

UNIVERSIDAD SAN FRANCISCO DE QUITO

Simulación numérica de integrales estocásticas y aplicaciones a
las Finanzas

Julio Fernando Ortega Andrade

Tesis de grado presentada como requisito para la obtención del
título de Maestría en Matemáticas Aplicadas

Quito

Abril del 2006

Universidad San Francisco de Quito
Colegio de Graduados

HOJA DE APROBACION DE TESIS

Simulación numérica de integrales estocásticas y aplicaciones a
las Finanzas

Julio Fernando Ortega Andrade

Carlos Jiménez, Ph.D
Director de Tesis (firma)

Valentina Aguilar ,Ph.D.
Miembro del Comité (firma)

Carlos Jiménez,Ph.D.
Director de la Maestría (firma)

Víctor Víteri,Ph.D.
Director del Colegio de Graduados (firma)

Quito, Abril del 2006

© Derechos de Autor
Julio Fernando Ortega Andrade
2006

Dedicatoria

A Mateo: mi alegría más profunda.

Agradecimiento

A todos los profesores de la Maestría en Matemáticas Aplicadas de la USFQ, de manera especial a Carlos Jiménez, Director del presente trabajo y a Valentina Aguilar, lectora del mismo, por su invaluable e incondicional ayuda.

Un agradecimiento especial a mi amigo David Vera por toda su ayuda y apoyo.

Julio Fernando Ortega Andrade

Quito, Abril del 2006

Resumen

Las *ecuaciones diferenciales estocásticas* juegan un rol prominente en áreas tales como: biología, química, epidemiología, mecánica, economía y finanzas. En lo que sigue las abreviaremos por EDE para referirnos a tales ecuaciones y usaremos las aplicaciones a finanzas (*precios de acciones*) para ejemplificarlas. El objetivo principal de este trabajo es presentar métodos de simulación para procesos estocásticos, útiles en finanzas, como son el movimiento browniano fraccionario y los procesos de Lévy tomando en cuenta que para estos procesos no es fácil encontrar la suficiente literatura en lo que se refiere a simulación. El objetivo específico es aplicar estos procesos estocásticos a las EDE que aparecen en finanzas. Para esto hace falta conocer el método de Euler para Ecuaciones Diferenciales Ordinarias (EDO), el concepto de variable aleatoria, movimiento browniano, movimiento browniano fraccionario, procesos de Levy y el método Euler-Maruyama. Las variables aleatorias son simuladas con un generador de números aleatorios y los valores esperados se aproximan usando promedios.

Abstract

The *stochastic differential equations* play a prominent role in areas like: biology, chemistry, epidemiology, mechanics, economy and finance. We'll refer to them as SDE and we are going to illustrate them through examples in finance. Taking into account that there is only limited information on the subject, the main goal of this work is to present simulation methods for stochastic processes, which are useful in financial analysis, such as fractional Brownian motion and Levy processes. More specifically the goal is to apply these processes to SDEs that appear in financial applications. In order to do that, we need to know the Euler's method for Ordinary Differential Equations (ODE) and the concept of random variable. We'll also discuss concepts like: brownian motion, fractional brownian motion, Levy processes and the Euler-Maruyama method. The random variables are generated with a random number generator and the expected values are approximated using averages.

Índice general

Dedicatoria	IV
Agradecimiento	V
Resumen	VI
Abstract	VII
1. Introducción	1
2. Movimiento browniano	2
2.1. Definición	2
2.2. Simulación numérica del movimiento browniano	3
2.2.1. Algoritmo 1	3
2.2.2. Programa bpath2.r	4
2.3. Aplicación del movimiento browniano a Mercados de Acciones	4
2.3.1. Proceso de Itô:	5
2.3.2. Lema de Itô	6
2.3.3. Demostración de la solución analítica	6
2.3.4. Simulación numérica de una función sobre un camino browniano	7
2.3.5. Algoritmo 2	8
2.3.6. Programa bpath3.r	9
2.4. Integrales estocásticas	10
2.4.1. Introducción	10
2.4.2. Integral de Itô e integral de Stratonovich	11
2.5. Simulación numérica de una integral estocástica	14
2.5.1. Algoritmo 3	14
2.5.2. Programa stint.r	14

	IX
2.6. Solución numérica de una EDE	16
2.7. Método de Euler-Maruyama EM	16
2.7.1. Algoritmo 4	17
2.7.2. Programa em.r	19
3. Movimiento browniano fraccionario	22
3.1. Definición	22
3.2. Ruido fraccionario Gaussiano	23
3.3. Autosimilaridad y memoria a largo plazo	23
3.4. Simulación del movimiento browniano fraccionario	24
3.4.1. Método de Weirstrass	24
3.4.2. Programa fbmweirstrass.r	25
3.4.3. Método de Hosking	25
3.4.4. Programa hosking.r	28
3.5. Aplicación del fBm a las finanzas	30
3.6. Funciones sobre caminos definidos por fBm	31
3.6.1. Simulación numérica de una función sobre caminos definidos por fBm	31
3.6.2. Algoritmo 5	32
3.6.3. Programa fBmf.R	33
3.7. Integración usando el fBm	33
3.8. Simulación numérica de una integral estocástica usando fBm	34
3.8.1. Programa intfBm.R	35
4. Proceso de Lévy	36
4.1. Definición de un Proceso de Levy	37
4.2. Simulación numérica de un proceso de Lévy	37
4.2.1. Proceso de Poisson	37
4.2.2. Propiedades del Proceso de Poisson	38
4.2.3. Proceso de Poisson compuesto	38
4.2.4. Algoritmo para la simulación de un proceso de Poisson (método 1)	38
4.2.5. Programa pprocc1	39
4.2.6. Algoritmo para la simulación de un proceso de Poisson (método 2)	40
4.2.7. Programa pprocc2	40
4.2.8. Algoritmo para la simulación de un proceso de Poisson compuesto	40

	x
4.2.9. Programa pprocce	41
4.2.10. Algoritmo para la simulación de un proceso simple de Lévy . . .	42
4.2.11. Programa levy1	43
4.3. Aplicación a los mercados de acciones	44
5. Conclusiones	45
Bibliografía	46
A. Método de Euler	47

Capítulo 1

Introducción

Las ecuaciones diferenciales estocásticas, que en lo que sigue las abreviaremos con EDE, juegan un rol importante en áreas tan diversas como biología, química, mecánica o finanzas.

El trabajo original de esta tesis se refiere a la presentación e implementación de varias formas de integrar numéricamente EDE usando varios tipos de procesos estocásticos.

También esta tesis muestra algunas posibles formas de resolver numéricamente ciertas ecuaciones diferenciales estocásticas y para esto se vale de aplicaciones a las finanzas.

En el **capítulo 2** se introduce el *movimiento browniano* y una forma de simularlo numéricamente, además se incluyen temas relacionados a este movimiento como el Proceso de Itô y las integrales estocásticas de Itô y Stratonovich; igualmente se presenta sus respectivas simulaciones numéricas con un análisis comparativo del error con respecto a su valor exacto.

En el **capítulo 3** se introduce el *movimiento browniano fraccionario*, como alternativa al movimiento browniano, igualmente se lo simula numéricamente, usando dos métodos para este fin, luego se calcula la integral estocástica respectiva y se lo usa para resolver cierto tipo de EDE que aparecen en las finanzas.

Finalmente el **capítulo 4** empieza definiendo los *procesos de Levy* y sus respectivas propiedades, lo que sigue es la simulación numérica de un proceso de Levy que incluye la simulación tanto de procesos de Poisson como de procesos de Poisson compuestos para terminar usando los procesos de Levy en la resolución de un EDE, surgida otra vez, en las finanzas.

Capítulo 2

Movimiento browniano

El movimiento browniano lleva su nombre gracias al botánico Robert Brown quien, en 1820, fue el primero observar el movimiento irregular de granos de polen inmersos en agua. En 1905 Einstein, a pesar de ignorar el descubrimiento, predijo su existencia de forma puramente teórica. Cinco años antes, L.Bachelier lo había empleado para modelar *precios de acciones* en el tiempo, a través de la fórmula :

$$S(t) = S(0) + \sigma W(t)$$

donde $S(0)$ es el *precio inicial* y σ es la *volatilidad del mercado* que será definida más adelante. Esto ejemplifica como decisiones individuales de mercado pueden determinar el precio final de este.

El movimiento browniano fue rigurosamente construído por N. Wiener en 1920 como una familia de funcionales sobre el espacio $C = C_0([0, \infty), \mathfrak{R})$ de funciones continuas a variable real sobre $[0, \infty)$ que tienden a cero. Equipó a C con una *medida* Gaussiana, llamada *de Wiener* en honor a su creador.

2.1. Definición

Un movimiento escalar Browniano estándar o proceso estándar de Wiener, sobre $[0, T]$ es un proceso estocástico $W(t)$ con índice $t \in [0, T]$ y que satisface las siguientes tres condiciones, ver [5]

1. $W(0) = 0$

2. Para

$$0 \leq s < t \leq T$$

la variable aleatoria dada por el incremento $W(t) - W(s)$ tiene una distribución normal con media *cerro* y varianza $t-s$ o lo que es lo mismo

$$W(t) - W(s) \sim \sqrt{t-s} N(0,1)$$

3. Para todo s, t, u, v tq.

$$0 \leq s < t < u < v \leq T$$

los incrementos $W(t) - W(s)$ y $W(v) - W(u)$ son independientes

Para propósitos de simulación es útil discretizar el movimiento browniano, es decir donde $W(t)$ se especifica para valores discretos de t .

Para esto sea $\delta t = T/N$ para algún entero positivo N y sea W_j que denota a $W(t_j)$ y $t_j = j\delta t$ con $j=0,1,2,\dots,N$.

La condición **1**, de la definición, dice que $W(0)=0$ con probabilidad 1, y las condiciones **2** y **3** nos dicen que se cumple:

$$W_j = W_{j-1} + dW_j, j = 1, 2, \dots, N \quad (2.1)$$

donde cada dW_j es una variable aleatoria independiente de la forma $\sqrt{\delta t}N(0,1)$, ver [7].

2.2. Simulación numérica del movimiento browniano

Antes de abordar las EDE en si, primero consideramos la discretización del movimiento browniano, lo cual nos servirá más adelante cuando construyamos las EDE. Para esto usemos el siguiente algoritmo, ver [4]. En los algoritmos que siguen en lugar de δt se usará dt .

Los programas requeridos en este trabajo están escritos en **R**, ver [8].

2.2.1. Algoritmo 1

1. Elegimos un intervalo $[0, T]$, $N =$ número de pasos y definimos $dt = T/N$ el tamaño del paso.
2. Con el propósito de que el experimento se pueda repetir se fija, con la instrucción `set.seed()`, la semilla inicial del generador de números aleatorios.
3. Creamos una secuencia $\{x\}$ de 0 a $T-dt$ con paso de tamaño dt

4. Luego generamos N números aleatorios $N(0,1)$, los escalamos, multiplicándolos por \sqrt{dt} , y los asignamos a la variable dw .
5. A continuación usamos la instrucción `cumsum` para construir el vector W que contiene los puntos del movimiento browniano generado, usando la definición de $W_j = W_{j-1} + dW_j$.
6. Finalmente graficamos el movimiento browniano, usando la secuencia x , de paso dt , y el vector W y unimos los puntos $(t_i, W(t_i))$ con segmentos de recta.

