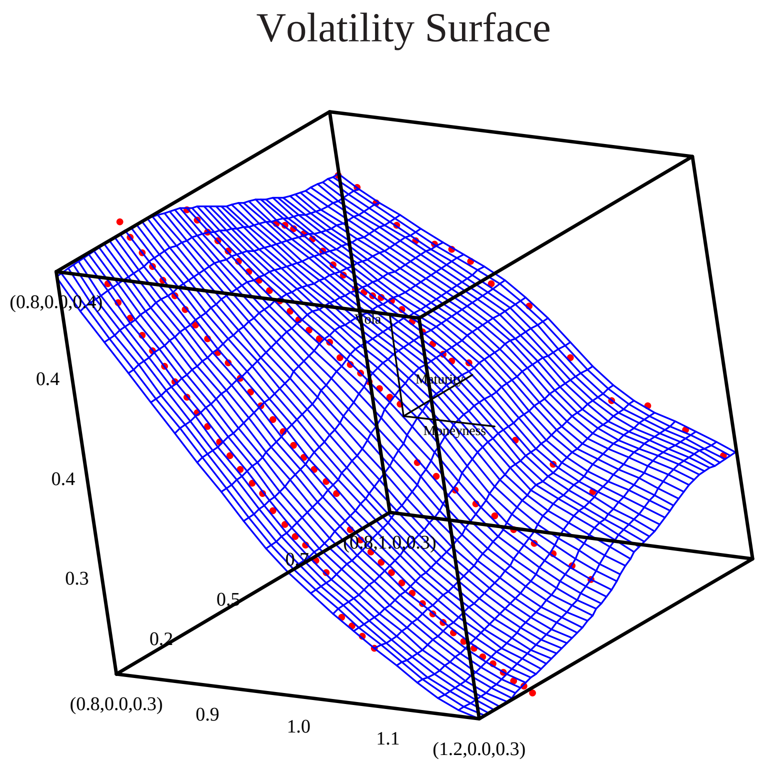

Figure 6.1: Implied volatility surface from Januar 4th, 1999 in moneyness metric. The red points in the graph denote the original options. The implied volatilities are computed with the bisection method. The implied volatility surface is constructed using the Nadaraya-Watson estimator.

`volsurf.xpl`

`volsurfplot.xpl`

A second example constructs the volatility surface also using the default Nadaraya-Watson estimator (`p` = `0`), but in strike metric (`metric` = `1`), for the same data set as in the previous example. The implied volatilities are now computed with the default Newton-Raphson algorithm.

```
library ("finance")
 data=read("volsurfdata2.dat") ; reads data
 sw=70|(1/52)                ; stepwidth
 bw=250|0.5                     ; bandwidth
 fXF=3500                        ; firstXF
 lXF=7000                       ; lastXF
 fMat=0                         ; firstMat
 lMat=1                         ; lastMat
 metric=1                      ; calculates in strike dimension
 AdjustToSurface=1
 IVSurface,IVpoints=volsurf(data,sw,fXF,lXF,fMat,lMat,metric,bw,1)
 volsurfplot(IVSurface,IVpoints,AdjustToSurface)
```

volsurf.xpl

volsurfplot.xpl

`volsurfplot` displays the implied volatility surface as a function of strike price and of time to expiration in years (Figure 6.2). The slight difference between the two surfaces relates to the different algorithms used for computing the implied volatilities. However, the profile in strike is mainly downward sloping and the profile in maturity shows a variable volatility term structure. Both examples confirm the well-known evidence that the implied volatility surface is other than flat, which would be the case if the assumption of constant volatility in the Black-Scholes model was correct.

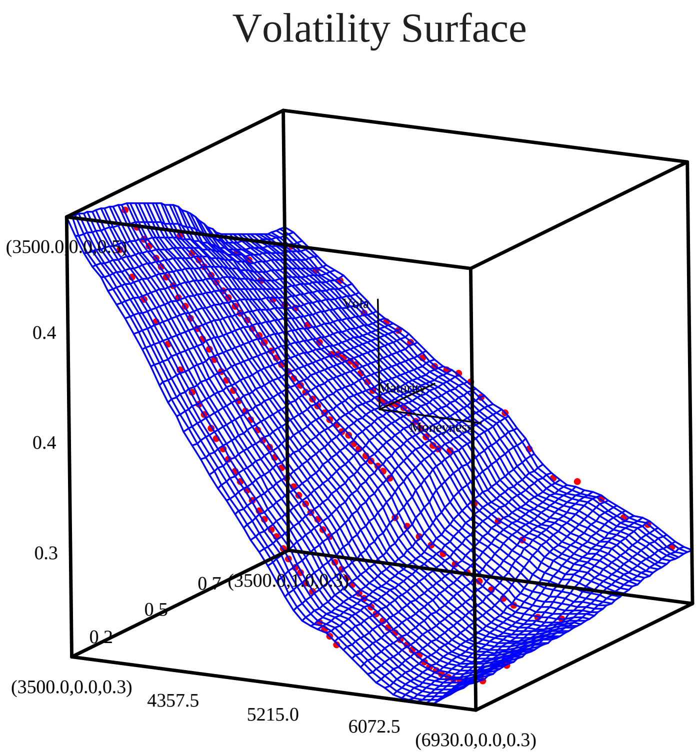

Figure 6.2: Implied volatility surface from Januar 4th, 1999 in strike metric. The red points in the graph denote the original options. The implied volatilities are computed with the Newton-Raphson algorithm. The implied volatility surface is constructed using the Nadaraya-Watson estimator.

