Implementación del Método de Metropolis–Hastings en una Aplicación Actuarial

Valoración Bayesiana de Opciones Asiáticas

Temas Selectos 1: Simulación Actuarial

22 de noviembre de 2025

Equipo: 1

Contenido

Introducción y Motivación

Modelo Estadístico

Método de Metropolis–Hastings

Resultados

Aplicación: Valoración de Opciones

Conclusiones

Introducción y Motivación

Contexto: Opciones Asiáticas

¿Qué son?

- Derivados cuyo *payoff* depende del promedio del precio del activo
- Más comunes que opciones europeas en ciertos mercados

Payoff (Call asiática):

$$\text{Payoff} = \text{máx}\left(\bar{S}_T - K, 0\right)$$

donde $\bar{S}_T = \frac{1}{T}\int_0^T S_t\,dt$ (promedio continuo)

Aplicaciones:

- Commodities: Petróleo, gas, metales
- Cobertura: Flujos continuos
- Gestión de riesgo: Reduce volatilidad

Ventaja clave
Menor costo que opciones europeas

Desafío Principal
Valorar opciones asiáticas cuando los parámetros del proceso estocástico (μ, σ) son desconocidos e inciertos.

Enfoque Clásico (MLE):

- × Estimaciones puntuales
- × Ignora incertidumbre
- × Subestima riesgo

Enfoque Bayesiano (MCMC):

- ✓ Distribuciones completas
- ✓ Cuantifica incertidumbre
- ✓ Propaga riesgo al precio

Objetivo del Proyecto
Implementar Metropolis–Hastings para inferencia bayesiana de parámetros y valoración robusta de opciones asiáticas.

Modelo Estadístico

Modelo del Activo Subyacente

Proceso estocástico: Geometric Brownian Motion (GBM)

$$dS_t = \mu S_t\, dt + \sigma S_t\, dW_t$$

- μ: drift (retorno esperado anualizado)
- σ: volatilidad (desviación estándar anualizada)
- W_t: Movimiento Browniano estándar

Discretización (Lema de Itô):

Aplicando a $\log S_t$:

$$r_t := \log\left(\frac{S_{t+\Delta t}}{S_t}\right) \sim N\left(\left(\mu - \frac{\sigma^2}{2}\right)\Delta t, \sigma^2 \Delta t\right)$$

Parámetros a estimar

$\theta = (\mu, \sigma)$ usando datos de log-retornos observados

Modelo Bayesiano

Verosimilitud
Dada una muestra de n log-retornos $\mathbf{r} = (r_1, \ldots, r_n)$:

$$p(\mathbf{r} \mid \mu, \sigma) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2 \Delta t}} \exp\left(-\frac{(r_i - m)^2}{2\sigma^2 \Delta t}\right)$$

donde $m = \left(\mu - \frac{\sigma^2}{2}\right) \Delta t$

Priors (no informativos):

$$\mu \sim N(0, 1)$$

$$\sigma \sim \mathrm{InvGamma}(2, 0{,}1)$$

Posterior (Teorema de Bayes):

$$p(\mu, \sigma \mid \mathbf{r}) \propto p(\mathbf{r} \mid \mu, \sigma) \times p(\mu) \times p(\sigma)$$

Problema

No tiene forma cerrada $\Rightarrow$ Requiere métodos MCMC

Método de Metropolis–Hastings

Fundamento del Algoritmo

Objetivo: Generar muestras de la distribución posterior $p(\theta \mid y)$ cuando solo conocemos $p(\theta \mid y)$ hasta una constante normalizante.

Idea Central
Construir una cadena de Markov cuya distribución estacionaria sea la posterior deseada.

Componentes clave:

1. **Distribución objetivo:** $\pi(\theta) = p(\theta \mid y)$ (posterior)
2. **Distribución de propuesta:** $q(\theta^* \mid \theta^{(t)})$
3. **Razón de aceptación:** α que garantiza balance detallado

Propiedad Clave
Bajo condiciones regulares, la cadena converge a la distribución posterior sin importar el punto inicial.