La figura (2.1) muestra el resultado obtenido para `set.seed=200`.

El programa `bpath2.r` realiza lo antes descrito.

2.2.2. Programa `bpath2.r`

```
bpath2=function(T=1,N=500) {
  set.seed(200)
  dt=T/N
  dw=0
  w=0
  dw=sqrt(dt)*rnorm(N)
  w=cumsum(dw)
  x=seq(0,T-dt,dt)
  return(T,N,w,dw,x,plot(x,w,type='l',main="Camino browniano
discretizado",xlab="t",ylab="W(t)"))
}
```

2.3. Aplicación del movimiento browniano a Mercados de Acciones

Una aplicación del movimiento browniano a las finanzas aparece en la EDE que modela el movimiento de *precios $S(t)$ de acciones*.

Este modelo asume que el cambio relativo o *retorno* (cantidad ganada en proporción directa al capital invertido) $\frac{dS}{S}$ de una *acción* (unidad de capital a nombre del poseedor y que indica propiedad sobre una empresa) en el intervalo de tiempo dt está compuesto

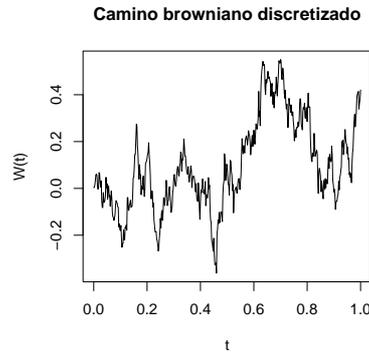


Figura 2.1: Camino browniano

de un término determinístico $\mu S(t)$ más fluctuaciones estocásticas de la forma $\sigma dW(t)$ como se muestra en la siguiente fórmula :

$$dS(t) = \mu S(t)dt + \sigma S(t)dW(t), S(0) = S_0 \quad (2.2)$$

Esta ecuación diferencial estocástica es lineal en $S(t)$, el término $\mu S(t)$ contiene μ : la *tasa esperada de retorno* que se define como el retorno de una inversión establecida como un porcentaje del monto invertido y $\sigma S(t)$ es el término que contiene σ : la *volatilidad* que se define como la desviación estándar del cambio en el valor de un instrumento financiero con un horizonte temporal específico del mercado.

En lo que sigue asumiremos que tanto μ como σ son constantes.

La solución analítica de la EDE (2.2) es :

$$S(t) = S(0)\exp\left(\left(\mu - \frac{\sigma^2}{2}\right)t + \sigma W(t)\right) \quad (2.3)$$

La justificación de esta solución analítica está en el lema de Itô, ver [9]. Antes de exhibir este lema, definamos primero que es un proceso de Itô.

2.3.1. Proceso de Itô:

$X(t)$ se dice sigue un proceso de Itô si se puede escribir como:

$$dX(t) = a(X(t), t)dt + b(X(t), t)dW(t), X(t_0) = X(0) \quad (2.4)$$

o en su forma integral

$$X(t) = X(0) + \int_{t_0}^t a(X(s), s)ds + \int_{t_0}^t b(X(s), s)dW(s) \quad (2.5)$$

La primera integral de la parte derecha de (2.5) es una integral de Reimann, mientras que la segunda es una integral estocástica que definiremos en la sección 2.4.

2.3.2. Lema de Itô

Supongamos que $X(t)$ sigue un proceso de Itô, es decir se puede escribir como sigue:

$$dX(t) = a(X(t), t)dt + b(X(t), t)dW(t)$$

y sea $g(x, t)$ una función con derivadas $\frac{\partial g}{\partial x}$, $\frac{\partial^2 g}{\partial x^2}$, $\frac{\partial g}{\partial t}$ continuas. Entonces $Y(t) = g(X(t), t)$ sigue un proceso de Itô con el mismo movimiento browniano $W(t)$:

$$dY(t) = \left(\frac{\partial g}{\partial x}a + \frac{\partial g}{\partial t} + \frac{1}{2} \frac{\partial^2 g}{\partial x^2} b^2 \right) dt + \frac{\partial g}{\partial x} b dW(t) \quad (2.6)$$

tanto las derivadas de g como los coeficientes a y b en general dependen de los argumentos $(X(t), t)$.

2.3.3. Demostración de la solución analítica

Veamos a continuación la demostración de (2.3), para esto sea $W(t)$ un movimiento browniano, se tiene que $W(t)$ es un proceso de Itô, además, de $X(t) = W(t)$ se sigue la EDE trivial $dX(t) = dW(t)$ y de aquí se tiene que los coeficientes del proceso son $a=0$ y $b=1$ por lo tanto sea $X(t) = W(t)$ y definamos $Y(t)$ como sigue:

$$Y(t) = g(X(t), t) = S(0) \exp \left(\left(\mu - \frac{1}{2} \sigma^2 \right) t + \sigma W(t) \right) \quad (2.7)$$

y por el Lema de Itô tenemos:

$$\begin{aligned} dY(t) &= \left(\mu - \frac{1}{2} \sigma^2 \right) Y(t) dt + \frac{1}{2} \sigma^2 Y(t) dt + \sigma Y(t) dW(t) \\ &= \mu Y(t) dt + \sigma Y(t) dW(t) \end{aligned}$$

Consecuentemente el proceso :

$$S(t) = S(0) \exp \left(\left(\mu - \frac{1}{2} \sigma^2 \right) t + \sigma W(t) \right) \quad (2.8)$$

resuelve la EDE lineal (2.2). Por lo tanto concluimos que la solución de $S(t)$ de (2.2) es una función del movimiento browniano $W(t)$:

$$u(W(t)) = S(0) \exp \left(\left(\mu - \frac{1}{2} \sigma^2 \right) t + \sigma W(t) \right) \quad (2.9)$$

2.3.4. Simulación numérica de una función sobre un camino browniano

Si, para efectos de la simulación, elegimos en (2.9), $\mu = \frac{9}{8}$, $\sigma = \frac{1}{2}$ y $S(0) = 1$ obtenemos la función

$$u(W(t)) = \exp\left(t + \frac{1}{2}W(t)\right) \quad (2.10)$$

por supuesto esta solución considera el movimiento browniano para t continuo; pero lo que nos interesa es calcular numéricamente $u(W_t)$ con el menor error de estimación posible. Este error se calcula sacando la raíz cuadrada a la suma de las diferencias entre el valor esperado y el valor promedio muestral de $u(W(t))$, en cada t , elevadas al cuadrado. El valor esperado antes mencionado es $E[u(W(t))] = \exp(9t/8)$ y se lo demostrará a continuación.

Dada la función

$$S(t) = S(0)\exp\left(\left(\mu - \frac{1}{2}\sigma^2\right)t + \sigma W(t)\right)$$

su valor esperado es:

$$E[S(t)] = S(0)\exp\left(\left(\mu - \frac{1}{2}\sigma^2\right)t + \sigma W(t)\right) = S(0)\exp\left(\left(\mu - \frac{1}{2}\sigma^2\right)t\right) * E[\exp(\sigma W(t))]$$

Si llamamos $X = W(t)$ de aquí se tiene que

$$\begin{aligned} E[\exp(\sigma W(t))] &= E[\exp(\sigma X)] = \int_{-\infty}^{\infty} e^{\sigma x} \frac{1}{\sqrt{2\pi t}} e^{-\frac{x^2}{2t}} dx = \\ &= \frac{1}{\sqrt{2\pi t}} \int_{-\infty}^{\infty} e^{\sigma x} e^{-\frac{x^2}{2t}} dx = \\ &= \frac{1}{\sqrt{2\pi t}} \sqrt{2\pi t} e^{\frac{\sigma^2}{2t}} = e^{\frac{\sigma^2}{2t}} \end{aligned}$$

Por lo tanto:

$$E[S(t)] = S(0)e^{(\mu - \frac{1}{2}\sigma^2)t} E[\exp(\sigma W(t))] = S(0)e^{\mu t} e^{-\frac{1}{2}\sigma^2 t} e^{\frac{1}{2}\sigma^2 t}$$

y de aquí :

$$E[S(t)] = S(0)e^{\mu t}$$

que es lo que queríamos probar.

El Algoritmo 2 nos permite evaluar la función $u(W(t)) = e^{(t + \frac{1}{2}W(t))}$ a lo largo de 1000 caminos brownianos discretizados al igual que el error en que se incurre.

2.3.5. Algoritmo 2

1. Fijamos la semilla, $set.seed()$, de la generación de los números aleatorios, para poder repetir el experimento.
2. Definimos T =extremo superior del intervalo $[0,T]$, N =número de pasos, $dt=T/N$ tamaño del paso y M =número de caminos brownianos.
3. Construimos una secuencia (t_j) de t que empiece en dt y termine en T , con pasos de tamaño dt .
4. Construimos la matriz dW de $M \times N$ tal que $dW(i,j)$ contiene el incremento dW_j para el camino i -ésimo.
5. Usando la función $cumsum$ sobre la dimensión M construimos la matriz W de dimensión $M \times N$ de tal forma que la i -ésima fila contenga el i -ésimo camino.
6. Ahora construimos la matriz U de dimensión $M \times N$ de tal forma que la i -ésima fila contenga la función $u(W(t))$ evaluada en el i -ésimo camino.
7. A continuación formamos el vector $Umean=mean(U)$ de $1 \times N$ que calcula los promedios por columna y cuya j -ésima entrada contiene el promedio de los $u(W(t_j))$.
8. Finalmente graficamos el promedio de las $u(W(t))$ sobre esos caminos y, la función $u(W(t_j))$ para cinco caminos cualesquiera (esto para mostrar que individualmente los caminos no son *suaves* pero en promedio si los son, tomando en cuenta que se sabe los caminos brownianos no son diferenciables en ningún punto) y la función $f(t)=exp(9t/8)$ que es el valor esperado de la solución de (2.9) para $\mu = \frac{9}{8}$. Una función se dice *lisa* o *suave* si su derivada es continua, ver [10].
9. También se calcula en **averr**, el error tal como se lo definió anteriormente. Es decir la raíz cuadrada de la suma de las diferencias entre el valor esperado exacto $f(t)=exp(9t/8)$ y la media muestral $u(W(t))$, para cada t , elevadas al cuadrado.

Se prueba, experimentalmente, que si se incrementa el tamaño de la muestra el error **averr** disminuye; esto nos sugiere usar el promedio sobre una muestra grande de caminos.

A continuación va el programa en **R**, sus correspondientes gráficos para $M=200$ y $M=1000$, donde M es el número de caminos, y el error **averr** para diferentes M .

2.3.6. Programa bpath3.r

```

bpath3=function(T=1,N=500,M=1000) {
  set.seed(100)
  dt=T/N
  t=seq(dt,1,by=dt)
  A=matrix(nrow=M,ncol=N)
  B=matrix(nrow=M,ncol=N)
  U=matrix(nrow=M,ncol=N)
  Umean=0 for (i in 1:M){
    dw=sqrt(dt)*rnorm(N)
    w=cumsum(dw)
    A[i,]=dw
    B[i,]=w
    f=exp(t+0.5*w)
    U[i,]=f }
  for (i in 1:N){
    Umean[i]=mean(U[,i])
  }
  dif=(Umean-exp(9*t/8))
  averr=sqrt(t(dif)%*(dif))
  return(plot(t,Umean,type="l",col="blue",xlim=c(0,1),
  ylim=c(-1,5)),lines(c(0,t), +c(1,U[1,])),lines(c(0,t),c(1,U[2,])),
  +lines(c(0,t),c(1,U[3,])), +lines(c(0,t),c(1,U[4,])),lines(c(0,t),
  +c(1,U[5,])), +lines(c(0,t),exp((9/8)*c(0,t)), col="red"))
}

```

	M=1000	M=2000	M=3000	M=4000
averr	1.014	0.598	0.225	0.097

Cuadro 2.1:

El cuadro (2.1) muestra que el error disminuye si incrementamos el número de caminos brownianos M .