`volsurf.xpl`

`volsurfplot.xpl`

6.2 Implied Binomial Trees

The variation of implied Black-Schloles volatilities, with both strike and expiration is currently a persistent feature of option markets. Jarrow and Rudd (1982) argue that the smile for a given maturity can be partially explained by departures from lognormality in underlying asset prices, particulary for out-of-the-money options. Researchers have attempted to enrich the Black-Scholes model to account for the smile. Extensions, such as jumps in the underling asset price (see subsection 2.2.2) or stochastic volatility factor (Hull and White, 1987), unfortunately cause several practical difficulties, for example the violation of the risk-neutral condition (Härdle and Zheng, 2001).

Implied binomial trees (IBT) proposed by Derman and Kani (1994), Dupire (1994), Rubinstein (1994), Barle and Cakici (1998) account for risk-neutrality and extend the Black-Scholes theory, making it consistent with the shape of the smile. This consistency is achieved by extracting the implied evolution of the stock price from the market prices of liquid European vanilla options on the underlying stock.

CRR is a binomial tree, which is a discrete version of the geometric Brownian motion (2.4). Similarly, IBT and any other multinomial tree can be viewed as discrete versions of the following diffusion process of the underlying asset:

$$\frac{dS_t}{S_t} = \mu_t \, dt + \sigma(S_t, t) \, dW(t) \tag{6.5}$$

The variable μ_t is the risk-neutral drift and $\sigma(S_t, t)$ is the instantaneous local volatility function, which is dependent on both underlying price and time. Models of this type usually involve a special parametric form of $\sigma(S_t, t)$. In contrast, the IBT approach deduces $\sigma(S_t, t)$ numerically from the smile. It ensures that local volatility varies from node to node, so that the market price of any plain vanilla option can be matched. Option prices for all strikes and expirations, obtained by interpolation from known option prices, will determine the position and the probability of reaching each node in the implied tree (Derman and Kani, 1994). The standard (CRR) binomial tree Figure (4.2) is then replaced by a distorted or implied tree as in Figure (6.3).

The IBT and any other implied tree should satisfy the following conditions:

- the tree correctly reproduces the volatility smile,
- the tree is risk-neutral,
- all transition probabilities are positive and less than one.

The last two conditions will eliminate arbitrage opportunities. The concept of constructing an implied binomial tree, based on the Derman and Kani (1994) algorithm, is explained briefly in subsection (6.2.2). It is also applied in XploRe for constructing implied binomial trees (see subsection 6.2.1). A detailed explanation of implied binomial trees and their construction within XploRe is provided by (Härdle and Zheng, 2001), in e-book Applied Quantitative Finance, ch.7.

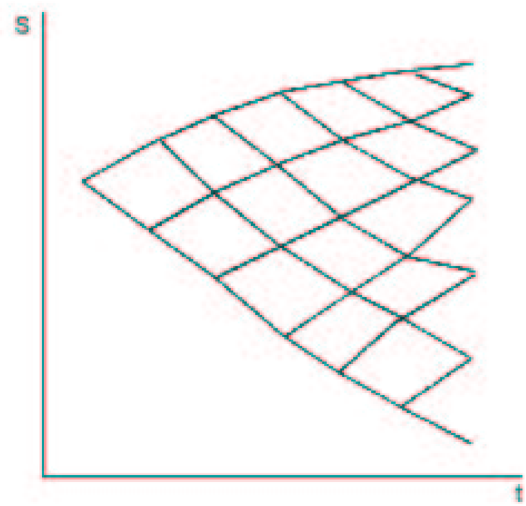

Figure 6.3: Implied binomial tree with volatility $\sigma(S_t, t)$
(Derman, Kani and Chriss, 1996)

When constructing an implied binomial tree, (see subsection 6.2.2) there is only one free parameter, which allows an arbitrary choice for the central node at each level of the tree. In a continuous limit, where there are an infinite number of nodes at each time step, this choice becomes irrelevant. Consequently, there is a unique implied binomial tree that fits option prices in any market. This feature can be disadvantageous, because there is no room for adjustment by inconsistency and/or arbitrage, or by implausible local volatility and probability distribution. One possible solution is to make the structure of the implied tree more flexible by using implied trinomial trees (see subsection 6.3).

