Delft University of Technology

Faculty EEMC
Delft Institute of Applied Mathematics

Pricing multi-dimensional American options using kernel ridge regression

by

Qianqian Chen

Submitted in partial fulfilment of the requirements for the degree of

Master of Science

in

Applied Mathematics with specialization Financial Engineering

at Delft University of Technology

Student number:	4593499
Thesis Committee:	Dr. N. Parolya (supervisor) Prof. Dr. A. Papapantoleon
Defence date:	January 12, 2023

January 4, 2023

TU Delft Delft University of Technology

Acknowledgement

I want to express my deepest appreciation to my supervisor Dr. N. Parolya for his guidance and feedback. I have learned a lot through this master end project under his supervision. I am very thankful to Prof. Dr. A. Papapantoleon for willing to be a member of my thesis committee.

I also want to thank my family, friends, and study buddies for supporting me through my master and this project. A special thanks to my lovely mother for her understanding and help.

Abstract

American option pricing has been an active research area in financial engineering over the past few decades. Since no analytic closed-form solution exists, various numerical approaches have been developed. Among all proposed methods, the least square Monte Carlo (LSMC) approach proposed by Longstaff and Schwartz (2001) is the most successful and popular. The LSMC utilizes linear regression for the estimation of the continuation values for the option. However, the accuracy of the LSMC is dependent on the chosen basis functions, where no objective strategy exists for the basis function selection process. Recently, Hu and Zastawniak (2020) proposed to use kernel ridge regression (KRR) with a bundling technique for high-dimensional American option pricing to avoid selecting the basis functions.

The Heston model is an example of the stochastic volatility model (Heston, 1993), where only limited literature can be found concerning multi-dimensional American option pricing under this model (Samimi and Mehrdoust, 2018). In this thesis, we reproduce the proposed KRR-based methods. Additionally, we extend the KRR-based methods for high-dimensional American options under the Heston model. We also involve the LSMC method for evaluating the pricing efficiency and accuracy of KRR-based methods. As no reliable benchmarks exist for the valuation of American options under the high-dimensional Heston model, we implement the primal-dual approach proposed by Andersen and Broadie (2004) and use its result as the reference price. Nevertheless, the obtained benchmarks are biased. Therefore, we can only conclude that the KRR-based methods apply to American option pricing under the high-dimensional Heston model. Moreover, the KRR-based techniques are computationally more efficient than the LSMC method.

Contents

Chapter 1

Introduction

An option allows the owner to trade a prescribed amount of underlying assets at a predetermined strike price before or on the expiration date. The American and European options can be distinguished according to exercise time. The former permits the owner to exercise at any time, including the day of expiration, whereas the latter allows the owner to exercise only at maturity.

The valuation of the options is crucial in the financial market. The traders can adjust their strategies and portfolios according to the option value estimate. Over the past few decades, the valuation of American options has been an active research area in financial engineering since it has been widely traded in the financial market. The value of an American option is equal to the achieved payoff when the holder optimally exercises it. This valuation problem is the same as determining the optimal exercise strategy, which leads to an optimal stopping time problem. In contrast to the valuation of European options, where a closed-form solution exists under many models (Oosterlee and Grzelak, 2020), American option pricing is generally solved using numerical approaches.

The Monte Carlo methods are the most feasible approach for high dimensional option pricing. Tilley (1993) first applied Monte Carlo simulation to American option valuation. Among all proposed Monte Carlo methods since then, the regression-based method is the most appealing for pricing high-dimensional American options. This approach involves backward recursion to solve the optimal stopping time problem, which was first proposed by Carriere (1996) and further developed by Tsitsiklis and Van Roy (2001) and Longstaff and Schwartz (2001). Especially the Least Squares Monte Carlo (LSMC) approach proposed by Longstaff and Schwartz (2001) remains the most successful simulation methodology for multidimensional American option pricing in both pricing accuracy and computational efficiency. One drawback of this approach is that the pricing accuracy depends on the chosen basis functions for linear regression. Han et al. (2008) suggested using kernel ridge regression instead of linear regression to avoid the selection process of the basis function. Hu and Zastawniak (2020) combined the kernel ridge regression with the bundling idea in the stochastic mesh method developed by Jain and Oosterlee (2015). This hybrid approach produces accurate pricing results and is computationally more efficient than LSMC for high-dimensional American options (Hu and Zastawniak, 2020).

Most of the methods mentioned above were applied to American options from which the underlying asset prices are generated by the Geometric Brownian motion (GBM) model (and Merton jump-diffusion in Hu and Zastawniak (2020)). Since the development of GBM, it has become a popular model for option pricing among researchers and practitioners. Nevertheless, the GBM model does not satisfy the properties of the real asset prices as the volatility is assumed to be a constant. To overcome this problem, stochastic volatility models are developed. The Heston model is an example of those developed models, where the volatility follows the Cox–Ingersoll–Ross(CIR) process (Heston, 1993). This CIR process ensures the underlying

asset prices have non-negative volatility. In addition, the asset prices generated by the Heston model converge to their long-term mean value over time, as the Heston model satisfies the mean-reverting property. Besides, the Heston model has a closed-form solution for European options. Those are the reasons for the popularity of the Heston model.

Those numerical methods for option pricing are also applied to the Heston model. However, only limited literature for American option pricing under the multi-dimensional Heston model can be found. In Samimi and Mehrdoust (2018), the LSMC methods are applied to multi-dimensional (up to dimension 3) assets under the Heston model. In this thesis, we will reproduce the kernel ridge regression (KRR)-based methods proposed by Hu and Zastawniak (2020). Furthermore, we will investigate whether the KRR-based method applies to American option pricing, where the asset prices are generated using the high-dimensional Heston model. We will simultaneously utilize LSMC methods for the high-dimensional Heston model and compare the LSMC-and KRR-based methods in terms of pricing accuracy and computational efficiency.

Main assumption

In this thesis, we consider the probability space $(\Omega, \mathcal{F}, \mathbb{Q})$ with a time horizon $[0, T]$, where $\mathbb{Q}$ denotes the risk-neutral measure. $\mathcal{F}$ is the σ-algebra of all events Ω and $\{\mathcal{F}_t | 0 \leq t \leq T\}$ is the corresponding filtration.

Outline of the thesis

The thesis is organized as follows. chapter 2 covers the mathematical description of American option pricing. The proposed kernel ridge regression-based (KRR-based) methods by Hu and Zastawniak (2020) are derived in chapter 3. The pricing results under high-dimensional geometric Brownian motion and Merton jump diffusion are evaluated in chapter 4. We elaborate on the Heston model in chapter 5, followed by numerical experiments of the Heston model using the KRR-based method in chapter 6. The conclusion and recommended future research are in chapter 7.

Chapter 2

Background

This chapter presents the fundamental concepts of American option pricing in a nutshell. We will begin with preliminary information on financial assets, followed by numerous asset price processes. Then we will cover the basic definitions of options. After the derivation of European option pricing, we will present the main topic of this thesis: the valuation of high-dimensional American options. Please keep in mind that, unless otherwise specified, all the fundamentals in this chapter are derived from Oosterlee and Grzelak (2020).

2.1 Preliminaries

A stochastic process $\{W(t) : t \geq 0\}$ is called a Wiener process (or Brownian motion) if the following conditions hold:

1. This process starts at 0 , i.e. $W_0 = W(0) = 0$.
2. For all $t > 0$, W_t is normally distributed with mean equal to zero and variance equal to t, i.e., $W_t \sim \mathcal{N}(0, t)$.
3. It has independent increments, i.e. for $0 \leq t_0 < t_1 < \cdots < t_n$, the random variables $Y_i = W(t_i) - W(t_{i-1})$ are independent for all $i \in \{1, ..., n\}$.
4. It is an almost surely continuous path.

The movement of asset prices is generally represented by a stochastic differential equation (SDE), which is commonly constructed using the Wiener process. Before presenting various examples of the SDE, two essential properties will be discussed to provide an intuition regarding the behavior of asset prices.

Definition 2.1.1 (Markov Process)**.** *Let $(\Omega, \mathcal{F}, \mathbb{Q})$ be a probability space with a time horizon $[0, T]$, where $\mathcal{F} = \{\mathcal{F}_t | 0 \leq t \leq T\}$ is the filtration. A stochastic process $S(t)$ defined on this probability space is a Markov process if following condition holds for any bounded and measurable function $g : \mathbb{R}^N \to \mathbb{R}$ with $s \leq t$:*

$$\mathbb{E}[g(S(t))|\mathcal{F}(s)] = \mathbb{E}[g(S(t))|S(s)].$$

This Markov property states that the current asset price contains all information about past prices. The other property is the martingale property, which states that the expectation of the future value is equal to the current value. If the martingale property holds, the model is free of arbitrage. Or, equivalently, there are no risk-free profits.

Definition 2.1.2 (Martingale)**.** *Given probability space $(\Omega, \mathcal{F}, \mathbb{Q})$ and define a stochastic process $S(t)$ on this probability space. $S(t)$ is a martingale if $\mathbb{E}[|S(t))|] < \infty$ for all $t < \infty$, and*

$$\mathbb{E}[S(t))|\mathcal{F}(s)] = S(s), \text{ for } s < t.$$

It is common to assume that the asset price follows the Itô's process. The Itô's process satisfies the Itô's lemma, which is fundamental for the derivation of the solutions of the SDE and for the option pricing.

Definition 2.1.3 (Itô's process)**.** *An Itô's process is a stochastic process that satisfies the following SDE,*

$$dX(t) = \tilde{\mu}(t, X(t))dt + \tilde{\sigma}(t, X(t))dW_t, \text{ with } X(t_0) = X_0$$

where $\tilde{\mu}(t, X(t))$ a general drift function that represents the growth rate of the asset prices and $\tilde{\sigma}(t, X(t)$ the volatility function which indicates the difference with the expected value and W_t is a Wiener process. The drift function and the volatility function must satisfy the following conditions:

$$|\tilde{\mu}(t,x) - \tilde{\mu}(t,y)|^2 + |\tilde{\sigma}(t,x) - \tilde{\sigma}(t,y)|^2 \leq K_1|x-y|^2$$
$$|\tilde{\mu}(t,x)|^2 + |\tilde{\sigma}(t,x)|^2 \leq K_2(1+|x|^2),$$

for some $x, y \in \mathbb{R}$ and $K_1, K_2 \in \mathbb{R}_+$. Those two conditions state that the volatility and the drift terms of a stock price should not change rapidly.

Before starting with Itô's lemma, we will introduce Itô's multiplication table (see Table 2.1) for the Wiener process, which is an essential tool in financial mathematics.

	dt	dW_t
dt	0	0
dW_t	0	dt

Table 2.1: Itô's multiplication table for Wiener process.

For the term $dtdW_t$, it is clear that its expectation is equal to zero as the Wiener increments follow the normal distribution with mean zero and variance dt. And the standard deviation of $dtdW_t$ is $dt^{\frac{3}{2}}$; this term vanishes rapidly to zero as dt goes to zero.

Remark. $(dW_t)^2 \to dt$ *as* $dt \to 0$.

Proof. Suppose the time horizon $[0, T]$ is divided into N equal-sized timesteps and the step size is $\Delta t = \frac{T}{N}$, then we can define $dW_t = W(t + \Delta t) - W(t)$

$$\mathbb{E}[(dW_t)^2] = \lim_{\Delta t \to 0} \mathbb{E}[(W(t+\Delta t) - W(t))^2] = \lim_{\Delta t \to 0} \text{Var}\left(W(t+\Delta t) - W(t)\right) = \lim_{\Delta t \to 0} \Delta t = dt.$$

The variance can be determined as follows

$$\begin{aligned}\text{Var}\left((dW_t)^2\right) &= \lim_{\Delta t \to 0} \mathbb{E}\left[(W(t+\Delta t) - W(t))^4\right] - \lim_{\Delta t \to 0} \left(\mathbb{E}\left[(W(t+\Delta t) - W(t))^2\right]\right)^2 \\ &= \lim_{\Delta t \to 0} 3(\Delta t)^2 - \lim_{\Delta t \to 0} (\Delta t)^2 = \lim_{\Delta t \to 0} 2(\Delta t)^2 = 2(dt)^2.\end{aligned}$$

As dt goes to zero, variance goes to zero faster than the expectation. Therefore, $(dW_t)^2 = dt$.

□

Lemma 2.1.1 (Itô's lemma)**.** *Suppose X(t) is a Itô's process, i.e.* $dX(t) = \tilde{\mu}(t, X(t))dt + \tilde{\sigma}(t, X(t))dW_t$*, and suppose* $Y(t) = g(t, X(t))$ *is a function of time t and* $X(t)$ *with continuous partial derivatives:* $\frac{\partial g}{\partial t}$, $\frac{\partial g}{\partial X}$ *and* $\frac{\partial^2 g}{\partial X^2}$*. Then the dynamic of* $g(t, X(t))$ *can be represented as:*

$$dY(t) = \left(\frac{\partial g}{\partial t} + \frac{\partial g}{\partial X}\tilde{\mu}(t, X(t)) + \frac{1}{2}\frac{\partial^2 g}{\partial X^2}\tilde{\sigma}^2(t, X(t))\right)dt + \frac{\partial g}{\partial X}\tilde{\sigma}(t, X(t))dW_t. \tag{2.1}$$

In the following section, we will present some commonly used models for generating asset prices. These models will later be used for the American option pricing.

2.2 Various asset price process

2.2.1 Geometric Brownian Motion (GBM)

The most commonly used asset price process is the Geometric Brownian motion (GBM), which satisfies the following SDE for $t \in [0, T]$:

$$\frac{\mathrm{d}S(t)}{S(t)} = (r - q)\mathrm{d}t + \sigma \mathrm{d}W_t, \tag{2.2}$$

where $S_t = S(t)$ is the asset price at time t, r is the annual risk-free interest rate, q the dividend rate, σ denotes the volatility parameter which indicates the difference with its expected value, and W_t is a Wiener process.

Let us consider the logarithmic process of the GBM, which is defined as $X_t = X(t) = \log(S(t))$ on $t \in [0, T]$. The corresponding partial derivatives of this process are as follows

$$\frac{\partial X_t}{\partial t} = 0, \ \frac{\partial X_t}{\partial S_t} = \frac{1}{S_t} \text{ , and } \frac{\partial^2 X_t}{\partial S_t^2} = -\frac{1}{S_t^2}.$$

By applying the Itô's lemma and then integrating, we get

$$\begin{aligned}
\mathrm{d}X_t &= \left(\frac{1}{S_t}(r - q)S_t - \frac{1}{2S_t^2}\sigma^2 S_t^2\right)\mathrm{d}t + \frac{1}{S_t}S_t\sigma \mathrm{d}W_t = \left((r - q) - \frac{1}{2}\sigma^2\right)\mathrm{d}t + \sigma \mathrm{d}W_t \\
\Rightarrow \int_0^t \mathrm{d}X_s &= \int_0^t \left((r - q) - \frac{1}{2}\sigma^2\right)\mathrm{d}s + \int_0^t \sigma \mathrm{d}W_s \\
X_t &= X_0 + \left((r - q) - \frac{1}{2}\sigma^2\right)t + \sigma\sqrt{t}Z,
\end{aligned} \tag{2.3}$$

with $Z \sim \mathcal{N}(0, 1)$. From this expression, the exact solution of Eq.(2.2) can be derived as follows

$$\begin{aligned}
\log(S_t) &= \log(S_0) + \log(e^{\left((r-q)-\frac{1}{2}\sigma^2\right)t+\sigma\sqrt{t}Z}) \\
&= \log(S_0 e^{\left((r-q)-\frac{1}{2}\sigma^2\right)t+\sigma\sqrt{t}Z}),
\end{aligned}$$

which gives

$$S_t = S_0 e^{\left((r-q)-\frac{1}{2}\sigma^2\right)t+\sigma\sqrt{t}Z}. \tag{2.4}$$

We can also consider the following for the simulation of the GBM process. Let us divide the time horizon into N time steps with step size $h = \frac{T}{N}$, i.e. $0 = t_0 < t_1 < ... < t_N = T$. Then the logarithmic asset price at each time step is

$$X_{t_{i+1}} = X_{t_i} + \left((r - q) - \frac{1}{2}\sigma^2\right)h + \sigma\sqrt{t_{i+1} - t_i}Z, \tag{2.5}$$

for $i \in \{0, ..., N - 1\}$ and $Z \sim \mathcal{N}(0, 1)$. Then $S_{t_{i+1}} = \exp(X_{t_{i+1}})$.