En la Figura (2.2) se nota que para $M=200$ el promedio $u(W(t))$ difiere ligeramente del

valor esperado $f(t)=exp(9t/8)$ mientras que la Figura (2.3) muestra que para $M=1000$ la curva aproximada se acerca a f .

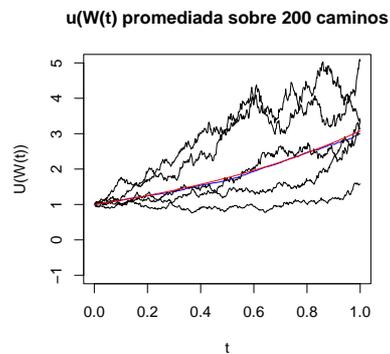


Figura 2.2: Función sobre 200 caminos brownianos

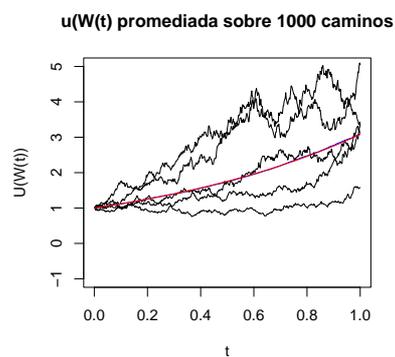


Figura 2.3: Función sobre 1000 caminos brownianos

2.4. Integrales estocásticas

2.4.1. Introducción

Sabemos, de la sección **2.3**, que el precio $S(t) = X(t)$ de una *acción* evoluciona de acuerdo a la EDE :

$$dX(t) = \mu X(t)dt + \sigma X(t)dW(t), X(0) = X_0 \quad (2.11)$$

La forma integral de esta EDE escalar puede ser escrita como:

$$X(t) = X(0) + \int_0^t f(X(s))ds + \int_0^t g(X(s))dW(s), 0 \leq t \leq T \quad (2.12)$$

donde $f = \mu X(t)$ y $g = \sigma X(t)$ son funciones escalares y $X(0)$ es una variable aleatoria. La primera integral está tomada en el sentido de Reimann y la segunda integral está tomada con respecto al camino browniano $W(t)$, sin embargo surge la pregunta sobre lo que significa $\int_0^t g(X(s))dW(s)$. Para entenderlo haremos a continuación una analogía con la Integral de Reimann.

2.4.2. Integral de Itô e integral de Stratonovich

Dada una función h continua en un intervalo cerrado y acotado, la integral $\int_0^T h(t)dt$ de Reimann, puede ser aproximada por la suma:

$$\sum_{j=0}^{N-1} h(t_j)(t_{j+1} - t_j) \quad (2.13)$$

donde $t_j = j\delta t$ con $\delta t = T/N$ y $j = 0, 1, \dots, N$, es la discretización del intervalo $[0, T]$. De hecho la integral puede ser definida tomando el límite $\delta t \rightarrow 0$ en (2.13). De forma similar, podemos considerar la suma de la forma:

$$\sum_{j=0}^{N-1} h(t_j)(W(t_{j+1}) - W(t_j)) \quad (2.14)$$

donde h es un proceso estocástico *progresivamente medible*, ver [4] para el cual si tomamos el límite, por analogía con (2.13), puede ser vista como una aproximación a la integral estocástica $\int_0^T h(t)dW(t)$. Aquí estamos integrando h con respecto al movimiento browniano estándar $W(t)$. Cuando estudiamos la integral de Reimann, vemos que una alternativa a (2.13) está dada por:

$$\sum_{j=0}^{N-1} h\left(\frac{t_j + t_{j+1}}{2}\right)(t_{j+1} - t_j) \quad (2.15)$$

en la cual se elige, en lugar del punto izquierdo del intervalo $[t_j, t_{j+1}]$ el punto medio de dicho intervalo y si tomamos el límite cuando $\delta t \rightarrow 0$, esta última expresión también es una aproximación a $\int_0^T h(t)dt$.

La correspondiente alternativa estocástica a (2.15) es :

$$\sum_{j=0}^{N-1} h\left(\frac{t_j + t_{j+1}}{2}\right)(W(t_{j+1}) - W(t_j)) \quad (2.16)$$

Si tomamos $h(t) \equiv W(t)$ en la suma (2.16) se requiere que $W(t)$ sea evaluada en $t = \frac{t_j+t_{j+1}}{2}$. Se puede mostrar que formando $\frac{W(t_j)+W(t_{j+1})}{2}$ y añadiendo un incremento independiente $N(0,\Delta t/4)$ da un valor para $W((t_j + t_{j+1})/2)$ que mantiene las tres condiciones de la definición de movimiento browniano dadas en la sección **2.1**.

Demostremos a continuación esta última afirmación; para esto sea Y el incremento mencionado que sigue una distribución normal $N(0,\Delta t/4)$ donde $\Delta t = T/N$ entonces podemos escribir:

$$W\left(\frac{t_j + t_{j+1}}{2}\right) = \frac{1}{2}(W(t_j) + W(t_{j+1})) + Y$$

ahora sea $\tilde{t}_j = \frac{t_j+t_{j+1}}{2}$ entonces el incremento Y va como sigue:

$$Y = W(\tilde{t}_j) - \frac{1}{2}(W(t_j) + W(t_{j+1}))$$

$$Y = W(\tilde{t}_j) - \frac{1}{2}W(t_j) - \frac{1}{2}W(t_{j+1})$$

$$Y = \frac{1}{2}W(\tilde{t}_j) + \frac{1}{2}W(\tilde{t}_j) - \frac{1}{2}W(t_j) - \frac{1}{2}W(t_{j+1})$$

$$Y = \frac{1}{2}(W(\tilde{t}_j) - W(t_j)) - \frac{1}{2}(W(t_{j+1}) - W(\tilde{t}_j))$$

de aquí se sigue, por propiedades de la distribución normal, que $\frac{1}{2}(W(\tilde{t}_j) - W(t_j))$ sigue una distribución $N(0,\Delta t/8)$ al igual que $\frac{1}{2}(W(t_{j+1}) - W(\tilde{t}_j))$ por lo tanto se concluye que,

$$Y = \frac{1}{2}(W(\tilde{t}_j) - W(t_j)) - \frac{1}{2}(W(t_{j+1}) - W(\tilde{t}_j))$$

es un incremento $N(0,\Delta t/4)$ que es lo que queríamos probar.

Hay que notar que los dos resultados -punto izquierdo y punto medio- para las integrales estocásticas, son diferentes, y se muestra experimentalmente que esta diferencia se reduce cuando $\delta t \rightarrow 0$. Esto subraya la diferencia entre integración determinística y estocástica; a la primera no le importa que *punto muestra* se tome en el intervalo $[t_j, t_{j+1}]$ mientras que para la segunda hay que precisar como la suma fue construída. La suma (2.14) que utiliza el punto **izquierdo** del intervalo $[t_j, t_{j+1}]$ da como resultado la integral denominada de **Itô** y la suma (2.16) que usa el **punto medio** $(t_i + t_{i+1})/2$ produce la integral denominada de **Stratonovich**.

Es posible evaluar exactamente las sumas (2.14) y (2.16) para obtener, a través del paso al límite, las respectivas integrales estocásticas de **Itô** como de **Stratonovich**.

La versión **Itô** es el caso límite de

$$\sum_{j=0}^{N-1} W(t_j)(W(t_{j+1}) - W(t_j)) = \frac{1}{2} \sum_{j=0}^{N-1} (W(t_{j+1})^2 - W(t_j)^2 - (W(t_{j+1}) - W(t_j))^2)$$

$$= \frac{1}{2} \left(W(T)^2 - W(0)^2 - \sum_{j=0}^{N-1} (W(t_{j+1}) - W(t_j))^2 \right)$$

con $W(0)=0$

De aqui, el valor esperado de

$$\sum_{j=0}^{N-1} (W(t_{j+1}) - W(t_j))^2$$

es igual a T, puesto que:

$$\begin{aligned} E \left[\sum_{j=0}^{N-1} (W(t_{j+1}) - W(t_j))^2 \right] &= \sum_{j=0}^{N-1} E[(W(t_{j+1}) - W(t_j))^2] = \\ &= \sum_{j=0}^{N-1} (\delta t) = T \end{aligned}$$

igualmente se prueba que su varianza es $O(\delta t)$ y de aqui haciendo que $\delta t \rightarrow 0$ se obtiene que:

$$\int_0^T W(t) dW(t) = \frac{1}{2} W(T)^2 - \frac{1}{2} T$$

La versión de **Stratonovich** es el caso límite de:

$$\sum_{j=0}^{N-1} \left(\frac{W(t_j) + (W(t_{j+1}))}{2} + \Delta Z_j \right) (W(t_{j+1}) - W(t_j))$$

donde cada ΔZ_j es un incremento independiente $N(0, \Delta t/4)$. Esta suma se convierte en:

$$\begin{aligned} &\sum_{j=0}^{N-1} \left(\frac{W(t_j) + (W(t_{j+1}))}{2} \right) (W(t_{j+1}) - W(t_j)) + \sum_{j=0}^{N-1} \Delta Z_j (W(t_{j+1}) - W(t_j)) \\ &= \sum_{j=0}^{N-1} \left(\frac{W^2(t_j) - W^2(t_{j+1}))}{2} \right) + \sum_{j=0}^{N-1} \Delta Z_j (W(t_{j+1}) - W(t_j)) \\ &= \frac{1}{2} (W(T)^2 - W(0)^2) + \sum_{j=0}^{N-1} \Delta Z_j (W(t_{j+1}) - W(t_j)) \end{aligned}$$

De aqui, el valor esperado de

$$\sum_{j=0}^{N-1} \Delta Z_j (W(t_{j+1}) - W(t_j)) = 0$$

por definición de movimiento Browniano y se prueba que su varianza es $O(\delta t)$, luego haciendo que $\delta t \rightarrow 0$ obtenemos:

$$\int_0^T W(t) dW(t) = \frac{1}{2} W(T)^2$$

2.5. Simulación numérica de una integral estocástica

Como vimos en la sección anterior, hay dos enfoques, entre otros, para abordar las integrales estocásticas: Itô y Stratonovich. Para calcular numéricamente dichas integrales usemos el siguiente algoritmo, en el cual la función $h(t)=W(t)$ donde $W(t)$ es un movimiento browniano cualquiera.

2.5.1. Algoritmo 3

1. Elegimos un intervalo $[0,T]$, $N =$ número de pasos y definimos $dt=T/N$.
2. Con el propósito de que el experimento se pueda repetir se fija, con la instrucción `set.seed()`, la semilla inicial del generador de números aleatorios.
3. Construimos un camino browniano cualquiera usando el **Algoritmo 1**
4. Luego construimos la suma $\sum_{j=0}^{N-1} W(t_j)(W(t_{j+1}) - W(t_j))$ y la asignamos a la variable *ito*.
5. A continuación construimos la suma

$$\sum_{j=0}^{N-1} \left(\frac{W(t_j) + W(t_{j+1})}{2} + \Delta Z_j \right) (W(t_{j+1}) - W(t_j))$$

- usando el mismo camino browniano - y la asignamos a la variable *strat*.

