



Instituto Nacional de Matemática Pura e Aplicada

Uma Abordagem Bayesiana para Calibragem de Superfície de Volatilidade Local

Autor: **Leonardo Fonseca Sarmanho**

Orientador: **Alfredo Noel Iusem**

Coorientador: **Vinícius Viana Luiz Albani**

Rio de Janeiro

Março de 2022

Para minha Família.

Agradecimentos

Gostaria de agradecer a todos que me ajudaram, direta ou indiretamente, no desenvolvimento deste trabalho.

Primeiramente, agradeço à minha mãe Cida e ao meu pai Luiz Osvaldo pelo amor e apoio durante toda minha vida e, por assertivamente, priorizarem os estudos em todos os momentos da minha vida. Obrigado pela esforço e abdicção que tiveram para criar meus irmãos e a mim. Sem esse suporte nada disso seria possível.

Agradeço à minha madrinha, Liliam, pelo conforto e refúgio em meio aos problemas que apareceram ao longo do caminho. Sem sua ajuda tudo seria muito mais difícil.

Agradeço à minha irmã Izabela pela ajuda, segurança e conforto que me foi proporcionado desde sempre. Tive o privilégio de poder contar com sua iniciativa e inteligência nas situações mais adversas em nossas vidas. Por isso, Obrigado.

Agradeço de forma especial ao meu irmão Gabriel por ser como um pai para mim, mesmo sem nenhuma obrigação; por ser espelho para mim, seja pessoal ou profissionalmente. Agradeço ao fato de você ter tornado a vida da nossa família mais fácil e mais leve. Obrigado, por ter me mostrado o caminho das exatas e por ter me apoiado nessa longa estrada, sem duvidar. Agradeço, também, à minha cunhada Thaís por todos os momentos de descontração e apoio ao longo desses anos, e, claro, pelo meu melhor presente do ano, minha afilhada Flora.

Agradeço, também, à minha esposa Alana pelo amor, paciência e ajuda diária na conclusão desta dissertação. Obrigado por estar ao meu lado, me apoiando, nos momentos mais complicados, de forma racional e madura. Sua generosidade, honestidade e alegria me inspiram e me ajudam a continuar em frente.

Agradeço ao meu grande amigo Mario pelo suporte, motivação e companhia desde o início da nossa graduação até o final desse mestrado.

Agradeço, também, a todos os professores do IMPA, em especial, aos orientadores Vinícius Albani, Alfredo Iusem e Jorge Zubelli pelo apoio e bizzos determinantes e valiosos no desenvolvimento deste trabalho e durante todo o curso.

Por fim, agradeço a todos os amigos, monitores e professores que estiveram presentes durante o desenvolvimento desta tese pelo apoio e ensinamentos fundamentais.

Resumo

A precificação de um instrumento financeiro e a estimação de sua medida de risco, a volatilidade, são questões fundamentais nas finanças corporativas, sendo o modelo de Black-Scholes (1973) um marco na teoria fundamental de apreçamento. Embora abrangente, algumas de suas premissas não refletem o funcionamento do mercado, como a de volatilidade constante. Ao supor que os coeficientes de difusão variem em função do tempo, origina-se o conceito de volatilidade local. Neste trabalho, um novo método de estimação da volatilidade local é proposto a partir de uma modificação de modelo proposto em Dupire (1994). No modelo proposto, os preços das opções de compra são descritos em função da densidade de máxima entropia (DME) em contraponto à densidade com difusão de acordo com a equação de Fokker-Planck de Dupire. A estimação da DME durante a interpolação dos preços é realizada segundo Neri-Schneider (2012), cujo processo de minimização é solucionado numericamente via o método de Newton-Raphson. O ajuste do problema inverso da calibragem da volatilidade local, para encontrar a superfície de volatilidade via o método Bayesiano, é resolvido via método dos mínimos quadrados, sendo a regularização de Tikhonov aplicada para fins de estabilização do problema mal-posto, no sentido de Hadamard. Por fim, o novo método proposto foi aplicado primeiramente a dados simulados para fins de validação dos algoritmos implementados e, em seguida, a dados reais de opções de compra S&P 500.

Palavras Chaves: Densidade de Máxima Entropia, Volatilidade Local, Métodos Bayesianos, Regularização de Tikhonov.

Abstract

The pricing of a financial instrument and the estimation of its measure of risk, the volatility, are fundamental problems in corporate finance, with the Black-Scholes (1973) model being a landmark in the standard fundamental theory of pricing. Although comprehensive, some of its assumptions, such as constant volatility, do not reflect the market's functioning. By assuming that the diffusion coefficients vary as a function of time, the concept of local volatility arises. In this work, a new method for estimating the local volatility is proposed based on a modification of the model introduced in Dupire (1994). In the new model, call option prices are modeled as a function of maximum entropy density (DME) as opposed to the diffusion density according to Dupire's Fokker-Planck equation. The DME estimation during the price interpolation is performed according to Neri-Schneider (2012) whose minimization process is numerically solved via the Newton-Raphson method. In order to find the volatility surface via the Bayesian paradigm, the adjustment of the inverse problem of calibrating the local volatility is solved by means of the least-square method, with the Tikhonov regularization applied to stabilize the ill-posed problem, in the sense of Hadamard. Finally, the new proposed method was firstly applied to simulated data for algorithms' validation purposes and then to S&P 500 call options.

Key words: Maximum Entropy Density, Local Volatility, Bayesian Methods, Tikhonov regularization.

Sumário

1	Introdução	3
1.1	Organização da Dissertação	4
1.2	Revisão Bibliográfica	4
2	Modelos de Volatilidade	7
2.1	O Modelo de Black-Scholes	7
2.1.1	A Volatilidade Implícita	10
2.2	O Modelo de Volatilidade Local	10
3	Distribuição de Máxima Entropia	13
3.1	DME Usando Opções de Compra	13
3.2	DME Usando Opções de Compra e Opções Digitais	14
4	Calibragem da Volatilidade Local	17
4.1	O Problema Inverso	17
4.1.1	O Problema Inverso e o Teorema de Bayes	18
4.1.2	Regularização de Tikhonov	19
4.1.3	O Critério da curva em L	20
4.2	O Processo de Calibragem	21
5	Resultados Numéricos	25
5.1	Dados Simulados	25
5.1.1	Resultados	28
5.2	Dados reais	33
5.2.1	Resultados	33
6	Conclusão	39
A	A Equação de Dupire	41
B	Códigos	43
B.1	MED	43
B.2	Calibragem	47
B.3	Vol Implícita	51
	Referências Bibliográficas	57

Capítulo 1

Introdução

No âmbito das finanças, um derivativo é um instrumento financeiro cujo preço é derivado de um ativo ou bem financeiro. Esses instrumentos são de extrema importância no cotidiano do mercado financeiro, sendo utilizados para mitigar o risco de portfólios e ativos causados pelas oscilações dos preços. Alguns tipos de derivativos são futuros, swaps e opções.

De um modo geral, um problema comum no contexto dos derivativos é a necessidade de precificar esses instrumentos financeiros. No caso das opções, instrumento que dá a seu detentor o direito, mas não a obrigação, de comprar (*call*) ou vender (*put*) um ativo financeiro (*underlying*) tem-se os mais variados métodos e modelos para calcular o seu preço.

Dentre esses modelos, o que mais se destaca é o modelo de Fischer Black e Myron Scholes apresentado em [Black and Scholes \(1973\)](#). Nesse trabalho os autores, após assumirem algumas suposições, demonstram que o preço dessa opção denotado por V , satisfaz a seguinte equação diferencial parcial:

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

Ou seja, o preço desse derivativo em função do preço do ativo objeto e do tempo, satisfaz uma EDP (equação de difusão). Em [Black and Scholes \(1973\)](#) os autores mostram que é possível chegar a esse resultado considerando uma "posição protegida", consistindo de um portfólio comprado no ativo objeto e vendida na opção de compra.

Apesar da utilidade e importância, algumas suposições do modelo de Black-Scholes vão de encontro com o que é visto nos dados reais, como por exemplo o fato de ser considerada a volatilidade constante. No mercado, observa-se que a volatilidade implícita varia com o *strike* e com o vencimento das opções, ficando aparente o *smile* nas curvas de volatilidade implícita. Dessa forma, outros modelos surgiram com o intuito de concordar com o que é visto na prática. Um desses modelos, o de volatilidade local, será o abordado ao longo desse texto.

O Modelo de volatilidade local é aquele que considera a volatilidade como função, determinística, do preço do ativo objeto e do tempo. Esse modelo apresenta uma série de características interessantes como por exemplo: É um modelo relativamente fácil de calibrar e, por não ter nenhum fator de risco adicional, a completude do mercado é mantida, permitindo o *Hedge* somente com o ativo objeto e opções do mesmo [Dupire \(1994\)](#).

De um modo geral, para encontrar a volatilidade local, é necessário resolver um problema inverso. Além disso, no caso apresentado ao longo do texto, esse problema inverso é mal-posto no sentido de Haddamard e para lidar com essa dificuldade, será utilizada a técnica de regularização de Tikhonov [Crepey \(2003\)](#); [Albani et al. \(2013\)](#).

Nesse trabalho uma abordagem diferente será utilizada a fim de calibrar a volatilidade local. A proposta desse trabalho de dissertação consiste em simplificar a dimensionalidade desse problema de calibragem e, para isso, será proposta uma abordagem Bayesiana para avaliar a densidade de um ativo e, para completar, será utilizada a distribuição de máxima entropia (DEM) para interpolar os preços das opções, técnica abordada em [Neri and Schneider \(2012\)](#). No passo da calibragem na equação do funcional será adicionado um termo, com o coeficiente de Tikhonov, para estabilizar o problema.

1.1 Organização da Dissertação

Esta dissertação está organizada em seis capítulos. No presente capítulo, além do texto introdutório acima, será feita, na seção [1.2](#) a seguir, uma revisão bibliográfica dos modelos e métodos utilizados até o momento para problemas correlatos. No segundo capítulo, os modelos de volatilidade serão abordados, com ênfase no modelo clássico de Black and Scholes e no de volatilidade local. No terceiro capítulo serão detalhados a intenção e o uso da distribuição de máxima entropia. No quarto capítulo serão abordados o passo a passo da calibragem e a obtenção da superfície de volatilidade local. No capítulo 5 serão apresentados os estudos feitos com dados sintéticos e uma aplicação a dados reais de opções. Por fim, no capítulo 6 (Conclusão) serão realizados os comentários finais a respeito do trabalho aqui proposto e as perspectivas de desenvolvimento futuro do tema.

1.2 Revisão Bibliográfica

O objetivo deste trabalho é propor uma abordagem para a obtenção e calibragem da volatilidade local a partir de um modelo Bayesiano utilizado para avaliar a densidade de um ativo.

Para obter a função de volatilidade local, esbarra-se em um problema inverso, que é ainda mal-posto no sentido de Hadamard, ou seja, não satisfaz uma das condições: de existência, unicidade e/ou dependência contínua dos dados iniciais. Para contornar esse problema será utilizada a técnica de regularização de Tikhonov, a fim de estabilizar o problema inverso em estudo [Tikhonov \(1943\)](#). Para encontrar os valores do coeficiente de regularização de Tikhonov, o critério da curva em L foi utilizado ([Hansen, 2001](#)).

A partir do modelo de Black-Scholes ([Black and Scholes, 1973](#)) diversos modelos foram desenvolvidos para estender e melhorar o modelo base para precificação de opções. O modelo de Heston [Steven \(1993\)](#), por exemplo, deixa de considerar a volatilidade constante ou determinística e assume que a volatilidade segue um processo estocástico. Já o modelo de volatilidade local considera a volatilidade como função do preço do ativo subjacente e do tempo. Atualmente, outros modelos também são bem discutidos e estudados como o modelo de volatilidade local estocástica, modelo SABR, dentre outros. A literatura é bem vasta se tratando desses

modelos. Gatheral (2006) Hagan et al. (2002) Gatheral and Jacquier (2014)

Para embasar o presente estudo foram pesquisadas referências de distribuição de máxima entropia a partir de preço de opções. De acordo com o princípio de máxima entropia, apresentado por Jaynes (1957), dada uma série de distribuições, existe apenas uma forma de selecionar uma distribuição consistentemente, a qual é baseada somente em informação dadas, ou seja, sem informações extras. Essa distribuição consiste na de máxima entropia (MED).

Utiliza-se então a distribuição de máxima entropia discutida em Buchen and Kelly (1996) e Neri and Schneider (2012). Em Buchen and Kelly (1996) é proposto um método de maximização para encontrar a distribuição de probabilidade de um ativo a partir de opções de compras. Neste método, foram encontrados um conjunto de multiplicadores de Lagrange no problema de maximização. Além disso, foram apresentados métodos para encontrar esses multiplicadores.

Diferente da proposta de Bunchen e Kelly, em que são resolvidas N equações para a determinação da distribuição de máxima entropia, foi apresentada uma nova técnica que permite reduzir o problema N -dimensional para unidimensional. Esta nova técnica, proposta por Neri and Schneider (2011) particiona os preços das opções em intervalos de *strikes* adjacentes. Além disso, são utilizadas opções digitais como restrições para o problema de otimização. Dessa forma o problema passa a ser entendido como a solução de $F(x) = \lambda$, que é solucionada utilizando o método de Newton–Raphson. Os autores propõem, também a utilização da técnica de *backtracking line search* (Boyd and Vandenberghe, 2004) para encontrar o melhor tamanho do passo.

A partir das densidades de máxima entropia geradas, foram utilizadas técnicas Bayesianas para reduzir a dimensionalidade do problema inverso, em si. Nesse caso, o teorema de Bayes Jari and Somersalo (2005) foi utilizado diretamente no cálculo do Funcional gerado para a minimização. Para a utilização do teorema, todos os parâmetros são vistos como variáveis aleatórias e deve-se encontrar a probabilidade a priori que reflete toda a informação anterior, encontrar a densidade conjunta do que é observado e do que é desconhecido.

Por fim, a calibragem do modelo será feita através de mínimos quadrados sendo esse funcional gerado a partir da Densidade de Máxima Entropia e de técnicas Bayesianas, adicionando o termo da regularização de Tikhonov para a estabilização do problema inverso, técnica essa com literatura bem vasta Engl et al. (1996) Egger and Engl (2005) Leitao and Baumeister (2005) Tarantola (2005).

Capítulo 2

Modelos de Volatilidade

Neste capítulo serão apresentados os modelos de precificação de opções necessários para o entendimento do modelo proposto neste trabalho. Inicialmente, na Seção 2.1, será abordado o modelo de Black-Scholes que foi pioneiro nesse contexto. Além da fórmula final do preço da opção, será mostrado também o conceito de volatilidade implícita, o valor de único de volatilidade usado no modelo de Black-Scholes para recuperar o preço da opção. Por fim, na seção 2.2 será abordado o conceito de volatilidade local, que atualiza o modelo de Black-Scholes, refletindo de maneira mais fiel o comportamento temporal da volatilidade dos ativos no mercado.

2.1 O Modelo de Black-Scholes

O modelo de **Black and Scholes (1973)** é considerado por muitos como o primeiro modelo na precificação de opções. A partir dele, é possível determinar o preço de uma opção de compra em função apenas do preço do ativo e de variáveis conhecidas. Para a construção desse modelo, são feitas algumas suposições:

- A volatilidade é constante;
- A tomada e o empréstimo de dinheiro é feita com taxa livre de risco constante;
- O ativo não realiza o pagamento de dividendos;
- Os preços se movem de acordo com um movimento browniano geométrico representado pela seguinte expressão:

$$dS_t = rS_t dt + \sigma S_t dW_t \quad , \quad (2.1.1)$$

onde r é a taxa livre de risco e σ a volatilidade do ativo.