Multi-dimensional GBM

Let us extend the single asset GBM to a multi-dimensional GBM with correlated asset prices of dimension d. The SDE of the multi-dimensional underlying asset $\mathbf{S}(t) = (S_1(t), ..., S_d(t)) \in \mathbb{R}^d$ is:

$$\frac{\mathrm{d}S_{t,v}}{S_{t,v}} = (r - q_v)\mathrm{d}t + \sigma_v \mathrm{d}W_{t,v}, \ v = 1, ..., d. \tag{2.6}$$

The Wiener processes $W_{t,v}$ and $W_{t,l}$ are correlated with correlation coefficient ρ_{vl}, for $v, l = 1, ..., d$ and $v \neq l$. Let $\Sigma^S \in \mathbb{R}^{d\times d}$ denote the covariance matrix which is defined as $(\Sigma^S)_{vl} = \sigma_v\sigma_l\rho_{vl}$ and we denote its Cholesky decomposition as $\Sigma^S = AA^T$. Then Eq.(2.6) can be rewritten as

$$\frac{\mathrm{d}S_{t,v}}{S_{t,v}} = (r - q_v)\mathrm{d}t + \sum_{l=1}^{d} A_{v,l}\mathrm{d}\widetilde{W}_{t,l}, \ v = 1, ..., d, \tag{2.7}$$

with independent $\widetilde{W}_{t,v}$ and $\widetilde{W}_{t,l}$ for all $v, l = 1, ..., d$ with $v \neq l$.

The multi-dimensional GBM can be simulated similarly as the one-dimensional case. By dividing the time horizon $[0, T]$ into N equal-sized time steps, the multi-dimensional logarithmic process $\mathbf{X}_t = \log(\mathbf{S}_t)$ is discretized as

$$X_{i+1,v} = \underbrace{X_{i,v} + (r - q_v - \frac{1}{2}\sigma_v^2)h}_{=\mu_{i+1,v}} + \sqrt{h}\sum_{l=1}^{d} A_{v,l}Z_{i,l}, \ \text{for } i = 0, ..., N-1, \tag{2.8}$$

where $h = \frac{T}{N}$ and $Z_{i,l}$ are identically independent standard normally distributed. The term t_i is denoted as i to simplify the notation. As a result, $\mathbf{X}_{i+1}|\mathbf{X}_i$ is normally distributed: $\mathbf{X}_{i+1}|\mathbf{X}_i \sim \mathcal{N}(\boldsymbol{\mu}_{i+1}, h\Sigma^S)$, with $\boldsymbol{\mu}_{i+1} = (\mu_{i+1,1}, ..., \mu_{i+1,d})$. Just like the single asset GBM, $S_{i+1,v} = \exp(X_{i+1,v})$.

Sudden jumps can be observed when looking at the movement of asset prices in the financial market. The GBM model does not capture this behavior which is a drawback of the GBM model. In the following subsection, we will introduce an example of the model that covers this phenomenon.

2.2.2 Merton Jump diffusion (MJD)

To investigate the discontinuity of the asset prices, Merton (1976) proposed the jump-diffusion model. The jump-diffusion models consist of one diffusion and one jump part. For the Merton jump-diffusion model(MJD), the diffusion part is generated from the GBM. Merton added the Poisson process to this diffusion model to describe the sudden asset price jumps by utilizing the discontinuous nature of the Poisson distribution.

Definition 2.2.1 (Poisson process)**.** *A Poisson process, $\{\Gamma(t), t \geq t_0 = 0\}$, is an integer-valued stochastic process with the following properties*

1. *It starts at 0, i.e. $\Gamma(0) = 0$.*
2. *It has independent increments: for all $t_0 = 0 < t_1 < \ldots < t_n$, the random variables $Y_i = \Gamma(t_i) - \Gamma(t_{i-1})$ are independent for all $i \in \{1, ..., n\}$.*
3. *The increments are Poisson distributed:*

$$\mathbb{Q}[Y_i = k] = \frac{\left(\lambda^J(t_i - t_{i-1})\right)^k \mathrm{e}^{-\lambda^J(t_i - t_{i-1})}}{k!}$$

for all $i \in \{1, ..., n\}$ and integers $k \geq 0$. The intensity $\lambda^J > 0$ denotes the expected number of jumps per time unit.

The underlying asset prices generated by an MJD model satisfy the following SDE under the risk-neutral measure $\mathbb{Q}$

$$\frac{\mathrm{d}S(t)}{S(t)} = \left(r - q - \kappa^J \lambda^J\right) \mathrm{d}t + \sigma \mathrm{d}W(t) + \left(\mathrm{e}^{N^J} - 1\right) \mathrm{d}\Gamma(t), \ t \in [0, T], \tag{2.9}$$

where r denotes the riskless interest rate, q the dividend rate and σ is the volatility of the diffusion part. $\Gamma(t)$ is a Poisson process with jump intensity λ^J and N^J is the jump sizes following normal distribution with mean μ^J and variance $(\sigma^J)^2$. To make sure that $\mathrm{e}^{-rt} S_t$ is a martingale under the risk-neutral measure, $\kappa^J = \mathbb{E}\left[\mathrm{e}^{N^J} - 1\right] = \exp\left(\mu^J + (\frac{1}{2}\sigma^J)^2\right) - 1$, for the detailed derivation see p.135 of Oosterlee and Grzelak (2020). We assume that the Poisson process $\Gamma(t)$, the Wiener process $W(t)$, and the jump size J are independent of each other.

The Itô's table and the Itô's lemma for MJD are slightly different than we have seen before. The Itô's table for the Poisson process is shown in the table below. For the proof that $(\mathrm{d}\Gamma_t)^2 = \mathrm{d}\Gamma_t$, we refer to p.125 in Oosterlee and Grzelak (2020).

	$\mathrm{d}t$	$\mathrm{d}W_t$	$\mathrm{d}\Gamma_t$
$\mathrm{d}t$	0	0	0
$\mathrm{d}W_t$	0	dt	0
$\mathrm{d}\Gamma_t$	0	0	$\mathrm{d}\Gamma_t$

Table 2.2: Itô's multiplication table for the Poisson process.

Single asset MJD

Now let us consider the logarithmic process $X(t) = \log(S(t))$. By using the Itô's table and the Itô's lemma for Poisson process (see P.124 in Oosterlee and Grzelak (2020)), the dynamics of $X(t)$ can be derived as

$$\mathrm{d}X_t = \left(r - q - \kappa^J \lambda^J - \frac{1}{2}\sigma^2\right) \mathrm{d}t + \sigma \mathrm{d}W_t + N^J \mathrm{d}\Gamma_t.$$

By integrating, the analytic solution of the logarithmic process is

$$\begin{aligned} X_t - X_0 &= \left(r - q - \kappa^J \lambda^J - \frac{1}{2}\sigma^2\right)(t - 0) + \sigma\left(W_t - W_0\right) + \sum_{m=1}^{\Gamma_t} N_m^J \\ \Rightarrow X_t &= X_0 + \left(r - q - \kappa^J \lambda^J - \frac{1}{2}\sigma^2\right) t + \sigma W_t + \sum_{m=1}^{\Gamma_t} N_m^J \end{aligned} \tag{2.10}$$

where Γ_t is the number of jumps between the time interval $[0, t]$ and N_m^J is the jump size of the m-th jump. Then the exact solution of the SDE in Eq.(2.13) can be obtained

$$S_t = S_0 \exp\left(\left(r - q - \kappa^J \lambda^J - \frac{1}{2}\sigma^2\right) t + \sigma W_t\right) \exp\left(\sum_{m=1}^{\Gamma_t} N_m^J\right). \tag{2.11}$$

When dividing the time horizon into N time steps with $h = \frac{T}{N}$ the step size, we assume that the number of jumps between each time step is defined as $\Delta\Gamma_i = k$, for $i = 0, ..., N - 1$. Then

Eq.(2.10) can be discretized as follows

$$X_{i+1} = X_i + \left(r - q - \kappa^J \lambda^J - \frac{1}{2}\sigma^2\right) t + \sigma(W_{t_{i+1}} - W_{t_i}) + \sum_{m=1}^{k} N_m^J.$$

Since the jump size $N_m^J \overset{iid}{\sim} \mathcal{N}(\mu^J, (\sigma^J)^2)$, it implies that $\sum_{m=1}^{k} N_m^J \sim \mathcal{N}(k\mu^J, k(\sigma^J)^2)$. As a result, the MJD is simulated using the following expression

$$X_{i+1} = X_i + \left(r - q - \kappa^J \lambda^J - \frac{1}{2}\sigma^2\right) h + \sigma\sqrt{h}Z + k\mu^J + \sigma^J\sqrt{k}Z^J, \tag{2.12}$$

where $Z, Z^J \overset{iid}{\sim} \mathcal{N}(0,1)$.

Multi-dimensional MJD

Given a d dimensional underlying asset prices $\mathbf{S}(t) = (S_1(t), S_2(t), ..., S_d(t))$ generated by the multi dimensional MJD. Then the dynamic of this multi-dimensional asset is

$$\frac{\mathrm{d}S_{t,v}}{S_{t,v}} = \left(r - q_v - \lambda^J \kappa_v\right) \mathrm{d}t + \sigma_v \mathrm{d}W_{t,v} + \left(\mathrm{e}^{N_v^J} - 1\right) \mathrm{d}\Gamma_{t,v}, \quad v = 1, \ldots, d. \tag{2.13}$$

Since the diffusion part follows the multi-dimensional GBM model, we have the same setting as in Eq.(2.6). For the jump part, we assume that the components of $\mathbf{S}(t)$ have the same jump numbers with different jump sizes $\mathbf{N}^J = (N_1^J, ..., N_d^J) \in \mathbb{R}^d$. This jump size $\mathbf{N}^J$ is multivariate normal distributed, i.e., $\mathbf{N}^J \sim \mathcal{N}(\boldsymbol{\mu}^J, \Sigma^J)$ with $\boldsymbol{\mu}^J = (\mu_1^J, ..., \mu_d^J)$. Its covariance matrix is defined as $(\Sigma^J)_{vl} = \sigma_v^J \sigma_l^J \rho_{vl}^J$, for $v, l = 1, ..., d$ with $v \neq l$. Moreover, we assume the Cholesky decomposition of the variance matrix is $\Sigma^J = A^J (A^J)^T$. And just as the one dimensional case, $\kappa_v = \exp\left(\mu_v^J + \frac{(\sigma_v^J)^2}{2}\right) - 1$.

Suppose the number of jumps between each time step with step size $h = \frac{T}{N}$ is defined as $\Delta\Gamma_i = k$, for $i = 0, ..., N-1$. Then similarly to the single asset case, the logarithmic process $\mathbf{X}_{i+1}$ given $\mathbf{X}_i$ and $\Delta\Gamma_i = k$ can be simulated as

$$X_{i+1,v} = X_{i,v} + \left(r - q_v - \kappa_v^J \lambda^J - \frac{1}{2}\sigma_v^2\right) h + \sqrt{h}\sum_{l=1}^{d} A_{v,l} Z_{i,l} + k\mu_v^J + \sqrt{k}\sum_{l=1}^{d} A_{v,l}^J Z_{i,l}^J, \tag{2.14}$$

for $v = 1, ..., d$. Note the matrix A is the Cholesky decomposition of the covariance matrix Σ^S for the Wiener process and $Z_{i,l}, Z_{i,l}^J \overset{iid}{\sim} \mathcal{N}(0,1)$. The expression in Eq.(2.14) is a summation of two independent normal distributions. Therefore, $(\mathbf{X}_{i+1} | \mathbf{X}_i, \Delta\Gamma_i = k)$ is multivariate normal distributed with the mean $\boldsymbol{\mu}_{i+1}^{JD}$ defined as

$$\mu_{i+1,v}^{JD} = X_{i,v} + \left(r - q_v - \kappa_v^J \lambda^J - \frac{1}{2}\sigma_v^2\right) h + k\mu_v^J, \ v = 1, ..., d$$

and the variance $\Sigma^{JD} = h\Sigma^S + k\Sigma^J$.

2.2.3 Cox–Ingersoll–Ross (CIR)

Up to now, we have only considered constant volatility. In this subsection, we will introduce the Cox–Ingersoll–Ross(CIR) model as an example of stochastic volatility. The CIR process was introduced by Cox et al. (1985) and was first used for the evaluation of interest rates. The SDE of this variance process is

$$\mathrm{d}v(t) = \kappa(\bar{v} - v(t))\mathrm{d}t + \gamma\sqrt{v(t)}\mathrm{d}W_v(t), \tag{2.15}$$

where $\kappa > 0$ is the speed of mean reversion, i.e., the rate at which the variance process reverts to the long-term mean $\bar{v} \geq 0$. The term $\gamma > 0$ denotes the volatility of the variance process. This variance process ensures non-negative volatility, which is desirable for simulating the underlying asset prices. If the Feller condition $2\kappa\bar{v} \geq \gamma^2$ is satisfied, then it is guaranteed that the volatility $v(t)$ stays strictly positive (Feller, 1951). Else, the variance process may reach zero.

Definition 2.2.2 (CIR)**.** *Given a CIR Process $v_t = v(t)$ for $t > 0$. Let $\chi^2(\delta, \bar{\kappa}(t,s))$ denote a non-central chi-squared random variable with degrees of freedom δ and the non-centrality parameter $\bar{\kappa}(t,s)$. Then $v_t \mid v_s$ is distributed as*

$$v_t \mid v_s \sim \bar{c}(t,s)\chi^2(\delta, \bar{\kappa}(t,s)), \quad t > s > 0,$$

with

$$\bar{c}(t,s) = \frac{1}{4\kappa}\gamma^2\left(1 - \mathrm{e}^{-\kappa(t-s)}\right), \quad \delta = \frac{4\kappa\bar{v}}{\gamma^2}, \quad \bar{\kappa}(t,s) = \frac{4\kappa v(s)\mathrm{e}^{-\kappa(t-s)}}{\gamma^2\left(1 - \mathrm{e}^{-\kappa(t-s)}\right)}.$$

The mean and variance are

$$\mathbb{E}[\nu(t) \mid \mathcal{F}(s)] = \bar{c}(t,s)(\delta + \bar{\kappa}(t,s)) \tag{2.16}$$

$$\mathrm{Var}[\nu(t) \mid \mathcal{F}(s)] = \bar{c}^2(t,s)(2\delta + 4\bar{\kappa}(t,s)). \tag{2.17}$$

When the underlying asset price $S(t)$ is generated using the GBM with the CIR model as the variance process, i.e. $\sigma = \sqrt{v(t)}$ in Eq.(2.2), then we get **the Heston model**

$$\begin{cases} \frac{dS(t)}{S(t)} = (r-q)dt + \sqrt{\nu_t}dW^S(t) \\ \mathrm{d}\nu(t) = \kappa(\bar{\nu} - \nu(t))\mathrm{d}t + \gamma\sqrt{\nu(t)}\mathrm{d}W^\nu(t) \end{cases}, t \in [0,T].$$

We have utilized the Euler discretization to simulate the previous two models. Nevertheless, this discretization scheme does not apply to the CIR model as it leads to an unrealistic negative variance process, although the model itself ensures non-negative variances (Alfonsi, 2015). We will provide more details about the simulation of the Heston model in chapter 5.