6. Usando las soluciones analíticas tanto de la integral de Itô como de Stratonovich, calculamos el error en el que se incurre en las variables *itoerr* y *straterr*
7. Finalmente calculamos para varios N , las integrales de Itô y Stratonovich, junto con sus errores respectivos, que no son sino los valores absolutos de la diferencia entre el valor analítico y el valor numérico.

El programa que va a continuación realiza lo antes descrito

2.5.2. Programa `stint.r`

```
stint=function(T=1,N=500)
{
  set.seed(100)
```

```

dt=T/N
dw=sqrt(dt)*rnorm(N)
w=cumsum(dw)
s1=c(0,w[1:(N-1)])*dw
ito=sum(s1)
s0=c(0,w[1:(N-1)])
s2=0.5*(s0+w)
s3=0.5*sqrt(dt)*rnorm(N)
s4=(s2+s3)*dw
strat=sum(s4)
itoe=0.5*(w[N]^2-T)
strate=0.5*(w[N])^2
itoerr=abs(ito-0.5*(w[N]^2-T))
straterr=abs(strat-0.5*(w[N])^2)
a=matrix(ncol=1,nrow=6)
a[1,1]=itoe
a[2,1]=ito
a[3,1]=itoerr
a[4,1]=strate
a[5,1]=strat
a[6,1]=straterr
return(a)
}

```

Por su naturaleza aleatoria tanto la integral Itô como la de Stratonovich tendrán diferentes valores cada vez que estas sean calculadas y a pesar de que no convergen ni dentro de cada integral ni la una hacia a la otra, como sucede con Reimann, al aumentar N el error de la aproximación - diferencia en valor absoluto del valor exacto con el valor aproximado- dentro de cada tipo de integral, tiende a cero cuando N aumenta como lo muestra la figura (2.4).

En lo que sigue se usará siempre la aproximación de Itô en lugar de la Stratonovich puesto que una transformación convierte la una en la otra.



Figura 2.4: Convergencia de los errores en Ito y Stratanovich

2.6. Solución numérica de una EDE

Como ya podemos evaluar numéricamente una integral estocástica, lo que nos falta ahora es un método numérico para hallar algunas soluciones aproximadas de una EDE y para esto recordemos que cuando tenemos una ecuación diferencial lineal determinística de la forma $y' = f(x, y)$, el método numérico más simple para hallar una solución aproximada es el Método de Euler, ver [Anexo 1]. Usando el mismo enfoque, ahora para una EDE, buscaremos una solución aproximada con el método denominado de Euler-Maruyama que no es sino una extensión del método de Euler pero para Ecuaciones Diferenciales Estocásticas.

2.7. Método de Euler-Maruyama EM

Una ecuación diferencial estocástica escalar puede ser escrita en su forma integral como:

$$X(t) = X(0) + \int_0^t f(X(s))ds + \int_0^t g(X(s))dW(s), 0 \leq t \leq T \quad (2.17)$$

donde f y g son funciones escalares y la condición inicial $X(0)$ es una variable aleatoria. La segunda integral en la parte derecha de (2.17) se toma con respecto al movimiento Browniano y se refiere a la integral de Itô.

La solución $X(t)$ de esta EDE es una variable aleatoria para cada t .

Es usual reescribir (2.17) en forma de ecuación diferencial :

$$dX(t) = f(X(t))dt + g(X(t))dW(t), X(0) = X_0, 0 \leq t \leq T \quad (2.18)$$

Si $g \equiv 0$ y $X(0)$ es una constante, entonces el problema se vuelve determinístico, y (2.18) se reduce a la ecuación diferencial $\frac{dX(t)}{dt} = f(X(t))$, con $X(0)=X_0$ y sabemos que para resolver esta ED podemos usar el conocido método de Euler.

Para aplicar el método de Euler-Maruyama a (2.18) necesitamos primero discretizar el intervalo $[0, T]$, para esto sea $\Delta t = T/L$ para algún entero L y sea $\tau_j = j\Delta t$ con $j=0, 1, \dots, L$; nuestra aproximación numérica se notará por X_j y el método toma la forma:

$$X_j = X_{j-1} + f(X_{j-1})\Delta t + g(X_{j-1})(W(\tau_j) - W(\tau_{j-1})), j = 1, 2, \dots, L \quad (2.19)$$

Para mostrar como funciona este método numérico usaremos el modelo con el cual hemos venido trabajando, es decir el de *mercado de acciones*. Este modelo asume que el cambio relativo (tasa de retorno dS/dt) de una *acción* en el intervalo de tiempo dt está compuesto por un término determinístico *drift* $\mu S(t)$ más una fluctuación estocástica de la forma $\sigma dW(t)$:

$$dS(t) = \mu S(t)dt + \sigma S(t)dW(t) \quad (2.20)$$

usando la notación usual tenemos que $:X(t) = S(t)$, $f(X(t)) = \mu S(t)$, $g(X(t)) = \sigma S(t)$ donde μ tasa esperada de retorno y σ la volatilidad, ver [9], donde μ y σ son constantes reales.

Vimos, en la sección **2.3**, que la solución analítica de esta ecuación es:

$$X(t) = X(0)\exp\left(\left(\mu - \frac{1}{2}\sigma^2\right)t + \sigma W(t)\right) \quad (2.21)$$

y la usaremos para compararla con la solución aproximada, generada por el método de Euler -Maruyama, en el punto $t=T$ a través del siguiente algoritmo.

2.7.1. Algoritmo 4

1. Elegimos un intervalo $[0, T]$, $L =$ número de pasos, μ la tasa de retorno, σ la volatilidad y definimos $\Delta t = T/L$ y $\tau_j = j\Delta t$ con $j=0, 1, \dots, L$. Por conveniencia se elige el paso Δt como un múltiplo entero $R \geq 1$ del incremento δt del camino Browniano. Esto asegura que el conjunto de puntos $\{t_j\}$ sobre el cual se basa el camino Browniano contengan los puntos $\{\tau_j\}$ para los cuales la solución **EM** es calculada.

2. Con el propósito de que el experimento se pueda repetir fijamos, con la instrucción `set.seed()`, la semilla inicial del generador de números aleatorios.
3. Construimos un camino browniano discreto cualquiera sobre $[0, T]$, usando el **Algoritmo 1**
4. Construimos una secuencia $\{t\}$ que empiece en 0 , termine en T con pasos de tamaño dt , donde dt es el tamaño del paso del camino browniano.
5. También construimos otra secuencia $\{t_1\}$, que empiece en 0 y termine en T con pasos de tamaño dt .
6. Evaluamos la solución exacta dada por

$$X(t) = X(0) \exp \left(\left(\mu - \frac{1}{2} \sigma^2 \right) t + \sigma W(t) \right)$$

para los valores de la secuencia t y la asignamos en la variable `Xtrue`.

7. A continuación definimos $\Delta t = Rdt$ y a $L=N/R$.
8. Como para cualquier paso del método EM se requiere el incremento $W(\tau_j) - W(\tau_{j-1})$ lo calculamos usando:

$$W(\tau_j) - W(\tau_{j-1}) = W(jR\delta t) - W((j-1)R\delta t) = \sum_{k=jR-R+1}^{jR} dW_k$$

y lo asignamos a la variable `Winc`

9. Hallamos la solución aproximada EM usando

$$X_j = X_{j-1} + f(X_{j-1})\Delta t + g(X_{j-1})(W(\tau_j) - W(\tau_{j-1}))$$

y la asignamos al vector de $1 \times L$ `Xem`.

10. La diferencia, en valor absoluto, entre la aproximación y la solución exacta, en el punto $t=T$, se guarda en `emerr`
11. Finalmente graficamos las soluciones, tanto la exacta como la aproximada.

En el programa `em.r` consideramos $\mu = 2$, $\sigma = 1$, $X_0 = 1$. El camino Browniano discreto se calcula sobre $[0, 1]$ con $\delta t = 2^{-8}$ y aplicamos EM para un paso $\Delta t = R\delta t$ con $R=4$.

2.7.2. Programa em.r

```

em=function(R=4) {
  set.seed(200)
  mu=2
  sigma=1
  Xzero=1
  T=1
  N=2^8
  dt=1/N
  dw=sqrt(dt)*rnorm(N)
  w=cumsum(dw)
  t=seq(dt,T,by=dt)
  t1=seq(0,T,by=dt)
  Xtrue=Xzero*exp((mu-0.5*sigma^2)*t+sigma*w)
  R=4
  Dt=R*dt
  L=N/R
  Xem=0
  Xtemp=Xzero
  for (j in 1:L) {
    Winc=sum(dw[(R*(j-1)+1):(R*j)])
    Xtemp=Xtemp + Dt*mu*Xtemp + sigma*Xtemp*Winc
    Xem[j]=Xtemp
  } t2=seq(0,T,by=Dt)
  emerr=abs(Xem[length(Xem)]-Xtrue[length(Xtrue)])
  return(emerr,plot(t1,c(Xzero,Xtrue),type="l",col="blue",xlab="t",ylab="X"),
+lines(t2,c(Xzero,Xem),col="red",lty=2)) }

```

	R=4	R=2	R=1
emerr	0.4246	0.2459	0.1659

Cuadro 2.2: Análisis del error

Si tomamos en cuenta que $\Delta t = Rdt$, donde Δt es el paso del método EM

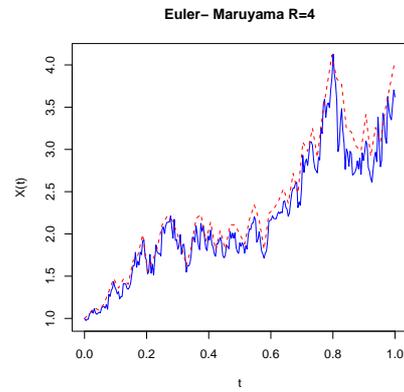


Figura 2.5: Aproximación numérica de una EDE

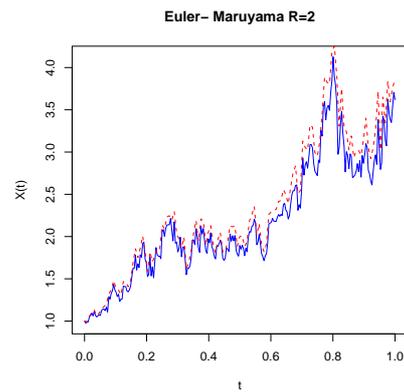


Figura 2.6: Aproximación numérica de una EDE

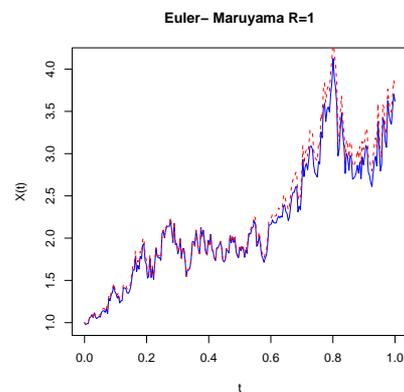


Figura 2.7: Aproximación numérica de una EDE

y dt el paso del movimiento browniano, podemos tomar pasos mas pequeños haciendo que $R=2$ o $R=1$, de aqui los gráficos (2.4) y (2.5) y (2.6) muestran que si usamos, en el algoritmo anterior, $R=2$ o $R=1$ en comparación a $R=4$, la solución aproximada se acerca más al la solución exacta.