6.2.1 Software Application

XploRe offers the possibility to generate the IBT by using either the quantlet `IBTdk`, which is based on the Derman and Kani (1994) method, or the quantlet `IBTbc` based on the Barle and Cakici (1998) method:

```
IBTdk(S,r,lev,expiration,volafunc)
     calculates the stock prices on the nodes of the
     implied tree, the transition probability tree and
     the tree of Arrow-Debreu prices, using Derman and
     Kani's method.

IBTbc(S,r,lev,expiration,volafunc)
     calculates the stock prices on the nodes of the
     implied binomial tree, the transition probability
     tree and the tree of Arrow-Debreu prices, using
     Barle and Cakici's method.

IBTlocsigma(ptree,prob,m,deltat)
     estimates the implied local volatility of each node
     in the implied binomial tree.

IBTvolaplot(loc,step,startpoint,endpoint,m)
     shows the implied local volatility surface
     sigma(S,tau) in the implied tree at different times
     to expiration and stock price levels.
```

In `IBTdk` and `IBTbc`, the input parameter `S` stands for the underlying asset price, `r` for the continuously compounded risk-free interest rate, `lev` for the number of time steps and `expiration` for time to expiration. The last parameter `volafunc` is a string, specifying the name of the function used to define how the Black-Scholes implied volatilities change with strike and expiration.

Both quantlets `IBTdk` and `IBTbc` output the tree of underlying asset prices, contained in a $(n+1)\times(n+1)$ dimensional matrix, the tree of transition probabilities contained in a $(n \times n)$ dimensional matrix and the tree of Arrow-Debreu prices contained in a $(n+1)\times(n+1)$ dimensional matrix.

The following example illustrates how to generate an IBT using the Derman and Kani method. The Black-Scholes implied volatility is assumed to be a linear function of $(S - K)/S$. The IBT corresponds to $\tau = 1$ year and $\Delta t = 0.25$ year.

```
library("finance")
 proc(sigma)=volafunc(K, S, time)
 sigma=0.1+(S-K)/S/10*0.5
 endp
 r=0.03           ; annualized risk-free interest rate
 S=100            ; underlying asset price
 lev=4            ; number of time steps
 expiration=1     ; annualized time to expiration
 ibtree=IBTdk(S,r,lev,expiration,"volafunc")
 ibtree
```

IBTdk.xpl

IBTbc.xpl

The output shows the one year stock price implied binomial tree (ibtree.Tree), the transition probability tree (ibtree.prob) and the Arrow-Debreu price tree (ibtree.lb). The elements at the nth column of the ibtree.Tree matrix correspond to the stock prices at the level $(n - 1)$ of the tree. The element (n, i) of the nth column and ith row of the ibtree.prob matrix corresponds to the transition probability of moving from node (n, i) to node $(n + 1, i + 1)$. Using the Arrow-Debreu prices from the ibtree.lb matrix together with the stock price at respective nodes, a discrete approximation of the implied distribution of the stock prices can be obtained.

```
Contents of ibtree.Tree

 [1,]      100     95.122   89.932    85.21    80.02
 [2,]        0    105.13   100        95.112   89.926
 [3,]        0      0      110.05    105.14   100
 [4,]        0      0        0       115.07   110.06
 [5,]        0      0        0         0      119.91


Contents of ibtree.prob

 [1,]      0.56274   0.58666   0.54526   0.58865
 [2,]      0         0.58921   0.56254   0.58582
 [3,]      0         0         0.57753   0.58956
```

```
 [4,]      0          0          0          0.59625


Contents of ibtree.lb

 [1,]      1          0.43399    0.17804    0.080359    0.032809
 [2,]      0          0.55854    0.48043    0.30495     0.17231
 [3,]      0          0          0.32664    0.40521     0.34238
 [4,]      0          0          0          0.18723     0.31214
 [5,]      0          0          0          0           0.1108
```

The quantlet `IBTlocsigma` needs the following parameters to compute the implied local volatilities in each node of the tree: `ptree` - the stock prices of the nodes generated by IBTdk or IBTbc, `prob` - the transition probability tree, `m` - the highest desired level, `deltat` - the annualized length of one time step. The output is a 3 column matrix, which consists of: i) the stock price at some nodes of the implied binomial tree, ii) time to expiration, and iii) the estimated implied local volatility at these nodes.

The quantlet `IBTvolaplot` plots the implied volatility in the implied binomial tree as a function of strike and time to expiration. The following input parameters are required: `loc` - implied local volatilities computed through `IBTlocsigma`, `step` - the bandwidth of the time interval, `startpoint` and `endpoint` - the lowest and the highest strike dimensions of the volatility surface, `m` - the number of steps to be estimated.

The following example fits an implied five-year tree with 20 levels. The implied local volatility $\sigma_{loc}(S,\tau)$ in the implied tree at different time to expirations and stock price levels is presented in Figure (6.4). The plot confirms the expected result, that the implied local volatility decreases with stock price and increases with time to expiration.

```
library("finance")
 proc(sigma)=volafunc(K, S, time)
 sigma=0.1+(S-K)/S/10*0.5
 endp
 r=0.03                  ; annualized risk-free interest rate
 S=100                   ; initial underlying asset price
 lev=20                  ; number of time steps
 expiration=5            ; annualized time to expiration
 ibtree=IBTdk(S,r,lev,expiration,"volafunc")
 ptree=ibtree.Tree       ; implied tree of stock prices
 prob=ibtree.prob        ; tree of transition probabilities
```

```
deltat=expiration/lev  ; bandwidth of one time step
m=20                   ; highest level
loc=IBTlocsigma(ptree,prob,m,deltat)

startpoint=50          ; lowest strike bound
endpoint=150           ; highest strike bound
n=10                   ; number of estimated steps
step=0.5               ; bandwidth of one time step
IBTvolaplot(loc,step,startpoint,endpoint,n)
```