A partir das suposições supracitadas, considerando argumentos de *hedging* e considerando ainda uma opção de compra (ou venda) com vencimento T e *strike* K , é possível demonstrar que o preço dessa opção satisfaz a seguinte equação diferencial parcial (EDP):

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + rS \frac{\partial V}{\partial S} - rV = 0 \quad (2.1.2)$$

a qual estará definida, no caso de uma opção de compra, a partir das condições de valor terminal:

$$C_{BS}(0, t) = 0 \text{ para todo } t \quad (2.1.3)$$

$$C_{BS}(S, t) \rightarrow S \text{ quando } S \rightarrow \infty \quad (2.1.4)$$

$$C_{BS}(S, T) = \max\{S - K, 0\} \quad (2.1.5)$$

em que $C_{BS}(S, t)$ é o preço da opção de compra quando o preço do ativo objeto é S no tempo t . A solução analítica dessa EDP é universalmente conhecida como fórmula de Black-Scholes, a qual é dada por (Wilmott, 2007):

$$C_{BS}(t, S_t, T, K, \sigma) = S_t \Phi(d_1) - K e^{-r(T-t)} \Phi(d_2) \quad , \quad (2.1.6)$$

onde em $C_{BS}(t, S_t, T, K, \sigma)$ são explicitados também os parâmetros *strike* (K), volatilidade (σ) e o preço do ativo objeto indexado no tempo (S_t).

$$d_1 = \frac{\ln(S_T/K) + (r + \frac{1}{2}\sigma^2)(T-t)}{\sigma\sqrt{T-t}} \quad \text{e} \quad d_2 = d_1 - \sigma\sqrt{T-t}, \quad (2.1.7)$$

Aqui, Φ é a densidade cumulativa da distribuição normal padrão:

$$\Phi(x) = P[Z \leq x | Z \sim N(0, 1)] \quad (2.1.8)$$

Além disso, para calcular o valor da opção de venda, utiliza-se a paridade *Put-Call* (Stoll, 1969). Sendo r a taxa de juros livre de risco, será dada por:

$$C_{BS}(t, S_t, T, K, \sigma, r) + e^{-r(T-t)} K = S_t + P_{BS}(t, S_t, T, K, \sigma) \quad , \quad (2.1.9)$$

onde, P_{BS} é o preço da opção de venda no modelo de Black-Scholes.

Pode-se definir também quantidades que indicam a sensibilidade do preço das opções com os parâmetros do modelo como a maturidade ou a própria volatilidade. Essas quantidades são chamadas de *greeks* ou simplesmente gregas - assim denominadas pela utilização de letras gregas para sua diferenciação.

São, essencialmente, indicadores ou medidas da performance de uma operação de uma opção de compra, sendo calculadas a partir da diferenciação do modelo de precificação de opções.

A grega mais utilizada é a Delta (Δ), dada pela derivada do preço da opção em relação ao preço do ativo objeto S . Seu valor absoluto é, aproximadamente, a probabilidade implícita de uma opção expirar no dinheiro *in the money*. Para o modelo de BS, é dada por:

$$\Delta = \frac{\partial V}{\partial S} = N(d_1) > 0 \quad (2.1.10)$$

A segunda derivada do preço da opção em relação ao preço do ativo objeto - ou simplesmente a derivada da Delta em relação a S - dá origem à grega Gama (Γ). Ela mede, portanto, a taxa de variação do Delta. Sua equação será:

$$\Gamma = \frac{\partial^2 V}{\partial S^2} = \frac{n(d_1)}{S_t \sigma \sqrt{T-t}} > 0 \quad (2.1.11)$$

A grega Lambda (Λ), por sua vez, mede a variação percentual no valor da opção por variação percentual no preço subjacente. É dita uma medida de alavancagem¹, por medir a quantidade ou índice de alavancagem empregada por uma opção. É chamada, ainda, de elasticidade, pelo paralelo ao termo comum em economia para a definição de variações percentuais relativas entre quantidades relacionadas.

$$\Lambda = \frac{\partial V}{\partial S} = n(d_1) S_t \sqrt{T-t} > 0 \quad (2.1.12)$$

A grega Theta (Θ), por sua vez, é dado pela derivada do preço da opção em relação ao tempo. Quanto maior, mais o valor da opção diminuirá, conforme se aproximar a data de vencimento. Para o modelo de BS, será dada por:

$$\Theta = \frac{\partial V}{\partial t} = -\frac{n(d_1) S_t \sigma}{2\sqrt{T-t}} - r K e^{-r(T-t)} N(d_2) < 0 \quad (2.1.13)$$

Quando o interesse² for na variação do preço da opção em relação à taxa de juros r (entre o dia atual e a data de vencimento da opção), utiliza-se a grega Rô (ρ). Nesse caso, considera-se que o custo de oportunidade ao longo do tempo, baseado na taxa de juros, afetará o valor da opção de maneira conjunta ao prêmio de risco.

$$\rho = \frac{\partial V}{\partial r} = K(T-t) e^{-r(T-t)} N(d_2) < 0 \quad (2.1.14)$$

A grega Vega³ (\mathcal{V}) reflete a variação do preço da opção em função da variação da sua volatilidade implícita σ . A Vega é útil em mercados de alta volatilidade, uma vez que mede a sensibilidade a elevadas variações da volatilidade da ação subjacente à opção.

$$\mathcal{V} = \frac{\partial^2 V}{\partial \sigma^2} = n(d_1) S_t \sqrt{T-t} \frac{d_1 d_2}{\sigma} \quad (2.1.15)$$

¹Pode ser definida como a utilização de um instrumento ou de recursos para multiplicar um determinado resultado, a partir de um esforço pequeno.

²Cabe ressaltar que a grega Rô possui maior validade e utilização em cenários de longo prazo, quando há grandes variações da taxa de juros. Nesse caso, a Rô é um bom termômetro, impactando positivamente o preço das *calls* e negativamente o preço das *puts*.

³Apesar de assim chamada, não é uma letra grega e sim uma forma de se chamar a letra V maiúscula. Utiliza-se também o κ

2.1.1 A Volatilidade Implícita

Assumindo as suposições da Seção 2.1, a fórmula de Black-Scholes fornece o preço da opção do ativo objeto. A volatilidade implícita (σ_{IV}) de uma opção é o valor único do parâmetro volatilidade que, ao ser inserido no modelo de Black-Scholes, retorna o preço de mercado da opção.

A abordagem mais comum para se calcular a volatilidade implícita em uma opção de um ativo é aplicar a metodologia reversa ao modelo de Black-Scholes. Dessa forma são utilizados métodos numéricos para encontrar a solução do problema a seguir:

$$\sigma_{IV} = \underset{\sigma}{\operatorname{argmin}}(C_{MKT} - C_{BS}(t, S_t, T, K, \sigma, r))^2 \quad , \quad (2.1.16)$$

onde C_{MKT} é o preço observado no mercado da opção e C_{BS} é o preço da opção encontrado em (2.1.6)

A volatilidade pode ser usada como um alerta a investidores para a possibilidade de mudanças irregulares no preço do ativo objeto e também para se testar a eficiência do mercado de opções, ou seja, se os preços das opções incorporam instantaneamente toda a informação disponível, a volatilidade implícita deve ser o melhor previsor da volatilidade futura segundo [Koopman et al. \(2005\)](#).

Diversos trabalhos tiveram como objetivo modelar diretamente a superfície de volatilidade implícita utilizando os mais variados modelos, como o modelo de volatilidade estocástica [Steven \(1993\)](#), o modelo SVI livres de arbitragem [Gatheral and Jacquier \(2013\)](#) e o de volatilidade local [Dupire \(1994\)](#), já estudados capazes de capturar todos os padrões encontrados no mercado real.

2.2 O Modelo de Volatilidade Local

Apesar do uso generalizado da fórmula de Black-Scholes na precificação de opções de mercado, observa-se que seu uso é impreciso por considerar a volatilidade do ativo como constante. Conforme discutido em [Dupire \(1994\)](#), a volatilidade apresenta dependência em relação ao tempo e ao preço do ativo objeto.

Para isso, um importante resultado apresentado em [Gyöngy \(1986\)](#) foi utilizado. Considere os processos estocásticos $\alpha(t)$ e $\beta(t)$ e um processo genérico X_t dado por:

$$dX_t = \alpha(t)dt + \beta(t)dW_t \quad , \quad (2.2.1)$$

Então, é mostrado em [Gyöngy \(1986\)](#) que existe um processo Markoviano Y_t com a mesmas distribuições marginais de X_t e esse processo é dado por:

$$dY_t = a(t, Y_t)dt + b(t, Y_t)dW_t \quad , \quad (2.2.2)$$

Onde:

$$a^2(t, x) = \mathbb{E}[\alpha^2(t, x) | X_t = x] \quad (2.2.3)$$

e

$$b(t, x) = \mathbb{E}[\beta(t, x) | X_t = x] \quad (2.2.4)$$

A partir dessa observação, surgiu o modelo de volatilidade local que considera a volatilidade como uma função do tempo e do preço do ativo subjacente. Com isso, o autor em Dupire (1994) mostra que existe um único parâmetro de difusão dado um conjunto de preços de opções europeias. Assim, seja o espaço de probabilidade $(\Omega, \mathcal{U}, \mathbb{Q})$ com \mathbb{Q} sendo a medida de probabilidade neutra ao risco. Considerando que o processo de preços pode ser calculada a partir da seguinte expressão:

$$dS_t = rS_t dt + \sigma(t, S_t)S_t dW_t \quad , \quad (2.2.5)$$

com dW_t sendo um movimento Browniano na medida \mathbb{Q} , r a taxa de juros (aqui assumida constante) e σ a volatilidade local.

A equação 2.2.5 implica que o preço de uma opção de compra será dado por um valor esperado que satisfaz a EDP a seguir, conhecida como Equação de Dupire (a dedução dessa fórmula encontra-se no apêndice A):

$$\frac{\partial C}{\partial T} = \frac{1}{2}\sigma^2 K^2 \frac{\partial^2 C}{\partial K^2} - rK \frac{\partial C}{\partial K}. \quad (2.2.6)$$

a qual estará definida, a partir da condição de contorno:

$$C(T = t, K) = (S_0 - K)^+ \quad (2.2.7)$$

Pela condição de não-arbitragem, tem-se que:

$$\frac{\partial^2 C}{\partial K^2} > 0 \quad (2.2.8)$$

Assim, tem-se que a volatilidade local $\sigma(T, K)$ em T pode ser escrita como:

$$\sigma(T, K) = \sqrt{2 \frac{\frac{\partial C}{\partial T} + rK \frac{\partial C}{\partial K}}{K^2 \frac{\partial^2 C}{\partial K^2}}}. \quad (2.2.9)$$

O lado direito da equação 2.2.9 pode ser encontrado utilizando preços de opções. Então, dado um conjunto de preços para *strikes* dessas opções, a volatilidade local é unicamente definida por (2.2.9).

Capítulo 3

Distribuição de Máxima Entropia

Um importante problema na precificação de opções consiste em encontrar a densidade de probabilidade dos preços para o ativo subjacente. A partir de um conjunto finito de preços em determinados *strikes*, é feita uma interpolação, a fim de calcular de forma contínua o preço em todos os valores necessários de *strikes*.

Nos modelos utilizados na literatura para precificação de opções, são assumidas diferentes densidades de probabilidade. O modelo de Black-Scholes, por exemplo, assume a distribuição log-normal (Black and Scholes, 1973). Neste trabalho, será utilizada a distribuição de máxima entropia (DME), amplamente abordada em Neri and Schneider (2012). Essa densidade de probabilidade possui o menor viés possível dado um conjunto de informações fornecidas segundo Buchen and Kelly (1996).

3.1 DME Usando Opções de Compra

No trabalho de Buchen and Kelly (1996) a construção da DME é feita a partir de preços de opções europeias. Seja S_t o preço do ativo objeto no tempo t , C_i o preço dessa opção no *strike* K_i e \tilde{C}_i o preço não descontado dessas opções em uma maturidade T com i entre 0 e n , podendo ser definido como:

$$\tilde{C}_i := C_{K_i, T} / DF(0, T), \quad (3.1.1)$$

onde $DF(0, T)$ é o fator de desconto na maturidade.

Define-se, então, a precificação da opção sob a medida neutra ao risco como:

$$\tilde{\mathbb{E}}[(S_T - K_i)^+] = \int_{K_i}^{\infty} (x - K_i)p(x)dx = \tilde{C}_i, \quad (3.1.2)$$

com $\tilde{\mathbb{E}}$ o valor esperado na medida neutra ao risco e $g(x)$ sendo a densidade do ativo objeto no tempo T . Logo, a entropia de $g(x)$ é dada por:

$$S(p) = - \int_0^{\infty} p(x) \log p(x) dx \quad , \quad (3.1.3)$$

de acordo com o princípio da máxima entropia, Buchnen-Kelly mostrou que o problema consistia em maximizar o funcional abaixo:

$$H(p) = - \int_0^{\infty} p(x) \log p(x) dx + (1 + \lambda_0) \int_0^{\infty} p(x) dx + \sum_{i=1}^m \lambda_i \int_0^{\infty} p(x) (x - K_i) dx \quad (3.1.4)$$

sendo λ_i os parâmetros de Lagrange com esse funcional maximizado quando a derivada de Fréchet com respeito a $p(x)$ (Cartan and Kouneihir, 2007) é zero, ou seja:

$$\delta H = - \int_0^{\infty} [\log p(x) dx + \lambda_0 + \sum_{i=1}^m \lambda_i c_i(x)] \delta p(x) dx = 0 \quad (3.1.5)$$

Dessa forma, para calcular a DEM, deve-se encontrar os n parâmetros de Lagrange, que consistirá em um problema multi-dimensional onde $g(x)$ tem a forma:

$$g(x) = \frac{1}{\mu} e^{p(x)} \quad , \quad (3.1.6)$$

em que:

$$\mu = \int_0^{\infty} \exp\left(\sum_{i=1}^m \lambda_i c_i(x)\right) dx \quad (3.1.7)$$

e

$$p(x) = \frac{1}{\mu} \exp\left(\sum_{i=1}^m \lambda_i c_i(x)\right) \quad (3.1.8)$$

3.2 DME Usando Opções de Compra e Opções Digitais

Continuando o trabalho de Bunchenn-Kelly, Neri and Schneider (2011) apresentou uma nova forma de se obter a DME utilizando opções digitais como restrições ao problema de otimização. Desse modo, a solução se torna mais robusta, fácil de implementar e estável. Foi utilizado o método de Newton-Rapson para resolver N problemas unidimensionais, diferentemente de Bunchen-Kelly que resolvia um problema N -dimensional permitindo paralelismo e evitando problemas mal condicionados (Neri and Schneider, 2012).