2.3 Options

2.3.1 Basic definitions of options

An option is a contract between two parties that gives the option holder the right to sell or buy a certain amount of the underlying assets for a strike price within a period. Let us denote the strike price as K and the maturity of the option as T. The holder's exercise strategy is determined by the performance of the underlying assets. If the holder decides to exercise the option, the option seller must abide by the contract and trade in the assets. An option is a ***call option*** if the holder has the right to buy the underlying assets, and a ***put option*** if the holder has the right to sell.

When considering an ***European option***, the option holder can only exercise the option at the maturity, $t = T$. In contrast, an ***American option*** can be exercised at any moment, including the maturity, i.e., $t \in [t_0, T]$. A ***Bermudan option*** can be seen as the intermediate style between the European and American options. It provides the option holder with a finite number of opportunities to decide whether to exercise the option, so $t \in \{t_1, ..., t_n = T\}$. When $n \to \infty$ and the step size $\frac{T}{n} \to 0$, the Bermudan option approximates the American option.

The exercise strategy depends on the payoff of the particular option. The option holder will exercise the call option if the asset price $S(t)$ is higher than the strike price K at the exercise time t. Since the holder can immediately sell assets in the market, the profit for the holder is $S(t) - K$. Conversely, the call option is valueless when the strike is higher than the asset price because the assets can be purchased at a lower price. Therefore, the profit is 0. In conclusion, the payoff function $h(t, S(t))$ of a call option at a certain exercise time t is

$$h(t, S(t)) = h_t(S_t) = \max\{S(t) - K, 0\}. \tag{2.18}$$

In the case of a put option, the holder will utilize his right as the asset price $S(t) < K$ at exercise time t. Since the option seller must purchase the underlying assets at the strike price, the holder will benefit by $K - S(t)$. When $S(t) > K$, selling the assets on the financial market is a wiser choice for the option holder than exercising the option. Therefore the payoff function of a put option is defined as

$$h(t, S(t)) = h_t(S_t) = \max\{K - S(t), 0\}. \tag{2.19}$$

The $S(t) - K$ and $K - S(t)$ are also called the *intrinsic option value* of the call and put options, respectively.

Definition 2.3.1 (ITM,ATM,OTM)**.** *Given the asset price $S(t)$ and the strike price K for an option. If the intrinsic option value corresponding to this option at time t is positive, then this option is in-the-money (ITM). If the intrinsic option value is close to zero, then the option is called at-the-money (ATM). When the payoff function of the option is equal to zero, the option is out-the-money (OTM).*

As mentioned, the Bermudan option approximates the American option as the number of exercise dates goes to infinity. Let us consider a Bermudan option with N exercise dates. The Bermudan can be treated as a European option between two exercise opportunities. Therefore, the valuation of a Bermudan option can be seen as finding the best stopping time.

In the following sections, we will examine European option pricing, followed by high-dimensional American option pricing. Finally, we explain the fundamental concept of the regression-based Monte Carlo method to American option pricing. Please keep in mind that we are just looking at the risk-neutral probability space.

2.3.2 European option pricing

The Feynman-Kac theorem can be used for obtaining a closed formed solution of the European option pricing.

Theorem 2.3.1 (Feynman-Kac theorem)**.** *Given the money-savings account $M(t)$ with constant interest rate r which is modeled as $\mathrm{d}M(t) = rM(t)\mathrm{d}t$. Let the underlying asset price $S = S(t)$ defined by*

$$\mathrm{d}S(t) = \tilde{\mu}(t, S(t))\mathrm{d}t + \tilde{\sigma}(t, S(t))\mathrm{d}W^{\mathbb{Q}}(t),$$

with $\tilde{\mu}(t,S)$ and $\tilde{\sigma}(t,S)$ are the drift and the volatility, respectively. Suppose $V(t,S)$ is a differentiable function of time t and $S(t)$ that satisfies the following PDE:

$$\frac{\partial V}{\partial t} + \tilde{\mu}(t,S)\frac{\partial V}{\partial S} + \frac{1}{2}\tilde{\sigma}^2(t,S)\frac{\partial^2 V}{\partial S^2} - rV = 0,$$

with $V(T,S) = h(T,S)$ and $h(t,S)$ is the payoff function. Then the solution $V(t,S)$ under the risk neutral measure $\mathbb{Q}$ is given by:

$$V(t,S) = e^{-r(T-t)}\mathbb{E}^{\mathbb{Q}}\left[h(T,S)|\mathcal{F}_t\right],$$

for all $t < T$.

Proof. See Appendix A. □

Suppose the underlying asset is generated by GBM with a dividend rate $q = 0$. It means

$$\frac{\mathrm{d}S(t)}{S(t)} = r\mathrm{d}t + \sigma\mathrm{d}W_t, \text{ for } t \in [0,T].$$

And suppose that our trading strategy is to hold one option with value $V(t,S_t)$ and continuously trade in the stock to hold an amount of n shares with price S_t. Then the value of our portfolio is $\Pi(t,S_t) = V(t,S_t) - nS(t)$. It means $\mathrm{d}\Pi(t,S_t) = \mathrm{d}V(t,S_t) - n\mathrm{d}S(t)$. By using the Itô's lemma, we get the following dynamic for $V(t,S_t)$

$$\mathrm{d}V = \left(\frac{\partial V}{\partial t} + \frac{\partial V}{\partial S}rS + \frac{1}{2}\sigma^2 S^2\frac{\partial^2 V}{\partial S^2}\right)\mathrm{d}t + \frac{\partial V}{\partial S}\sigma\mathrm{d}W_t,$$

therefore,

$$\begin{aligned}\mathrm{d}\Pi &= \left(\frac{\partial V}{\partial t} + \frac{\partial V}{\partial S}rS + \frac{1}{2}S^2\sigma^2\frac{\partial^2 V}{\partial S^2}\right)\mathrm{d}t + \frac{\partial V}{\partial S}\sigma\mathrm{d}W_t - n\left[r\mathrm{d}t + \sigma\mathrm{d}W_t\right]S \\ &= \left(\frac{\partial V}{\partial t} + \left(\frac{\partial V}{\partial S} - n\right)rS + \frac{1}{2}\sigma^2 S^2\frac{\partial^2 V}{\partial S^2}\right)\mathrm{d}t + \left(\frac{\partial V}{\partial S} - n\right)\sigma\mathrm{d}W_t. \qquad (2.20)\end{aligned}$$

Let us assume that $\frac{\partial V}{\partial S} = n$. This assumption can be made because we want to hedge the risky position. Then Eq.(2.20) becomes

$$\mathrm{d}\Pi = \left(\frac{\partial V}{\partial t} + \frac{1}{2}S^2(t)\sigma^2\frac{\partial^2 V}{\partial S^2}\right)\mathrm{d}t. \qquad (2.21)$$

Moreover, the value of the portfolio should generate the same return as the money-saving account $M(t)$, as the growth speed has to be equivalent (Oosterlee and Grzelak, 2020). Let us assume that the interest rates r are constant, then the time value of money can be determined by using the compounded interest, which gives us:

$$\mathrm{d}M(t) = rM(t)\mathrm{d}t.$$

The solution of this equation is $M(t) = M_0\exp\left(\int_0^T r dt\right)$, with M_0 is the constant of integration. Then, $\Pi(t) = M(t)$ implies $\mathrm{d}\Pi = r\Pi\mathrm{d}t$. If we combine this with the equation above, we get the Black-Scholes PDE:

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2\frac{\partial^2 V}{\partial S^2}S^2(t) = r\Pi.$$

Note that $\Pi = V(t, S_t) - \frac{\partial V}{\partial S} S(t)$, then we get

$$\frac{\partial V}{\partial t} + \frac{1}{2} S^2(t)\sigma^2 \frac{\partial^2 V}{\partial S^2} = r(V(t, S_t) - \frac{\partial V}{\partial S} S(t)) \Rightarrow \frac{\partial V}{\partial t} + \frac{\partial V}{\partial S} rS(t) + \frac{1}{2}\sigma^2 S^2(t)\frac{\partial^2 V}{\partial S^2} - rV = 0.$$

It means the Feynman-Kac theorem applies to the GBM model. The Feynman-Kac theorem can be adjusted for the MJD models. See chapter 5 of Oosterlee and Grzelak (2020). The verification for the Heston model is similar to the procedure of the GBM model. For the derivation for the Heston model, we refer to section 8.2.1 of Oosterlee and Grzelak (2020).

2.3.3 American option pricing

Let $(\Omega, \mathcal{F}, \mathbb{Q})$ be a probability space with a time horizon $[0, T]$, where $\mathcal{F} = \{\mathcal{F}_t | 0 \leq t \leq T\}$ is the filtration and $\mathbb{Q}$ the risk-neutral measure. Let us define the moving saving account with constant interest rate r as

$$M_t = \exp\left(\int_0^t r dt\right)$$

and the discount factor $D_{s,t} = \frac{M_s}{M_t} = e^{-r(s-t)}$. Assume the underlying asset prices $\mathbf{S}_t \in \mathbb{R}^d$ with $d \in \mathbb{Z}^+$ follow one of the models introduced in the previous section, which is a Markov process. Moreover, we define the payoff at time t as $h_t(\mathbf{S}_t) = h(t, \mathbf{S}_t)$. Since an American option can be exercised at any time before the expiration time T, the problem of American option pricing is to find the best stopping time such that it optimizes the expected discounted payoff:

$$V_0 = V(\mathbf{S}_{t_0}) = \sup_{\tau \in \mathcal{T}} e^{-r(\tau - t_0)} \mathbb{E}[h_\tau(\mathbf{S}_\tau)] = D_{t_0,\tau^*} \mathbb{E}[h_{\tau^*}(\mathbf{S}_{\tau^*})], \text{ with } t_0 = 0 \tag{2.22}$$

where $\mathcal{T}$ is the set of all stopping times and $\tau^* \in \mathcal{T}$ is the optimal stopping time. We call a random variable τ_s as a stopping time if $\{\tau_s \leq t\}$ is $\mathcal{F}_t$-measurable for all $t \geq 0$.

In practice, a Bermudan option can be used to approximate the American option. As mentioned before, a Bermudan option gives the holder the right to exercise at discrete time steps. If we subdivide the time horizon $[0, T]$ as $0 = t_0 < t_1 < \cdots < t_N = T$ with step size $h = \frac{T}{N}$, then the Bermudan option can be exercised at any $t \in \{t_0, t_1, \cdots, t_N\}$. From now, we will denote t_i as i to simplify the notations with $i \in \{1, ..., N\}$.

Regression-based Monte Carlo approach

The basic idea of the regression-based Monte Carlo approach consists of the following steps:

1. Simulate M paths of underlying asset prices $\mathbf{S} = (\mathbf{S}_0, ..., \mathbf{S}_N)$ over the time horizon. This can be done via the simulation process stated in the previous section.
2. Evaluate the discounted cashflow of the American option on each simulated path using regression.
3. Average the obtained cashflows.

The main focus of the second step is to find the optimal exercise strategy with the corresponding discounted cashflow. To accomplish this, we start from the maturity and perform a backward recursive process. Let $\mathbf{S}_i$ be an arbitrary Monte Carlo path at timestep i, simulated in step 1. At every time step $i \in \{N-1, ..., 0\}$, the option holder has two choices:

1. Exercise the option, which means the option value is equivalent to the payoff at the current timestep, i.e. $V_i(\mathbf{S}_i) = h_i(\mathbf{S}_i)$,

2. Hold the option to the next time step, then the option can be considered as newly issued at timestep i. Therefore, the continuation value is defined as

$$Q_i(\mathbf{S}_i) = \mathbb{E}[D_{i,i+1}V_{i+1}|\mathcal{F}_i] = \mathbb{E}[D_{i,i+1}V_{i+1}|\mathbf{S}_i]. \tag{2.23}$$

The second equality is due to the Markov property of the underlying asset prices.

The holder's decision is based on which choice can maximize the profit. As a result, the option value at time step i is

$$V_i(\mathbf{S}_i) = \max\{h_i(\mathbf{S}_t), Q_i(\mathbf{S}_i))\}, i = 0, ..., N-1.$$

Note $V_N = h_T(\mathbf{S}(T))$, and together with the equation above, it forms the dynamic programming principle (DPP) for solving the optimal stopping problem stated in Eq.(2.22). To obtain the true continuation value is as hard as solving the optimal stopping time problem of the American option itself. The regression-based approaches estimate the continuation value using regression. The DPP stated above uses the estimation of the continuation value directly in computing the option value, which can result in less accurate results during the backward recursion process (Hu and Zastawniak, 2020). To limit the influence of the estimated continuation value on the valuation of the option, we will reformulate this DPP by using the stopping time.

The optimal stopping time τ^* is the first time from the start such that the exercise payoff is greater or equal to the continuation value

$$\tau_0^* = \inf\{t \geq t_0 : h_i(\mathbf{S}_t) \geq Q_t(\mathbf{S}_t)\}.$$

Then we can interpret the options that were not exercised before current timestep i as newly issued. Thus, $\tau_i^* = \inf\{t \geq t_i : h_i(\mathbf{S}_t) \geq Q_t(\mathbf{S}_t)\}$. The DPP in stopping time form is then as follows:

$$\begin{cases} \tau_N^* = N = T \\ \tau_i^* = \begin{cases} i, \text{ if } h_i(\mathbf{S}_i) \geq Q_i(\mathbf{S}_i) \\ \tau_{i+1}^*, \text{ otherwise} \end{cases} \text{, for } i = N-1, N-2, ..., 0 \end{cases}. \tag{2.24}$$

The estimation of the continuation values is the most essential step in the DPP. There are two ways to estimate $Q_i(\mathbf{S}_i)$ in Eq.(2.23) using regression (Glasserman and Yu, 2004):

1. Regression now: this method approximates $Q_i(\mathbf{S}_i)$ directly. Given a finite number of m basis functions $\boldsymbol{\phi}(\mathbf{x}) = [\phi_1(\mathbf{x}), \cdots, \phi_m(\mathbf{x})]^T$ with $\mathbf{x} \in \mathbb{R}^d$, the estimation of $Q_i(\mathbf{S}_i)$ is determined by

$$\hat{Q}_i(\mathbf{S}_i) = \sum_{j=1}^{m} \beta_{i,j}\phi_j(\mathbf{S}_i),$$

where the coefficients $\beta_{i,j}$ are the solution of the following ordinary least squares

$$\min_{\boldsymbol{\beta}_i} \|Q_i(\mathbf{S}_i) - \hat{Q}_i(\mathbf{S}_i)\|^2,$$

with $Q_i(\mathbf{S}_i)$ the cashflows. For an arbitrary path $l \in \{1, ..., M\}$, its cashflow is defined as $Q_i^l(\mathbf{S}_i^l) = D_{i,\tau_{i+1}^*} h_{\tau_{i+1}^*}(\mathbf{S}_{\tau_{i+1,l}^*}^l)$. The Least-Square Monte Carlo (LSMC) approach (Longstaff and Schwartz, 2001), widely used for the American option pricing, is an example of the regression now method.