IBTlocsigma.xpl

IBTvolaplot.xpl

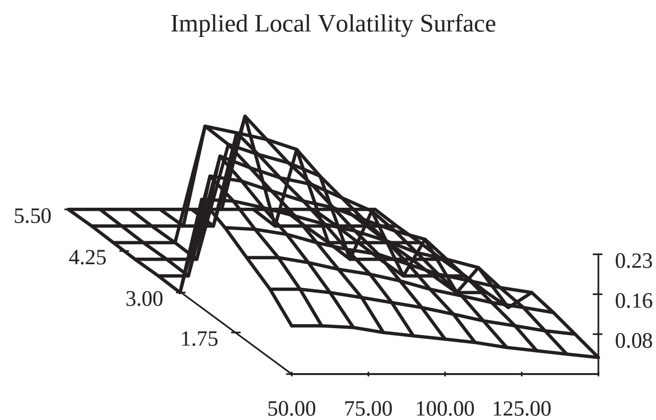

Figure 6.4: Implied binomial local volatility surface using Derman and Kani IBT.

The Barle and Cakici (1998) method can be used in exactly the same way as the Derman and Kani (1994) method. The following output shows the one year stock price tree, the transition probability tree and the tree of Arrow-Debreu prices, when as in the above example, the quantlet `IBTdk` is replaced by `IBTbc` and all other parameters remain the same. The local volatility generated by Barle and Cakici IBT is plotted in Figure (6.5).

```
Contents of ibtree.Tree

 [1,]      100    96.827   90.526   87.603   82.002
 [2,]        0   104.84   101.51    97.731   93.077
 [3,]        0     0      112.23   107.03   103.05
 [4,]        0     0        0      117.02   112.93
 [5,]        0     0        0        0      123.85


Contents of ibtree.prob

 [1,]      0.49006   0.63991   0.35597   0.56528
 [2,]      0         0.38389   0.48864   0.54064
 [3,]      0         0         0.60523   0.48462
 [4,]      0         0         0         0.45512


Contents of ibtree.lb

 [1,]      1         0.50613   0.18089   0.11563    0.049891
 [2,]      0         0.4864    0.61889   0.37802    0.23722
 [3,]      0         0         0.18533   0.37277    0.39353
 [4,]      0         0         0         0.11133    0.23951
 [5,]      0         0         0         0          0.050289
```

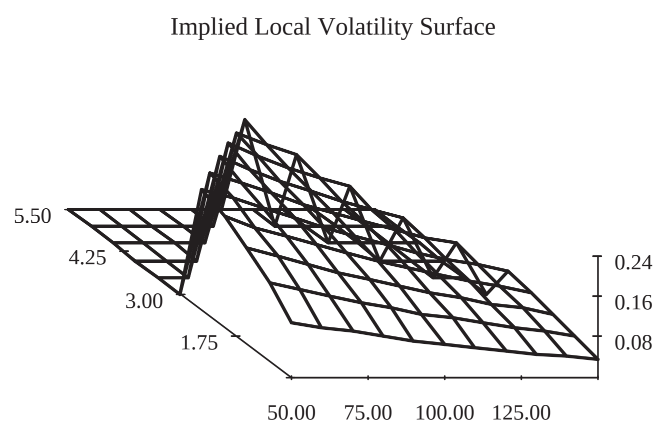

Figure 6.5: Implied binomial local volatility surface using Barle and Cakici IBT.

6.2.2 Derman and Kani Algorithm

Within an implied binomial tree framework, stock prices, transition probabilities, and Arrow Debreu prices at each node are calculated iteratively level by level. The following describes the construction of implied binomial trees using the Derman and Kani (1994) approach.

Assuming that n levels of the tree have already been constructed, the following explains the construction of the next level, $(n+1)$, of the implied binomial tree, which is illustrated in Figure 6.6. The node i, for $i = 1...n$, at level n of the implied tree is denoted with (n, i). As in the case of the regular binomial tree, the asset price $S_{n,i}$ in node (n, i) at time t_n, can either branch upwards to the node (i+1) with the stock price $S_{n+1,i+1}$, or downwards to the node i with the stock price $S_{n+1,i}$ moving from level n to level $n+1$ of the tree.

The Arrow-Debreu price at node (n, i), denoted as $\lambda_{n,i}$ (Figure 6.6), is the price of an option that pays one unit payoff in one and only one state i at level n, otherwise it pays zero. The Arrow-Debreu price is computed by forward induction, as the sum over all paths, from the root of the tree to node (n, i), of the product of transition probabilities at each node, in each path, leading to node (n, i), discounted with the risk-free interest rate.

The next step involves:

1. choosing the positions of the new $n+1$ nodes at time t_{n+1},
2. choosing the n "up" probabilities: $p_{n,n}$, $p_{n,n-1}$,.., $p_{n,i}$,.., $p_{n,1}$, between times t_n and t_{n+1}.