Dessa forma, Neri-Scheneider definem como \tilde{C}_i e \tilde{D}_i os preço das opções de compra e das opções digitais, respectivamente, sem desconto. As opções digitais são aquelas com pagamento “binário”, ou seja, ou em uma quantidade fixada ou sem pagamento algum. Necessitam, portanto, de uma previsão precisa do preço de exercício. Em Neri and Schneider (2012) é mostrado

que existe uma única densidade g que maximiza a entropia e que combina, tanto com a opção como com a opção digital, as quais são dadas, respectivamente, por:

$$\int_{K_i}^{\infty} (x - K_i)g(x)dx = \tilde{C}_i \quad (3.2.1)$$

e

$$\int_{K_i}^{\infty} g(x)dx = \tilde{D}_i \quad (3.2.2)$$

Além disso, é mostrado que g maximiza $H(g)$ sob as restrições (3.2.1) e (3.2.2) se e somente se maximiza $H_i(g)$ localmente em cada intervalo $[K_i, K_{i+1}[$ sob as restrições:

$$\int_{K_i}^{K_{i+1}} xg(x)dx = (\tilde{C}_{i+1} + K_{i+1}\tilde{D}_{i+1}) - (\tilde{C}_i + K_i\tilde{D}_i) \quad (3.2.3)$$

e

$$\int_{K_i}^{K_{i+1}} g(x)dx = \tilde{D}_{i+1} - \tilde{D}_i \quad (3.2.4)$$

Com isso, definindo \bar{K}_i como:

$$\bar{K}_i := \frac{(\tilde{C}_i + K_i\tilde{D}_i) - (\tilde{C}_{i+1} + K_{i+1}\tilde{D}_{i+1})}{\tilde{D}_i - \tilde{D}_{i+1}} \quad , \quad (3.2.5)$$

tem-se que em cada intervalo $[K_i, K_{i+1}[$, a DME admite a forma exponencial:

$$g(x) = \alpha_i e^{\beta_i x} \quad (3.2.6)$$

com parâmetros dados por:

$$\alpha_i = p_i e^{c_i(\beta_i)} \quad \text{e} \quad c'(\beta_i) = \bar{K}_i \quad , \quad (3.2.7)$$

onde:

$$c_i(\beta) = \begin{cases} c \log\left(\frac{e^{\beta K_{i+1}} - e^{\beta K_i}}{\beta}\right) & \text{for } i < n \text{ e } \beta \neq 0, \\ \log(K_{i+1} - K_i) & \text{for } i < n \text{ e } \beta = 0, \\ \log\left(-\frac{e^{\beta K_i}}{\beta}\right) & \text{for } i = n \text{ e } \beta < 0, \end{cases} \quad (3.2.8)$$

$$c'_i(\beta) = \begin{cases} \frac{K_{i+1}e^{\beta K_{i+1}} - K_i e^{\beta K_i}}{e^{\beta K_{i+1}} - e^{\beta K_i}} - \frac{1}{\beta} & \text{for } i < n \text{ e } \beta \neq 0, \\ \frac{K_{i+1} + K_i}{2} & \text{for } i < n \text{ e } \beta = 0, \\ K_i - \frac{1}{\beta} & \text{for } i = n \text{ e } \beta < 0, \end{cases} \quad (3.2.9)$$

$$c''_i(\beta) = \begin{cases} -(K_{i+1} - K_i)^2 \frac{e^{\beta(K_{i+1} + K_i)}}{(e^{\beta K_{i+1}} - e^{\beta K_i})^2} + \frac{1}{\beta^2} & \text{for } i < n \text{ e } \beta \neq 0, \\ \frac{(K_{i+1} - K_i)^2}{12} & \text{for } i < n \text{ e } \beta = 0, \\ \frac{1}{\beta^2} & \text{for } i = n \text{ e } \beta < 0, \end{cases} \quad (3.2.10)$$

No algoritmo 1 encontra-se o pseudocódigo para a resolução passo a passo para encontrar os coeficientes α e β da equação (3.2.6) dado os valores de *strike* dentro de um intervalo:

Algoritmo 1: ESTIMAÇÃO DOS PARÂMETROS DA DME

Entrada: $x_0, K, C, err, c, c', c''$

Saída: Parâmetros da DME

início

$x_n = x_0$

para cada $i, k \in K$ **faça**

enquanto $|c'(x_n)| > err$ **faça**

$x_n \leftarrow x_n - c'(x_n)/c''(x_n)$

fim

$\beta_i \leftarrow x_n$

$\alpha_i \leftarrow p_i \exp(-c(x_n))$

fim

fim

retorna α, β

Com os valores dos coeficientes encontrados utilizando o método de Newton-Raphson descrito no algoritmo no algoritmo 1, pode-se determinar a DME para valores dentro do intervalo (no caso do Algoritmo indicado por K).

Capítulo 4

Calibragem da Volatilidade Local

Neste capítulo será abordado, de forma detalhada, o problema de calibragem da volatilidade local utilizando preços de opções europeias observados no mercado, o qual encontra-se presente na classe dos problemas inversos abordado na Seção 4.1. Nesta seção, o problema de estimação de parâmetros será abordado de forma detalhada, desde a definição do problema inverso mal-posto no sentido de Hadamard e passando pela explicação do método de regularização de Tikhonov (Subseção 4.1.2). Por fim, será detalhada uma abordagem alternativa para a solução desses problemas, utilizando técnicas Bayesianas (Subseção 4.1) e conceitos de Distribuição de Máxima Entropia (MED) discutidas no capítulo 3.

4.1 O Problema Inverso

Considere o problema de se encontrar, dado um $y \in \mathbb{R}^n$, um $x \in \mathbb{R}^m$, satisfazendo a seguinte equação:

$$y = f(x) \quad , \quad (4.1.1)$$

Esse tipo de problema é denominado problema inverso, em que y denota os dados observados e x é a incógnita ou os parâmetros a serem estimados/identificados. No presente caso, y representa os preços de opções observados e x a superfície de volatilidade local.

Quando o problema é o contrário, isto é, quando o objetivo é encontrar y a partir de x , esse será chamado de problema direto. Em geral, mesmo que o problema direto esteja bem definido e que a função f seja suave, o problema inverso pode não estar definido. Caso y não esteja na imagem de f , y pode determinar mais de um x , isto é, pode não haver unicidade de solução, ou ainda, o problema inverso pode não ser contínuo com relação a y .

Outra dificuldade notória é o caso em que $n > m$, na qual o sistema passa a ser indeterminado devido ao maior número de incógnitas quando comparado ao de equações.

Por fim, quando considera-se a existência de um ruído associado aos dados, isto é, quando se reescreve o Modelo 4.1.1 como:

$$y = f(x) + e \quad , \quad (4.1.2)$$

em que $e \in \mathbb{R}^m$ representa o erro ou ruído aleatório observado. Nesse caso, y não pertencerá à imagem de f e, dessa forma, inverter não é suficiente para a resolução do problema.

Uma abordagem tradicional para encontrar uma solução para o problema 4.1.2 é minimizar a norma L^2 do resíduo, este dado pela diferença entre o valor observado e o esperado do modelo, $y - f(x)$. A minimização é então descrita por meio de um problema de mínimos quadrados em que a função objetivo é a função de mérito:

$$\min_x \|y - f(x)\|_{L^2}^2 \quad (4.1.3)$$

Outra possível abordagem para resolver esse problema é determinar as componentes do ruído que não mudam durante diferentes medições, porém o ruído pode mudar nos instantes das medições, o que inviabiliza essa abordagem (Jari and Somersalo, 2005). Dessa forma, na próxima seção será sugerida uma outra abordagem para contornar essas dificuldades.

4.1.1 O Problema Inverso e o Teorema de Bayes

Como citado anteriormente, a solução de problemas inversos contém algumas dificuldades ao se considerar a abordagem tradicional, por isso será aqui proposta uma nova abordagem utilizando técnicas Bayesianas para encontrar o modelo inverso do problema proposto.

Para tal, assume-se que x, y e e na equação 4.1.2 são variáveis aleatórias (contínuas) e também que, antes de se ter o dado observado y , tem-se a informação sobre a incógnita x , ou seja, tem-se a informação da incógnita a *Priori* da informação observada (do termo obtido na medição).

Além disso, denotando-se por $\Pi(x)$ e $\Pi(x, y)$, respectivamente, a densidade marginal de x e a densidade conjunta de x e y , e considerando os resultados teóricos do cálculo de probabilidades, tem-se que a densidade marginal de x (a priori, pois se sabe a informação) pode ser obtida pela integração da $\Pi(x, y)$ em relação a y , isto é:

$$\Pi_{pr}(x) = \int_{\mathbb{R}^m} \Pi(x, y) dy \quad (4.1.4)$$

Outro resultado importante são as distribuições condicionais, também dadas por um simples resultado do cálculo de probabilidades:

$$\Pi(y|x) = \frac{\Pi(x, y)}{\Pi(x)} \quad \text{e} \quad \Pi(x|y) = \frac{\Pi(x, y)}{\Pi(y)} \quad , \quad (4.1.5)$$

sendo válidas para $\Pi_{pr}(x) \neq 0$ e $\Pi_{pr}(y) \neq 0$, respectivamente.

O objetivo final é a obtenção da distribuição condicional de x dado o conhecimento de y , ou seja, $\Pi(x|y)$. Para tanto, sob a ótica Bayesiana utiliza-se o teorema de Bayes para a obtenção

dessas quantidades. Nele, a informação a *Priori* $\Pi(x)$ é atualizada pela verossimilhança do modelo $\Pi(x|y_{obs})$, de modo a conjuntamente serem utilizadas para a obtenção da condicional $\Pi(x|y)$, da seguinte forma:

$$\Pi(x|y_{obs}) = \frac{\Pi(x, y_{obs})}{\Pi(y_{obs})}, \text{ se } \Pi(y_{obs}) = \int_{\mathbb{R}^n} \Pi(x, y_{obs}) dx \neq 0. \quad (4.1.6)$$

Em que assume-se que a informação observada é denotada por $y = y_{obs}$. Essa densidade é chamada de *Posteriori*, ou seja, é a densidade do que se sabe da incógnita x após a observação de y .

Por fim, pode-se escrever o resultado do teorema de Bayes para problemas inversos (Jari and Somersalo, 2005) da seguinte forma:

$$\Pi_{post}(x) = \Pi(x|y_{obs}) = \frac{\Pi_{pr}(x)\Pi(y_{obs}|x)}{\Pi(y_{obs})}, \quad (4.1.7)$$

em que a densidade marginal de y , $\Pi(y)$, a exemplo da de x , pode ser calculada novamente via integração da distribuição conjunta $\Pi(x, y)$, isto é:

$$\Pi(y) = \int_{\mathbb{R}^n} \Pi(x, y) dx = \int_{\mathbb{R}^n} \Pi(y|x)\Pi_{pr}(x) dx \quad (4.1.8)$$

Dessa forma, dado um processo estocástico $S_{n_0}^N$ e um passo inicial s_0 , denota-se, para todo $n = 1, 2, \dots, N$, a densidade de S_n por $\Pi(s_n) = \Pi(s_n|S_0)$. Logo, pela fórmula (4.1.8), pode-se escrever:

$$\Pi(s_{n+1}) = \int_{\mathbb{R}^n} \Pi(s_{n+1}|s_n)\Pi(s_n) dx \quad (4.1.9)$$

A equação 4.1.9 é um importante resultado para a construção do funcional a ser minimizado na calibragem da volatilidade local.

4.1.2 Regularização de Tikhonov

Para obter a volatilidade local, é preciso resolver um problema inverso mal-posto no sentido de Haddamard, ou seja, que não atende a pelo menos um dos requisitos:

1. Existência de uma solução;
2. Unicidade dessa solução;
3. Dependência contínua dos dados.

Dessa forma, para a estabilização desse tipo de problema, será utilizada uma forma de regularização¹, mais especificamente a regularização de Tikhonov (Tikhonov et al., 2013).

Sendo $\alpha > 0$ e o problema inverso mostrado na Seção 4.1, a solução regularizada de x_α é o minimizador do funcional:

$$F_\alpha(x) = \|f(x) - y\|^2 + \alpha\|x - x_0\|^2 \quad , \quad (4.1.10)$$

onde x_0 é um valor dado previamente e α é denominado parâmetro de regularização.

Observa-se que a adição desse parâmetro é similar ao papel de um multiplicador de Lagrange, ou seja, a solução de uma minimização com a restrição de $\|x - x_0\|^2 = R$ para algum $R > 0$ (Jari and Somersalo, 2005).

Além disso, para um operador linear A é possível mostrar que a solução do funcional de Tikhonov x_α existe e é única, sendo dada por:

$$x_\alpha = (AA + \delta I)^{-1}Ay \quad (4.1.11)$$

O detalhamento desse resultado será omitido por não ser o objetivo central do texto, mas sua literatura é vasta (Tikhonov et al., 2013; Jari and Somersalo, 2005).

4.1.3 O Critério da curva em L

A escolha do parâmetro de regularização em problemas inversos é de fundamental importância e tem grande impacto no resultado final. Para o método proposto no texto, foi utilizado o critério da curva em L, descrito em Hansen and O'leary (1993). Além desse método, outras técnicas são bastante utilizadas para encontrar esse parâmetro como por exemplo o princípio da discrepância de Morozov (Pereverzyev and Schock, 2000) e o método de validação cruzada generalizada (Caterina Fenu, 2017)

A curva em L é um gráfico, em escala logarítmica do resíduo da solução contra a norma da solução parametrizada pelo parâmetro de regularização, isto é, a curva:

$$(\|f(x_\lambda) - y\|^2, \|x_\lambda - x_0\|^2) \quad (4.1.12)$$

Dessa forma, a ideia do critério em L é escolher o valor do parâmetro como o valor que corresponde ao ponto do canto (*corner point*) da curva “L”, ilustrado na Figura 4.1. Assim, é possível perceber que, um parâmetro de regularização muito baixo acarretaria em resíduos elevados e um parâmetro elevado, não ocorreria um bom ajuste.

¹Técnicas de regularização são úteis para trazer maior estabilidade à modelos estatísticos, sobretudo na resolução de problemas mal-postos no sentido de Hadamard, ou ainda quando modelos estatísticos enfrentam situações não triviais como a esparsidade e a alta esparsidade dos dados, além do sobreajuste em problemas de regressão.

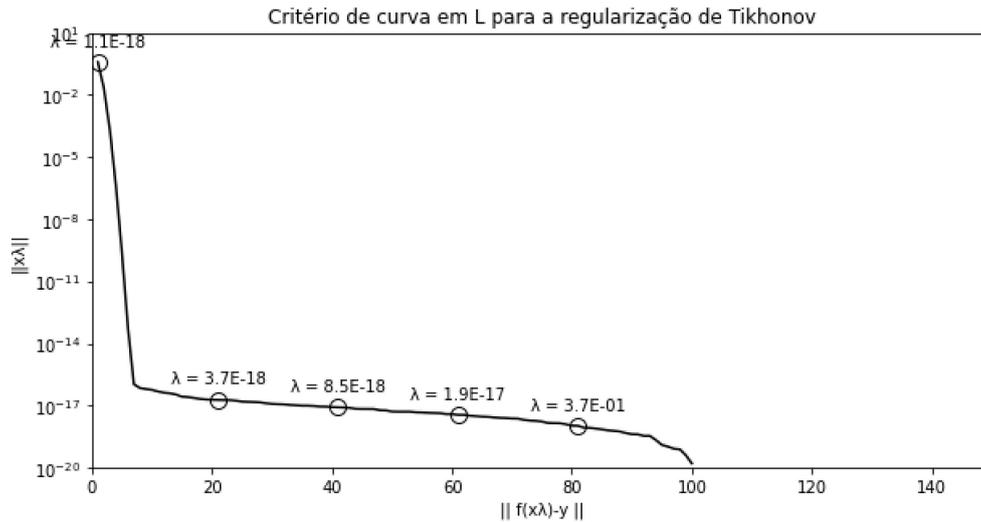


Figura 4.1: Curva em L genérica para problema inverso utilizando a regularização de Tikhonov.

4.2 O Processo de Calibragem

Conforme mostrado na seção 2.2, a volatilidade local é função do tempo t e do ativo objeto. Assim, será desenvolvida a solução de precificação de opções de compra a partir da MED, discutida na Seção 3 a partir de uma abordagem Bayesiana, mostrada em 4.1.1.