Igualmente el cuadro (2.2) muestra que el error entre la aproximación numérica y la solución exacta $emerr$ disminuye cuando Δt lo hace.

Capítulo 3

Movimiento browniano fraccionario

Hasta aquí hemos trabajado con el movimiento browniano clásico para resolver cierto tipo de EDE que aparecen en Finanzas, sin embargo el movimiento browniano carece de ciertas propiedades como *memoria a largo plazo*, que se definirá en (3.3) y que es una característica de los mercados de acciones. Además el consenso general es asumir que toda la información sobre estos mercados está contenida dentro de los mismos, por lo que es razonable asumir que son procesos markovianos ver [4], característica de la cual carece también el movimiento browniano. Estas son las razones por lo que abordaremos el mismo problema -*precios de acciones*- pero ahora con el *movimiento browniano fraccionario* el cual si posee las características anotadas.

3.1. Definición

Un movimiento browniano fraccionario normalizado

$$B_H = B_H(t) : 0 \leq t \leq \infty$$

con $0 < H < 1$ (llamado parámetro de Hurst) está únicamente caracterizado por las siguientes propiedades, ver [3]:

1. $B_H(t)$ tiene incrementos estacionarios, es decir que la distribución del número de eventos que ocurren en cualquier intervalo de tiempo depende solamente de la longitud del intervalo de tiempo.
2. $B_H(0) = 0$ y $E[B_H(t)] = 0$ para $t \geq 0$

3. $E[B_H^2(t)] = t^{2H}$ para $t \geq 0$

4. $B_H(t)$ tiene una distribución normal para $t > 0$

Vamos a suponer siempre que estamos trabajando en el caso que $B_H(t)$ tiene trayectorias continuas (esto lo justifica el criterio de Kolmogorov para caminos continuos, ver [4]).

De las primeras tres propiedades se sigue que la función de covarianza está dada por :

$$\rho(s, t) = E[B_H(s)B_H(t)] = \frac{1}{2}t^{2H} + s^{2H} - (t - s)^{2H} \quad (3.1)$$

para $0 < s < t$. Como el movimiento browniano fraccionario no es sino un proceso gaussiano con incrementos estacionarios, veamos antes de seguir, que es un ruido fraccionario gaussiano.

3.2. Ruido fraccionario Gaussiano

Definamos el proceso de incrementos $X = \{X(k) : k = 0, 1, \dots\}$ de un movimiento browniano fraccionario, al que llamaremos *ruido fraccionario gaussiano*, por:

$$X(k) = B_H(k + 1) - B_H(k) \quad (3.2)$$

Es claro que $X(k)$ tiene una distribución normal estándar para cada k , pero en general no existe independencia. Para ser precisos, la correspondiente función de autocovarianza es de la forma :

$$\gamma(k) = \frac{1}{2}[|k - 1|^{2H} - 2|k|^{2H} + |k + 1|^{2H}] \quad (3.3)$$

para k en los enteros. Si $H = \frac{1}{2}$ todas las covarianzas son cero y estamos ante el movimiento browniano estándar.

Para seguir necesitamos dos conceptos importantes que son : *autosimilaridad y memoria a largo plazo*.

3.3. Autosimilaridad y memoria a largo plazo

En lugar de definirlos en el contexto del movimiento browniano fraccionario, definamos estas propiedades en el marco de los procesos estocásticos estacionarios generales.

Sea $X = \{X(t) : t = 0, 1, 2, \dots\}$, un proceso estocástico *estacionario* discreto, esto significa que los vectores $(X(t_1), \dots, X(t_d))$ y $(X(t_1 + n), \dots, X(t_d + n))$ tienen la

misma distribución para todos los enteros $d, n \geq 1$ y $t_1, \dots, t_d \geq 0$. Para un proceso gaussiano esto es equivalente a exigir que $\gamma(k) = \text{Cov}(X(n), X(n+k))$ no dependa de n .

Decimos ahora que un proceso estocástico a tiempo discreto es *autosimilar* si los procesos $\{X(ct)\}$ y $\{kX(t)\}$, donde c y k son constantes reales, tiene la misma distribución. Análogamente decimos que un proceso estocástico a tiempo discreto es *autosimilar* con parámetro de Hurst $0 < H < 1$ si $k = c^H$. Este concepto puede ser extendido a procesos estocásticos continuos en el tiempo.

Un proceso estocástico a tiempo continuo $Y = \{Y(t) : 0 \leq t < \infty\}$ es llamado *autosimilar* con parámetro de Hurst $0 < H < 1$ si $\{Y(at) : 0 \leq t < \infty\}$ y $\{a^H Y(t) : 0 \leq t < \infty\}$ tiene distribuciones finito-dimensionales idénticas para todo $a > 0$.

Introduzcamos ahora el concepto de *memoria a largo plazo*: un proceso estacionario a tiempo discreto se dice con *memoria a largo plazo* cuando su función de autocovarianza $\gamma(\cdot)$ decae tan lentamente que $\sum_{k=0}^{\infty} \gamma(k) = \infty$.

Intuitivamente cuando la *memoria a largo plazo* está presente "high-lag" correlaciones pueden ser individualmente pequeñas pero su efecto acumulativo es significativo.

3.4. Simulación del movimiento browniano fraccionario

Existen muchos métodos para simular este movimiento entre estos están: los métodos exactos como Hosking, Cholesky y el de Davies y Hart. También existen métodos aproximados como: el método de representación estocástica, el método de Random Midpoint Displacement, el de Paxon, el que usa Wavelets, etc.

Sin embargo un método fácil de implementar es el método de Weirstrass.

3.4.1. Método de Weirstrass

Este método simula un *movimiento browniano fraccionario*, para esto definimos los números $b > 1$, $0 < H < 1$ (parámetro de Hurst) y n para luego construir la función $fBm(t)$ dada por, ver [6]:

$$fBm(t) = c_1 b^{-H} \sin(b^1 t + d_1) + c_2 b^{-2H} \sin(b^2 t + d_2) + c_3 b^{-3H} \sin(b^3 t + d_3) + \dots$$

donde los $c_i, i = 1, \dots, n$ siguen una distribución normal $N(0,1)$ y los $d_i, i = 1, \dots, n$ una distribución uniforme en el intervalo $[0, 2\pi]$.

A continuación va el código del programa usado para la simulación y tres gráficos

correspondientes a $H=0.25$, $H=0.5$ y $H=0.75$, donde H es el parámetro de Hurst.

3.4.2. Programa fbmweirstrass.r

```
fbmweirstrass=function(n=8,H=0.75,b=2,N=500){
  set.seed(100)
  t=seq(0,1,length=N)
  s=0 for(i in 1:n){
    s=s+rnorm(1)*b^(-i*H)*sin(b^(i)*t+runiform(1,min=0,max=2*pi)) }
  plot(t,s,type="l",main="fBm")
  return(s)
}
```

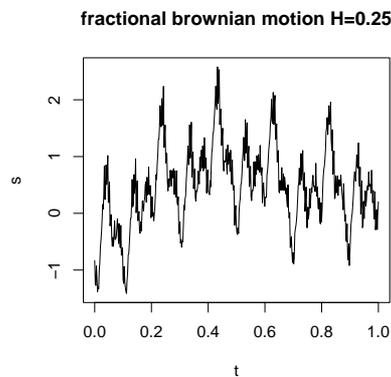


Figura 3.1: Movimiento browniano fraccionario $H=0.25$

Un método más complicado, pero en el cual está justificada la construcción teórica es el método de Hosking descrito a continuación.

3.4.3. Método de Hosking

El método de Hosking es un algoritmo para simular un proceso estacionario gaussiano general; por lo tanto nos enfocaremos en la simulación de un *ruido fraccionario gaussiano* X_0, X_1, \dots para luego usar el hecho que el *movimiento browniano fraccionario*

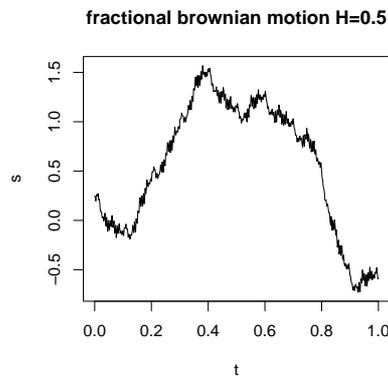


Figura 3.2: Movimiento browniano fraccionario $H=0.5$

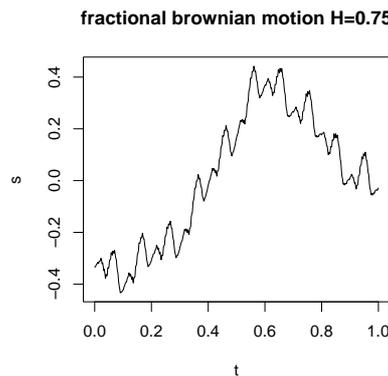


Figura 3.3: Movimiento browniano fraccionario $H=0.75$

Y_0, Y_1, \dots se obtiene del *ruido fraccionario gaussiano* tomando sumas acumulativas, es decir:

$$B_H(k) = X_0 + X_1 + \dots + X_{k-1}$$

El método genera X_{n+1} dados X_n, \dots, X_0 y su fortaleza radica en que la distribución de X_{n+1} , dado su pasado, se puede calcular explícitamente, ver [3].

Para esto sea $\gamma(\cdot)$ la función de covarianza del proceso de media cero, esto es

$$\gamma(k) = E(X_n X_{n+k})$$

para $n, k=0, 1, 2, \dots$, asumamos por conveniencia que $\gamma(0) = 1$.

Además sea $\Gamma(n) = \gamma(i - j)$ con $i, j=0, 1, \dots, n$ la matriz de covarianza y $c(n)$ el vector columna cuyos elementos son: $c(n)_k = \gamma(k + 1)$ con $k = 0, 1, \dots, n$.