These choices provide $2n+1$ degrees of freedom. To fulfill the risk-neutrality condition, the expected value of the underlying price $E[S_{n,i}]$ for the next period must equal its forward price:

$$E[S_{n,i}] = p_{n,i}S_{n+1,i+1} + (1-p_{n,i})S_{n+1,i} = F_i = exp\{(r-\delta)\Delta t\}S_{n,i}, \quad (6.6)$$

where $F_i = exp\{(r-\delta)\Delta t S_{n,i}$ is the forward price corresponding to $S_{n,i}$, r is the continuous interest rate, δ is the dividend yield and Δt is the time step from t_n to t_{n+1}. There are n of these forwards equations, one for each i. This uses n degrees of freedom.

The tree is also constructed to insure that n independent European vanilla options - $C(K, t_{n+1})$ for call and $P(K, t_{n+1})$ for put - with strike $K = S_{n,i}$ and expiring at time t_{n+1} are priced correctly, i.e. the theoretical values of these options should match their market prices (Derman and Kani, 1994), which uses an additional n degrees of freedom. The theoretical binomial

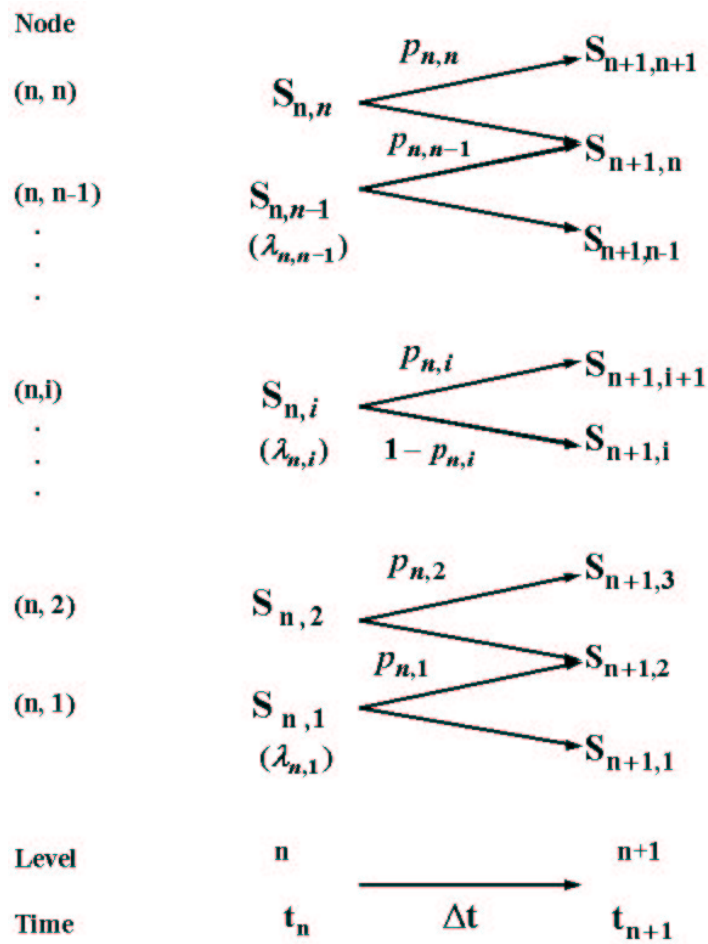

Figure 6.6: Constructing level $(n+1)$ of the implied binomial tree.

value of any option with strike K and expiration at t_{n+1}, is the sum over all nodes i at the (n+1)th level, of the transition probabilities of reaching each node $(n+1, i)$ multiplied by the payoff there and discounted with the risk-free interest rate:

$$C(K, t_{n+1}) = e^{-r\Delta t} \sum_{i}^{n+1} \{\lambda_{n,i} p_{n,i} + \lambda_{n,i+1}(1 - p_{n,i+1})\}\, max\,(S_{n+1,i+1} - K, 0) \tag{6.7}$$

$$P(K, t_{n+1}) = e^{-r\Delta t} \sum_{i}^{n+1} \{\lambda_{n,i} p_{n,i} + \lambda_{n,i+1}(1 - p_{n,i+1})\}\, max\,(K - S_{n+1,i+1}, 0) \tag{6.8}$$

All $\lambda_{n,1}$ are known, because early tree nodes and their transition probabilities have already been implied out at level n. The market price of each option, $C(K, t_{n+1})$ or $P(K, t_{n+1})$, is obtained by interpolation based on a CRR binomial tree with constant volatility. This constant volatility, is the implied Black-Scholes volatility from known market option prices.

This construction leads to $2n$ equations: n for the expected value of the underlying as in (6.6) and n for the option prices as in (6.7) and (6.8). There are $2n + 1$ unknowns: (n+1) stock prices in nodes at level (n+1) of the tree and n transition probabilities. Hence, there is only one degree of freedom left, which is used to ensure that the center node in level n equals the current underlying price. Solving the equations, the tree is advanced only by one time step. The ITB is constructed by repeating this procedure for each time step. Then all stock prices, transition probabilities and Arrow-Debreu prices at any node in the tree will be known.

The implied local volatility $\sigma_{loc}(S_{n,i}, m\Delta t)$ that describes the structure of the second moment of the underlying process at any level m of the tree can then be calculated as a discrete approximation of the following conditional variance

$$\sigma_{loc}(S, m\Delta t) = Var\left(logS_{t+\tau}|S_t = S\right),$$

at $S = S_{n,i}$ and $\tau = m\Delta t$. The formulas for the discrete estimation of this implied volatility, together with a detailed explanation is given in Härdle and Zheng (2001).