Algoritmo de Metropolis–Hastings

Algorithm 1 Metropolis–Hastings

Inicializar $\theta^{(0)}$
for $t = 1$ to N **do**
    Proponer $\theta^* \sim q(\theta^* \mid \theta^{(t-1)})$
    Calcular razón de aceptación:

$$\alpha = \text{mín}\left(1, \frac{p(\theta^* \mid y)\, q(\theta^{(t-1)} \mid \theta^{)}}{p(\theta^{(t-1)} \mid y)\, q(\theta | \theta^{(t-1)})}\right)$$

    Generar $u \sim \text{Uniform}(0, 1)$
    if $u < \alpha$ **then**
        $\theta^{(t)} = \theta^*$ (aceptar)
    else
        $\theta^{(t)} = \theta^{(t-1)}$ (rechazar)
    end if
end for

Propuesta Random Walk: $\theta^* = \theta^{(t-1)} + \epsilon, \quad \epsilon \sim N(0, \Sigma)$

$\Rightarrow$ Simétrica: $q(\theta^* \mid \theta) = q(\theta \mid \theta^*) \Rightarrow$ Se simplifica α

Configuración de la Implementación

Datos Simulados:

- $\mu_{\text{true}} = 0{,}08$
- $\sigma_{\text{true}} = 0{,}25$
- $S_0 = 100$
- $T = 1$ año
- $n = 252$ observaciones

Parámetros MCMC:

- Iteraciones: 30,000
- Burn-in: 5,000
- Propuesta SD: [0.01, 0.01]

Resultado
Tasa de aceptación: **72.1 %** ✓Óptimo (rango ideal: 40–70 %)

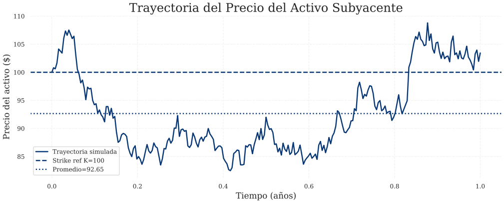

Figura 1: Trayectoria simulada del activo subyacente

Resultados

Diagnósticos MCMC: Convergencia

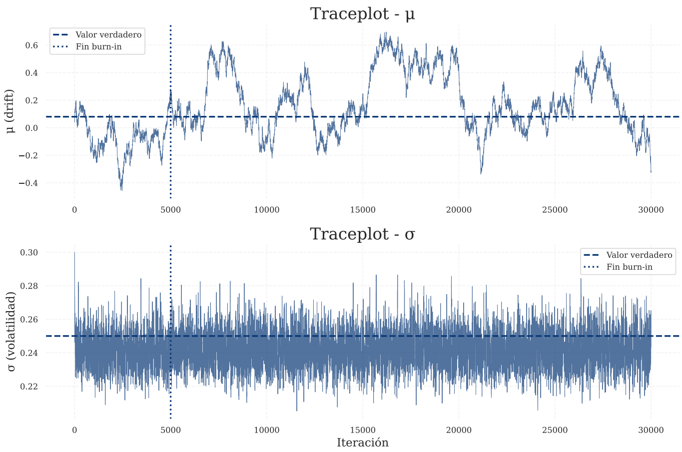

Figura 2: Traceplots de los parámetros μ y σ. Convergencia alcanzada después del burn-in.

- ✓ Cadena de σ mezcla bien
- ! Cadena de μ muestra alta autocorrelación (exploración lenta)

Distribuciones Posteriores

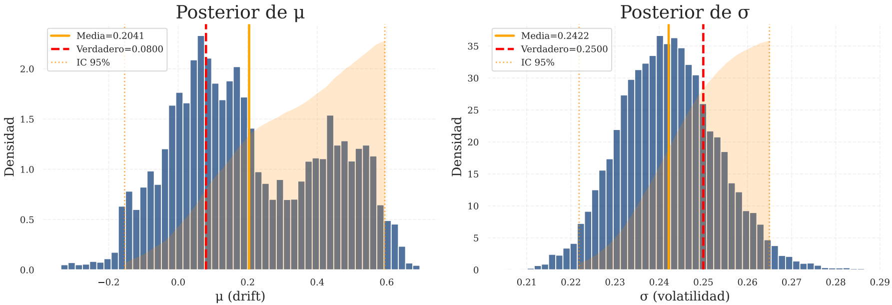

Figura 3: Distribuciones posteriores marginales con intervalos creíbles del 95 %

- **Volatilidad σ:** Estimación muy precisa, IC estrecho
- **Drift μ:** Mayor incertidumbre, IC amplio (consistente con teoría)