2. Regression later: this method first approximates V_{i+1} by applying regression at time step $i+1$, i.e.

$$\hat{V}_{i+1}(\mathbf{S}_{i+1}) = \sum_{j=1}^{m} \beta_{i,j}\phi_j(\mathbf{S}_{i+1})$$

and then calculate the continuation value as

$$\hat{Q}_i(\mathbf{S}_i) = D_{i,i+1}\sum_{j}^{m} \beta_{i,j}\mathbb{E}[\phi_j(\mathbf{S}_{i+1})|\mathbf{S}_i].$$

The coefficients are obtained by solving

$$\min_{\boldsymbol{\beta}_i} \|V_{i+1}(\mathbf{S}_{i+1}) - \hat{V}_{i+1}(\mathbf{S}_{i+1})\|^2,$$

where $V^l_{i+1}(\mathbf{S}^l_{i+1}) = h_{\tau^*_{i+1}}(\mathbf{S}^l_{\tau^*_{i+1,l}})$ for $l = 1, ..., M$.

The output of this backward recursion process is the optimal stopping times for the American options with simulated sample asset prices. Then the value of the option using the DPP in Eq.(2.24) is given by

$$\hat{V}_0 = \frac{1}{M}\sum_{j=1}^{M} D_{0,\tau^*_j} h_{\tau^*_j}(\mathbf{S}^j(\tau^*_j)), \tag{2.25}$$

for $\mathbf{S}^j$ the j-th MC sample path and τ^*_j the corresponding best exercise time.

As we can see, the regression-based methods mainly depend on the selected basis functions. Especially for the regression later method, the conditional expectations of the chosen basis functions must have analytical expressions for estimating the continuation values. Hu and Zastawniak (2020) propose using the kernel ridge regression instead of the linear regression to avoid selecting the basis functions. In the next chapter, we will derive how to estimate the continuation value by utilizing kernel ridge regression.

Chapter 3

Methodology

In this chapter, we will present the methodology of kernel ridge regression-based American option pricing proposed by Hu and Zastawniak (2020). We will begin with some fundamental knowledge about kernel ridge regression before introducing the kernel ridge regression now (NKRR) and kernel ridge regression later (LKRR) methods. The NKRR and LKRR will be examined in this thesis for high-dimensional American option pricing under various asset price models.

3.1 Kernel Ridge Regression (KRR)

Given a data set $\mathcal{D} = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \cdots, (\mathbf{x}_n, y_n)\}$ with $\mathbf{x}_i \in \mathbb{R}^d$ the input vector and $y_i \in \mathbb{R}$ the corresponding output. Let $\boldsymbol{\phi}(\mathbf{x}) = [\phi_1(\mathbf{x}), \cdots, \phi_m(\mathbf{x})]^T$ be a feature vector associated with the input $\mathbf{x}$. Note for all $j \in \{1, ..., m\}$, the function $\phi_j(\mathbf{x})$ is the selected basis function. There are various possibilities for the basis functions, e.g., the polynomial basis function and the Hermite functions. The Hermite function is defined as $H_{n+1}(x) = xH_n(x) + H_n^{'}(x)$, where $H_n^{'}(x)$ denotes the derivative of $H_n(x)$ for $n \in \mathbb{N}_0$. Moreover, $H_0(x) = 1$.

Now let us consider the following linear regression model (Bishop, 2007, Chapter 3):

$$\mathbf{y} = \Phi\boldsymbol{\beta} + \epsilon,$$

where $\Phi \in \mathbb{R}^{n\times m}$ the feature matrix with $\Phi_{ij} = \phi_j(\mathbf{x_i})$ for $j \in \{1, ..., m\}$ and $i \in \{1, ..., n\}$. The output vector is denoted as $\mathbf{y} \in \mathbb{R}^n$, $\boldsymbol{\beta} \in \mathbb{R}^m$ is the coefficients vector, and $\epsilon \in \mathbb{R}^n$ is the residual which is an independent random noise with zero mean. For the linear regression problem, the goal is to find $\boldsymbol{\beta}$ such that the residual sum of squares(RSS) is minimized:

$$\min_{\boldsymbol{\beta}\in\mathbb{R}^m} \|\mathbf{y} - \Phi\boldsymbol{\beta}\|^2 = \min_{\boldsymbol{\beta}\in\mathbb{R}^m} (\mathbf{y} - \Phi\boldsymbol{\beta})^T(\mathbf{y} - \Phi\boldsymbol{\beta}).$$

The solution to this minimization problem is

$$\hat{\boldsymbol{\beta}} = (\Phi^T\Phi)^{-1}\Phi^T\mathbf{y} \tag{3.1}$$

since

$$\frac{\partial(\mathbf{y} - \Phi\boldsymbol{\beta})^T(\mathbf{y} - \Phi\boldsymbol{\beta})}{\partial\boldsymbol{\beta}} = 0 \;\Rightarrow\; (\Phi^T\Phi)\boldsymbol{\beta} = \Phi^T\mathbf{y}$$

and the second partial derivative

$$\frac{\partial^2(\mathbf{y} - \Phi\boldsymbol{\beta})^T(\mathbf{y} - \Phi\boldsymbol{\beta})}{\partial\boldsymbol{\beta}^2} = \Phi^T\Phi$$

is positive definite.

To overcome overfitting or prevent the problem becomes ill-posed, a ridge penalty term $\lambda\boldsymbol{\beta}^T\boldsymbol{\beta}$, with $\lambda > 0$, can be added to the linear regression, which results in a ***ridge regression*** problem:

$$\min_{\boldsymbol{\beta}\in\mathbb{R}^m} R(\boldsymbol{\beta}) = \min_{\boldsymbol{\beta}\in\mathbb{R}^m} (\mathbf{y} - \Phi\boldsymbol{\beta})^T(\mathbf{y} - \Phi\boldsymbol{\beta}) + \lambda\boldsymbol{\beta}^T\boldsymbol{\beta}. \tag{3.2}$$

By working out the brackets, we obtain $R(\boldsymbol{\beta}) = \mathbf{y}^T\mathbf{y} - 2\Phi^T\mathbf{y}\beta + \boldsymbol{\beta}^T(\Phi^T\Phi + \lambda I_m)\boldsymbol{\beta}$ and the ordinary least squared(OLS) estimate is obtained by solving

$$\frac{\partial R(\boldsymbol{\beta})}{\partial\boldsymbol{\beta}} = -2\Phi^T\mathbf{y} + 2(\Phi^T\Phi + \lambda I_m)\boldsymbol{\beta} = 0 \;\Rightarrow\; (\Phi^T\Phi + \lambda I_m)\boldsymbol{\beta} = \Phi^T\mathbf{y}$$

which gives us

$$\hat{\boldsymbol{\beta}} = (\Phi^T\Phi + \lambda I_m)^{-1}\Phi^T\mathbf{y}. \tag{3.3}$$

Since $\frac{\partial^2 R(\boldsymbol{\beta})}{\partial\boldsymbol{\beta}^2} = \Phi^T\Phi + \lambda I_m$ is positive definite, the above expression is the estimate of the solution to the optimization problem in Eq.(3.2). The linear regression model satisfies the so-called learning subspace property (LSP) (Kung, 2014), therefore $\hat{\boldsymbol{\beta}} \in span\{\Phi\}$. By definition, there is some $\hat{\boldsymbol{\alpha}} \in \mathbf{R}^n$ such that $\hat{\boldsymbol{\beta}} = \Phi^T\hat{\boldsymbol{\alpha}}$.

Remark. *Let $A \in \mathbb{R}^{m\times n}$, $B \in \mathbb{R}^{n\times m}$ and $I_m \in \mathbb{R}^{m\times m}$ the identity matrix, then*

$$B(I_m + AB)^{-1} = (I_n + BA)^{-1}B.$$

Proof. $B(I_m + AB) = (I_n + BA)B \Rightarrow (I_n + BA)^{-1}B(I_m + AB) = B, so$
$(I_n + BA)^{-1}B = B(I_m + AB)^{-1}$. □

The expression for $\hat{\boldsymbol{\alpha}}$ can be derived using this remark, since

$$\begin{aligned}(\Phi^T\Phi + \lambda I_m)^{-1}\Phi^T &= \left(\frac{1}{\lambda}\Phi^T\Phi + I_m\right)^{-1}\frac{1}{\lambda}\Phi^T = \frac{1}{\lambda}\Phi^T\left(\frac{1}{\lambda}\Phi\Phi^T + I_n\right)^{-1}\\ &= \Phi^T(\Phi\Phi^T + \lambda I_n)^{-1}.\end{aligned}$$

The Eq.(3.3) can be rewritten as $\hat{\boldsymbol{\beta}} = \Phi^T(\Phi\Phi^T + \lambda I_n)^{-1}\mathbf{y}$ and $\hat{\boldsymbol{\beta}} = \Phi^T\hat{\boldsymbol{\alpha}}$, we obtain

$$\hat{\boldsymbol{\alpha}} = (\Phi\Phi^T + \lambda I_n)^{-1}\mathbf{y}. \tag{3.4}$$

Now, let us define a kernel function $\mathcal{K} : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$, such that $\mathcal{K}(\mathbf{x}_i, \mathbf{x}_j) = \boldsymbol{\phi}^T(\mathbf{x}_i)\boldsymbol{\phi}(\mathbf{x}_j)$ for all $i, j = 1, ..., n$. Then the Gram matrix $\mathbf{K} \in \mathbb{R}^{n\times n}$ is given by:

$$\mathbf{K} = \begin{bmatrix}\mathcal{K}(\mathbf{x}_1, \mathbf{x}_1) & \cdots & \mathcal{K}(\mathbf{x}_1, \mathbf{x}_n)\\ \vdots & \ddots & \vdots\\ \mathcal{K}(\mathbf{x}_n, \mathbf{x}_1) & \cdots & \mathcal{K}(\mathbf{x}_n, \mathbf{x}_n)\end{bmatrix}.$$

Note $(\Phi\Phi^T)_{ij} = \boldsymbol{\phi}^T(\mathbf{x}_i)\boldsymbol{\phi}(\mathbf{x}_j)$, which means that $\mathbf{K} = \Phi\Phi^T$ and $\hat{\boldsymbol{\alpha}} = (\mathbf{K} + \lambda I_n)^{-1}\mathbf{y}$. The corresponding prediction $\hat{\mathbf{y}}$ is then as follows:

$$\hat{\mathbf{y}} = \Phi\hat{\boldsymbol{\beta}} = \Phi\Phi^T\hat{\boldsymbol{\alpha}} = \mathbf{K}\hat{\boldsymbol{\alpha}},$$

with

$$\hat{y}_i = \sum_{m=1}^{n}\hat{\alpha}_m\mathcal{K}(\mathbf{x}_m, \mathbf{x}_i) \tag{3.5}$$

for $i \in \{1, ..., n\}$.

There are various options for kernel functions. In this thesis, we will use the Gaussian kernel as suggested by Hu and Zastawniak (2020), which is given by:

$$\mathcal{K}(\mathbf{x}, \mathbf{y}) = e^{-\frac{1}{C}(\mathbf{x}-\mathbf{y})^T(\mathbf{x}-\mathbf{y})}, C > 0. \tag{3.6}$$

This kernel is chosen because it can cover any continuous basis function (Exterkate, 2013).

3.2 KRR-based American option pricing

Now that we have gained more insight into kernel ridge regression(KRR), we will apply this method to estimate the continuation values. Suppose M Monte Carlo paths of the asset prices are simulated and let $\mathbf{S}_t \in \mathbb{R}^d$ be an arbitrary simulated sample path at time $t \in [0, T]$. Since this time horizon is divided into N equal-sized time steps, we denote $\mathbf{S}_i \in \mathbb{R}^d$ as the asset price at time step i. Recall that the continuation value at time step i is given by

$$Q_i(\mathbf{S}_i) = \mathbb{E}[D_{i,i+1}V_{i+1}(\mathbf{S}_{i+1})|\mathbf{S}_i], \text{ for } i = N-1, ..., 1. \tag{3.7}$$

The NKRR method estimates the continuation values directly by applying KRR at time step t_i. The regression later method LKRR first estimates $V_{i+1}(\mathbf{S}_{i+1})$ by regression on the data at time step t_{i+1} and then substitutes it into the above equation to compute the desired continuation value at t_i. We will explain those two methods in detail in the following two subsections.

3.2.1 Regression-Now: NKRR

At maturity, the continuation value is equal to zero, which implies that the cashflow is equal to its payoff $V_N(\mathbf{S}_T) = h_T(\mathbf{S}_T)$. Since NKRR applies regression directly for the estimation of the continuation value $Q_i(\mathbf{S}_i)$, it gives us the following by using Eq.(3.5) for $i = N-1, .., 1$:

$$\hat{Q}_i(\mathbf{S}_i) = D_{i,i+1} \sum_{m=1}^{M} \hat{\alpha}_{i,m} \mathcal{K}(\mathbf{S}_i^{(m)}, \mathbf{S}_i), \tag{3.8}$$

with $\mathbf{S}_i$ an arbitrary asset price, $\mathbf{S}_i^{(m)}$ the m-th simulated Monte Carlo path and $\hat{\alpha}_{i,m}$ the m-th coefficient at time step i. The coefficients vector $\hat{\boldsymbol{\alpha}}_i \in \mathbb{R}^M$ can be calculated by using Eq.(3.4) as

$$\hat{\boldsymbol{\alpha}}_i = (\mathbf{K} + \lambda I_M)^{-1} \boldsymbol{Q}_i(\mathbf{S}_i),$$

with $\mathbf{K} \in \mathbb{R}^{M \times M}$ is the Gram matrix of the Gaussian kernel. The vector $\boldsymbol{Q}_i(\mathbf{S}_i) \in \mathbb{R}^M$ is the discounted cashflow. According to the framework of regression now method using KRR, it is defined as

$$Q_i^l(\mathbf{S}_i) = D_{i,\tau^*_{i+1,l}} h_{\tau^*_{i+1,l}}(\mathbf{S}^l_{\tau^*_{i+1,l}}), \ l = 1, ..., M, \tag{3.9}$$

where $\tau^*_{i+1,l}$ is as in Eq.(2.24). The discounted cashflow $Q_i^l(\mathbf{S}_i)$ can also be interpreted as the value of a newly issued option at the time step i where the optimal exercise strategy is $\tau^*_{i+1,l}$.

3.2.2 Regression-Later: LKRR

The LKRR method has two differences in comparison with the NKRR. Instead of $\mathbf{S}_i$, the logarithmic process $\mathbf{X}_i = \ln(\mathbf{S}_i)$ is used for the regression. Moreover, we apply regression on the asset prices at time step $i+1$ for computing the continuation values. It means for $i = N-1, ..., 1$

$$\hat{V}_{i+1}(\mathbf{X}_{i+1}) = \sum_{m=1}^{M} \hat{\alpha}_{i+1,m} \mathcal{K}(\mathbf{X}_{i+1}^{(m)}, \mathbf{X}_{i+1}), \tag{3.10}$$

where $\hat{\boldsymbol{\alpha}}_{i+1} = (\mathbf{K} + \lambda I_M)^{-1} \mathbf{V}_{i+1}(\mathbf{X}_{i+1})$ using Eq.(3.4). The term $\mathbf{V}_{i+1}$ is the payoff of optimally exercising the option. Despite the use of the log prices, the payoffs remain the same. Thus,

$$V_{i+1}^{l}(\mathbf{S}_{i+1}^{l}) = h_{\tau_{i+1}^*}(\mathbf{S}_{\tau_{i+1,l}^*}^{l}), l = 1, ..., M.$$

Then the estimation of the continuation value of any underlying asset $\mathbf{X}_i$ is given by:

$$\begin{aligned} \hat{Q}_i(\mathbf{X}_i) &= D_{i,i+1} \mathbb{E}\left[\sum_{m=1}^{M} \hat{\alpha}_{i+1,m} \mathcal{K}(\mathbf{X}_{i+1}, \mathbf{X}_{i+1}^{(m)}) | \mathbf{X}_i \right] \\ &= D_{i,i+1} \sum_{m=1}^{M} \hat{\alpha}_{i+1,m} \mathbb{E}\left[\mathcal{K}(\mathbf{X}_{i+1}, \mathbf{X}_{i+1}^{(m)}) | \mathbf{X}_i \right]. \end{aligned} \tag{3.11}$$

The conditional expectations of the Gaussian kernel function depend on the model from which the asset prices are generated. In this thesis, we are considering process $\mathbf{S}_i$ that follows the multidimensional GBM, MJD, or Heston models. The evaluation of the Heston model will be done in a later Chapter. In the previous chapter, we established that logarithmic processes are both multivariate normal under the former two models. Then the following lemma holds.