One problem with this approach, is that negative probabilities sometimes arise. When a particular probability turns out to be negative, it is necessary to introduce a rule to override the option price responsible for this negative probability. Another shortcoming is that calculating interpolated option prices by the CRR is computationally intensive.

Barle and Cakici (1998) proposed an improvement on the Derman and Kani (1994) algorithm. The major modification is that for the choice of the central node, their algorithm takes the risk-free interest rate into account. A detailed explanation of the Barle and Cakici (1998) algorithm can be found in Härdle and Zheng (2001).

6.3 Implied Trinomial Trees

Implied tree models account for the volatility smile and attempt to price options consistent with the market price. They can be constructed in various ways. Implied binomial trees as discussed in subsection (6.2) have just enough parameters, node prices and transition probabilities to fit the smile. In contrast, a trinomial tree has by construction more parameters, since within one single node the stock price can move to one of three possible future values, each with its own respective probability (Figure 6.7). For example at node i, at time t_n, there are five unknown parameters: two transition probabilities $p_{n,i}$, $q_{n,i}$ and three new node prices S_i, S_{i+1}, and S_{i+2}.

In a risk-neutral trinomial tree, there are two constrains concerning the expected value and the variability of the stock price for these five unknown parameters (Derman, Kani and Chriss, 1996). Consequently, there are three degrees of freedom, which can be used to freely specify three of the parameters, S_i, S_{i+1}, and S_{i+2}, required to fix the tree. No unique trinomial tree, but many equivalent trinomial trees exist, which as Δt goes to zero represent the same continuous theory.

The three degrees of freedom can be used to conveniently specify the state space - the underlying price in every node - and allow the transition probabilities to vary as smoothly as possible across the tree. In other words, there is total freedom over the choice of the state space of an implied trinomial tree. This flexibility is a major advantage of using trinomial trees.

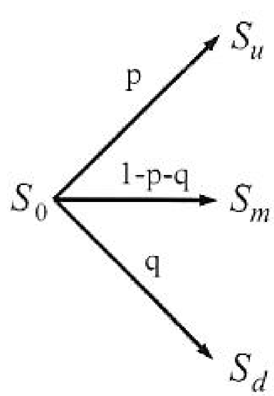

Figure 6.7: A single step of the trinomial tree at t_0. The sum of the three transition probabilities equals one.

A standard trinomial tree represents a constant volatility world and is constructed out of a regular mesh (Figure 6.8). An implied trinomial tree has an irregular mesh, confirming the variation of local volatility with level and time across the tree (Figure 6.9).

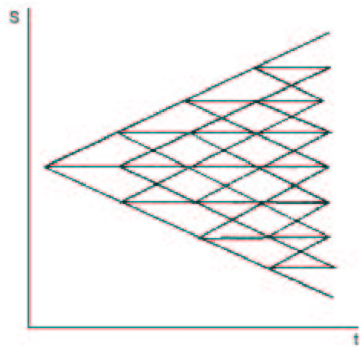

Figure 6.8: Standard trinomial tree. (Derman, Kani and Chriss, 1996)

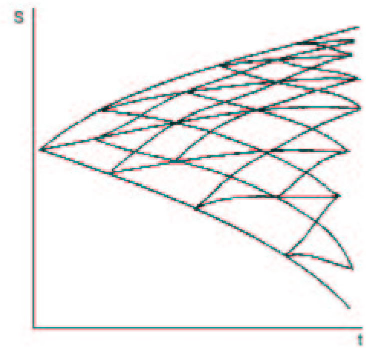

Figure 6.9: Implied trinomial tree. (Derman, Kani and Chriss, 1996)

An implied trinomial tree is usually constructed by two steps (Derman, Kani and Chriss, 1996). In the first step, the initial state space is selected. When implied volatility varies slowly with strike and expiration, a constant volatility trinomial tree can be used. This can be done by combining the two steps of the CRR binomial tree into a single step of a trinomial tree as illustrated in Figure (6.10). Other methods used to build a constant volatility trinomial tree are presented in Derman, Kani and Chriss (1996). If volatility varies significantly with strike and expiration, a trinomial space with proper skew and term structure must be chosen.

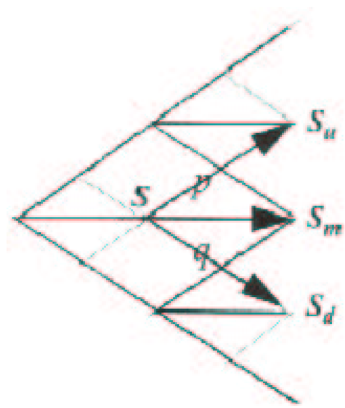

Figure 6.10: Combining two steps of a CCR binomial tree. (Derman, Kani and Chriss, 1996)

By knowing the location of every node, the market forwards and option prices are used in the second step to fix transition probabilities. This is done iteratively to ensure that all European vanilla options will have theoretical values that match their market prices.