Seja, então, o processo de preço desse ativo dado por:

$$dS_t = rS_t dt + \sigma(t, S_t)S_t dW_t \quad , \quad (4.2.1)$$

mantendo as definições da Seção 2.2. Reescrevendo a equação anterior na sua forma integral, tem-se que:

$$S_t = S_s + r \int_s^t S_u du + \int_s^t \sigma(S_u, u)S_u d\tilde{W}_u \quad (4.2.2)$$

Assumindo t suficientemente próximo a s e assumindo que sigma é localmente constante, tem-se:

$$\int_s^t \sigma(S_u, u)S_u d\tilde{W}_u \approx \sigma(S_s, s) \int_s^t S_u d\tilde{W}_u \quad (4.2.3)$$

Dessa forma pode-se escrever a dinâmica de preço como:

$$S_t \approx S_s \exp\left[\left(r - \frac{1}{2}\sigma^2\right)(t-s) + \sigma(\tilde{W}_t - \tilde{W}_s)\right] \quad (4.2.4)$$

Então, dada uma partição $[0, T]$ em N intervalos de tamanho $\Delta_t = T/N$ e definindo $t_n = n\Delta_t$, tem-se:

$$S_{n+1} = S_n \cdot \exp \left[\left(r - \frac{1}{2} \sigma^2(t_n, S_n) \right) \Delta t + \sigma(t_n, S_n) \left(\tilde{W}_{n+1} - \tilde{W}_n \right) \right]. \quad (4.2.5)$$

Assim, ajustando a volatilidade, S_n possui distribuição *log-normal*. Então, a densidade de probabilidade de S_{n+1} pode ser escrita como:

$$\Pi(s_{n+1}|s_n) := \frac{1}{\sqrt{2\pi\Delta t}} \cdot \exp \left[-\frac{1}{2} \frac{\left(\ln \left(\frac{s_{n+1}}{s_n} \right) - \left(r - \frac{1}{2} \sigma^2(t_n, s_n) \right) \Delta t \right)^2}{\sigma^2(t_n, s_n) \Delta t} \right] \quad (4.2.6)$$

Agora, seja $C(t_{n+1}, K)$ o preço de uma opção europeia de compra com o tempo para o vencimento $t_{n+1} = (n+1)\Delta t$ e *strike* K . Então, na medida neutra ao risco, pode-se escrever:

$$C(t_{n+1}, K) = e^{-rt_{n+1}} \mathbb{E}^{\mathbb{Q}}[\max(S_{n+1} - K, 0)] = e^{-rt_{n+1}} \int_0^{\infty} \max(S_{n+1} - K, 0) \Pi(s_{n+1}) ds_{n+1} \quad (4.2.7)$$

Assim, utilizando a equação encontrada em (4.1.9):

$$\begin{aligned} C(t_{n+1}, K) &= e^{-rt_{n+1}} \int_0^{\infty} \max(0, s_{n+1} - K) \int_0^{\infty} \Pi(s_{n+1}|s_n) \Pi(s_n) ds_n ds_{n+1} \\ &= e^{-rt_n} \int_0^{\infty} \Pi(s_n) \int_0^{\infty} e^{-r\Delta t} \max(0, s_{n+1} - K) \Pi(s_{n+1}|s_n) ds_{n+1} ds_n \\ &= e^{-rt_n} \int_0^{\infty} \Pi(s_n) C_{BS}(s_n, K, r, \Delta t, \sigma(t_n, s_n)) ds_n \quad , \end{aligned} \quad (4.2.8)$$

onde C_{BS} é o preço da opção europeia pelo modelo de Black-Scholes.

O intervalo de análise será restringido, em um intervalo $[0, \bar{K}]$, com \bar{K} suficientemente grande pois a MED tem um decaimento exponencial. Então para um tempo t_{n+1} e um K no intervalo $[0, \bar{K}]$, o preço da opção será definido como:

$$C(t_{n+1}, K) = e^{-rt_n} \int_0^{\bar{K}} \Pi(s_n) C_{BS}(s_n, K, r, \Delta t, \sigma(t_n, s_n)) ds_n \quad (4.2.9)$$

Dessa forma, a equação (4.2.9) fornece o preço da opção de compra a partir de métodos Bayesianos, o qual é encontrado interpolando a DME.

O próximo passo é calibrar o modelo, isto é, resolver um problema de mínimos quadrados em que a função objetivo é um determinado funcional, conforme discutido na Seção 4.1. No presente caso, dado o preço da opção e o problema inverso envolvendo a volatilidade local, tem-se que o funcional é dado por:

$$F(\sigma) = \int_0^{\bar{K}} |C_{mkt}(t_{n+1}, K) - C(t_{n+1}, K)|^2 dK \quad , \quad (4.2.10)$$

onde C_{mkt} é o preço de mercado da opção de compra, C é o preço encontrado em (4.2.9).

Todavia, minimizar apenas o funcional $F(\sigma)$, ou seja, encontrar σ no espaço \mathcal{H}_σ é um problema mal-posto no sentido de Hadamard. É necessário, portanto, adicionar o termo $F_R(\sigma) = \alpha\|\sigma - \sigma_0\|^2$, denominado funcional de Tikhonov, em que α é o parâmetro de regularização de Tikhonov. Assim, o problema de minimização da calibragem será dado por:

$$\min_{\sigma \in \mathcal{H}_\sigma} F(\sigma) + F_R(\sigma) \quad (4.2.11)$$

Para a implementação do algoritmo da calibragem, foi utilizada a função `least_squares()` da biblioteca `SciPy` da linguagem `Python`.

Capítulo 5

Resultados Numéricos

Neste capítulo serão apresentados os resultados numéricos e uma descrição detalhada dos métodos apresentados nesse trabalho, desde a implementação dos métodos descritos por Neri-Schneider (Neri and Schneider, 2011) até a solução do problema de calibragem da volatilidade local. Na realização dos testes, foram utilizados dados sintéticos com o cuidado para evitar o crime inverso (Colton, 2003). Por fim, foram utilizados dados reais de opções de S&P para encontrar a superfície de volatilidade local calibrada.

5.1 Dados Simulados

Esta etapa de validação do modelo apresentado consiste na geração de dados de uma superfície de volatilidade arbitrariamente definida e na posterior calibragem da volatilidade local. Por se tratar da geração de dados sintéticos a partir de modelos bem definidos, é preciso cautela para que esse processo seja válida. Um dos principais problemas que podem acontecer nessa etapa é o chamado de “crime inverso”.

Esse termo, introduzido em Colton (2003), faz referência ao fato dos métodos numéricos englobarem características que resolvem problemas inversos “menos mal-postos” do que eles realmente são - ou seja, não se assemelham a um problema inverso real.

Na prática, os crimes inversos aparecem em duas situações (Baumeister, 2010):

1. Quando os dados simulados são produzidos pelo mesmo modelo que é usado para inverter os dados;
2. Ou quando a discretização na simulação numérica é a mesma que a utilizada na inversão.

No problema apresentado nesse texto, para contornar o primeiro caso de crime inverso, os dados gerados de preços de opções, no problema direto, são obtidos através da solução numérica da Equação de Dupire via o método de Crank-Nicolson (Albani et al., 2008) que será explicitado logo adiante.

Para contornar o segundo ponto, foi utilizada uma malha mais fina na geração da superfície de volatilidade em comparação à malha utilizada na resolução do problema inverso - Dessa forma claramente evitando eventuais crimes inversos.

Além disso, foi inserido um ruído branco gaussiano com média $\mu = 0$ e desvio padrão $\sigma_r = 10^{-5}$ à curva de preços com a mesma malha utilizada para gerar a curva de volatilidade local. Esse ruído está demonstrado na figura 5.1

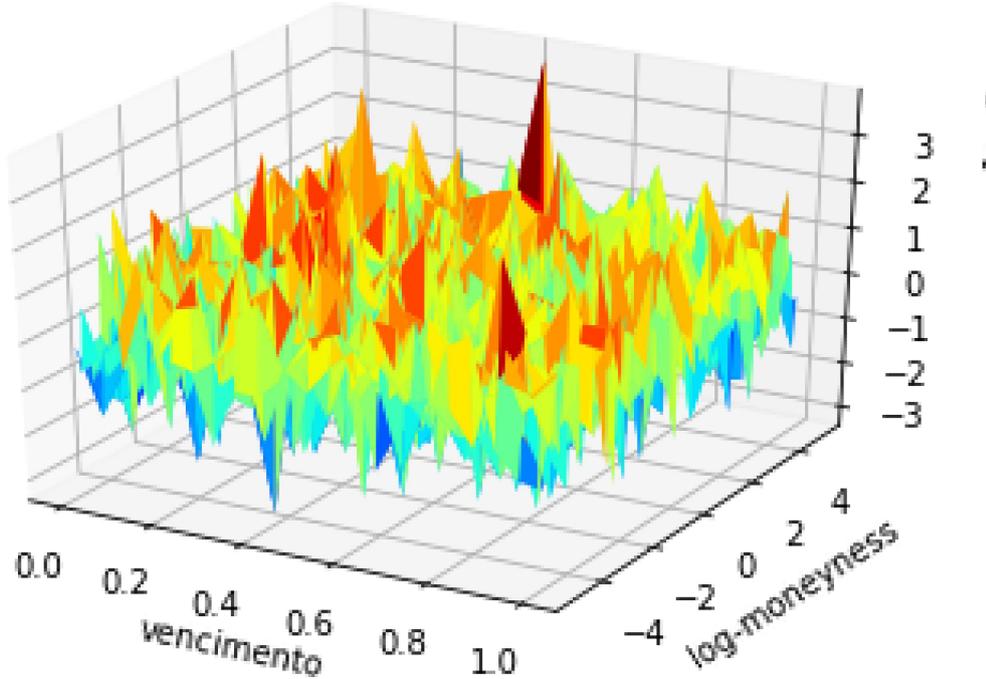


Figura 5.1: Ruído branco gaussiano ($\mu = 0, \sigma_r = 10^{-5}$) adicionado aos dados simulados.

A superfície de volatilidade local, utilizada para essa validação, foi construída a partir da função dada a seguir, introduzida por [Albani and Zubelli \(2014\)](#):

$$\sigma(t, y) = \begin{cases} \frac{2}{5} - \frac{4}{25}e^{-t/2}\cos\left(\frac{5\pi y}{4}\right) & \text{para } -2/5 \leq y \leq 2/5 \\ \frac{2}{5} & \text{caso contrário} \end{cases} \quad (5.1.1)$$

Para a obtenção dos preços de opções foi resolvida a equação de Dupire já discutida em [Dupire \(1994\)](#) a partir método de Crank-Nicolson [Nocedal and Wright \(2006\)](#).

Na geração da superfície, foram considerados os seguintes parâmetros:

- Passo do tempo: $\Delta t = 0.001$;
- Passo do *log-moneyness*¹: $\Delta y = 0.05$;
- Taxa de juros: $r = 0.02$;
- Domínio: $t \times y$: $[0, 1] \times [-4, 4]$.

¹Aqui definido como o logaritmo da razão entre o preço do ativo objeto e o *strike* da opção: $\log \frac{S}{K}$

No caso do problema inverso, a malha foi gerada a partir dos seguintes parâmetros:

- Passo do tempo: $\Delta t = 0.002$;
- Passo do *log-moneyness*: $\Delta y = 0.1$.

A Figura 5.2 mostra a superfície de volatilidade local utilizada na simulação via equação (5.1.1). Essa função foi escolhida devido a característica do *smile* apresentada na superfície de volatilidade e por apresentar uma componente de decaimento temporal em relação ao vencimento da opção, o que permite representar de forma mais fiel o que é encontrado nos dados reais de mercado.

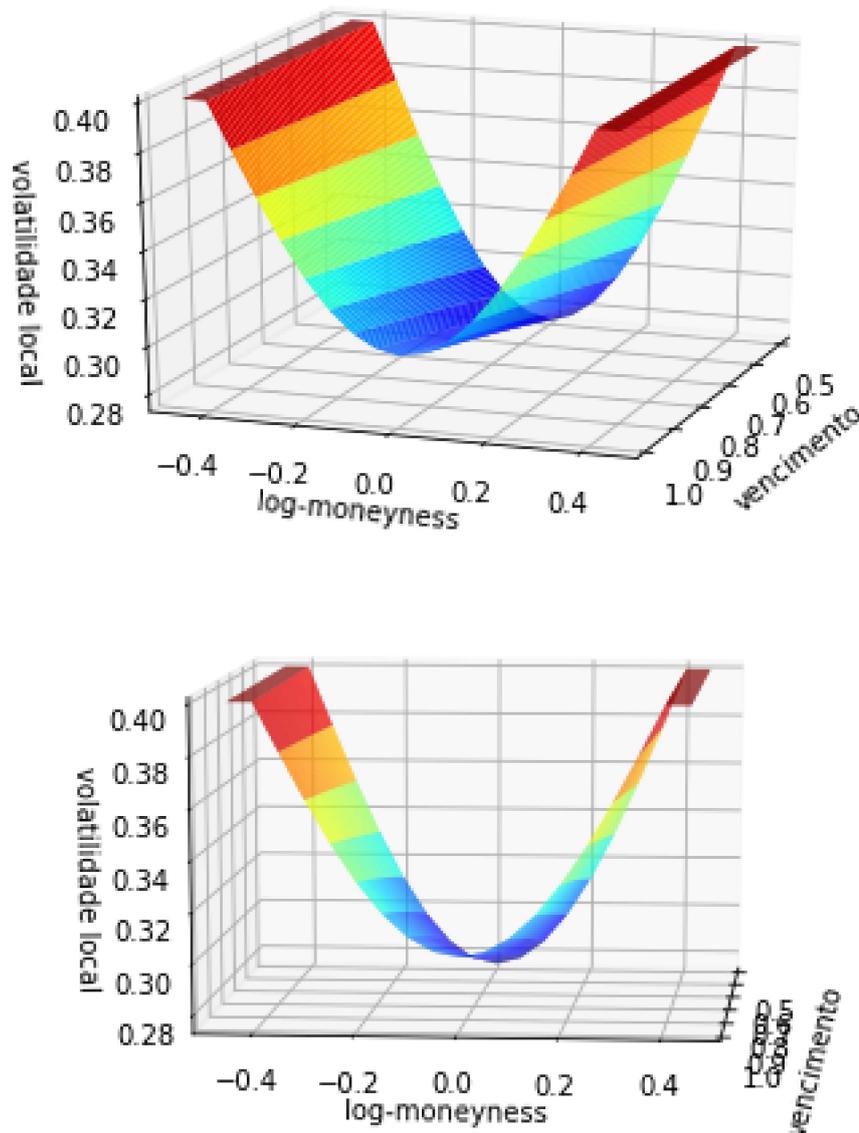


Figura 5.2: Superfície de volatilidade gerada para a simulação dos dados observada em dois ângulos distintos.

Finalizando a etapa de simulação, na Figura 5.3 encontra-se a curva de preços gerada a partir do método de Crank-Nicolson.