Lemma 3.2.1. *(Hu and Zastawniak, 2020) Suppose* $\mathbf{X} \sim \mathcal{N}(\boldsymbol{\mu}, \Sigma)$*, then*

$$\mathbb{E}[e^{-\frac{1}{C}\mathbf{X}^T\mathbf{X}}] = \frac{1}{|\frac{2}{C}\Sigma + I|^{\frac{1}{2}}} e^{-\frac{1}{C}\boldsymbol{\mu}^T(\frac{2}{C}\Sigma + I)^{-1}\boldsymbol{\mu}},$$

with I *the identity matrix and* $|\cdot|$ *is the determinant function.*

Proof. See Appendix B. □

LKRR under GBM

Let $\mathbf{X}_{i+1}$ be an arbitrary logarithmic asset price path at timestep $i+1$ simulated using the multidimensional GBM model and $\mathbf{X}_{i+1}^{(m)}$ be the m-th simulated MC path. By using Eq.(3.6), the Gaussian kernel is

$$\mathcal{K}\left(\mathbf{X}_{i+1}, \mathbf{X}_{i+1}^{(m)}\right) = e^{-\frac{1}{C}\left(\mathbf{X}_{i+1} - \mathbf{X}_{i+1}^{(m)}\right)^T \left(\mathbf{X}_{i+1} - \mathbf{X}_{i+1}^{(m)}\right)} \text{ with } C > 0$$

and the logarithmic process under GBM is normal distributed, $\mathbf{X}_{i+1}|\mathbf{X}_i \sim \mathcal{N}(\boldsymbol{\mu}_{i+1}, h\Sigma^{\mathbf{S}})$ with $\boldsymbol{\mu}_{i+1} \in \mathbb{R}^d$ and $\Sigma^{\mathbf{S}} \in \mathbb{R}^{d \times d}$ are defined as in Eq.(2.8). It implies that

$$\left(\mathbf{X}_{i+1} - \mathbf{X}_{i+1}^{(m)}\right) | \mathbf{X}_i \sim \mathcal{N}(\boldsymbol{\mu}_{i+1} - \mathbf{X}_{i+1}^{(m)}, h\Sigma^{\mathbf{S}}).$$

By applying Lemma 3.2.1, we get the following expression for the continuation value using LKRR with the Gaussian kernel

$$\begin{aligned}\hat{Q}_i(\mathbf{X}_i) &= D_{i,i+1}\sum_{m=1}^{M}\hat{\alpha}_{i+1,m}\mathbb{E}\left[\mathcal{K}(\mathbf{X}_{i+1},\mathbf{X}_{i+1}^{(m)})|\mathbf{X}_i\right]\\ &= D_{i,i+1}\sum_{m=1}^{M}\hat{\alpha}_{i+1,m}\frac{1}{|\frac{2}{C}h\Sigma^{\mathbf{S}}+I_d|^{\frac{1}{2}}}e^{-\frac{1}{C}\left(\boldsymbol{\mu}_{i+1}-\mathbf{X}_{i+1}^{(m)}\right)^T(\frac{2}{C}h\Sigma^{\mathbf{S}}+I_d)^{-1}\left(\boldsymbol{\mu}_{i+1}-\mathbf{X}_{i+1}^{(m)}\right)}\\ &= \frac{D_{i,i+1}}{|\frac{2}{C}h\Sigma^{\mathbf{S}}+I_d|^{\frac{1}{2}}}\sum_{m=1}^{M}\hat{\alpha}_{i+1,m}e^{-\frac{1}{C}\left(\boldsymbol{\mu}_{i+1}-\mathbf{X}_{i+1}^{(m)}\right)^T(\frac{2}{C}h\Sigma^{\mathbf{S}}+I_d)^{-1}\left(\boldsymbol{\mu}_{i+1}-\mathbf{X}_{i+1}^{(m)}\right)}.\end{aligned}\tag{3.12}$$

LKRR under MJD

As derived in subsection 2.2.2, the logarithmic process $\mathbf{X}_{i+1}$ given $\mathbf{X}_i$ and the number of jumps $\Delta\Gamma_i = k$ within the time interval $(i, i+1]$ is normal distributed with mean $\boldsymbol{\mu}_{i+1}^{JD}$ and variance Σ^{JD}. Note the number of jumps is Poisson distributed, therefore the expectation of the Gaussian kernel under the multi-assets MJD is

$$\begin{aligned}\mathbb{E}\left[\mathcal{K}(\mathbf{X}_{i+1},\mathbf{X}_{i+1}^{(m)})|\mathbf{X}_i\right] &= \mathbb{E}\left[\mathbb{E}\left[\mathcal{K}(\mathbf{X}_{i+1},\mathbf{X}_{i+1}^{(m)})|\mathbf{X}_i,\Delta\Gamma_i=k\right]\right]\\ &= \sum_{k=0}^{\infty}\mathbb{E}\left[\mathcal{K}(\mathbf{X}_{i+1},\mathbf{X}_{i+1}^{(m)})|\mathbf{X}_i,\Delta\Gamma_i=k\right]\mathbb{P}(\Delta\Gamma_i=k)\\ &= \sum_{k=0}^{\infty}\frac{\left(\lambda^J h\right)^k \mathrm{e}^{-\lambda^J h}}{k!}\mathbb{E}\left[\mathcal{K}(\mathbf{X}_{i+1},\mathbf{X}_{i+1}^{(m)})|\mathbf{X}_i,\Delta\Gamma_i=k\right].\end{aligned}\tag{3.13}$$

By substituting the above expression into Eq.(3.11) and then applying Lemma 3.2.1, the estimate of the continuation value can be obtained:

$$\begin{aligned}\widehat{Q}_i\left(\mathbf{X}_i\right) =& D_{i,i+1}\sum_{m=1}^{M}\widehat{\alpha}_{i+1,m}E\left[\mathcal{K}\left(\mathbf{X}_{i+1}^{(m)},\mathbf{X}_{i+1}\right)\mid\mathbf{X}_i\right]\\ =& D_{i,i+1}\sum_{m=1}^{M}\widehat{\alpha}_{i+1,m}\sum_{k=0}^{\infty}\frac{\left(\lambda^J h\right)^k \mathrm{e}^{-\lambda^J h}}{k!}\mathbb{E}\left[\mathcal{K}(\mathbf{X}_{i+1},\mathbf{X}_{i+1}^{(m)})|\mathbf{X}_i,\Delta\Gamma_i=k\right]\\ =& D_{i,i+1}\sum_{k=0}^{\infty}\frac{\mathrm{e}^{-\lambda^J h}(\lambda^J h)^k}{k!}\sum_{m=1}^{M}\widehat{\alpha}_{i+1,m}\mathbb{E}\left[\mathcal{K}\left(\mathbf{X}_{i+1}^{(m)},\mathbf{X}_{i+1}\right)\mid\mathbf{X}_i,\Delta\Gamma_i=k\right]\\ =& D_{i,i+1}\sum_{k=0}^{\infty}\frac{\mathrm{e}^{-\lambda^J h}(\lambda^J h)^k}{k!}\sum_{m=1}^{M}\widehat{\alpha}_{i+1,m}\mathrm{e}^{-\frac{1}{C}\left(\boldsymbol{\mu}_{i+1}^{JD}-\mathbf{X}_{i+1}^{(m)}\right)^{\top}(\frac{2}{C}\Sigma^{JD}+\mathrm{I})^{-1}\left(\boldsymbol{\mu}_{i+1}^{JD}-\mathbf{X}_{i+1}^{(m)}\right)}.\end{aligned}\tag{3.14}$$

Hu and Zastawniak (2020) suggest truncating the infinite summation for $k > 2$, as this truncation still provides sufficient accuracy. We will adapt this truncation in our numerical experiments in the next chapter.

3.2.3 Bundling

Since Monte Carlo techniques necessitate the simulation of large samples, the regression phase entails computing the inverse of a large-scale matrix to compute the regression coefficients. Moreover, regression has to be applied at each time step. As a result, the computation time

will increase with the number of exercise opportunities. In general, the computational cost of regression-based Monte Carlo also increases with the dimension, which is also a drawback of the LSMC approach proposed by Longstaff and Schwartz (2001). The NKRR and LKRR also suffer from a high computational effort if KRR is applied to the whole simulated data for high dimensional option pricing.

To improve computational efficiency, Hu and Zastawniak (2020) proposes utilizing the bundling technique for NKRR and LKRR. The main idea of this hybrid method is to partition the data set into P non-overlapping bundles and then apply KRR to data in the same bundle. This application strategy will reduce the time complexity of regression from $O(M^3)$ to $O(M^3/P^2)$ (Hu and Zastawniak, 2020). As proven in Jain and Oosterlee (2015), local regression using bundling provides more accurate pricing results than global regression. It means KRR with bundling will also improve the pricing accuracy with a reasonable P. There are several approaches for bundling, such as K mean clustering. In this thesis, we will use the payoff at each time step as bundling references. In other words, options with similar payoffs are divided into the same bundle. Moreover, we only use In the Money (ITM) paths in each bundle for the regression to reduce computational costs further.

Because the data used for regression now and regression later procedures differ, so do the bundling references. Suppose the current step is i for $i \in \{N-1, ..., 1\}$. For the regression now methods, the payoff at $i-1$ is used as a bundling reference to partition the underlying asset prices at $i-1$ into P non-overlapping bundles. Then we perform the regression on the paths at i that belong to the same bundle. The regression later approaches apply the regression to data in the same bundle at $i+1$, where the payoff at i serves as a bundling reference.

3.2.4 Hyperparameter optimization

Three parameters need to be determined from the data: the ridge parameter λ, the Gaussian kernel parameter C, and the number of bundles P. Those hyperparameters influence the performance of the NKRR and LKRR methods. The former two hyperparameters are related to the KRR. The ridge parameter λ shrinks the singular values (van Wieringen, 2015). Moreover, the Gaussian kernel parameter C is the scale parameter that determines the prediction's smoothness and depends on the dimension of the input data (Exterkate, 2013). As for both methods, we perform local regression at the bundle level. It is reasonable that the number of paths M/P belonging to one bundle should not be too small for regression. Otherwise, it will lead to poor pricing accuracy(Jain and Oosterlee, 2015). We aim to avoid hyperparameter optimization at each time step to keep the computing cost to a minimum. Our fundamental goal is to find a set of hyperparameters that optimizes the pricing result. A reference price V^{ref} is required for the hyperparameter tuning process, which might be the estimated price using LSMC or the true price if it is known.

The most straightforward hyperparameter tuning technique is the grid search. The fundamental idea is to test every possible combination of the sets of specified hyperparameters and select the one that produces the best results. However, this approach suffers from the curse of dimensionality. As the number of hyperparameters that need to be tuned increases, the number of combinations becomes large. This results in expensive computational costs and can become impractical. Since we only have three hyperparameters, this approach is feasible in our case. We first apply this approach to low-dimensional options and then find the hyperparameter set, which makes the relative error between the estimated price (using NKRR/LKRR) and V^{ref} below 0.5%. When this stopping criterion is satisfied, we evaluate whether the found hyperparameters are also suitable for high-dimensional options. If not, we return to the previous step

and test other sets of hyperparameters. The grid area for search is crucial for this approach. To limit the running time of hyperparameter tuning, we will only use this approach if some prior information about the hyperparameters is known.

Hu and Zastawniak (2020) suggest calibrating the Gaussian kernel hyperparameters C in the following way. We consider C as $C = C_0$ with $C_0 > 0$ is a constant. The cross-validation strategy is as follows

1. Choose the tolerance value ϵ and the maximum number of iterations.
2. Find C_0 such that the mean squared error(MSE) of V^{ref} and the estimated price is below the tolerance.
3. Validate C_0 on options of higher dimension(e.g. 100). If the MSE is smaller than ϵ, then $C = C_0$ is found. Else, we have to repeat steps 2 and 3.

If C_0 cannot be found, then we will attempt to seek $C = C_1 d$ in the same way. This cross-validation strategy is also applicable to the ridge parameter λ. In this thesis, we fix $\lambda = 1$ and $P = 100$ as in Hu and Zastawniak (2020).

3.3 Algorithm

This section will present the summary of the NKRR and LKRR methods with bundling for multi-dimensional American option pricing. We mainly use the grid search approach for the Gaussian kernel hyperparameter tuning for the numerical experiments. Note that $\lambda = 1$ and $P = 100$ are fixed in this thesis.

Algorithm 1 KRR based multi-dimensional American option pricing

1: Determine C as stated in the previous section.

2: Simulate M underlying assets $\mathbf{S} \in \mathbf{R}^d$ of N time steps.

3: Generate a $M \times N$ zero cash flow matrix with the option value $h_T(\mathbf{S}_N)$ as the cash flows at t_N.

4: For $i = N-1, ..., 1$:

(i) Bundling. Use the payoff at t_{i-1}(regression now) or t_i(regression later) to partition the corresponding grid points into P bundles.

(ii) For bundle $p = 1, ..., P$, perform KRR on in-the-money paths at t_i(regression now) or t_{i+1}(regression later) to obtain $\hat{\boldsymbol{\alpha}}_i$ or $\hat{\boldsymbol{\alpha}}_{i+1}$ and then compute $\hat{\boldsymbol{Q}}_i$ by using equation (3.8) or (3.11).

(iii) Update cash flows in the cash flow matrix by comparing the estimated continuation values with the exercise payoff.

5: Compute the average of the discounted cash flows to get the option price.

Chapter 4

Experiments: GBM and MJD

In this chapter, we will conduct various numerical experiments to investigate the accuracy, efficiency, and robustness of the NKRR and LKRR approaches for American option pricing. The underlying asset prices are constructed using the multidimensional GBM and MJD models. The numerical tests are primarily concerned with three aspects:

1. The pricing accuracy and the computational costs under a range of dimensions $d = \{5, 10, 15, 20, 30, 40, 60, 80, 100\}$;

2. The influence of the hyperparameters P, λ and C on the pricing accuracy;

3. The robustness of those two methods under varying parameters such as the initial asset price S_0, the drift term μ, etc.

In addition, we include the least square Monte Carlo (LSMC) method for the experiments about the first aspect. The main reason is to investigate whether NKRR and LKRR improve the pricing results in terms of accuracy and computational efficiency compared to the LSMC-based method. It should be noted that the LSMC is an example of a regression now method. The only distinction between LSMC and NKRR is the regression part, where LSMC utilizes linear regression and uses Eq.(3.1) for calculating the regression coefficients. We consider two different sets of basis functions for LSMC. The first one employs polynomial basis functions up to the second degree, and we denote it as LSM. The other set has the payoff function as an additional basis function, and we denote it as LSMP.