Definamos la matriz $F(n)$ de $(n+1) \times (n+1)$ como $F(n) = \mathbf{1}_{(i=n-j)}$ con $i, j=0, 1, \dots, n$, donde $\mathbf{1}$ denota la función indicatriz. La matriz $\Gamma(n+1)$ puede ser separada de la siguiente forma:

$$\begin{aligned} \Gamma(n+1) &= \begin{pmatrix} 1 & c(n)' \\ c(n) & \Gamma(n) \end{pmatrix} \\ &= \begin{pmatrix} \Gamma(n) & F(n)c(n) \\ c(n)'F(n) & 1 \end{pmatrix} \end{aligned}$$

El próximo paso es calcular la distribución condicional de X_{n+1} dados X_0, X_1, \dots . En efecto vamos a mostrar que esta distribución es normal con esperanza μ_n y varianza σ_n^2 dadas por:

$$\mu_n = c(n)' \Gamma(n)^{-1} \begin{pmatrix} X_n \\ \cdot \\ \cdot \\ X_1 \\ X_0 \end{pmatrix}, \sigma_n^2 = 1 - c(n)' \Gamma(n)^{-1} c(n)$$

Por conveniencia definamos $d(n) = \Gamma(n)^{-1} c(n)$ y de aquí decir que :

$$\mu_n = d(n)' \begin{pmatrix} X_n \\ \cdot \\ \cdot \\ X_1 \\ X_0 \end{pmatrix}, \sigma_n^2 = 1 - d(n)c(n)$$

Se puede probar que :

$$\Gamma(n+1)^{-1} = \frac{1}{\sigma_n^2} \begin{pmatrix} 1 & -d(n)' \\ -d(n) & \sigma_n^2 \Gamma(n)^{-1} + d(n)d(n)' \end{pmatrix} \quad (3.4)$$

es lo mismo que tener :

$$\Gamma(n+1)^{-1} = \frac{1}{\sigma_n^2} \begin{pmatrix} \sigma_n^2 \Gamma(n)^{-1} + F(n)d(n)d(n)'F(n) & -F(n)d(n) \\ -d(n)'F(n) & 1 \end{pmatrix} \quad (3.5)$$

De aquí se puede deducir que para cada $x \in R^{n+1}$ y para cada $y \in R$ se tiene que :

$$\begin{pmatrix} y & x' \end{pmatrix} \Gamma(n+1)^{-1} \begin{pmatrix} y \\ x \end{pmatrix} = \frac{(y - d(n)'x)^2}{\sigma_n^2} + x' \Gamma(n)^{-1} x$$

Esto implica que la distribución de X_{n+1} dados X_0, X_1, \dots es, de hecho, normal con esperanza μ_n y varianza σ_n^2 antes definidas.

Ahora que conocemos la distribución de X_{n+1} la muestra requerida se encuentra generando una variable aleatoria X_0 normal estandar y simulando X_{n+1} recursivamente para $n=0,1,2,\dots$. El problema a solucionar es evitar el cálculo de la matriz inversa en cada paso y para esto procedemos como sigue, ver [3].

Supongamos conocidos μ_n, σ_n^2 y $\tau_n = d(n)'F(n)c(n) = c(n)'F(n)d(n)$. Usando (3.5) es fácil ver que σ_n^2 satisface la recursión :

$$\sigma_{n+1}^2 = \sigma_n^2 - \frac{(\gamma(n+2) - \tau_n)^2}{\sigma_n^2}$$

La recursión para $d(n+1) = \Gamma(n+1)^{-1}c(n+1)$ también se obtiene de (3.5):

$$\left(d(n) - \phi_n F(n)d(n) \quad \phi_n \right)'$$

donde

$$\phi_n = \frac{\gamma(n+2) - \tau_n}{\sigma_n^2}$$

Los primeros n elementos de la fila $(n+2)$ del vector $d(n+1)$ pueden así ser calculados de $d(n)$ y ϕ_n .

Dado $d(n+1)$ se calcula $\mu_{n+1}, \sigma_{n+1}^2$ y τ_{n+1} . Comenzamos la recursión con $\mu_0 = \gamma(1)X_0, \sigma_0^2 = 1 - \gamma(1)^2$ y $\tau_0 = \gamma(1)^2$.

Finalmente calculamos el *movimiento browniano fraccionario* simplemente haciendo las sumas acumulativas.

3.4.4. Programa `hosking.r`

A continuación va el programa en R `hosking.r` que simula un *movimiento browniano fraccionario* para un parámetro de Hurst igual a 0.75 , $n=1024$ y el gráfico respectivo.

```
covar=function(i,H=3/4) {
  if (i == 0)
    return(1)
```

```

else
  return(((i-1)^(2*H)-2*i^(2*H)+(i+1)^(2*H))/2);
}

```

```

hosking=function(n=6,H=3/4,L=1) {
m=2^n
output=numeric()
cov=numeric()
phi=numeric()
psi=numeric()
output[1]=rnorm(1)
v=1
phi[1]=0

for (i in 1:m)
cov[i]=covar(i,H)

for (i in 2:m) {
phi[i-1]=cov[i]
for (j in 1:(i-1)) {
psi[j]=phi[j]

phi[i-1]=phi[i-1]-psi[j]*cov[i-j]
}
phi[i-1]=phi[i-1]/v
for (j in 1:(i-1))
phi[j]=psi[j]-phi[i-1]*psi[i-j]
v=v*(1-phi[i-1]*phi[i-1])
output[i]=0
for (j in 1:(i-1))
output[i]=output[i]+phi[j]*output[i-j]
output[i]=output[i]+sqrt(v)*rnorm(1)
}
br=cumsum(output)
}

```

```
return(plot(br,type="l",main="fractional brownian
motion",xlab="t")) }
```

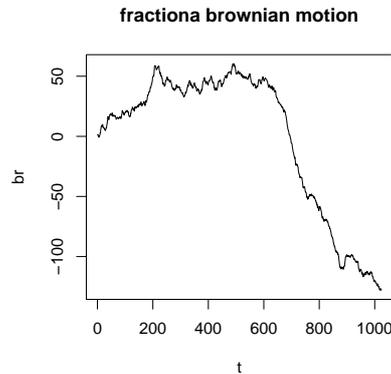


Figura 3.4: Simulación de **fBm** usando Hoskings

3.5. Aplicación del fBm a las finanzas

Algunos de los fenómenos que aparecen en los mercados financieros, necesitan para ser resueltos el enfoque de las Ecuaciones Diferenciales Estocásticas que usan el movimiento fraccionario browniano, a continuación describamos uno de ellos.

Mercado de Bonos y Acciones. Este tipo de mercado, en el cual participan dos tipos de *papeles: bonos y acciones*, está gobernado por la siguiente estructura, ver [11]:

1. Cuenta bancaria o *bono* para el cual su precio es $\mathbf{B}(t)$ y que está descrito por la ecuación diferencial ordinaria :

$$dB(t) = rB(t), B(0) = 1, 0 \leq t \leq T, r > 0 \quad (3.6)$$

donde r es la tasa de interés del bono. La solución de esta última ecuación diferencial es: $B(t) = \exp(rt)$

2. *Acción* para el cual su precio está dado por $\mathbf{S}(t)$ y que satisface la ecuación :

$$dS(t) = \mu S(t)dt + \sigma S(t)dB_H(t) \quad (3.7)$$

con condición de valor inicial $S(0) = S_0$ con μ, σ constantes reales , $0 \leq t \leq T$ y donde $B^H(t)$ es un movimiento browniano fraccionario con $H \in (\frac{1}{2}, 1)$

Como se verá más adelante, la ecuación (3.7) tiene como solución analítica a :

$$S(t) = S_0 \exp(\mu t + \sigma B_H(t)) \quad (3.8)$$

Y es justamente usando esta última solución que compararemos el error en que se incurre al buscar una solución numérica para el Mercado de Bonos y Acciones.

3.6. Funciones sobre caminos definidos por fBm

Acabamos de ver que en el contexto del *movimiento fraccionario browniano*, el precio $S(t)$ en el modelo de mercados de bonos y stocks, evoluciona de acuerdo a la EDE:

$$dS(t) = \mu S(t)dt + \sigma S(t)dB_H(t), S(0) = S_0 \quad (3.9)$$

donde μ es la tasa promedio de retorno, σ la volatilidad y $B^H(t)$ es un movimiento browniano fraccionario.

La solución analítica de esta EDE es:

$$S(t) = S(0) \exp(\mu t + \sigma B_H(t)) \quad (3.10)$$

Esto significa que esta solución es una función de los caminos fraccionarios brownianos. El siguiente paso es simular estas funciones para valores específicos de μ y σ .

3.6.1. Simulación numérica de una función sobre caminos definidos por fBm

Si para efectos de la simulación, elegimos en (3.10), $\mu = \frac{9}{8}$, $\sigma = \frac{1}{2}$ y $S(0)=1$ obtenemos la función $u(B_H(t)) = \exp(\frac{9}{8}t + \frac{1}{2}B_H(t))$, pero lo que nos interesa es calcular numéricamente la solución del modelo, con el menor error de estimación posible; este error se calcula sacando la raíz cuadrada a la suma de las diferencias entre el valor esperado y el valor promedio muestral de $u(B_H(t))$, en cada t , elevadas al cuadrado.

Para esto usemos el siguiente algoritmo, que nos permite evaluar la función

$$u(B_H(t)) = \exp\left(\frac{9t}{8} + \frac{1}{2}B_H(t)\right)$$

a lo largo de 1000 caminos fraccionarios brownianos discretizados.

Se puede probar, ver [11], que el valor esperado de la solución (3.10) es

$$E[S(t)] = S_0 \exp\left(\mu t + \frac{\sigma^2 t^{2H}}{2}\right)$$

Por lo tanto el valor esperado exacto para $\mu = \frac{9}{8}$ y $\sigma = \frac{1}{2}$ sería:

$$E[S(t)] = \exp\left(\frac{9t}{8} + \frac{1}{8}t^{\frac{3}{2}}\right)$$

3.6.2. Algoritmo 5

1. Definimos T =extremo superior del intervalo $[0,T]$, N =número de pasos, dt =tamaño del paso y M =número de caminos brownianos fraccionarios.
2. Construimos una secuencia $\{t\}$ que empiece en dt y termine en T , con pasos de dt .
3. Construimos la matriz B de $M \times N$ tal que $B(i,j)$ contiene el camino browniano fraccionario i -ésimo.
4. Ahora construimos la matriz U de dimensión $M \times N$ de tal forma que la i -ésima fila contenga la función $u(B_H(t))$ evaluada en el i -ésimo camino.
5. A continuación formamos el vector $U_{mean} = mean(U)$ de $1 \times N$ que calcula los promedios por columna y cuya j -ésima entrada contiene el promedio de los $u(B_H(t_j))$.
6. Finalmente graficamos el promedio de las $u(B_H(t))$ sobre esos caminos y, la función $u(B_H(t_j))$ para cinco caminos cualesquiera (esto para mostrar que individualmente los caminos no son *suaves* pero en promedio si los son) y la función

$$f(t) = \exp\left(\frac{9t}{8} + \frac{1}{8}t^{\frac{3}{2}}\right)$$

que es el valor esperado de la solución de (3.10).

7. También se calcula en **averr**, el error tal como se lo definió anteriormente, es decir la raíz cuadrada de la suma de las diferencias entre el valor esperado

$$E[S(t)] = \exp\left(\frac{9t}{8} + \frac{1}{8}t^{\frac{3}{2}}\right)$$

y la media muestral $u(B_H(t))$, para cada t , elevadas al cuadrado.

A continuación va el programa respectivo y su gráfico correspondiente.