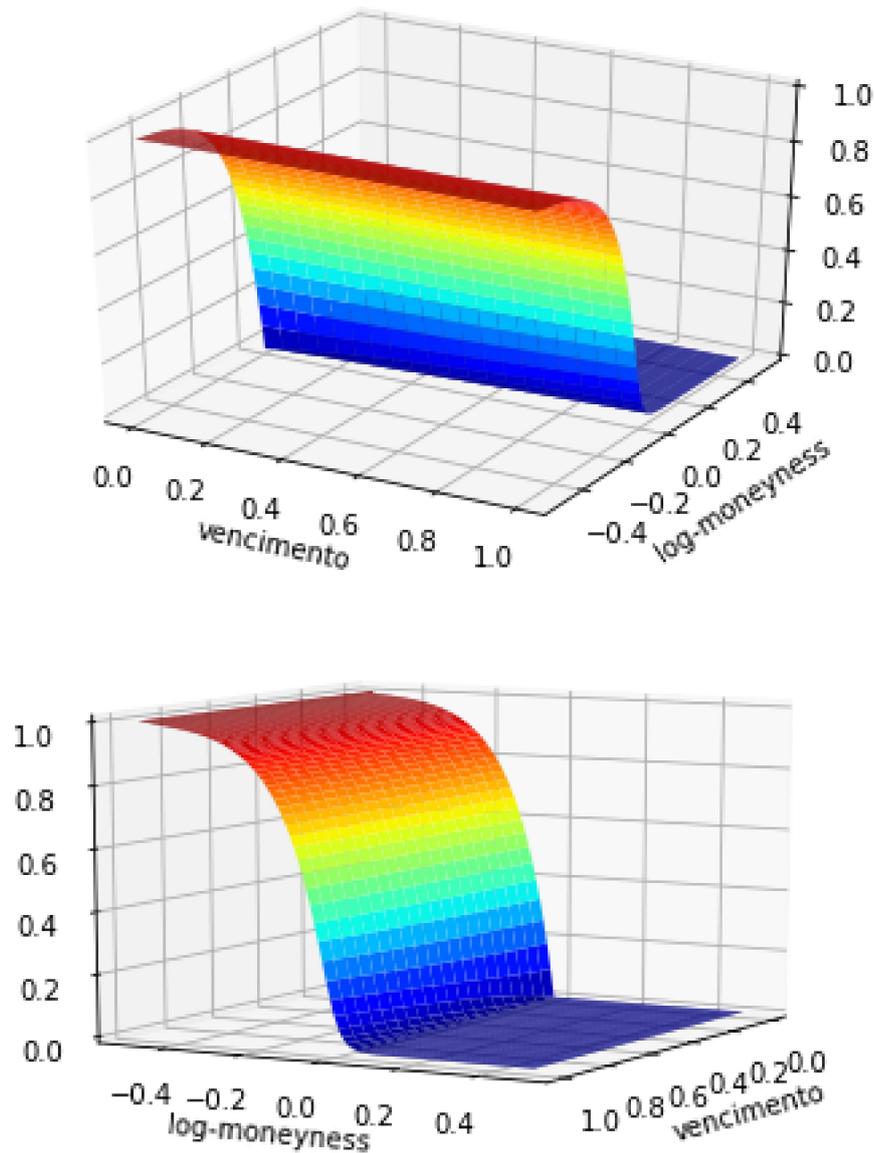


Figura 5.3: Superfície de preços gerada a partir do método de Crank-Nicolson observada em dois ângulos distintos.

5.1.1 Resultados

Uma vez simulados os dados sintéticos, foi realizada a implementação do método proposto nesta dissertação para a estimação da curva de volatilidade local.

Para isso, a primeira etapa foi, a partir da superfície de preços simulada, gerar uma nova curva de preços ($\log\text{-moneyness}$, vencimento) utilizando a distribuição de máxima entropia (DME), conforme equação. (4.2.9). Essa nova superfície é necessária para a calibragem. Na Figura 5.4 há uma comparação dessa nova curva, em um única maturidade, em relação aos dados originais.

Pelo observado no gráfico à esquerda, o preços ficaram muito próximos, mostrando que a interpolação levou a preços bem aproximados, o que foi também evidenciado pela curva de erro

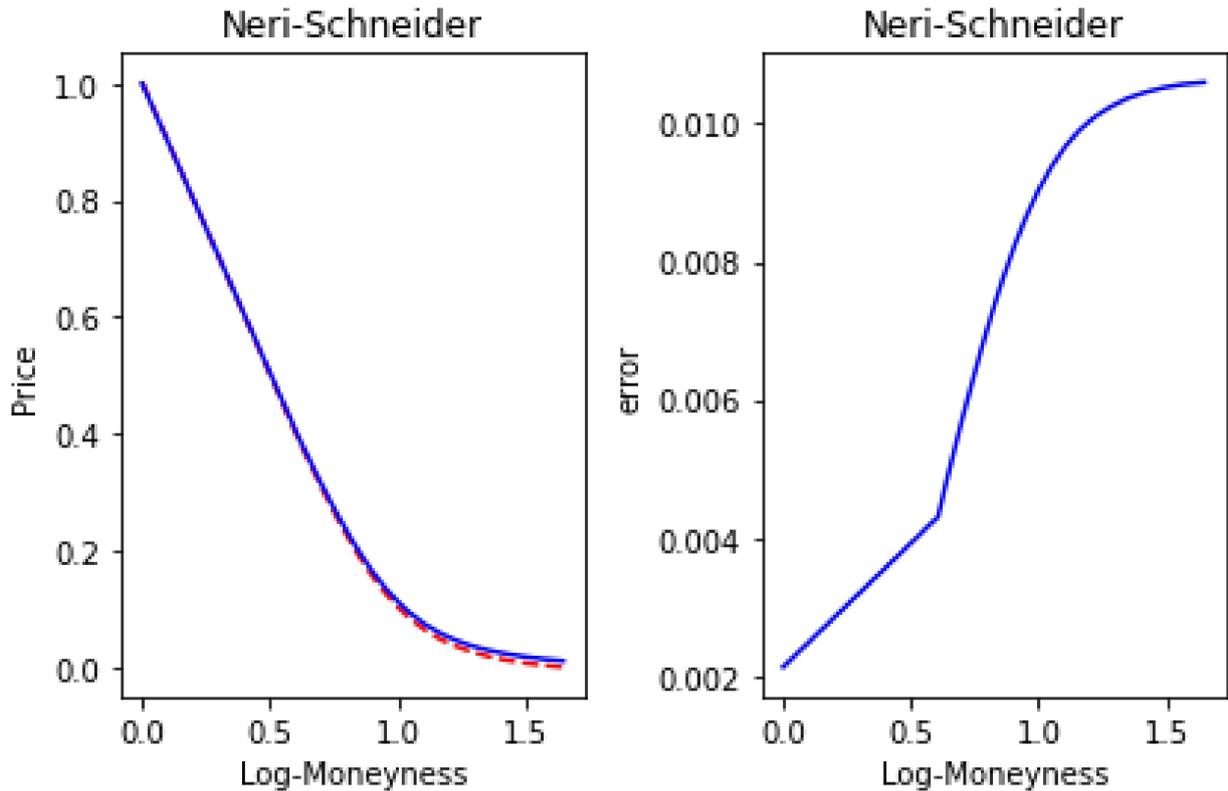


Figura 5.4: Representação do resultado da interpolação dos preços comparando somente uma maturidade. Esquerda: comparação do preço das opções de compra com os dados interpolados utilizando a distribuição de máxima entropia. Direita: erro relativo dos dados interpolados com os dados reais.

à direita em que a magnitude das diferenças é pequena.

O passo seguinte foi a obtenção da superfície de volatilidade local a partir da superfície de preços. Para isso, foi realizada a calibração do modelo a partir da minimização do funcional, conforme 4.2.10. Para a parcela referente à regularização, foi estimado o parâmetro de Tikhonov utilizando o critério da curva em L, conforme Figura 5.5. Foi escolhido um parâmetro perto da quebra da curva “L”, conforme discutido na Seção 4.1.3.

A tabela 5.1 resume os valores dos parâmetros utilizados na calibragem.

r	α	S_0	σ_0
0.02	2E-5	1	0.4

Tabela 5.1: Parâmetros da calibragem.

Na Figura 5.6 é mostrada a superfície de volatilidade local encontrada após a calibragem. De maneira geral, a ela se assemelha à curva de volatilidade teórica simulada, em forma e intervalo de variação.

Para melhor compará-las, na Figura 5.7 encontra-se o superfície do erro relativo entre ambas, a fim de se observar a magnitude dessas diferenças ponto a ponto e melhor analisar o grau de similaridade entre elas.

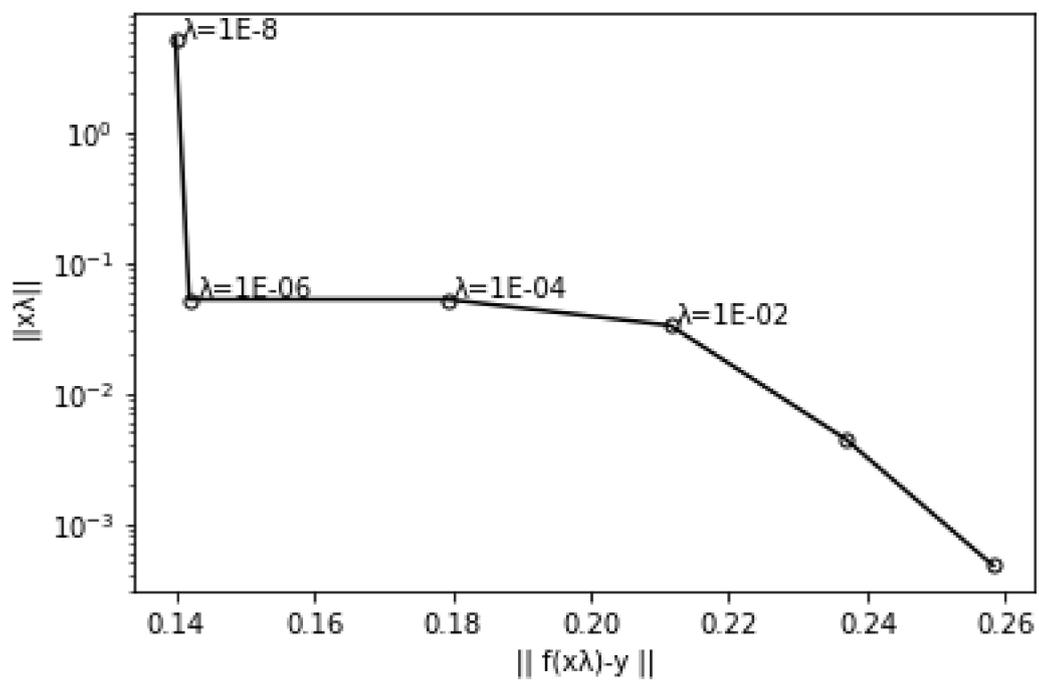


Figura 5.5: Curva em L gerada a partir dos dados simulados.

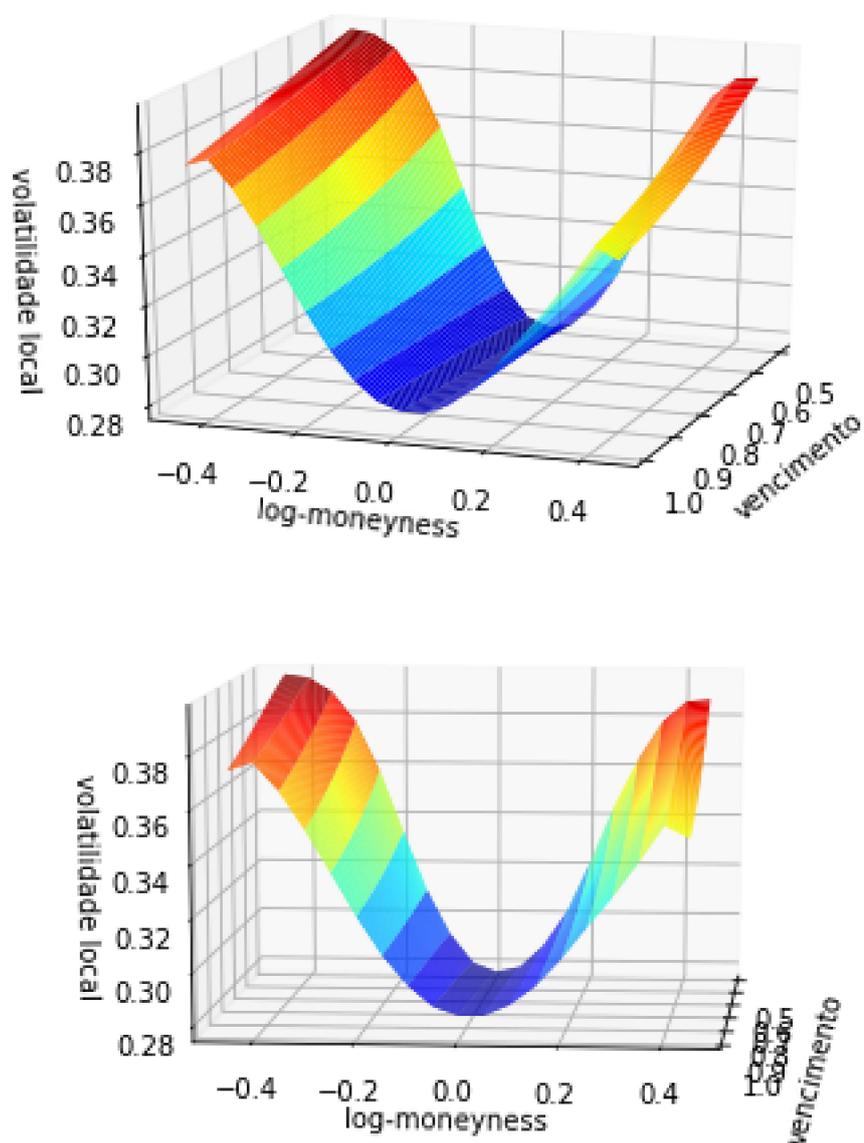


Figura 5.6: Superfície de Volatilidade resultante do problema inverso simulado

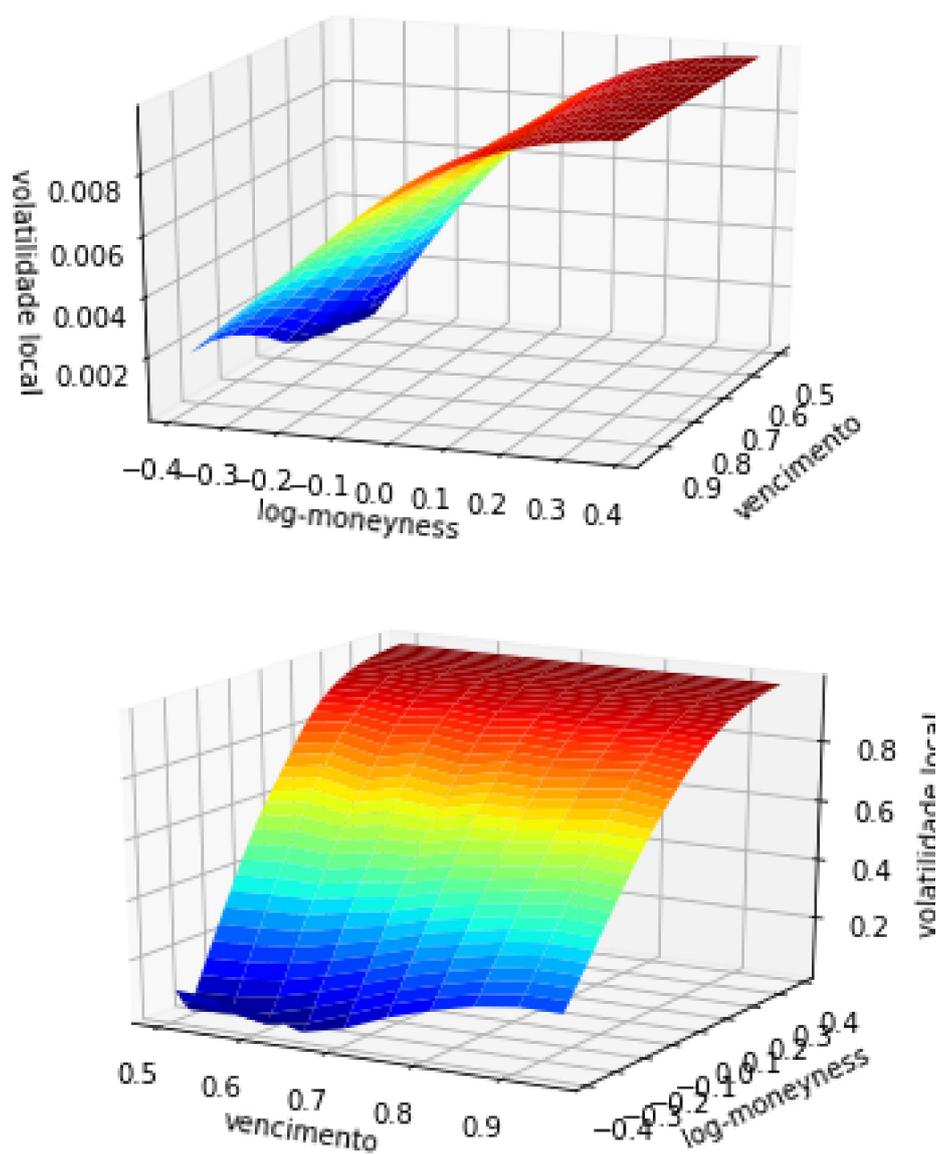


Figura 5.7: Superfície do erro relativo, em dois ângulos distintos, entre a curva simulada e a curva estimada da volatilidade segundo o método proposto.