For the KRR-based methods, we will use fixed $P = 100$ and $\lambda = 1$ except for the experiments about the second aspect. The bundling technique is not applied to the LSMC method since bundling does not improve the pricing accuracy, and it increases the computational time (Hu and Zastawniak, 2020).

Since no reliable results exist for high-dimensional American option pricing, we will use the results from Premia[1] as benchmarks for the numerical experiments. All the experiments are performed using Matlab 2020a on an Intel Core i5-8265U and 8 GB of RAM laptop in a Windows 10 environment.

[1]Premia is a numerical platform designed for option pricing, hedging, and financial model calibration.https://www.rocq.inria.fr/mathfi/Premia/index.html.

4.1 Experiments GBM: max call American options

Let us consider a max call American option with the underlying asset prices following the multidimensional GBM model. The payoff function of a max call option is defined as

$$h(\mathbf{S}_t) = \left(\max\left(S_{t,1}, ..., S_{t,d}\right) - K_s\right)^+, \tag{4.1}$$

where $\mathbf{S}_t = (S_{t,1}, ..., S_{t,d})$ is generated using GBM and K_s denotes the strike price. We employ the identical set of parameters as the study case of Hu and Zastawniak (2020) to verify the correctness of our implementation of NKRR and LKRR. Table 4.1 shows the parameters, where $v, l = 1, ..., d$ with $v \neq l$ and $d \in \{5, 10, 15, 20, 30, 40, 60, 80, 100\}$. The terms r and q_v are the annual risk-free interest and dividend rates, respectively. The experiments utilize M Monte Carlo simulations. The term N refers to the number of exercise opportunities, and T is the maturity of the max call American option. Furthermore, we assume that the underlying assets are not correlated. We will also use the same Gaussian kernel parameters for NKRR and LKRR, which are $C_{nkrr} = 10^5$ and $C_{lkrr} = 30$, respectively.

$S_{0,v}$	K_s	σ_v	r	q_v	ρ_{vl}	T	P	λ	N	M
100	100	0.2	5%	10%	0	3	100	1	3	10,000

Table 4.1: Parameters for the tests under GBM.

4.1.1 Pricing accuracy and computational time

Figure 4.1 and Table 4.2 present the computational time of those four methods against the range of dimensions. As mentioned, the running time of the LSMC approach increases in dimension. This phenomenon is also visible in the plot below, as the computational times of LSM and LSMP appear to increase exponentially for $d > 40$. For LKRR, the increase in running times is roughly linear concerning the dimensions. On the other hand, the dimension of the options seems to have a negligible influence on the computational cost for NKRR.

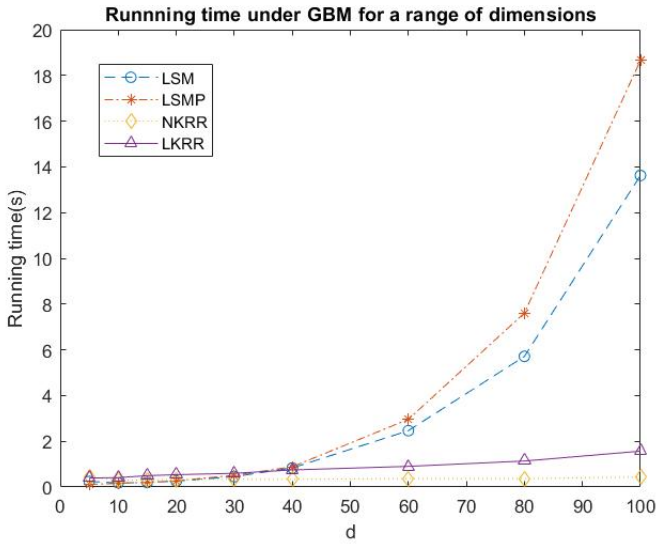


Figure 4.1: Computational time under GBM.

Considering computing performance for high-dimensional option pricing, the KRR-based approaches surpass the LSMC method. NKRR is the best approach in terms of computing time

in all dimensions, as indicated in the table below. Among those four methods, LKRR is the slowest for dimensions $d \leq 30$. NKRR uses less running time than LKRR, which is reasonable, as LKRR involves computing the conditional expectation of the Gaussian kernel to estimate continuation values. A similar phenomenon exists between LSMP and LSM for $d \geq 60$. LSMP has an additional basis function, which might lead to a higher computational cost for computing the regression coefficients.

d	5	10	15	20	30	40	60	80	100
LSM	0.1308	0.1481	0.1926	0.2523	0.4499	0.8214	2.3913	5.6007	13.4351
LSMP	0.1108	0.1501	0.2076	0.2685	0.4624	0.8397	2.6818	7.3514	18.5348
NKRR	0.0572	0.0573	0.0567	0.0592	0.0711	0.0780	0.0945	0.1044	0.1186
LKRR	0.1641	0.2785	0.2605	0.2910	0.4694	0.6028	0.8091	1.1086	1.4836

Table 4.2: GBM: Running time in (s) over dimensions.

Additionally, Table 4.3 and Figure 4.2 present the pricing accuracy of those four methods. The benchmark prices acquired from Premia using the primal-dual approach of Andersen and Broadie (2004) are listed in the column Premia. The primal-dual method provides a lower and upper bound for the true option value. The lower bound price is achieved using the LSMC method with $M = 50000$ Monte Carlo simulations and Hermite polynomials of degree $0 - 3$ as the basis functions. The upper bound computation involves 500 outer simulation and 100 nested simulations. This primal-dual method will be elaborated in more detail in subsection 6.2.1 when dealing with the high-dimensional American option pricing under the Heston model. Premia only provides the point estimates of the lower and upper bounds. Moreover, Premia does not provide a confidence interval for their results. We will utilize the 95% confidence interval of Premia constructed by Hu and Zastawniak (2020), see Appendix C.

The pricing results of those four methods are calculated using 60 independent iterations. Between the brackets are the corresponding standard deviations. Moreover, the pricing errors are the relative error with respect to the benchmarks.

d	Premia	LSM	Error(%)	LSMP	Error(%)	NKRR	Error(%)	LKRR	Error(%)
5	25.306	25.268(0.188)	0.150	25.294(0.177)	0.048	25.456(0.179)	0.593	25.487(0.189)	0.716
10	37.698	37.318(0.211)	1.008	37.363(0.213)	0.889	37.575(0.224)	0.325	37.697(0.227)	0.003
15	45.569	45.168(0.259)	0.880	45.261(0.262)	0.676	45.495(0.265)	0.161	45.652(0.263)	0.182
20	51.443	50.888(0.263)	1.079	51.029(0.251)	0.804	51.236(0.257)	0.403	51.428(0.252)	0.029
30	59.775	59.041(0.259)	1.228	59.245(0.264)	0.887	59.454(0.255)	0.537	59.622(0.267)	0.256
40	65.525	65.059(0.258)	0.711	65.286(0.251)	0.365	65.482(0.265)	0.066	65.597(0.257)	0.109
60	73.900	73.718(0.283)	0.247	73.913(0.287)	0.018	73.930(0.262)	0.041	73.964(0.277)	0.087
80	79.908	80.051(0.240)	0.179	80.136(0.240)	0.287	79.769(0.253)	0.174	79.812(0.248)	0.120
100	84.501	85.164(0.264)	0.784	85.200(0.260)	0.827	84.248(0.256)	0.299	84.410(0.252)	0.108

Table 4.3: Pricing results under GBM with $C_{nkrr} = 10^5$ and $C_{lkrr} = 30$.

The pricing accuracy of LSM and LSMP is superior to KRR-based methods for d=5. While NKRR and LKRR generally perform better than the LSMC approach for other dimensions. Furthermore, LKRR is generally more accurate than NKRR, and LSMP is slightly better than LSM in most dimensions. When only considering the accuracy, LKRR is the best pricing method in most cases for $d \geq 10$.

When taking the computational cost into account, LKRR is the best choice for dimensions in most cases for $d \leq 30$ (except for d= 5) since the computing time of LKRR is similar to the other methods but has the best pricing accuracy. NKRR is superior for $40 \leq d \leq 80$ because

NKRR and LKRR have comparable pricing errors, but NKRR outperforms LKRR in running time. A trade-off between pricing accuracy and efficiency needs to be made for $d = 100$.

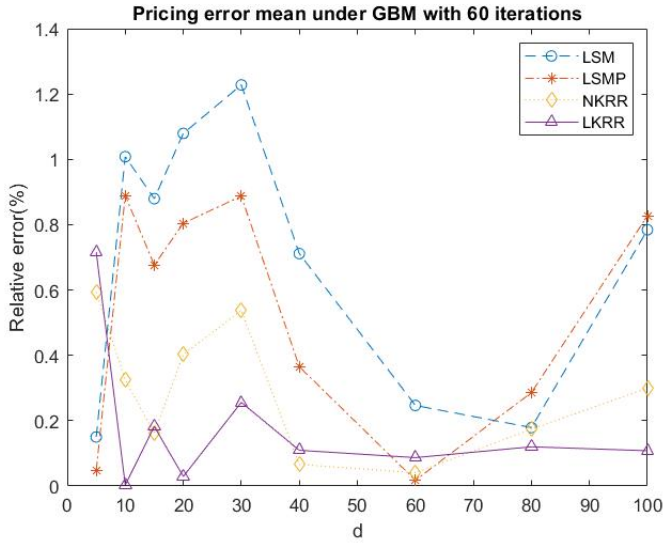


Figure 4.2: Pricing error under GBM using 60 iterations.

Our pricing results are similar to the one of Hu and Zastawniak (2020), which verifies the validity of our implementation. The computational costs of our implementation are generally higher than theirs. It implies there is still room for improvement in efficiency.

4.1.2 The influence of the hyperparameters

In this subsection, we will vary the hyperparameters P, λ, and the Gaussian kernel parameters C_{nkrr} and C_{lkrr}. As we aim to investigate the influence of these parameters on the pricing accuracy of options with low and high dimensions, we will use $d = 20$ and $d = 100$ for all three test cases. When varying one of those parameters, all other parameters remain the same as in the experiments of the previous subsection. Moreover, the test results are based on 20 independent iterations to limit the total computational time of our experiments.

The number of bundles P

Since we are using $M = 10000$ Monte Carlo simulations, let us consider this set for the number of bundles, $P = [20, 50, 80, 100, 200, 400]$. The number of bundles P should not be too great, as this would reduce the sample size for regression and result in poor pricing accuracy. The computational time should decrease in P because the sample size of local regression decreases (Zhang et al., 2015), which should improve the computational efficiency. This phenomenon can be observed in the left subfigures of Figure 4.3.

The pricing errors of both methods decrease as P increases to 100 for both low and high dimensions. For $P = 200$, the pricing result of LKRR is affected more significantly for low dimensions than high dimensional options, while we observe the opposite for NKRR. Both methods become less accurate with $P = 400$ for both low and high dimensions. This might be because fewer data points are used for regression when the number of bundles grows, which could result in erroneous regression coefficients and lead to greater relative errors.

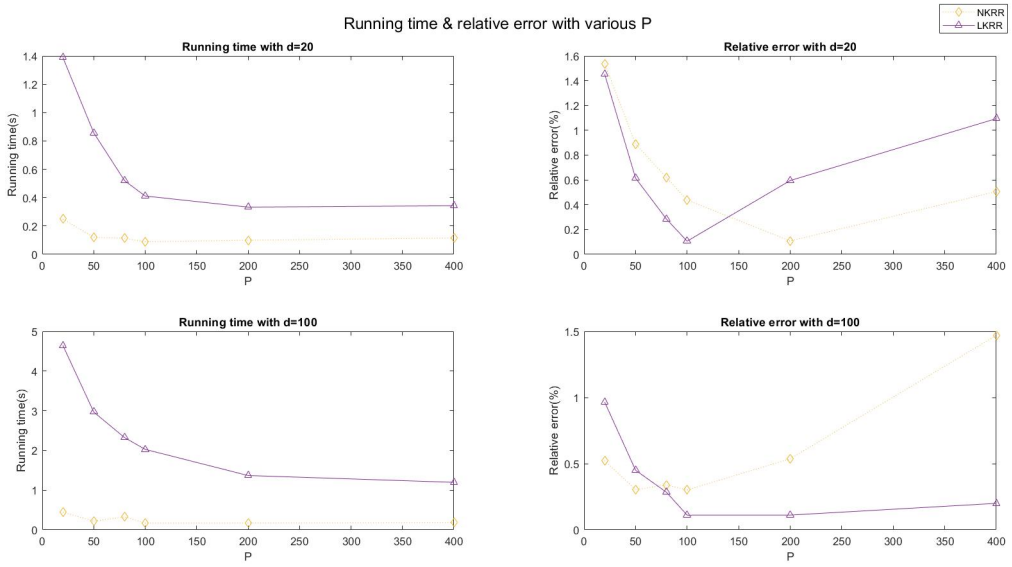
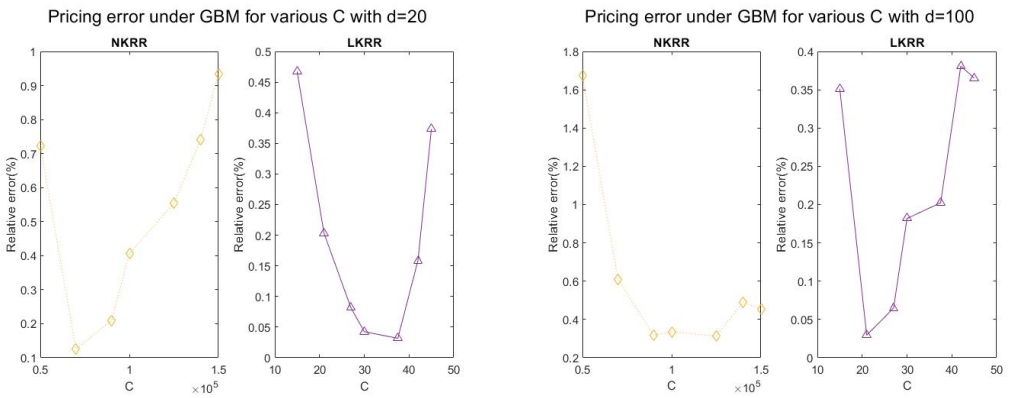


Figure 4.3: Running time and pricing error under GBM with various P.

Figure 4.3 also indicates the existence of optimal choice P. However, it seems to be different for different dimensions and methods. Because the optimal $P = 100$ for LKRR while $P = 200$ is the best choice for NKRR at $d = 20$. Both methods have the most accurate pricing results at high dimensions with $P = 100$. It implies the primary choice of $P = 100$ suggested by Hu et al. is suitable for both low and high-dimension option pricing. In a later chapter, we will use this choice for other numerical experiments under the Heston model.

The Gaussian kernel parameters

To evaluate the influence of the Gaussian kernel parameters for NKRR and LKRR, we multiply C with a scale vector $s = [0.5, 0.7, 0.9, 1, 1.25, 1.4, 1.5]$. Recall that the previous tests employed values of $C = 10^5$ and $C = 30$ for NKRR and LKRR, respectively. This hyperparameter does not influence computational efficiency. Therefore, we will not perform experiments on the running time. The figure below shows the pricing error of this experiment under 20 independent iterations.