Cuadro 1: Estimaciones Bayesianas vs. Clásicas (MLE)

Parámetro	**Verdadero**	**Media Post.**	**IC 95 %**	**MLE**
μ (drift)	0.0800	0.2041	$[-0{,}15, 0{,}59]$	0.0629
σ (vol.)	0.2500	0.2422	$[0{,}22, 0{,}26]$	0.2413

Observaciones Clave

- MLE y media bayesiana casi **idénticos para** σ
- Gran diferencia en μ: refleja incertidumbre no capturada por MLE
- IC para μ muy amplio: difícil estimar retorno esperado con datos limitados

Diagnósticos MCMC: Autocorrelación

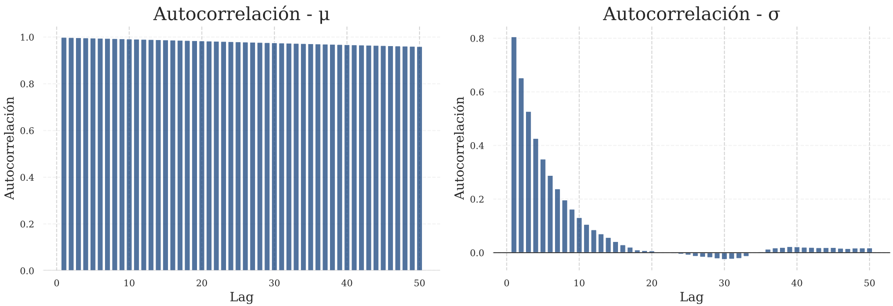

Figura 4: Funciones de autocorrelación para ambos parámetros

Tamaño Efectivo (ESS):

- μ: 258 / 25,000
- σ: 2,601 / 25,000

Implicación:

× Alta autocorr. en μ

✓ Buena mezcla en σ

Distribución Conjunta Posterior

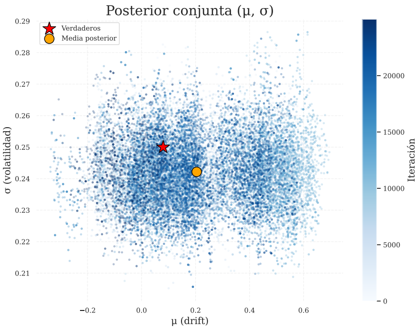

Figura 5: Distribución conjunta posterior (μ, σ). Correlación débil entre parámetros.

- Estrella roja: Valores verdaderos
- Círculo naranja: Media posterior
- Nube de puntos: Región de alta probabilidad posterior

Aplicación: Valoración de Opciones

Valoración de Opción Asiática

Especificación de la opción:

- Tipo: Call asiática (promedio aritmético continuo)
- $S_0 = 100$, $K = 100$ (at-the-money), $T = 1$ año, $r = 3\,\%$

Metodología:

1. Para cada muestra posterior $(\mu^{(i)}, \sigma^{(i)})$
2. Simular 1,000 trayectorias del precio (Monte Carlo)
3. Calcular precio de la opción
4. Obtener distribución posterior del precio

Cuadro 2: Resultados de Valoración

Método	Precio ($)	SD/SE	IC 95 %
Parámetros Verdaderos	7.74	0.10	—
Estimación MLE	7.16	0.10	—
Inferencia Bayesiana	14.11	9.76	[2.39, 35.11]

Distribución Posterior del Precio

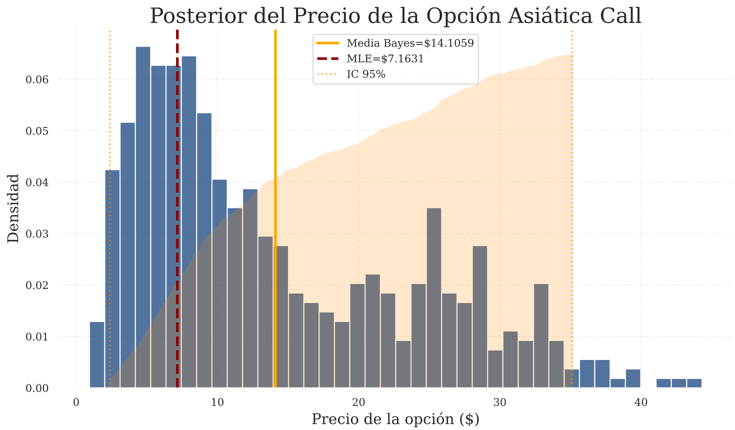

Figura 6: Distribución posterior del precio de la opción asiática. Bimodal y muy dispersa.