5.2 Dados reais

Para validação dos métodos apresentados foram utilizados dados reais de opções de compra européias de *Standard & Poor's 500* (S&P 500)² obtidas em [Achdou and Pironneau \(2002\)](#) e datadas de outubro de 1995. A superfície de preços está disposta para valores de *log-moneyness* no intervalo de 0.9 a 1.4 e para valores de vencimento entre 0 e 5 anos. Na Figura 5.8 a superfície é ilustrada.

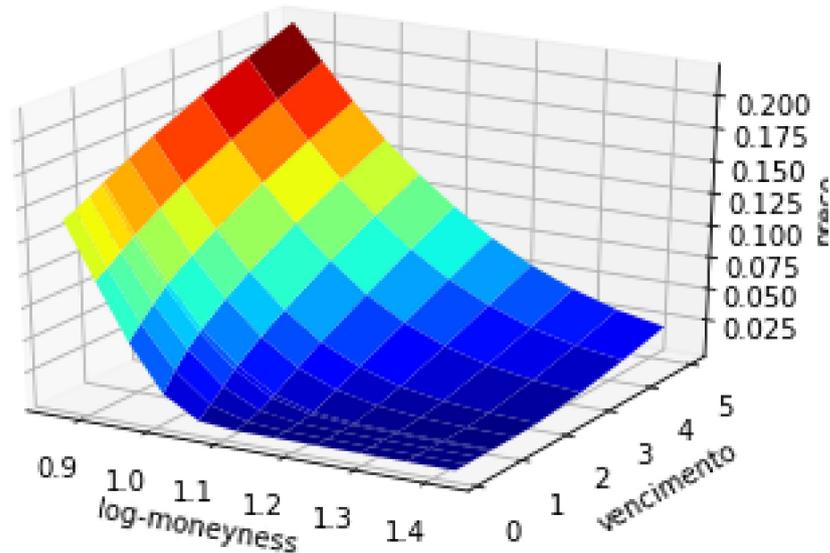


Figura 5.8: Superfície de preços das opções de compra européias da S&P 500 (outubro de 1995) em função do vencimento e do *log-moneyness*.

5.2.1 Resultados

O processo de obtenção da curva de volatilidade local para os dados reais segue as mesmas etapas seguidas na seção anterior (dados simulados).

Novamente, o primeiro passo foi a obtenção de uma nova curva de preços a partir da interpolação considerando a DME, conforme ilustrado na Figura 5.9. As curvas são bem similares, conforme esperado.

Para melhor análise visual, a diferença entre as superfícies original e interpolada pode ser vista na Figura 5.8. Para contratos com vencimentos mais longos e opções fora do dinheiro observou-se maior diferença relativa entre as superfícies de preços.

Posteriormente, na calibragem, foi realizada a minimização do funcional 4.2.10 e obtida a superfície de volatilidade local, conforme Figura 5.9. Assim como caso dos dados simulados, foi adicionada a parcela referente à regularização a qual requereu a estimação do parâmetro de Tikhonov obtida pela curva em L, conforme Figura 5.11.

²Índice composto pela ponderação das ações das 500 maiores empresas do mundo listadas na NYSE e na Nasdaq, uma medida padrão do desempenho médio do mercado de ações norte-americano.

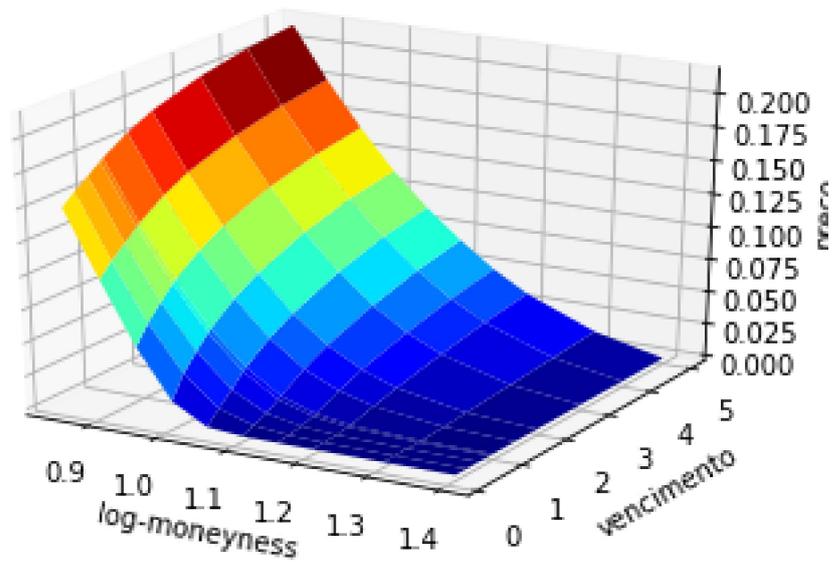


Figura 5.9: Superfície de preços das opções de compra interpolados com a Distribuição de Máxima Entropia.

O resultado da estimação da superfície de volatilidade local foi satisfatório pois houve a captura do *smile* do mercado.

Por fim, foi também gerada a superfície de volatilidades implícitas de Black and Scholes tanto para o problema calibrado quanto para os dados reais, ilustradas na Figura 5.13. Conforme observado, as curvas são bem próximas entre si, o que mostra um resultado satisfatório.

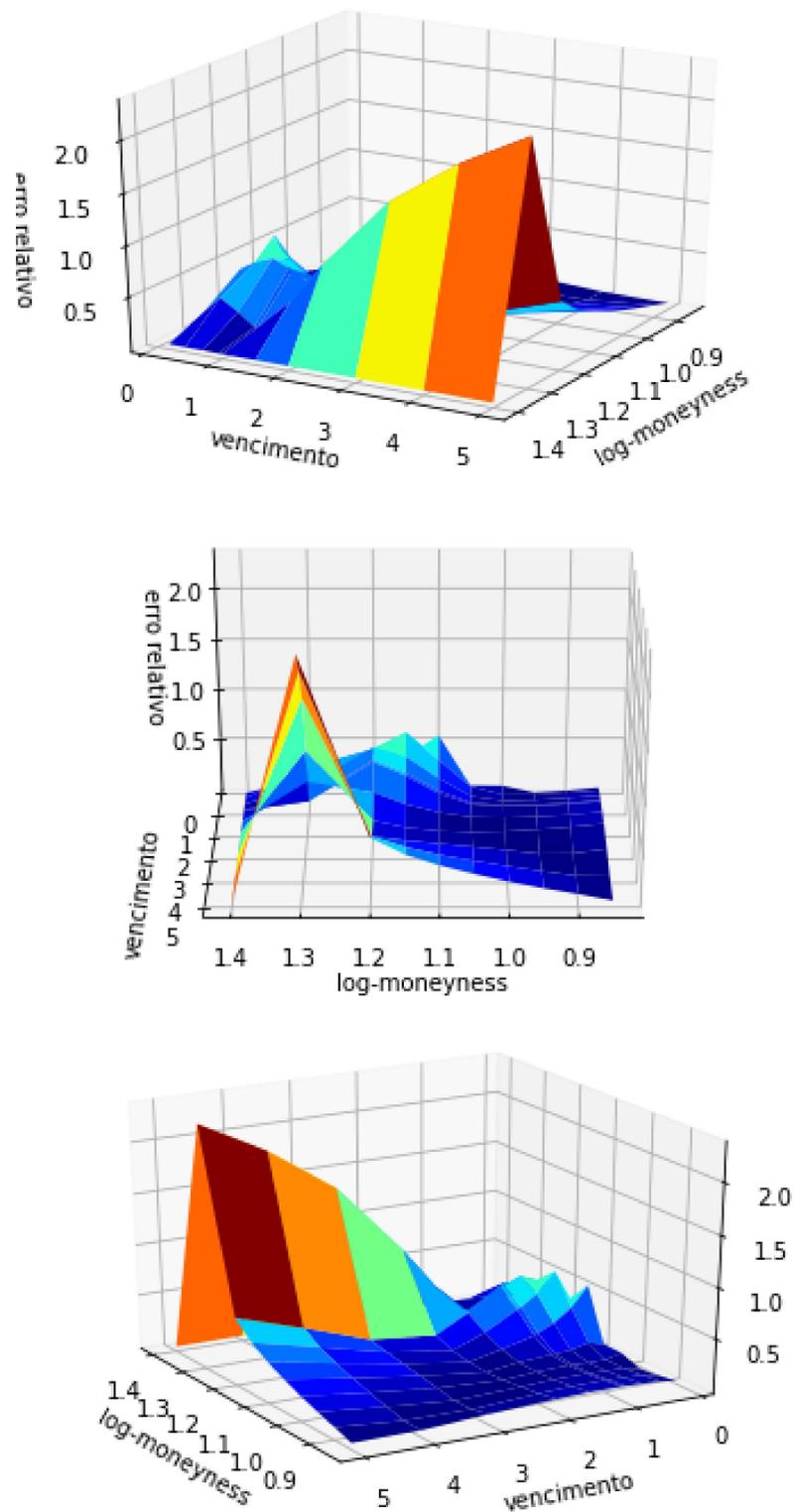


Figura 5.10: Diferença relativa entre as superfícies de preços dos dados reais de opções de compra de S&P e interpolada pelo método de Neri-Schneider (usando a MED), para diferentes ângulos.

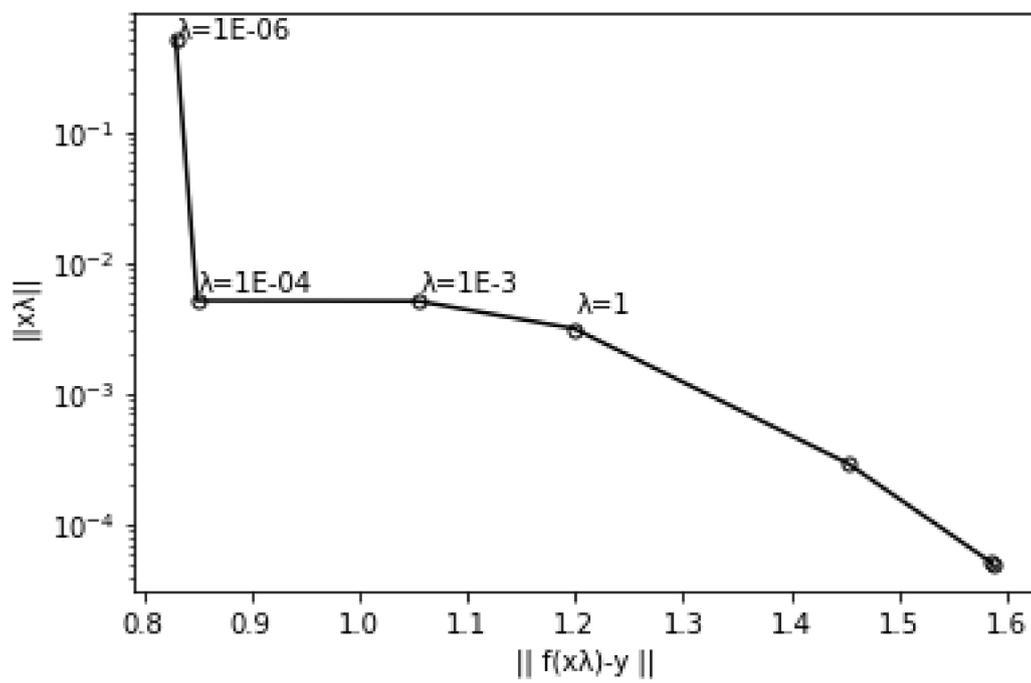


Figura 5.11: Curva em L gerada a partir dos dados reais.

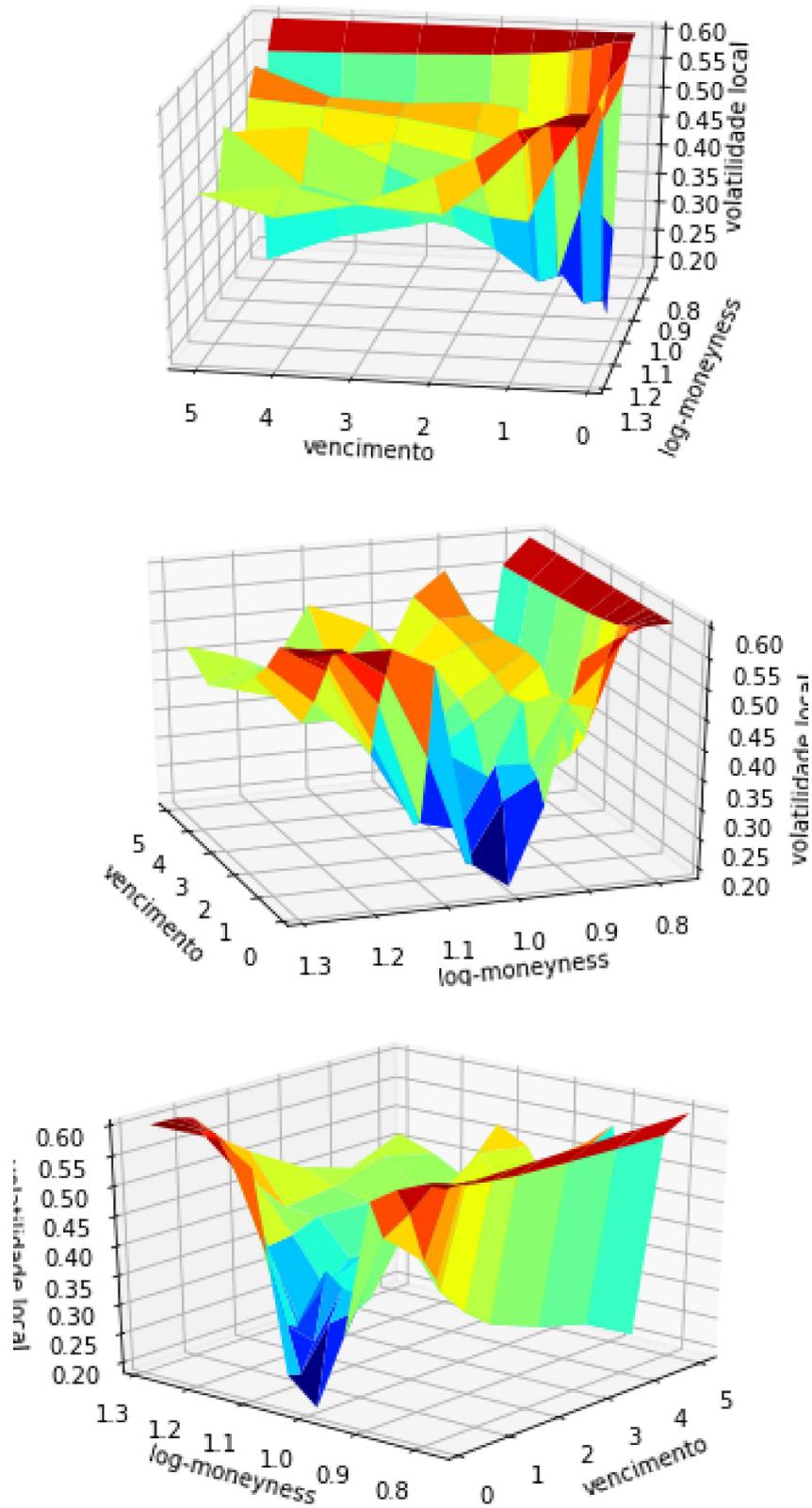


Figura 5.12: Diferentes ângulos da superfície de volatilidade local para preços de opções de S&P obtidas como solução do problema de calibragem, segundo o método proposto.