(a) $d = 20$ (b) $d = 100$

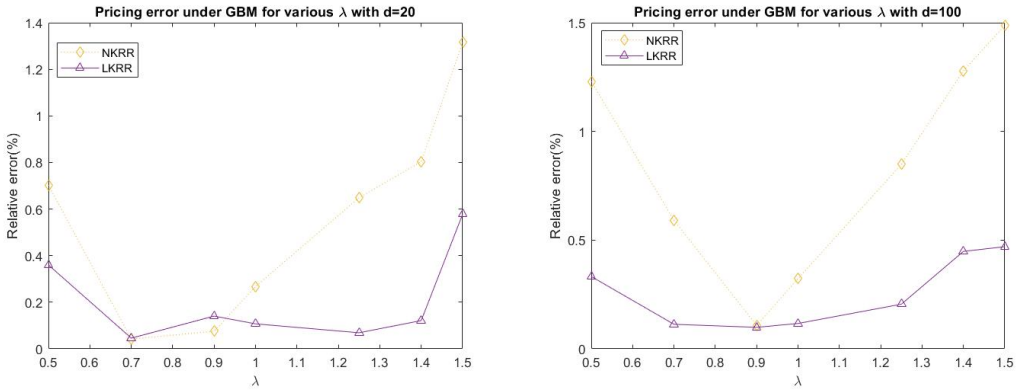
Figure 4.4: Pricing accuracy under GBM for various C.

As can be seen, there are better choices C than what we selected for both NKRR and LKRR. This indicates that pricing results in Table 4.3 can be further improved. Furthermore, the optimal values of this hyperparameter also depend on the dimension of the options. Because when d varies, we have different optimal C for both NKRR and LKRR. To limit the computational costs, we aim to avoid the calibration process of C at each dimension. It implies that a compromise must be made and choose C that provides accurate results in all dimensions.

The pricing error of LKRR is not sensitive to varying Gaussian kernel parameters as the error rates vary between 0.025% and 0.5% for both dimensions. NKRR seems to be more affected by the Gaussian kernels from the plot. However, keep in mind that the initial $C_{nkrr} = 10^5$, so C_{nkrr} is varied between a broader range of parameters than C_{lkrr}. It means a wider range of pricing errors is reasonable for NKRR in our experiment. If we zoom in on the interval for C_{nkrr}, we could expect a similar phenomenon as for C_{lkrr}.

The ridge parameter λ

We modify the value of λ using the same scale vector s as the previous experiment. This parameter does not affect the running time as well. Therefore, we only focus on its effect on pricing accuracy. The outcomes are depicted in the figure below.



(a) $d = 20$ (b) $d = 100$

Figure 4.5: Pricing accuracy under GBM for various λ.

A similar conclusion can be drawn as the previous experiment, that the optimal λ depends on the option dimension. The optimal choice is $\lambda = 0.9$ for both NKRR and LKRR when $d = 100$. Although $\lambda = 0.7$ is the best at $d = 20$, this selection leads to a worse precision for $d = 100$. Thus, the compromise choice is $\lambda = 0.9$ as it is the second best for $d = 20$. The pricing result presented in Table 4.3 can be further improved since we used a nearly optimal $\lambda = 1$.

Moreover, the pricing error of LKRR fluctuates between 0.05% and 0.6%, while the error rate of NKRR is between 0.1% and 1.5%. In other words, LKRR is more robust than NKRR as the influence of varying ridge parameters is less significant for both dimensions.

4.1.3 Robustness under varying parameters

To figure out how other parameters affect the pricing results, we have plotted in Figure 4.6 for various σ, S_0, r and ρ with $d = 30$. Since the result of various ρ is not available in Premia, we use the pricing values of LSM and LSMP as benchmarks. The differences between the benchmarks and the results of NKRR and LKRR are insignificant. Only for the correlations,

the results of NKRR are slightly higher than the other methods. This experiment indicates the hyperparameters are quite robust with varying parameters.

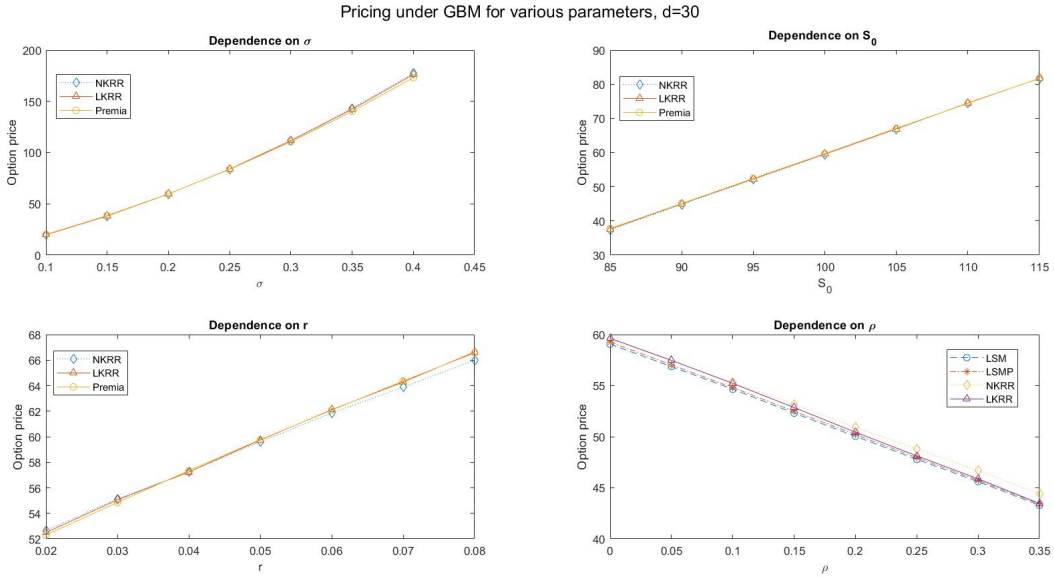


Figure 4.6: Pricing error under GBM with various choices of parameters.

4.2 Experiments MJD: geometric put American options

This section will conduct various experiments with geometric put American options under the multidimensional MJD. The payoff function is as follows

$$h(\mathbf{S}_t) = \left(K_s - \Pi_{v=1}^{d} S_{t,v}^{\frac{1}{d}}\right)^{+}. \tag{4.2}$$

where $\mathbf{S}_t$ is generated by the MJD model as in Eq.(2.13). Although Premia does not provide results for the high-dimensional option under MJD, the multidimensional problem of MJD can be converted into a one-dimensional problem. Recall the analytic solution of one-dimensional MJD is as in Eq.(2.11) and is given by

$$\widetilde{S}_t = \widetilde{S}_0 \exp\left(\left(r - \widetilde{q} - \widetilde{\kappa}^J\widetilde{\lambda}^J - \frac{1}{2}\widetilde{\sigma}^2\right)t + \widetilde{\sigma}W_t\right)\exp\left(\sum_{m=1}^{\Gamma_t} Z_m^J\right).$$

The parameters can be obtained using the parameters of the multi-MJD models (Hu and Zastawniak, 2020):

$$\widetilde{S}_t = \prod_{v=1}^{d} S_{t,v}^{\frac{1}{d}}\ ; \quad \widetilde{\mu}^J = \frac{1}{d}\sum_{v}^{d}\mu_v^J\ ; \quad \widetilde{\sigma}^J = \frac{1}{d}\left(\sum_{v,l}^{d}\sigma_v^J\sigma_l^J\rho_{vl}^J\right)^{1/2},\ \widetilde{\lambda}^J = \lambda^J;$$

$$\widetilde{\kappa} = \exp\left(\widetilde{\mu}^J + \frac{1}{2}\left(\widetilde{\sigma}^J\right)^2\right) - 1\ ; \quad \widetilde{\sigma} = \frac{1}{d}\left(\sum_{v,l}^{d}\sigma_v\sigma_l\rho_{vl}\right)^{1/2},$$

$$\widetilde{q} = \frac{1}{d}\sum_{v}^{d}\left(q_v + \frac{1}{2}\sigma_v^2 + \lambda^J\kappa_v\right) - \frac{1}{2}\widetilde{\sigma}^2 - \lambda^J\widetilde{\kappa}.$$

We can evaluate those four approaches in high-dimensional settings using the results of one-dimensional MJD as the benchmark, provided we obtain the parameters for multi-dimensional MJD and their corresponding ones for one-dimensional MJD.

To investigate the correctness of our implementation of LKRR under the MJD model, the parameters for the single MJD are the same as the study case in Hu and Zastawniak (2020), see Table 4.4.

$\widetilde{S}_0$	K_s	$\widetilde{\sigma}$	r	$\widetilde{q}$	$\widetilde{\mu}^J$	$\widetilde{\sigma}$	$\widetilde{\kappa}$	$\widetilde{\lambda}^J$	T	N	M
40	40	$\sqrt{0.05}$	8%	0	-0.025	$\sqrt{0.05}$	0	5	1	10	10,000

Table 4.4: Parameters for the one-dimensional MJD.

The benchmark of this set of parameters is 6.995 (Broadie and Yamamoto, 2003), which remains the same for all dimension values d. Then according to Hu and Zastawniak (2020), the corresponding parameters for the multi-dimensional MJD are

$$S_{0,v} = \tilde{S}_0, \quad \sigma_v = \sigma_v^J = a\tilde{\sigma}, \quad r = 8\%, \quad \mu_v^J = \tilde{\mu}^J, \lambda^J = \tilde{\lambda}^J,$$
$$q_v = \frac{1}{2}\left(\tilde{\sigma}^2 - \sigma_v^2\right) - \lambda^J \kappa_v, \quad \rho_{vl} = \rho_{vl}^J = \frac{d/a^2 - 1}{d-1}, \quad a > 1,$$

where $a = 1.5$ is chosen for the experiments.

As in the GBM experiments, we set $P = 100$ and $\lambda = 1$. The Gaussian kernel parameters are as follows: $C_{nkrr} = C_{lkrr} = d \times 10^4$ (Hu and Zastawniak, 2020). Furthermore, no reliable results exist for the multi-MJD with different parameters. We will drop the experiment that investigates the influence of parameters(such as the mean, the variance, etc.) on the pricing accuracy.

4.2.1 Pricing accuracy and computational time

Figure 4.7 and Table 4.5 present the computing time of those four pricing methods against dimensions. The relationship between the dimension and computing time is comparable to the GBM model, despite the running time being longer for all methods under the MJD model.

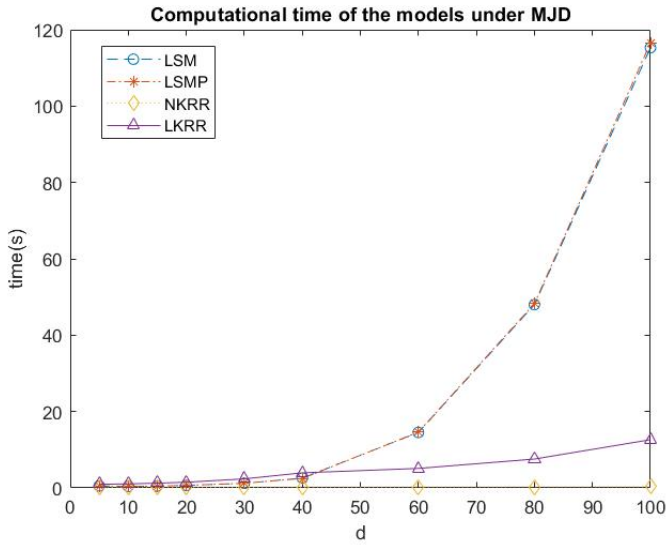


Figure 4.7: Computational time under MJD.

The computing time of all four methods is below 5s for options with $d \leq 40$, where the LKRR approach is the slowest. For higher dimensions $d \geq 40$, the KRR-based methods perform much better in computational costs than the LSMC methods. As in the GBM experiment, the running time of LSM/LSMP increases exponentially in dimension, while it is approximately linear for LKRR. Although the NKRR approach is also slightly affected by increasing dimensions, it is still the most computationally efficient method, regardless of the dimensions.

d	5	10	15	20	30	40	60	80	100
LSM	0.5245	0.3977	0.4734	0.6477	1.2058	2.5791	14.5253	48.0104	115.3042
LSMP	0.3071	0.3655	0.4843	0.6566	1.2269	2.4729	14.6209	48.2768	116.5671
NKRR	0.2617	0.1763	0.1649	0.1676	0.1864	0.2130	0.2300	0.2729	0.3188
LKRR	0.9584	1.0391	1.2562	1.4841	2.3745	3.9503	5.1054	7.5635	12.6317

Table 4.5: MJD: Running time in (s) over dimensions.

The pricing result of those four methods is presented in Table 4.6 and Figure 4.8. As the running time of LSM/LSMP is above 100s at $d = 100$, we only run 20 independent iterations to limit the total running time of this numerical experiment. The values between the brackets are the corresponding standard deviations.

d	LSM	Error(%)	LSMP	Error(%)	NKRR	Error(%)	LKRR	Error(%)
5	6.884(0.057)	1.585	7.015(0.053)	0.282	6.965(0.056)	0.432	7.058(0.057)	0.897
10	6.956(0.087)	0.552	7.070(0.083)	1.071	7.017(0.070)	0.313	7.049(0.074)	0.777
15	7.011(0.066)	0.231	7.123(0.060)	1.833	7.011(0.065)	0.233	7.028(0.057)	0.470
20	7.143(0.061)	2.114	7.232(0.052)	3.385	7.041(0.062)	0.660	7.054(0.054)	0.840
30	7.121(0.070)	1.796	7.464(0.068)	6.709	7.059(0.064)	0.929	7.055(0.064)	0.858
40	7.440(0.057)	6.359	7.727(0.054)	10.467	7.064(0.046)	0.990	7.043(0.052)	0.685
60	8.162(0.039)	16.678	8.331(0.052)	19.096	7.053(0.045)	0.824	7.032(0.035)	0.533
80	8.892(0.053)	27.120	8.900(0.055)	27.240	7.098(0.057)	1.470	7.065(0.059)	1.002
100	9.511(0.060)	35.968	9.517(0.060)	36.053	7.088(0.065)	1.323	7.065(0.065)	1.007

Table 4.6: Pricing results under MJD with $C = d \times 10^4$ and benchmark is 6.995.

The KRR-based methods provide more accurate results than the LSMC methods, as indicated in Table 4.6. LSM and LSMP produce acceptable results for low dimensions $d < 20$. However, the relative error of the LSMC method increases in d, and it becomes more significant for $d \geq 20$. Therefore, the LSMC method is unsuitable for the high-dimensional American option under the MJD model. NKRR is generally the best method of all for low dimensions. Furthermore, LKRR outperforms NKRR when $d \geq 30$. Nevertheless, the pricing results of NKRR and LKRR under the MJD model are generally less accurate than under the GBM model.

Regarding computing efficiency and pricing precision for $d \leq 30$, NKRR is superior to LKRR. Although LKRR has smaller relative errors than NKRR for $d > 30$, the computational cost for LKRR is more expensive. Therefore, a trade-off between pricing efficiency and accuracy must be made when the dimension increases. The LSMC-based approach requires more running time while producing less convincing pricing results than KRR-based methods. Therefore, LSMC is not the best pricing method under the MJD model in computing efficiency and pricing accuracy.

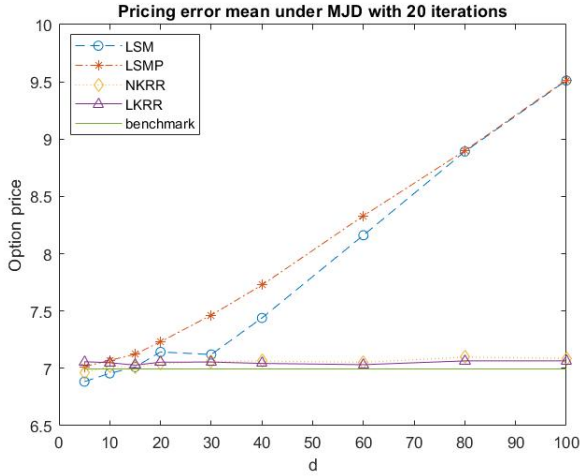


Figure 4.8: Option value under MJD.