Constructing the tree may result in transition probabilities being negative or greater than one, which is inconsistent with rational option prices and allows arbitrage. In this case, a rule must be defined for overwriting the option price which produces incorrect probabilities.

Komorád (2002) explains in detail how to use XploRe for constructing implied trinomial trees. Since this is included in Tutorials-Finance in XploRe, these quantlets are introduced briefly.

`ITT` is used to compute the state space of an implied tree, the probability matrices, the Arrow-Debreu prices and the local volatility matrix. It uses the quantlet `ITTcrr` to compute the option prices. `ITTcrr` builds up a constant volatility trinomial tree by combining two steps of a CRR binomial tree.

`ITT.xpl`

`ITTcrr.xpl`

The simplest way to display results is to apply the quantlet `plotITT`. Once the ITT is constructed, `plotITT` offers the possibility to plot the state space of the ITT, the tree of transition probabilities, the tree of local volatilities, the tree of Arrow Debreu prices and the state price density.

`plotITT.xpl`

More advanced features allow the user to integrate the plot of any trinomial tree with other graphical objects, into one graph. `grITTstsp` returns the state space of an implied trinomial tree with transition probabilities as a graphical object (Komorád, 2002), whereas `grITTspd` generates a state price density of an implied trinomial tree (see Komorád (2002) for a detailed explanation and examples).

`grITTstsp.xpl`

`grITTspd.xpl`

7 Conclusion

This thesis has outlined several fundamental concepts underlying option theory. It consists of four main parts: asset price dynamics, pricing models for plain vanilla options, sensitivities and implied volatilities. There are undoubtedly numerous books explaining these concepts, the logic behind and the mathematics. However, the objective of this thesis has been to focus on the subject of incorporating computational tools within option theory.

It is therefore an interactive tutorial, which offers the reader not only computational examples, but more importantly, the possibility to directly switch to XploRe. The reader is then able to substantiate theoretical results asserted in the thesis, or alternatively is able to apply XploRe in order to pose questions and receive satisfactory answers. By using graphical tools within XploRe for sensitivities and volatility surfaces, it is then possible for the reader to obtain a feeling for the impact of different factors, especially volatility, on the option price.

This thesis has taken for granted the fact that the reader has prior knowledge on options and therefore does not address several basic but important questions, for example: 'why do options exists', 'what gives them value' and 'how are these products used for trading and hedging'? It also does not cover extensions of the classical Black-Scholes model to incorporate such features as stochastic volatility, or extensions to higher dimensionality for options on more than one underlying asset. It neither discusses exotic options, although there is an awareness of their recent expansion in trading, or current pricing theory that is based on martingale methods, which adopt a unified approach to all forms of derivatives, options included. Nevertheless, it covers the basics of option evaluation, introduced in the equity world, including numerous examples from XploRe.

Bibliography

Alexander, C. (1996). *Risk Management and Analysis* (1 ed.), John Wiely & Sons, New York.

Barley, S. and Cakici, N. (1998). How to Grow a Smiling Tree, *The Journal of Financial Engineering* 7, pp. 127–146.

Barone-Adessi, G. and Whaley, R.E. (1987). Efficient Analytic Approximation of American Option Values, *Journal of Finance* 42, pp. 301–320.

Black, F. and Scholes, M. (1973). The pricing of option and corporate liabilities, *Journal of Political Economy* 81, pp. 637–659.

Campa, J. M. and Chang, K. (1998). The Forecasting Ability of Correlations Implied in Foreign Exchange Options, *Journal of International Money and Finance* 17, pp. 855–880.

Chiras, D. P. and Manaster, S. (1976). The Information Content of Option Prices and a Test of Market Efficiency, *Journal of Financial Economics* 6, pp. 213–234.

Cox,C. J., Ross A. S. and Rubinstein, M. (1979). Option Pricing: A Simplified Approach, *Journal of Financial Economics* 7, pp. 229–263.

Cont, R. and da Fonseca, J. (2002). Dynamics of implied volatility surfaces, *Quantitative finance* 2, pp. 45–60.

Cuthbertson, B. (1996). *Quantitative Financial Economics*, John Wiely & Sons, New York.

Derman, E. and Kani, I. (1994). The Volatility Smile and Its Implied Tree, *Quantitative Strategies Research Notes* Goldman Sachs, http://www.gs.com/qs/

Derman, E. and Kani, I. (1996). Implied Trinomial Trees of Volatility Smile, *Quantitative Strategies Research Notes* Goldman Sachs, http://www.gs.com/qs/

Dumas B., Fleming J. and Whaley, R. E. (1996). Implied Volatility Functions: Empirical Tests, *NBER Working Paper* 5500.

Dupire, B. (1994). Pricing with a Smile, *Risk* 7, pp. 18–20, http://www.gs.com/qs/

Franks, J.R. and Schwarz, E. J. (1991). The stochastic behaviour of market variance impied in the price of index options, ***Economic Journal* 101, pp. 1460–1475.

Fengler M. R., Härdle W. and Schmidt, P. (2002). The Analysis of Implied Volatilities, in: W. Härdle, T. Kleinow, G. Stahl, *Applied Quantitative Finance*, Springer e-book, http://www.xplore-stat.de/ebooks/ebooks.html

Franke, J., Härdle, W. and Hafner, Ch. (2001). *Statistics of Financial Market* (1. ed.), Springer e-book, http://www.xplore-stat.de/ebooks/ebooks.html

Gibson, R. (1993). *Option Valuation* (1 ed.), McGraw-Hill series in finance.