3.6.3. Programa fBmf.R

```
fBmf=function(T=1,N=500,M=1000) {
dt=T/N
t=seq(dt,1,by=dt)
B=matrix(nrow=M,ncol=N)
U=matrix(nrow=M,ncol=N)
Umean=0
ve=0
for(i in 1:M){
B[i,]=fBmweirstrass()
U[i,]=exp(1.125*t+0.5*B[i,]) }
  for (i in 1:N){
    Umean[i]=mean(U[,i])
  }
ve=exp(9*t/8+(1/8)*t^(3/2))
dif=(Umean-ve)
t1=c(0,t)
averr=sqrt(t(dif)%*%(dif))
return(plot(t,Umean,type="l",col="blue",xlim=c(0,1),ylim=c(-1,5),
main="Funcion sobre fBm",ylab="S(t)",lines(t1,c(1,U[1,])),
lines(t1,c(1,U[2,])),lines(t1,c(1,U[3,])),lines(t1,c(1,U[4,])),
lines(t1,c(1,U[5,])),lines(t1,c(1,ve),col="red",lty=2))
}
```

3.7. Integración usando el fBm

Varios métodos se han desarrollado para integrar usando el **fBm** como integrador, sin embargo son dos los que tienen aplicaciones a finanzas. El primero es el **fractional pathwise integral** el cual tiene la siguiente forma:

$$\int_a^b f(t, w) \delta B_H(t) = \lim_{|\pi| \rightarrow 0} \sum_{k=0}^{n-1} f(t_k, w) (B_H(t_{k+1}) - B_H(t_k))$$

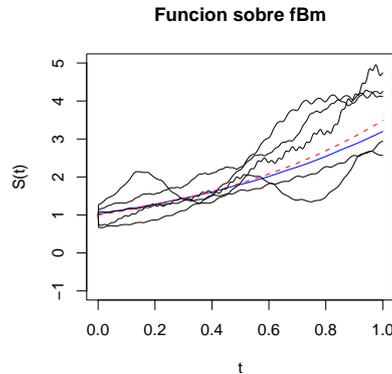


Figura 3.5: Función sobre caminos fBm

donde π es una partición del intervalo $[a,b]$, sin embargo este enfoque provoca **arbitraje**, ver [2]. *El arbitraje es la práctica de obtener ventaja de un estado en desequilibrio entre dos o más mercados, esta ventaja se la logra a través de operaciones de negocios que explotan este desequilibrio y la ganancia es la diferencia de precios entre estos mercados.*

El otro camino es la **fractional Wick-Ito integral** la cual tiene la forma:

$$\int_a^b f(t, w) \delta B_H(t) = \lim_{|\pi| \rightarrow 0} \sum_{k=0}^{n-1} f(t_k, w) \diamond (B_H(t_{k+1}) - B_H(t_k))$$

donde \diamond representa el producto de Wick y $1/2 < H < 1$. Este nuevo tipo de integración produce un mercado **sin arbitraje**, ver [2].

En nuestro trabajo, solamente abordaremos el primer método a pesar del arbitraje.

3.8. Simulación numérica de una integral estocástica usando fBm

El método que usaremos para hacerla simulación será el **pathwise integral**, vimos anteriormente que este enfoque dice que la integral con respecto al movimiento browniano fraccionario se define como:

$$\int_a^b f(t, w) \delta B_H(t) = \lim_{|\pi| \rightarrow 0} \sum_{k=0}^{n-1} f(t_k, w) (B_H(t_{k+1}) - B_H(t_k))$$

donde π es una partición del intervalo $[a,b]$ y la función $f(t_k, w)$ que utilizaremos será el *movimiento browniano* $W(t)$ definido al principio de este trabajo, además usaremos

$H=0.5$ para el cual el *movimiento browniano fraccionario* se vuelve el *movimiento browniano* .

El programa a continuación muestra dicha simulación.

3.8.1. Programa intfBm.R

```
intfBm=function(a=0,b=1,No=492){
t=seq(a,b,length=No)
Bt=weirstrass(N=No)
w=bpath2(N=No)
suma=0
for (i in 1:(No-1))
suma=suma+w[i]*(Bt[i+1]-Bt[i])
return(suma)
}
```

Al igual que con la integral estocástica que usa el *movimiento browniano* esta integral no converge, debido a su naturaleza aleatoria, cuando se afina la partición, como lo muestra el cuadro (3.1).

	N=500	N=1000	N=1500	N=2500
int	-0.04166	-0.08940	0.77614	-0.62087

Cuadro 3.1: Integral estocástica usando fBm

Capítulo 4

Proceso de Lévy

Hemos visto en el **Capítulo 2** que el modelo clásico de *precio de acciones* está basado en el camino browniano, esto es:

$$S(t) = S(0) \exp \left(\mu - \frac{\sigma^2}{2} \right) t + \sigma W(t) \quad (4.1)$$

Debido a que el camino browniano es continuo, también lo es el precio que aparece en el modelo. Sin embargo los precios de acciones solo se ven continuos cuando se consideran en un período largo de tiempo. Para períodos de tiempo cortos, dichos precios lucen discontinuos, es decir ellos tienen saltos; esto hace que tanto el *el movimiento browniano* como el *el movimiento browniano fraccionario* no sean buenos modelos para describir el comportamiento de estos precios.

Los procesos de Lévy son una clase de procesos que tienen saltos y que son muy populares a la hora de modelar *precios de acciones*, estos procesos se dividen en dos clases, ver [1]:

1. *jump diffusion models*, por ejemplo aquellos modelos que contienen un camino browniano:

$$X(t) = W(t) + C(t), t \geq 0$$

donde $W(t)$ es un camino browniano y $C(t)$ es un procesos de Poisson.

2. *infinite activity models* es decir modelos sin camino browniano, por ejemplo los procesos *Variance-Gamma*

En este trabajo consideraremos los modelos que involucran a los primeros.

4.1. Definición de un Proceso de Levy

Un proceso estocástico $X = (X(t), t \geq 0)$ se dice de Lévy si satisface las siguientes condiciones:

1. X tiene incrementos estacionarios e independientes
2. $X(0) = 0$ con probabilidad 1.
3. X es *estocásticamente continuo*, es decir para todo $a > 0$ y para todo $s \geq 0$ se cumple que:

$$\lim_{t \rightarrow s} P(|X(t) - X(s)| > a) = 0$$

4. Los saltos de $X(t)$ no deben pasar en tiempos determinísticos.

Para poder usar el proceso de Levy en las diferentes aplicaciones, necesitamos primero simularlo numéricamente y eso haremos a continuación.

4.2. Simulación numérica de un proceso de Lévy

El proceso de Lévy que simularemos a continuación se denomina *proceso simple de Lévy*, está caracterizado por la tripleta (a, ν, γ) y se define a través de:

$$X(t) = \gamma t + aW(t) + C(t) \tag{4.2}$$

donde $W(t)$ es un movimiento browniano cualquiera, $C(t)$ es un proceso de Poisson compuesto y ν es el número esperado de saltos por unidad de tiempo.

Antes de seguir definamos que es un proceso de Poisson y que es un proceso de Poisson compuesto.

4.2.1. Proceso de Poisson

Sea $\{\tau_i\}$ una secuencia de variables aleatorias exponenciales de parámetro λ , es decir cuya *densidad* de distribución es $f(x) = \lambda e^{-\lambda x}$, $x > 0$ y sea $T_n = \sum_{i=1}^n \tau_i$, ver [7].

El proceso $N(t)$, $t \geq 0$ definido por:

$$N(t) = \sum_{n \geq 1} 1_{t \geq T_n} \tag{4.3}$$

se denomina *proceso de Poisson* con intensidad λ .

4.2.2. Propiedades del Proceso de Poisson

1. $N(0) = 0$
2. Los caminos de muestra son continuos por la derecha con límites por la izquierda (cadlag).
3. Para cualquier t , $N(t)$ sigue una distribución de Poisson con parámetro λt , es decir:

$$P(N(t) = n) = e^{-\lambda t} \frac{(\lambda t)^n}{n!} \quad (4.4)$$

4. $N(t)$ tiene incrementos independientes y estacionarios.

4.2.3. Proceso de Poisson compuesto

Un proceso de Poisson compuesto con intensidad λ es un proceso estocástico $X(t)$ definido como:

$$X(t) = \sum_{i \geq 1}^{N(t)} Y_i \quad (4.5)$$

donde las Y_i son variables aleatorias *i.i.d* con distribución f y $N(t)$ es un proceso de Poisson con intensidad λ independiente de las Y_i . Cuando $Y_i = 1$ obtenemos el *proceso de Poisson estándar*, ver [1].

Por lo tanto antes de presentar el algoritmo para el proceso de Lévy, necesitamos los algoritmos tanto para el proceso de Poisson (dos métodos) como para el proceso compuesto de Poisson.

4.2.4. Algoritmo para la simulación de un proceso de Poisson (método 1)

1. Elegimos un intervalo $[0, T]$ y un cierto N para establecer una partición de tal forma que tengamos $N+1$ puntos: $0 = t_0 < t_1 < \dots < t_N = T$ y tal que $N_0 = 0$
2. Simulamos las T_k , variables aleatorias que siguen una distribución exponencial de parámetro λT , mientras $\sum_{i=1}^k T_i < T$
3. Finalmente ponemos $N(t_i) = \sup\{k : \sum_{j=1}^k T_j < t_i\}$

4.2.5. Programa pproc1

```

pproc1=function(T=1,N=100,M=1000,lambda=5){
  sq=seq(0,T,length=N)
  Nt=numeric(0)
  Tk=numeric(0)
  repeat{
    b=rexp(1,lambda)
    Tk=c(Tk,b)
    x=cumsum(Tk)
    if (max(x)>1)
      break
  }
  for(j in 1:length(sq)){
    Nt[j]=max(c(0,which(x<sq[j])))
  }

  return(Nt,plot(stepfun(sq,Nt),verticals=F,do.points=F,
  main="Proceso de Poisson",xlab="t",ylab="Nt"))
}

```

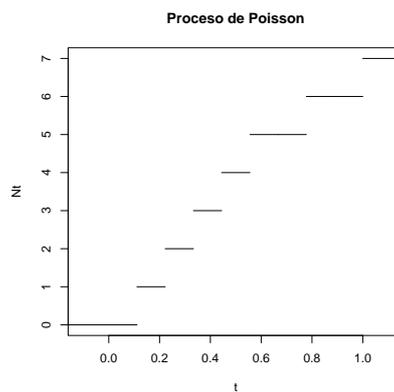


Figura 4.1: Poisson método 1

4.2.6. Algoritmo para la simulación de un proceso de Poisson (método 2)

1. Elegimos un intervalo $[0, T]$ y un cierto N para establecer una partición de tal forma que tengamos $N+1$ puntos: $0 = t_0 < t_1 < \dots < t_N = T$ y tal que $N(0) = 0$
2. Simulamos N (*saltos*) variables aleatorias que siguen una distribución de Poisson de parámetro λT
3. Simulamos N variables aleatorias uniformemente distribuidas, U_j , sobre el intervalo $[0, T]$. Estas variables corresponden a los tiempos de salto.
4. Finalmente, sea $N(t_i) = \sup\{k : \sum_{j=1}^k U_j < t_i\}$

4.2.7. Programa pprocc2

```
pprocc2=function(T=1,N=10,M=1000,lambda=5){
  set.seed(100)
  sq=seq(0,T,length=N)
  lambda1=lambda*T
  Nt=numeric(0)
  a=rpois(1,lambda1)
  b=runif(a,0,T)
  b1=sort(b) x=b1
  for(j in 1:length(sq)){
    Nt[j]=max(c(0,which(x<sq[j])))
  }
  Nt=c(0,Nt)
  return(sq,a,x,Nt,plot(stepfun(sq,Nt),verticals=F,do.points=F,
  main="Proceso de Poisson",xlab="t",ylab="Nt")) }
```

4.2.8. Algoritmo para la simulación de un proceso de Poisson compuesto

1. Elegimos un intervalo $[0, T]$ y un cierto N para establecer una partición de tal forma que tengamos $N+1$ puntos $0 = t_0 < t_1 < \dots < t_N = T$ y pongamos $C_0 = 0$