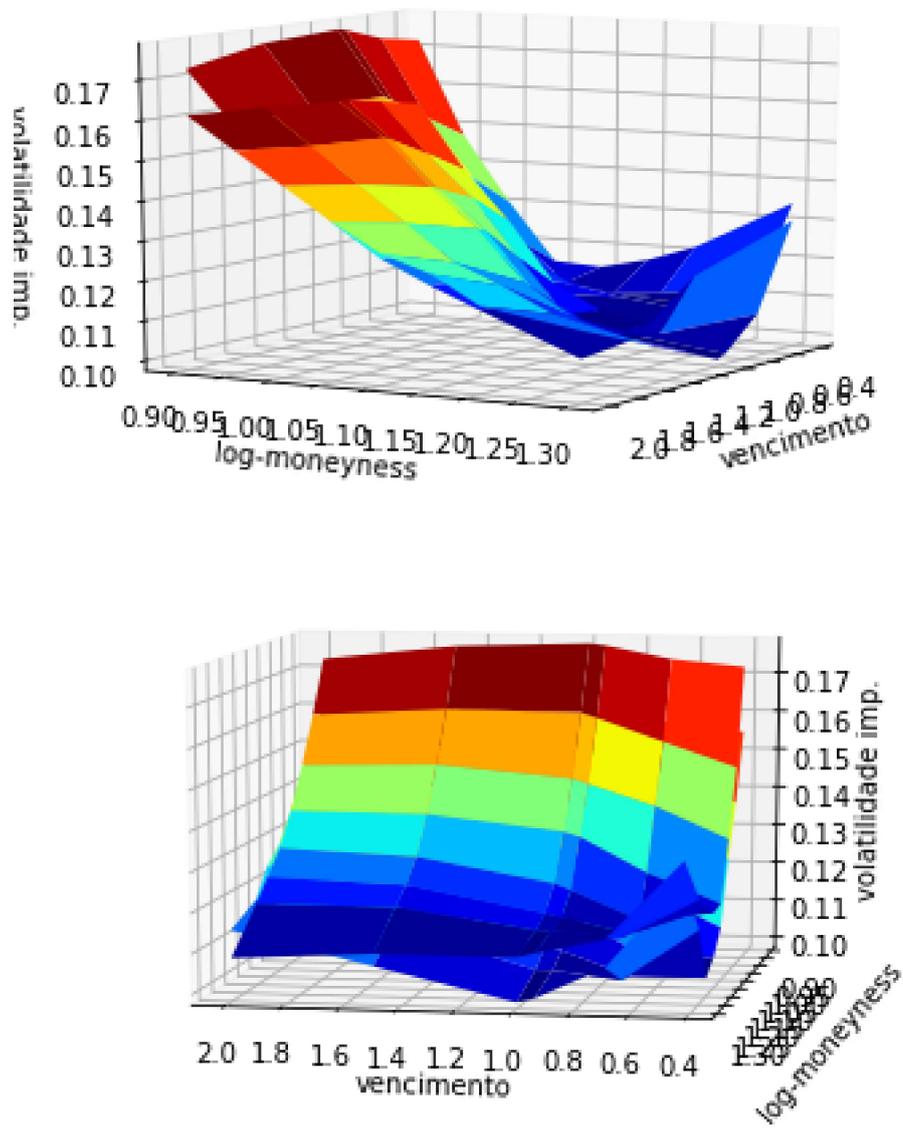


Figura 5.13: Diferentes ângulos das superfícies de volatilidade implícita de Black and Scholes para a solução do problema inverso e para os dados reais.

Capítulo 6

Conclusão

Nesta dissertação foi proposta uma nova forma de se estimar a volatilidade, a medida de risco de um ativo financeiro. Mais precisamente, o objetivo foi a construção de uma superfície de volatilidade local, assim chamada por não tratar os parâmetros de difusão do modelo de precificação do ativo de maneira constante, ao contrário do tradicional e importante modelo de [Black and Scholes \(1973\)](#).

Para tanto, foi considerada uma modificação do modelo proposto em [Dupire \(1994\)](#), sendo necessário o entendimento e a obtenção da densidade de máxima entropia (DME) discutida em [Neri and Schneider \(2012, 2011\)](#) e [Buchen and Kelly \(1996\)](#), para interpolação dos preços das opções de compra dos ativos. Para a estimação da DME durante a interpolação dos preços foi necessária a solução numérica de um processo de minimização via método de Newton-Raphson.

Essa abordagem se contrapõe a da equação de Fokker-Planck de Dupire e objetivou, fundamentalmente, a simplificação da dimensionalidade do problema da calibragem da volatilidade local. Tal redução permitiu a paralelização do problema de identificação da volatilidade local para diferentes vencimentos, o que não é possível ao se usar diretamente a equação de Dupire. Dessa forma, é possível dizer que houve melhora em comparação ao uso do método de Buchen and Kelly e de outros métodos numéricos para a resolução de uma EDP.

Para o ajuste do problema inverso da calibragem da volatilidade local foi utilizado o paradigma de estimação Bayesiano, em que o método de mínimos quadrados é empregado para encontrar a superfície de volatilidade. Esse método se mostrou eficiente ao reduzir a complexidade e aumentar a eficiência computacional na obtenção dos resultados.

Ademais, por ser um problema mal-posto no sentido de Hadamard, foi necessária a utilização da regularização de Tikhonov na calibragem da volatilidade local, a qual se mostrou eficiente para a estabilização do problema de minimização. O parâmetro de Tikhonov foi estimado por meio do critério da curva em L, tendo essa abordagem gerado resultados condizentes com o esperado. De maneira geral, todos os parâmetros do modelo podem ser identificados de forma simples e robusta.

Por fim, além da aplicação do novo método proposto primeiramente a dados simulados, para fins de validação dos algoritmos implementados, ele foi também aplicado a dados reais de opções de compras S&P 500. Para essa aplicação foram implementadas funções e métodos utilizando a linguagem Python™, sendo que todo o código desenvolvido se encontra disponibilizado no apêndice [B](#) da dissertação para fins de validação e disseminação do conhecimento.

Há que ressaltar que o método aqui proposto permite ir além do modelo de Black e Scholes e das interpolações de preços de opções, uma vez que a volatilidade local estimada permite uma aderência ao *smile* e captura o risco de forma muito mais precisa que modelos menos complexos. Permite, ainda, casar a volatilidade local com técnicas de interpolação de preço de forma eficiente, reduzindo o número de incógnitas no problema de calibragem.

Apêndice A

A Equação de Dupire

Suponha que a difusão de preço de um ativo, na medida neutra ao risco, seja dada pela seguinte equação:

$$\frac{dS}{S} = \mu(t)dt + \sigma(S, t)dZ \quad (\text{A.0.1})$$

Onde $\mu(t)$ é o *drift* e $\sigma(S, t)$ a volatilidade local.

Seja $C(S_0, K, T)$ o preço de uma opção Europeia, na media neutra ao risco, com *strike* K e maturidade T . Esse preço pode ser escrito como:

$$C(S_0, K, T) = \int_K^\infty dS_T \varphi(S_T, T; S_0) (S_T - K) \quad (\text{A.0.2})$$

Aqui $\varphi(S_T, T; S_0)$ é densidade de probabilidade no tempo T . Essa densidade tem a difusão, de acordo com a equação de Fokker-Planck, dada por:

$$\frac{1}{2} \frac{\partial^2}{\partial S_T^2} (\sigma^2 S_T^2 \varphi) - S \frac{\partial}{\partial S_T} (\mu S_T \varphi) = \frac{\partial \varphi}{\partial T} \quad (\text{A.0.3})$$

Diferenciando com respeito a K tem-se:

$$\begin{aligned} \frac{\partial C}{\partial K} &= - \int_K^\infty dS_T \varphi(S_T, T; S_0) \\ \frac{\partial^2 C}{\partial K^2} &= \varphi(K, T; S_0) \end{aligned} \quad (\text{A.0.4})$$

Agora, diferenciando [A.0.2](#) com respeito ao tempo:

$$\begin{aligned} \frac{\partial C}{\partial T} &= \int_K^\infty dS_T \left\{ \frac{\partial}{\partial T} \varphi(S_T, T; S_0) \right\} (S_T - K) \\ &= \int_K^\infty dS_T \left\{ \frac{1}{2} \frac{\partial^2}{\partial S_T^2} (\sigma^2 S_T^2 \varphi) - \frac{\partial}{\partial S_T} (\mu S_T \varphi) \right\} (S_T - K) \end{aligned} \quad (\text{A.0.5})$$

Integrando por partes:

$$\begin{aligned} \frac{\partial C}{\partial T} &= \frac{\sigma^2 K^2}{2} \varphi + \int_K^\infty dS_T \mu S_T \varphi \\ &= \frac{\sigma^2 K^2}{2} \frac{\partial^2 C}{\partial K^2} + \mu(T) \left(C - K \frac{\partial C}{\partial K} \right) \end{aligned} \tag{A.0.6}$$

Que é a equação de Dupire quando o ativo objeto tem o *drift*, na medida neutra ao risco, μ .

Apêndice B

Códigos

Neste apêndice estão disponibilizados os códigos na linguagem Python utilizados para a implementação do modelo proposto nesta dissertação, com todos os gráficos e estimativas apresentadas no capítulo de resultados. Na seção B.1 encontra-se a classe MED para cálculo da distribuição de máxima entropia (MED) via método de Newton-Raphson.

B.1 MED

```
1 import numpy as np
2 import numba as nb
3
4
5 class MED:
6
7     @nb.njit
8     def med_newton(self, x0, K_i, K_i1, C_i, C_i1, D_i, D_i1, i, n, max_iter=5000, tol
9         =1E-15, is_backtrack=False):
10         """
11             Maximum entropy distribution calculation with Newton-Raphson Method.
12
13             C. Neri and L. Schneider, Maximum entropy distributions inferred
14             from option portfolios on an asset
15
16             Parameters
17             -----
18
19             x0 : float
20                 initial value.
21             K_i : float
22                 Strike time i.
23             K_i1 : float
24                 Strike time i+1.
25             C_i : float
26                 Call price time i.
27             C_i1 : float
28                 Call price time i+1.
29             D_i : float
30                 Digital time i.
```

```

30     D_i1 : float
31         Digital time i+1.
32     i : float
33         time.
34     n : float
35         size.
36
37     max_iter : float, optional
38         Maximum iteration newton. The default is 5000.
39     tol : float, optional
40         error tolerance. The default is 1E-15.
41     is_backtrack : float, optional
42         Using backtrack algorithm. The default is False.
43
44     Returns
45     -----
46     float, float
47         alpha and beta coef Neri-Schneider.
48
49     """
50
51     error = 1 # error in iteration
52
53     p_i = D_i - D_i1 # den
54
55     Ktil_i = (C_i+K_i*D_i - (C_i1+K_i1*D_i1))/(p_i) # some problem if
the interval is very close to 0
56
57     # define functino to get MED NS definition
58     def med_funcs(xk):
59         # 2 cases as defined in paper
60         if(i==n):
61             if(xk<0):
62                 f = lambda x: np.log(np.exp(x*K_i)/(-x))
63                 Df = lambda x: K_i-1/x - Ktil_i
64                 DDf = lambda x: (1/x)**2
65
66                 return f, Df, DDf
67             else:
68                 return None, None, None
69
70         else: #i<n
71             if(xk==0):
72                 f = lambda x: np.log(K_i1-K_i)
73                 Df = lambda x: ((K_i1+K_i)/2) - Ktil_i
74                 DDf = lambda x: (((K_i1-K_i)**2)/12)
75
76             else:
77                 f = lambda x: np.log((np.exp(x*K_i1) - np.exp(x*K_i))/(x
78 ))
79                 Df = lambda x: (K_i1*np.exp(K_i1*x) - K_i*np.exp(K_i*x))
80 /((np.exp(K_i1*x) - np.exp(K_i*x)) - (1/x) - (Ktil_i)
81
82                 DDf = lambda x: (((-K_i1-K_i)**2)*((np.exp(x*(K_i1+K_i))
83 )/((np.exp(K_i1*x) - np.exp(K_i*x))**2)))) + (1/x**2)
84
85                 return f, Df, DDf

```

```

84     xn = x0
85
86     count = [xn]
87
88     # iterate to error tol
89     while error >= tol:
90         f, Df, DDf = med_funcs(xn)
91         if(f is not None):
92
93             # get functions
94             if(is_backtrack):
95                 xn = self.backtrack(x0, f, Df(xn), DDf(xn), i, xn)
96
97             else:
98                 xn = xn - Df(xn)/DDf(xn)
99
100            count = count + [xn]
101
102            # here there is an updet in derivative function
103            Df = lambda x: (K_i1*np.exp(K_i1*x) - K_i*np.exp(K_i*x))/(np
.exp(K_i1*x) - np.exp(K_i*x)) - (1/x) - (Ktil_i)
104            error = abs(Df(xn))
105
106            else:
107                Df = lambda x: (K_i1*np.exp(K_i1*x) - K_i*np.exp(K_i*x))/(np
.exp(K_i1*x) - np.exp(K_i*x)) - (1/x) - (Ktil_i)
108                error = abs(Df(xn))
109
110            if(len(count)==max_iter):
111                break
112
113            if((np.isnan(xn)) or (~np.isfinite(xn))):
114                beta_i = 0
115
116            else:
117                beta_i = xn
118
119            if(np.isnan(beta_i)):
120                alpha_i = 0
121
122            elif(i==n and beta_i>=0):
123                alpha_i=0
124
125            else:
126                f, Df, DDf = med_funcs(xn)
127                alpha_i = p_i*np.exp(-f(xn))
128
129            if(~np.isfinite(alpha_i)):
130                alpha_i = 0
131
132            return alpha_i, beta_i
133
134
135     def backtrack(self,x0, f, dfx, ddfx, t, beta):
136         """
137
138         Backtrack Newton-Raphson algorithm
139

```

```

140     Parameters
141     -----
142
143     x0 : TYPE
144         initial value.
145     f : TYPE
146         f.
147     dfx : TYPE
148         f'(xn).
149     ddfx: TYPE
150         f''(xn).
151     t : TYPE
152         time.
153     beta : TYPE
154         x n-1.
155
156     """
157
158     while (f(x0) - (f(x0 - t*np.array([dfx(x0), ddfx(x0)])) * t * np.dot
159 (np.array([dfx(x0), ddfx(x0)]), np.array([dfx(x0), ddfx(x0)])))) < 0:
160         t *= beta
161     return t
162
163 @nb.njit
164 def build_digital_array(self,C,K):
165     """
166
167     Calculate digital array
168
169     Parameters
170     -----
171
172     C : numpy.ndarray
173         Call prices array.
174
175     K : numpy.ndarray
176         Strike array.
177
178     Returns
179     -----
180     D : numpy.array
181         Artificial digital options prices.
182     """
183
184     if(K.shape!=C.shape):
185         print("Different shape sizes")
186         return np.array([])
187
188     # init digital array
189     D = np.zeros(K.shape)
190     D[0] = -(C[1] - C[0])/(K[1]-K[0])
191
192     for i in range(1,len(K)-1):
193         D[i] = -(C[i+1]-C[i-1])/(K[i+1]-K[i-1])
194
195     return D
196