¡Enorme Incertidumbre!
IC 95 %: [$2.39, $35.11] $\Rightarrow$ El precio podría ser hasta 5 veces mayor que la estimación MLE

Interpretación Actuarial

Implicaciones para Gestión de Capital

1. **MLE subestima riesgo**: Precio puntual \$7.16 ignora incertidumbre
2. **Capital requerido**: Usar percentil 95 % (\$35.11) $\Rightarrow$ 5× mayor
3. **Pricing conservador**: Percentil 75–90 (\$17–\$25) como precio de venta
4. **Reservas**: Necesarias significativamente mayores que sugiere MLE

Medidas de Riesgo:

- $\text{VaR}_{95\,\%} = \$35.11$
- $\text{ES}_{95\,\%} \approx \$38$

Fuente de incertidumbre:

- Principalmente: drift μ
- μ alto $\Rightarrow$ trayectorias crecientes $\Rightarrow$ payoffs grandes

Comparación: Bayesiano vs. Clásico

Cuadro 3: Enfoque Bayesiano vs. MLE en Valoración

Aspecto	**MLE (Clásico)**	**Bayesiano (MCMC)**
Estimación	Puntual: \$7.16	Distribución: \$14.11 $\pm$ \$9.76
Incertidumbre	No cuantificada	IC: [\$2.39, \$35.11]
Capital requerido	\$7.16	\$35.11 (perc. 95 %)
Decisiones	Potencialmente riesgosas	Conservadoras y robustas
Gestión de riesgo	Inadecuada	Completa

Conclusión Clave
El enfoque clásico subestima dramáticamente el riesgo real de la posición.

Conclusiones

Conclusiones Principales

1. Éxito de la Implementación

- ✓ Algoritmo M–H implementado **desde cero** en Python
- ✓ Convergencia verificada con tasa de aceptación óptima (72 %)
- ✓ Diagnósticos MCMC confirman calidad de las muestras

2. Valor del Enfoque Bayesiano

- ✓ **Cuantifica incertidumbre** paramétrica completa
- ✓ **Revela riesgos ocultos** no capturados por MLE
- ✓ **Permite decisiones robustas** en gestión de capital

3. Implicación Actuarial Crítica

La valoración clásica (MLE) puede llevar a reservas insuficientes y capital inadecuado, exponiendo a la institución a riesgos catastróficos.

Limitaciones y Trabajo Futuro

Limitaciones:

- Alta autocorrelación en μ
- Datos sintéticos (idealmente: datos reales de mercado)
- Modelo simple (GBM)

Mejoras Posibles:

- Hamiltonian MC (HMC)
- Reparametrización
- Propuestas adaptativas

Extensiones:

- Volatilidad estocástica (Heston)
- Saltos (Merton)
- Opciones sobre canastas
- Modelos multivariados

Aplicaciones Adicionales:

- Reserving bayesiano
- Modelos de credibilidad
- Riesgo de crédito
- Pérdidas agregadas

Relevancia Actuarial del MCMC

¿Por qué MCMC es esencial en Actuaría moderna?

1. **Complejidad creciente**: Modelos sin soluciones analíticas
2. **Requerimientos regulatorios**: Solvencia II, ORSA
3. **Gestión de riesgos**: Cuantificación completa de incertidumbre
4. **Robustez**: Decisiones que resisten incertidumbre de modelo

Áreas de aplicación:

- Reservas técnicas (Chain Ladder bayesiano)
- Pricing de productos complejos
- Modelos de dependencia (cópulas)
- Riesgo operacional y catastrófico
- Stress testing y análisis de escenarios

Mensaje Final
Metropolis–Hastings es una herramienta fundamental que todo actuario debe dominar para enfrentar los desafíos del siglo XXI.