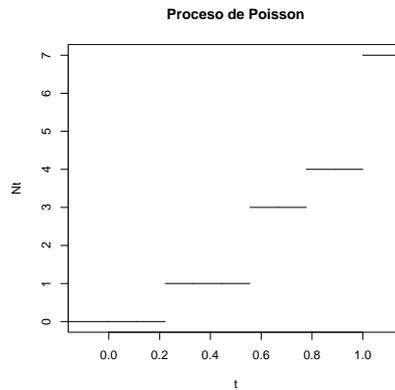


Figura 4.2: Poisson método 2

2. Generamos N (*saltos*) y N *tiempos de salto* J_1, J_2, \dots, J_N como en el caso del proceso de Poisson.
3. Simulamos N variables aleatorias idénticamente distribuidas Y_1, Y_2, \dots, Y_N de una distribución dada, por ejemplo normales.
4. Finalmente, sea $C(t_i) = \sum_{j=0}^{N(t_i)} Y_j$ donde $Y_0 = 0$

4.2.9. Programa pproccc

```

pproccc=function(T=1,N=10,M=1000,lambda=5){
  set.seed(100)
  sq=seq(0,T,length=N)
  lambda1=lambda*T
  Nt=numeric(0)
  Ct=numeric(0)
  a=rpois(1,lambda1)
  b=runif(a,0,T)
  b1=sort(b)
  x=b1
  Rc=rnorm(a)
  for(j in 1:length(sq)){
    Nt[j]=max(c(0,which(x<sq[j])))
  }
}

```

```

s=0
for(j in 1:length(sq)){
Ct[j]= sum(Rc[0:Nt[j]])
}
return(sq,a,x,Nt,Rc,Ct,plot(stepfun(sq,c(0,Ct)),
verticals=F,do.points=F,main="Proceso compuesto de
Poisson",xlab="t",ylab="y"))
}

```

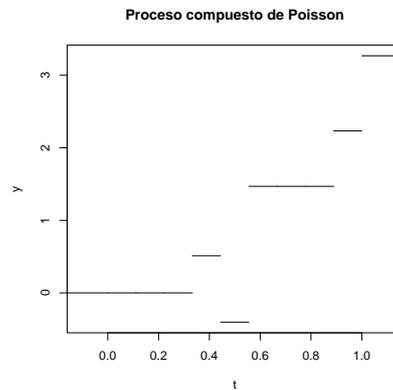


Figura 4.3: Poisson compuesto

4.2.10. Algoritmo para la simulación de un proceso simple de Lévy

1. Elegimos un intervalo $[0,T]$ y un cierto N para establecer una partición de tal forma que tengamos $N+1$ puntos: $0 = t_0 < t_1 < \dots < t_N = T$ y pongamos $X(0) = 0$
2. Generamos un movimiento browniano $W(t)$ y un proceso de Poisson compuesto $C(t)$.
3. Finalmente, definimos para cada t_i :

$$X(t_i) = aW(t) + C(t) + \gamma(t_i - t_{i-1})$$

4.2.11. Programa levy1

```

levy1=function(T=1,N=500,lambda=5) {
  set.seed(200)
  dt=T/N
  dw=0
  w=0
  dw=sqrt(dt)*rnorm(N)
  w=cumsum(dw)
  x=seq(0,T-dt,dt)
  sq=seq(0,T,length=N)
  lambda1=lambda*T Nt=numeric(0)
  Ct=numeric(0)
  a=rpois(1,lambda1)
  b=runif(a,0,T)
  b1=sort(b)
  x=b1
  Rc=rnorm(a)
  for(j in 1:length(sq)){
    Nt[j]=max(c(0,which(x<sq[j])))
  }

  s=0
  for(j in 1:length(sq)){
    Ct[j]= sum(Rc[0:Nt[j]])
  }
  Lt=sq+w+Ct
  a=list(Proceso=Lt,a=plot(sq,Lt,type="l",main="Proceso de Levy",
xlab="t",ylab="Y")) return(a)
}

```

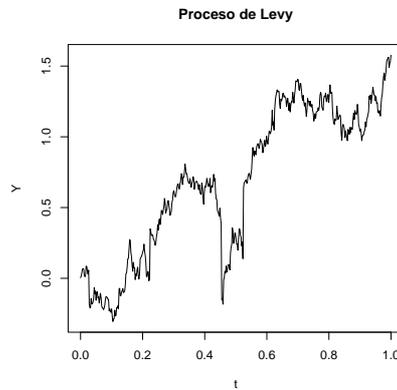


Figura 4.4: Proceso de Levy

4.3. Aplicación a los mercados de acciones

La aplicación correspondiente del Proceso de Levy a los *mercados de acciones* se denomina modelo de Merton, que se describe a través de la siguiente ecuación:

$$S(t) = S(0)exp(\gamma t + \sigma W(t) + \sum_{i \geq 1}^{N(t)} Y_i) \quad (4.6)$$

donde:

- $W(t)$ es un movimiento browniano cualquiera
- $N(t)$ es un proceso de Poisson con intensidad λ
- $Y_i \sim N(\mu, \sigma^2)$ son variables aleatorias i.i.d independientes de $W(t)$ y $N(t)$
- $\gamma = r - \frac{\sigma^2}{2} - \lambda(e^{\mu + \frac{\sigma^2}{2}} - 1)$
- r tasa de interés cualquiera

Capítulo 5

Conclusiones

Del trabajo precedente se puede concluir lo siguiente:

1. En la integral de Reimann se puede elegir cualquier punto de muestra, ya sea el extremo izquierdo del intervalo, el derecho o el punto medio y cuando se afina la partición las diferentes integrales convergen a un solo valor. Sin embargo para las integrales estocásticas tratadas, esto es, *Itô* y *Stratonovich*, el punto elegido es muy importante puesto que define integrales diferentes, además no importa cuanto se afine la partición las integrales no convergen entre si, sin embargo dentro de cada integral , cuando la partición se afina, el error entre el valor exacto y el valor encontrado a través de la simulación disminuye lo que provoca que cada integral converja a su valor exacto.
2. A diferencia de las integrales tratadas en este trabajo que usan como integrador el *movimiento browniano* (integrales de *Itô* y *Stratonovich*), la integral que usa como integrador el *movimiento browniano fraccionario* (fractional pathwise integral) no converge, no importa cuanto se afine la partición .
3. Al integrar usando el *movimiento browniano fraccionario* el enfoque del *fractional pathwise* provoca *arbitraje*, una sugerencia para trabajos posteriores sería usar el enfoque de *Wick-Itô* que no provoca tal arbitraje.
4. A la hora de construir los modelos aplicados a finanzas se descubre que los modelos que usan los procesos de Lévy representan mejor a la realidad puesto que a diferencia del *movimiento browniano* y del *movimiento browniano fraccionario* los procesos de Lévy tiene saltos, los cuales representan mejor el comportamiento de los *precios de acciones*.

Bibliografía

- [1] Szymon Borak. Levy process,lecture 1.
- [2] Zhongyin Jhon Daye. Introduction to fractional brownian motion in finance. Mayo,2003.
- [3] Ton Dieker. *Simulation of fractional Brownian motion*. PhD thesis, University of Twente, Amsterdam,The Netherlands, 2004.
- [4] Laurence C. Evans. *An Introduction to Stochastic Differential Equations*. Department of Mathematics ,U.C.Berkeley.
- [5] Desmond J. Higham. An algorithmic introduction to numerical simulation of stochastic differential equations. *SIAM Review*, 43(3):525–546, 2001.
- [6] Desmond J. Higham. Fractional brownian motion. 2001.
- [7] Sheldon M.Ross. *Introduction to Probability Models*. Harcourt,Academic Press.
- [8] R Development Core Team. *R: A language and environment for statistical computing*. R Foundation for Statistical Computing, Vienna, Austria, 2005. ISBN 3-900051-07-0.
- [9] Rudiger Seydel. *Tools for Computational Finance*. Springer.
- [10] James Stewart. *Cálculo de una variable,trascedentes tempranas,Cuarta Edición*. Thomson Learning.
- [11] A.V.Swishchuk Y.S.Mishura. Stochastic stability of fractional (b,s)-securities markets. 2001.

Apéndice A

Método de Euler

Expondremos brevemente el método de Euler para Ecuaciones Diferenciales Ordinarias, ver [10].

Para describir este método partamos del problema de valor inicial

$$\frac{dy}{dt} = f(t, y), y(t_0) = y_0$$

Como $f(t, y)$ está dada, podemos trazar su **campo de direcciones** en el plano t-y.

La idea del método es empezar en el punto (t_0, y_0) y dar pequeños pasos dictados por las tangentes de esta.

Primero elegimos un tamaño de paso Δt . La pendiente de la solución aproximada es recalculada cada Δt de t.

Comenzando en (t_0, y_0) , nuestro primer paso es ir a (t_1, y_1) donde $t_1 = t_0 + \Delta t$ y (t_1, y_1) es un punto sobre la línea que pasa por (t_0, y_0) y cuya pendiente es proporcionada por el campo de direcciones en (t_0, y_0) . En (t_1, y_1) repetimos el procedimiento. Dando un paso de tamaño Δt y cuya dirección está determinada por el campo de direcciones en (t_1, y_1) , llegamos al nuevo punto (t_2, y_2) . El nuevo tiempo está dado por $t_2 = t_1 + \Delta t$ y está sobre el segmento que pasa por (t_1, y_1) y tiene pendiente $f(t_1, y_1)$. De la misma manera, usamos el campo de direcciones en (t_k, y_k) para calcular el siguiente punto (t_{k+1}, y_{k+1}) . La secuencia de valores y_1, y_2, \dots sirve como aproximación a la solución en los tiempos t_1, t_2, \dots . Para poner en práctica el Método de Euler, necesitamos una fórmula que determine (t_{k+1}, y_{k+1}) a partir de (t_k, y_k) . Encontrar t_{k+1} es fácil puesto que

$$t_{k+1} = t_k + \Delta t$$

Para obtener y_{k+1} usamos la ecuación diferencial. Sabemos que la pendiente de la solución a la ecuación $\frac{dy}{dt} = f(t, y)$ en el punto (t_k, y_k) es $f(t_k, y_k)$ y el método de Euler

usa esta pendiente para calcular y_{k+1} .

Usando la fórmula para pendientes obtenemos

$$\frac{y_{k+1} - y_k}{t_{k+1} - t_k} = f(t_k, y_k)$$

como $\Delta t = t_{k+1} - t_k$ obtenemos:

$$\frac{y_{k+1} - y_k}{\Delta t} = f(t_k, y_k)$$

$$y_{k+1} - y_k = f(t_k, y_k)\Delta t$$

$$y_{k+1} = y_k + f(t_k, y_k)\Delta t$$

Esta última fórmula resume el método de Euler.

Método de Euler para $dy/dt=f(t,y)$ Dada la condición inicial $y(t_o) = y_o$ y el tamaño del paso Δt , calcule el punto (t_{k+1}, y_{k+1}) a partir del punto precedente (t_k, y_k) como sigue:

1. Use la ecuación diferencial para determinar la pendiente $f(t_k, y_k)$.
2. Calcule el siguiente punto (t_{k+1}, y_{k+1}) mediante las fórmulas

$$t_{k+1} = t_k + \Delta t$$

y

$$y_{k+1} = y_k + f(t_k, y_k)\Delta t$$