```

```

197
198 @nb.jit
199 def ns_density(self,K,s,alpha,beta):
200     """
201
202     Build Neri-Schneider density
203
204     Parameters
205     -----
206
207     K : np.array
208         strike array.
209     K2 : np.array
210         integrate strike array.
211     alpha : np.array
212         alpha coefficient array Neri-Schneider.
213     beta : np.array
214         beta coefficient array Neri-Schneider.
215
216     Returns
217     -----
218     g : np.array
219         MED density array.
220
221     """
222
223     # init density array
224     g = np.zeros(s.shape)
225
226     # iterate strike intervals
227     for ii in range(0,len(K)-1):
228         aux = (alpha[ii]*np.exp(beta[ii]*s))*(s >= K[ii])*(s<K[ii+1])
229         aux[np.isnan(aux)] = 0
230         g = g + aux
231
232     return g

```

B.2 Calibragem

```

1 import numpy as np
2 import numba as nb
3 from scipy import interpolate
4 from bls.bls import bls
5
6
7 class util:
8
9     def CrankNicolson_Dupire(self,V,X,T,S0,r,a,l,nt,nx):
10         """
11
12         Solve Dupire equation using Crank-Nicolson method
13
14         Parameters
15         -----
16

```

```

17     V : np.array
18         Vol surface.
19     X : np.array
20         strike mesh.
21     T : np.array
22         Time mesh.
23     S0 : float
24         initial price.
25     r : float
26         rate.
27     a : np.array
28         time array.
29     l : np.array
30         strike array.
31     nt : float
32         time size.
33     nx : float
34         strike size.
35
36     Returns
37     -----
38     U : np.array
39         Price matrix.
40
41     """
42
43     # calc variance surface
44     A = (V**2)/2
45
46     # initial condition
47     U0 = np.maximum(0,(1 - np.exp(X)))
48
49     # boundary condition
50     BC = np.array([1,0])
51
52     # initialize variables
53     U = np.zeros((nx,nt))
54
55     # boundary conditions
56     U[:,0] = U0.T
57     U[0,:] = BC[0]
58     U[-1,:] = BC[1]
59
60     for j in range(1,nt):
61         # mat 1
62         A1 = np.diag(1 + A[:,j]*a[:,j]) - \
63
64             0.5*np.concatenate((np.zeros((nx,1)),np.concatenate((np.diag((a
65 [0:-1,j]-0.5*l[0:-1,j])*A[0:-1,j]-0.5*r*1[0:-1,j]),np.zeros((1,nx-1))),
66 axis=0)),axis=1)-\
67
68             0.5*np.concatenate((np.zeros((nx,1)),np.concatenate((np.diag((a
69 [1:,j]+0.5*l[1:,j])*A[1:,j]+0.5*r*1[1:,j]),np.zeros((1,nx-1))),axis=0)),
70 axis=1).T
71
72         # mat 2
73         A2 = np.diag(1 - A[:,j-1]*a[:,j-1]) + \

```

```

71         0.5*np.concatenate((np.zeros((nx,1)),np.concatenate((np.diag((a
[0:-1,j-1]-0.5*1[0:-1,j-1])*A[0:-1,j-1]-0.5*r*1[0:-1,j-1]),np.zeros((1,nx
-1))),axis=0)),axis=1)+\
72
73         0.5*np.concatenate((np.zeros((nx,1)),np.concatenate((np.diag((a
[1:,j-1]+0.5*1[1:,j-1])*A[1:,j-1]+0.5*r*1[1:,j-1]),np.zeros((1,nx-1))),
axis=0)),axis=1).T
74
75
76
77         # build aux matrices
78         A1A = A1[1:nx-1,1:nx-1]
79         b = np.matmul(A2,U[:,j-1])
80
81         # get temp mat
82         Utemp = b[1:nx-1]
83         Utemp[0] = Utemp[0] + 0.5*(A[0,j]*(0.5*1[0,j]+a[0,j])+0.5*1[0,j
]*r)*BC[0]
84         Utemp[-1] = Utemp[-1] + 0.5*(A[-1,j]*(-0.5*1[-1,j]+a[-1,j])-0.5*
1[-1,j]*r)*BC[1]
85         Utemp = np.linalg.lstsq(A1A,Utemp,rcond=None)[0]
86         U[1:nx-1,j] = Utemp
87
88         return U
89
90
91     @nb.jit
92     def ns_density(self,K,s,alpha,beta):
93
94         """
95         Build Neri-Schneider density
96
97         Parameters
98         -----
99
100        K : np.array
101            strike array.
102        K2 : np.array
103            integrate strike array.
104        alpha : np.array
105            alpha coefficient array Neri-Schneider.
106        beta : np.array
107            beta coefficient array Neri-Schneider.
108
109        Returns
110        -----
111        g : np.array
112            MED density array.
113
114        """
115
116        #init density vector
117        g = np.zeros(s.shape)
118
119        # iterate in strike axis
120        for ii in range(0,len(K)-1):
121            aux = (alpha[ii]*np.exp(beta[ii]*s))*(s >= K[ii])*(s<K[ii+1])
122            aux[np.isnan(aux)] = 0

```

```

123         g = g + aux
124
125     return g
126
127
128 @nb.jit
129 def NS_Pricing(self,r,S0,K,T,phi,vol0,dt):
130
131     """
132     Calculate option price using Neri-Schneider density
133
134     Parameters
135     -----
136
137     r : float
138         rate.
139     S0 : float
140         initial price.
141     K : np.array
142         strike array.
143     T : np.array
144         time array.
145     phi : function
146         Neri-Schneider density function.
147     vol0 : np.array
148         initial local-vol.
149     dt : TYPE
150         time step.
151
152     Returns
153     -----
154     C : np.array
155         option price.
156
157     """
158
159     # init params
160     dK = 0.001
161     K2 = np.arange(1E-3,K[-1]+10,dK)
162
163     # calculate density with function
164     density = phi(K2)
165     density[np.isnan(density)] = 0
166
167     # init idxs
168     idx1 = np.arange(2,len(density)-2,2)
169     idx2 = np.arange(1,len(density)-2,2)
170
171     # const to integrate
172     const = dK/3*(density[0] + density[-1] + 2*sum(density[idx1]) + 4*
173 sum(density[idx2]))
174
175     # aux vecs
176     vaux = np.array(list(vol0) + [vol0[-1]])
177     kaux = np.array(list(K)+[K[-1]])
178     set_interpolate = interpolate.interp1d(kaux,vaux)
179
180     # init result vec

```

```

180     CM = np.zeros(K.shape)
181
182     # interpolate and get idx to integrate
183     v = set_interpolate(K2)
184     idx1 = np.arange(2, len(K2)-2, 2)
185     idx2 = np.arange(1, len(K2)-2, 2)
186
187     # integrate with respect to K
188     for ll in range(0, len(K)):
189         aux = bls.blsprice(K2, K[ll], dt, r, v, 0)
190         aux2 = density*aux/const
191
192         # set val on result integration
193         CM[ll] = dK/3*(aux2[0] + aux2[-1] + 2*sum(aux2[idx1]) + 4*sum(
aux2[idx2]))
194
195     C = CM*np.exp(r*(dt))
196
197     return C

```

B.3 Vol Implícita

```

1 import numpy as np
2 from scipy import stats
3 from scipy import optimize
4 import numba as nb
5
6
7 class bls:
8
9     @staticmethod
10    @nb.jit
11    def blsprice(S, K, T, r, sigma, q=0, option_type='call'):
12
13        """
14        Function to return Black-Scholes option price
15
16        Parameters
17        -----
18
19        S : float
20            underlying price.
21        K : float
22            Strike.
23        T : float
24            Time to maturity.
25        r : float
26            interest rate.
27        sigma : float
28            volatility.
29        q : float, optional
30            dividend. The default is 0.
31        option_type : string, optional
32            option type 'call' or 'put'. The default is 'call'.
33

```

```

34
35     Returns
36     -----
37     float
38         option price.
39
40     """
41
42     # calculate bs params
43     d1 = (np.log(S/K) + (r - q + sigma ** 2 * 0.5)*T)/(sigma*np.sqrt(T))
44     d2 = d1 - sigma * np.sqrt(T)
45
46     if option_type.lower() in ['c', 'call']:
47         return np.exp(-r*T) * (S * np.exp((r - q)*T) * stats.norm.cdf(d1
) - \
48             K * stats.norm.cdf(d2))
49
50     elif option_type.lower() in ['p', 'put']:
51         return np.exp(-r*T) * (K * stats.norm.cdf(-d2) - \
52             S * np.exp((r - q)*T) * stats.norm.cdf(-d1))
53
54     else:
55         print('No such option type {}'.format(option_type))
56         print('Options are call, put')
57
58         return np.nan
59
60
61
62     @staticmethod
63     @nb.jit
64     def impvol(S, K, T, r, V, q=0, option_type='call', a=1e-10, b=10):
65
66         """
67         Calculate implied volatility with brentq method
68
69         Parameters
70         -----
71
72         S : float
73             underlying price.
74         K : float
75             Strike.
76         T : float
77             Time to maturity.
78         r : float
79             interest rate.
80         V : float
81             Option price.
82         q : float, optional
83             dividend. The default is 0.
84         option_type : string, optional
85             option type 'call' or 'put'. The default is 'call'.
86         a : float, optional
87             vol inf range. The default is 1e-10.
88         b : float, optional
89             vol max ange. The default is 10.
90

```

```
91     Returns
92     -----
93     root_vol : float
94         implied volatility.
95     """
96
97     # root function
98     root_vol_fun = lambda sigma: bls.blsprice(S, K, T, r, sigma, q,
option_type) - V
99
100     # if vol is close to 0
101     if root_vol_fun(a)*root_vol_fun(b) > 0:
102         a = -1
103
104     # finding root of fun
105     root_vol = optimize.brentq(root_vol_fun, a, b)
106
107     return root_vol
```


Referências Bibliográficas

- Achdou, Y. and O. Pironneau (2002). Volatility smile by multilevel least square. *International Journal of Theoretical and Applied Finance* 5(06), 619–643.
- Albani, V., U. Ascher, and J. P. Zubelli (2008). Local volatility models in commodity markets and online calibration. *Journal of Computational Finance*.
- Albani, V., A. Cezaro, and J. Zubelli (2013, 08). Convex regularization of local volatility estimation in a discrete setting.
- Albani, V. and J. Zubelli (2014). Online local volatility calibration by convex regularization. *Applicable Analysis and Discrete Mathematics* 8(2), 243–268.
- Baumeister, J. (2010). Inverse probleme der finanzmathematik wise 2010/11.
- Black, F. and M. Scholes (1973). The pricing of options and corporate liabilities. *Journal of Political Economy* 81(3), 637–654.
- Boyd, S. and L. Vandenberghe (2004). *Convex Optimization*. Cambridge University Press.
- Buchen, P. and M. Kelly (1996). The maximum entropy distribution of an asset inferred from option prices. *J. Financ. Quant. Anal.* 31, 143–159.
- Cartan, H. and J. Kouneiher (2007). *Cours de calcul différentiel*. Collection Méthodes. Editions Hermann.
- Caterina Fenu, Lothar Reichel, G. R. (2017). Morozov’s discrepancy principle for tikhonov regularization of severly ill-posed problems in finite-dimensional subspaces. *it Numer Math* 21, 1019–1039.
- Colton, D. (2003). Inverse acoustic and electromagnetic scattering theory. In *Inside Out: Inverse Problems and Applications*, Volume 47, pp. 67–110. Cambridge University Press.
- Crepey, S. (2003). Calibration of the local volatility in a generalized Black-Scholes model using Tikhonov regularization. *SIAM Journal of Mathematical Analysis* 34, 1183–1206.
- Dupire, B. (1994). Pricing with a smile. *Risk Magazine* 7, 18–20.
- Egger, H. and H. Engl (2005). Tikhonov Regularization Applied to the Inverse Problem of Option Pricing: Convergence analysis and Rates. *Inverse Problems* 21, 1027–1045.
- Engl, H., M. Hanke, and A. Neubauer (1996). *Regularization of Inverse Problems*, Volume 375 of *Mathematics and its Applications*. Kluwer Academic Publishers Group, Dordrecht.

- Gatheral, J. (2006). *The Volatility Surface: A Practitioner's Guide*. Wiley Finance. John Wiley & Sons.
- Gatheral, J. and A. Jacquier (2013). Arbitrage-free svi volatility surfaces.
- Gatheral, J. and A. Jacquier (2014). Arbitrage-free svi volatility surfaces. *Quantitative Finance* 14(1), 59–71.
- Gyöngy, I. (1986). Mimicking the one-dimensional marginal distributions of processes having an ito differential. *Probability Theory and Related Fields* 71, 501–516.
- Hagan, P., D. Kumar, A. Lesniewski, and D. Woodward (2002, 01). Managing smile risk. *Wilmott Magazine* 1, 84–108.
- Hansen, P. C. (2001, 01). *The L-Curve and Its Use in the Numerical Treatment of Inverse Problems*, Volume 4, pp. 119–142.
- Hansen, P. C. and D. O'leary (1993, 11). The use of the l-curve in the regularization of discrete ill-posed problems. *SIAM J. Sci. Comput.* 14, 1487–1503.
- Jari, K. and E. Korn (2004). *Statistical and Computational Inverse Problems*, Volume 31 of *Graduate Studies in Mathematics*. Springer.
- Jari, K. and E. Somersalo (2005). *Statistical and Computational Inverse Problems*. Springer-Verlag New York.
- Jaynes, E. T. (1957, May). Information theory and statistical mechanics. *Phys. Rev.* 106, 620–630.
- Koopman, S. J., B. Jungbacker, and E. Hol (2005). Forecasting daily variability of the s&p 100 stock index using historical, realised and implied volatility measurements. *Journal of Empirical Finance* 12(3), 445–475.
- Leitao, A. and J. Baumeister (2005, 07). *Topics in Inverse Problems*.
- Neri, C. and L. Schneider (2011). A family of maximum entropy densities matching call option prices. *Applied Mathematical Finance* 20, 548 – 577.
- Neri, C. and L. Schneider (2012). Maximum entropy distributions inferred from option portfolios on an asset. *Finance and Stochastics* 16(2), 293–318.
- Nocedal, J. and S. J. Wright (2006). *Numerical Optimization* (second ed.). New York, NY, USA: Springer.
- Pereverzyev, S. and E. Sock (2000, 11). Gcv for tikhonov regularization by partial svd. *Numerical Functional Analysis and Optimization* 21.
- Steven, H. (1993). A closed-form solution for options with stochastic volatility with applications to bond and currency options. *The review of financial studies* 6, 327–343.
- Stoll, H. R. (1969). The relationship between put and call option prices. *The Journal of Finance* 24(5), 801–824.
- Tarantola, A. (2005). *Inverse Problem Theory and Methods for Model Parameter Estimation*. Other titles in applied mathematics. Society for Industrial and Applied Mathematics.

- Tikhonov, A. N. (1943). On the stability of inverse problems. *Doklady Akademii Nauk SSSR*, 195–198.
- Tikhonov, A. N., A. Goncharsky, V. Stepanov, and A. G. Yagola (2013). *Numerical methods for the solution of ill-posed problems*, Volume 328. Springer Science & Business Media.
- Wilmott, P. (2007). *Paul Wilmott Introduces Quantitative Finance* (2 ed.). USA: Wiley-Interscience.