Gourieroux J.J. (2001). *Financial Econometrics* (1 ed.), Princeton University Press.

Härdle, W.and Müller, M. and Sperlich, S. and Werwatz, A. (2001). *Non- and Semiparametric Modelling*, e-book, http://www.xplore-stat.de/ebooks/ebooks.html

Härdle, W. and Zheng, J. (2001). How presice are pricing distributions predicted by implied binomial trees? in: *Applied Quantitative Finance*, pp. 145-170, e-book, http://www.xplore-stat.de/ebooks/ebooks.html

Härdle, W. and Schmidt, P. (2000). Common factors governing VDAX movements and the maximum loss, *Humboldt University Working Paper****.

Härdle, W. (1994). *Applied Nonparametric regression*, Cambridge University Press, e-book, http://www.xplore-stat.de/ebooks/ebooks.html

Heynen, R. (1993). An empirical investigation of observed smile paterns, *Review Future Markets* 13, pp. 317–353.

Heynen, R. and Kemma, K. and Vorst, T. (1994). Analysis of the term structure of implied volatilities, *J. Financial Quant. Anal.* 29, pp. 31-56.

Hodges, H. M. (1996). Arbitrage bounds on implied volatility strike and term structure of European style options, *Journal of Derivatives* 3, pp. 23–35.

Hull, J.C. (2000). *Options, Futures, and Other Derivative Securities* (4 ed.), Prentice-Hull International, Inc.

Hull, J. C. and White, A. (1987). The pricing of Options on Assets with Stochastic Volatilities, *Journal of Finance* 42, pp. 281–300.

Jarrow, R. and Turnbull, S. (1996). newblock *Derivative Securities*(1 ed.), International Thomson Publishing.

Jarrow, R. C. and Rudd, A. (1982). newblock Approximate option valuation for arbitrary stochastic processes, *Journal of Financial Economics* 10, pp. 347–369.

Jiang, G. J. (1998). Jump-Diffusion Model of Exchange Rate Dynamics - Estimation via inderect inference, *University of Groningen, Netherlands, Working Papers*, http://www.ub.rug.nl/eldoc/som/a/98A40

Jorion, Ph. (1995). Predicting Volatility in the Foreign Exchange Market *Journal of Finance* 50, 507–528.

Jorion, Ph. (2001). *Value at Risk: The new Benchmark for Managing Financial Risk* (2 ed.), John Wiely & Sons, New York.

Komorád K. (2002). *Implied Trinomial Trees and Their Implementation with XploRe*, http://www.xplore-stat.de/tutorials/quantstart.html

Küchler U. and Sørensen, M. (1997). *Exponential Families of Stochastic Processes* (1 ed.), Springer.

Kwock, K. Y. (1998). *Mathematical Models of financial Derivatives* (1 ed.), Springer.

Latane, H. and Rendleman, R. J. (1976). Standard Deviation on Stock Price Ratios Implied by Option Premia, *Journal of Finance* 31, pp. 369–382.

Lo, A. (1988). Maximum Likelihood Estimation of Generalized Ito processes with discretely sampled data, *Econometric Theory* 4, pp. 231–247.

MacMillan, L. W. (1986). Analytic Approximation for the American Put Option, *Advances in Futures and Options Research* 1, pp. 119–139.

Merton, R. C. (1973). Theory of rational option pricing, *Bell Journal of Economics and Managment Sciences* 4, pp. 141–183.

Merton, R. C. (1976). Option pricing with discontinuous returns, *Bell Journal of Financial Economics* 3, pp. 145–166.

Neftci, N. S. (1998). *An Introduction to the Mathematics of financial Derivatives*
(2 ed.), Academic Press.

Nelder, J. A. and Mead, R. (1965). A simplex Method for Function Minimization, *Computer J.* 7, pp. 308–318.

Press, J. (1967). A compound events model for security prices, *Journal of Business* 40, pp. 317–335.

Rebonato, R. (1999). *Volatility and Correlation in the Pricing of Equity, FX and Interest Rate Options* John Wiely & Sons, New York.

Rubinstein, M. (1994). Implied Binomial Trees, *Journal of Finance* 49, 3 (July 1994), pp. 771–818.

Tompkins, R. (1994). *Options Explained*, (1. ed.) MacMillan Press Ltd.

Whaley, R. E. (1982). Valuation of American Call Options on Dividend-Paying Stocks: Empirical Tests, *Journal of Financial Economics* 10, pp. 29–58.

Wilmott, P., (1997). *Derivatives: The Theory and Practice of Financial Engineering* (2 ed.), John Wiley & Sons Ltd.

Wilmott, P., Howison S. and Dewynne, J. (1997). *The Mathematics of financial Derivatives: a Student Introduction* (2 ed.), Cambridge University Press.

Zhu, Y. and Avellaneda, M. (1997). An E-ARCH model for the term structure of the implied volatility of FX options. *Appl. Math. Finsnce* 4, pp. 81–100.