Contribuciones del Proyecto

Logros Técnicos

1. Implementación completa de M–H sin librerías especializadas
2. Aplicación a problema actuarial realista (opciones asiáticas)
3. Análisis exhaustivo con 6 visualizaciones de calidad
4. Comparación rigurosa: Bayesiano vs. Clásico

Aportaciones Conceptuales

- Demostración de **subestimación de riesgo** en métodos puntuales
- Cuantificación de **impacto financiero** de incertidumbre paramétrica
- Metodología replicable para otros derivados y productos

Código y Documentación

- Código Python documentado (500+ líneas)
- Reporte técnico LaTeX (18 páginas)
- 9 referencias académicas

¡Gracias!

Preguntas y Comentarios

Proyecto: Método de Metropolis–Hastings
Aplicación: Valoración Bayesiana de Opciones Asiáticas
Materia: Temas Selectos 1 - Simulación Actuarial

Referencias i

- Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., & Teller, E. (1953). *Equation of state calculations by fast computing machines*. The Journal of Chemical Physics, 21(6), 1087–1092.
- Hastings, W. K. (1970). *Monte Carlo sampling methods using Markov chains and their applications*. Biometrika, 57(1), 97–109.
- Robert, C. P., & Casella, G. (2004). *Monte Carlo Statistical Methods* (2nd ed.). Springer.
- Hull, J. C. (2018). *Options, Futures, and Other Derivatives* (10th ed.). Pearson.
- Glasserman, P. (2004). *Monte Carlo Methods in Financial Engineering*. Springer.
- Klugman, S. A., Panjer, H. H., & Willmot, G. E. (2012). *Loss Models: From Data to Decisions* (4th ed.). Wiley.

Apéndice: Detalles Técnicos del Código

Estructura del código Python (500+ líneas):

1. **Generación de datos**: Simulación GBM con parámetros conocidos
2. **Función log-posterior**: Likelihood + Priors
3. **Algoritmo M–H**: Implementación desde cero con Random Walk
4. **Diagnósticos**: Traceplots, autocorrelación, ESS
5. **Valoración**: Monte Carlo para opciones asiáticas
6. **Visualización**: 6 gráficas de alta calidad

Herramientas utilizadas:

- NumPy (cálculos numéricos y álgebra lineal)
- SciPy (distribuciones de probabilidad)
- Matplotlib (visualización)
- Pandas (manejo de datos tabulares)

Apéndice: Fragmento de Código (Log-Posterior)

```python
def log_posterior(theta, log_retornos, dt):
    """
    Calcula el log de la posterior (hasta constante normalizante)
    theta = [mu, sigma]
    """
    mu, sigma = theta

    # Restricción: sigma > 0
    if sigma <= 0:
        return -np.inf

    # Log-likelihood
    media_retornos = (mu - 0.5 * sigma**2) * dt
    var_retornos = sigma**2 * dt

    log_lik = -0.5 * len(log_retornos) * np.log(2 * np.pi * var_retornos)
    log_lik -= 0.5 * np.sum((log_retornos - media_retornos)**2) / var_retornos

    # Log-prior para mu: N(0, 1)
    log_prior_mu = -0.5 * np.log(2 * np.pi) - 0.5 * mu**2

    # Log-prior para sigma: InverseGamma(2, 0.1)
    alpha, beta = 2, 0.1
    log_prior_sigma = alpha * np.log(beta) - (alpha + 1) * np.log(sigma) - beta / sigma

    return log_lik + log_prior_mu + log_prior_sigma
```

Apéndice: Fórmulas Adicionales

Razón de aceptación (Random Walk simétrico):

Como $q(\theta^* \mid \theta^{(t)}) = q(\theta^{(t)} \mid \theta^*)$, se simplifica a:

$$\alpha = \text{mín}\left(1, \frac{p(\theta^* \mid y)}{p(\theta^{(t)} \mid y)}\right) = \text{mín}\left(1, \exp\left(\log p(\theta^* \mid y) - \log p(\theta^{(t)} \mid y)\right)\right)$$

Tamaño Efectivo de Muestra (ESS):

Aproximación mediante autocorrelación:

$$\text{ESS} \approx \frac{N}{1 + 2\sum_{k=1}^{K} \rho_k}$$

donde ρ_k es la autocorrelación en lag k.