4.2.2 The influence of the hyperparameters

This subsection examines the influences of the bundle number P, the Gaussian kernel parameter C, and the ridge parameter λ on the pricing accuracy of NKRR and LKRR for both low-and high-dimension scenarios. We select $d = 20$ and $d = 100$ as representatives for low and high dimensions. As in GBM experiments, the latter two parameters are multiplied with the scale vector $s = [0.5, 0.7, 0.9, 1, 1.25, 1.4, 1.5]$. The test results are also based on 20 independent iterations.

The number of bundles P

We use the same set of $P = [20, 50, 80, 100, 200, 400]$ as in the GBM experiment. The running time is decreasing in P for both dimensions as indicated in the left subfigures of Figure 4.9. The influence of P becomes less significant for the required computing time in both dimensions when $P \geq 100$.

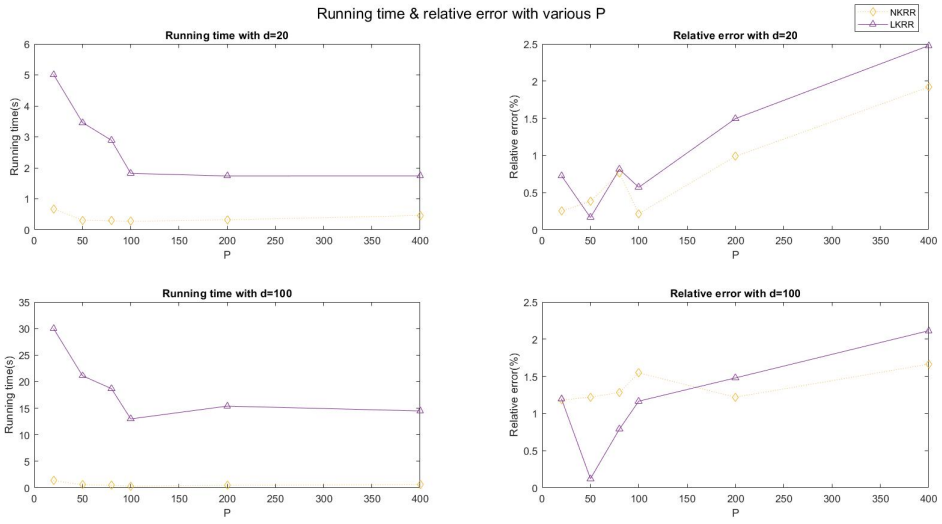


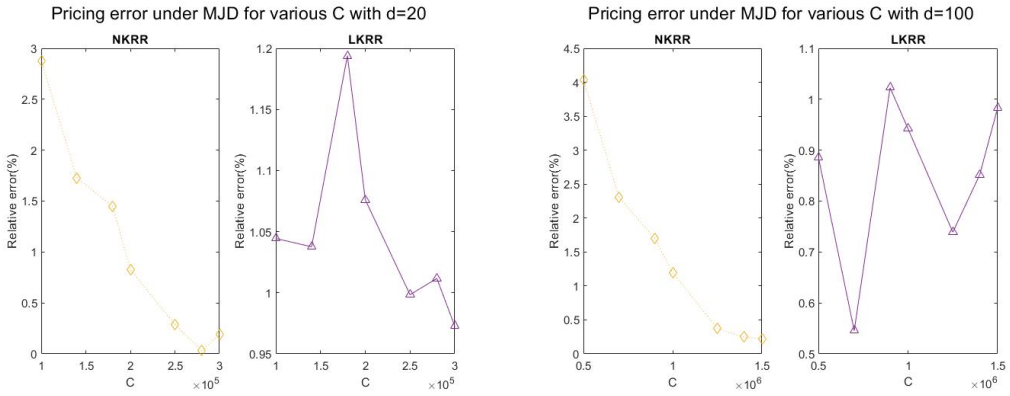
Figure 4.9: Running time and pricing error under MJD with various P.

Considering the pricing error, $P = 50$ is a better choice for both methods in low and high dimensions. The pricing result of both methods is significantly affected when $P > 100$, as the error rates become above 2% in both dimensions. Recall the optimal $P = 100$ under GBM using equal-sized Monte Carlo simulations. In other words, the used model for generating the asset prices seems to impact the optimal choice of P in addition to the data size.

The Gaussian kernel parameter

The pricing results are shown in Figure 4.10. Both methods' initial Gaussian kernel parameters are equivalent and $C = d \times 10^4$. As this parameter depends on the dimension, we observe different x-axis for $d = 20$ and $d = 100$.

NKRR is more sensitive to the change of the Gaussian kernel parameters than LKRR, as the relative errors of NKRR for both dimensions decrease significantly for a greater C value. While for LKRR, the relative errors are below 1.2% for both dimensions and fluctuate within a small range. The same conclusion can be drawn as in GBM tests, that the optimal choice of C depends on dimensions. For LKRR, the best $C = 2d \times 10^4$ in $d = 20$ while $C = 0.7d \times 10^4$ when $d = 100$. The choice of $C = d \times 10^4$ is clearly not optimal for both methods at low and high dimensions, which means the pricing results presented in the table above still have room for improvement. The better choice at both dimensions is $C = 1.25d \times 10^4$ as it reduces the pricing error for both methods.

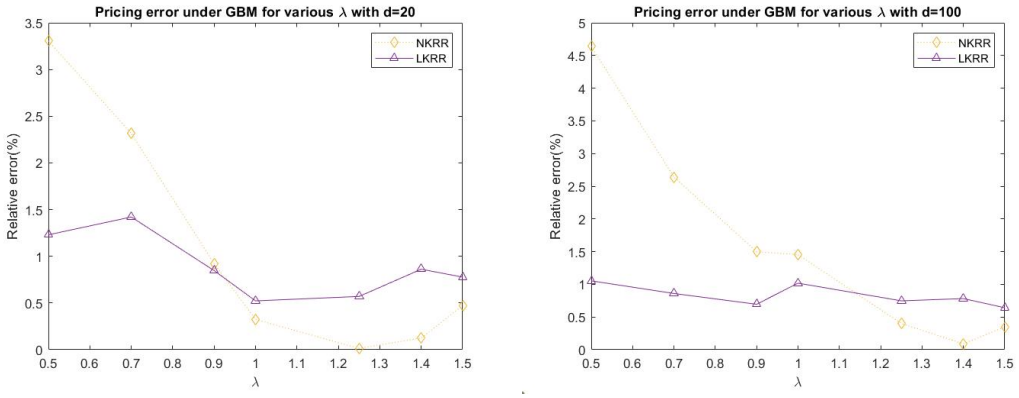


(a) $d = 20$ (b) $d = 100$

Figure 4.10: Pricing accuracy under MJD for various C.

The ridge parameters

The pricing accuracy of these two dimensions under varying ridge parameters is depicted in the image below. For NKRR, we observe a familiar phenomenon as in the previous experiment about C: the pricing error reduces in λ. Additionally, the error fluctuation of LKRR is less significant than NKRR. In conclusion, LKRR is more resistant to changes in the ridge parameter. An optimal λ exists for both methods. Nevertheless, the current choice, $\lambda = 1$, produces satisfactory outcomes among this range of λ.



(a) $d = 20$ (b) $d = 100$

Figure 4.11: Pricing accuracy under MJD for various λ.

4.3 Conclusion

In this chapter, we have conducted various experiments to investigate the proposed NKRR and LKRR methods by Hu and Zastawniak (2020) for multidimensional American options. For comparison purposes, we have also implemented the LSMC methods with two sets of basis functions, where one with polynomial functions up to degree 2(LSM) and the other has the payoff function as an additional basis function(LSMP). Several experiments were conducted to determine which method is the best choice under various asset price models for a range of dimensions in the sense of computational efficiency and pricing accuracy. Furthermore, the robustness of the NKRR and LKRR is examined using various (hyper)parameters at $d = 20$ and $d = 100$.

Regarding computational costs and pricing accuracy, the performance of those four methods differs under various asset price models. All four methods provide accurate results for asset prices generated using multidimensional GBM. The KRR-based approach generally outperforms LSMC-based methods as $d > 5$, just as shown in Figure 4.2. Among the KRR-based methods, the pricing error of LKRR is smaller than NKRR in most tested dimensions. Therefore, LKRR is the best method concerning pricing accuracy under the GBM model. When taking computational costs into account, we suggest using LKRR for $d < 40$ and NKRR for $40 \leq d \leq 80$ as NKRR provides similar pricing results to LKRR but requires less running time, see Figure 4.1. At $d = 100$, a trade-off between pricing accuracy and running time must be made. The LSMC-based methods are not the best choice for high dimensional option pricing since their running time increases exponentially in dimensions.

For asset prices following the multidimensional MJD model, the pricing accuracy of those four methods is worse than under GBM. The LSMC method does not perform well in both computational efficiency and accuracy; see Figure 4.7 and Figure 4.8. LKRR provides better pricing results for $d \geq 30$ than NKRR but with a more expensive computational cost. As the running time of NKRR is only slightly affected by the dimension and LKRR generally requires much more time, NKRR is a better choice under high dimensional MJD.

The optimal choice of hyperparameters varies with the dimensions. To avoid the calibration process of the hyperparameters for each dimension, we must choose compromise hyperparameters that provide accurate pricing results among both low and high dimensions. In general, LKRR is less sensitive to varying hyperparameters compared with NKRR. We can also conclude that the choice of $\lambda = 1$ and $P = 100$ produces satisfactory results, and we will use those two parameters for the experiments under the Heston model.

Chapter 5

The Heston model

This chapter will elaborate on the Hetson model, an example of the stochastic volatility model. Since the Euler discretization scheme of CIR generates unrealistic sample realizations (Oosterlee and Grzelak, 2020), we will first introduce the so-called Quadratic Exponential scheme proposed by Andersen (2008) for the simulation of the CIR process. The simulation of the one-dimensional Heston model will then be generalized to the multi-dimensional Heston. Eventually, we will derive an expression for estimating the continuation value using LKRR under the Heston model.

5.1 One dimensional Heston

The Heston model describes the prices of the underlying assets as the GBM with stochastic volatility $\sigma^2 = \nu_t$, where ν_t follows the CIR process as in Eq.(2.15). Then the SDE of the asset price $S(t)$ under the Heston model is given as follows (Heston, 1993)

$$\begin{cases} \frac{dS(t)}{S(t)} = (r-q)dt + \sqrt{\nu_t} dW^S(t) \\ \mathrm{d}\nu(t) = \kappa(\bar{\nu} - \nu(t))\mathrm{d}t + \gamma\sqrt{\nu(t)}\mathrm{d}W^\nu(t) \end{cases}, t \in [0, T]$$

where r is the annual risk-free interest rate, q the dividend rate, $\kappa > 0$ is the speed of mean reversion. The long-term mean is denoted as $\bar{\nu}$, which is nonnegative. $\gamma > 0$ denotes the volatility of the variance process. Further, $W^S(t)$ and $W^\nu(t)$ are standard Brownian motions under risk neutral measure with correlation $dW^S(t)dW^\nu(t) = \rho dt$. If we define $W^\nu(t) = \widetilde{W}^\nu(t)$ and $W^S(t) = \rho d\widetilde{W}^\nu(t) + \sqrt{1-\rho^2} d\widetilde{W}^S(t)$ with independent $\widetilde{W}^S(t)$ and $\widetilde{W}^\nu(t)$, then we obtain the following SDE with independent Brownian motions

$$\begin{cases} \dfrac{\mathrm{d}S(t)}{S(t)} = (r-q)\mathrm{d}t + \sqrt{\nu_t}\left(\rho\mathrm{d}\widetilde{W}^\nu(t) + \sqrt{1-\rho^2}\mathrm{d}\widetilde{W}^S(t)\right), & \text{(5.1a)} \\ \mathrm{d}\nu(t) = \kappa(\bar{\nu} - \nu(t))\mathrm{d}t + \gamma\sqrt{\nu(t)}\mathrm{d}\widetilde{W}^\nu(t). & \text{(5.1b)} \end{cases}$$

Suppose we discretize the time interval $[0, T]$ into N time steps with step size $h = \frac{T}{N}$ and denote the time step t_i as i for $i = 0, ..., N$ with $t_0 = 0$ and $t_N = T$. Moreover, let us consider the log prices $X_i = \log S_i$, the dynamic can be derived using the Itô's lemma

$$\mathrm{d}X_i = \left(r - q - \frac{1}{2}\nu_i\right)\mathrm{d}t + \sqrt{\nu_i}\left(\rho\mathrm{d}\widetilde{W}_i^\nu + \sqrt{1-\rho^2}\mathrm{d}\widetilde{W}_i^X\right).$$

By integrating $\mathrm{d}X_i$and $\mathrm{d}\nu_i$, the following formulas yield, which give us an intuition for the

discretization

$$X_{i+1} = X_i + \int_i^{i+1} \left(r - q - \frac{1}{2}\nu(s)\right) \mathrm{d}s + \rho \int_i^{i+1} \sqrt{\nu(s)} \mathrm{d}\widetilde{W}^{\nu}(s) + \sqrt{1-\rho^2} \int_i^{i+1} \sqrt{\nu(s)} \mathrm{d}\widetilde{W}^{X}(s) \tag{5.2}$$

and

$$\nu_{i+1} = \nu_i + \int_i^{i+1} \kappa(\bar{\nu} - \nu(s))\mathrm{d}s + \gamma \int_i^{i+1} \sqrt{\nu(s)} \mathrm{d}\widetilde{W}^{\nu}(s).$$

Then by rearranging and simplifying the expression of the CIR process, we get

$$\int_i^{i+1} \sqrt{\nu(s)} \mathrm{d}\widetilde{W}^{\nu}(s) = \frac{1}{\gamma}\left[\nu_{i+1} - \nu_i - \kappa\bar{\nu}h + \kappa \int_i^{i+1} \nu(s)\mathrm{d}s\right]. \tag{5.3}$$

Finally, the substitution of Eq.(5.3) into Eq.(5.2) gives us

$$X_{i+1} = X_i + (r-q)h + \frac{\rho}{\gamma}\left[\nu_{i+1} - \nu_i - \kappa\bar{\nu}h\right] + \left(\frac{\kappa\rho}{\gamma} - \frac{1}{2}\right) \int_i^{i+1} \nu(s)\mathrm{d}s + \sqrt{1-\rho^2} \int_i^{i+1} \sqrt{\nu_i} \mathrm{d}\widetilde{W}^{X}(s).$$

Note that $\int_i^{i+1} \sqrt{\nu(s)} \mathrm{d}\widetilde{W}^{X}(s)$ is an Itô's integral, which is normal distributed with zero mean and variance $\int_i^{i+1} \nu(s)\mathrm{d}s$, given ν_i. From the expression above, it is clear that the simulation of the samples ν_{i+1} and $\int_i^{i+1} \nu(s)\mathrm{d}s$ is essential. We will use the Quadratic Exponential (QE) scheme for the simulation of ν_{i+1} in this thesis since it is the most efficient and robust in accuracy even if the Feller condition is violated (Oosterlee and Grzelak, 2020; Wadman, 2010). The QE scheme will be discussed further in the following subsection.

5.1.1 The Quadratic Exponential scheme of CIR

The Quadratic Exponential (QE) discretization scheme for the simulation of the CIR process is proposed by Andersen (2008). As the name suggests, this scheme is composed of two different algorithms. The main idea behind this scheme is the switching between these two algorithms based on the parameters of the CIR process.

The first algorithm of the QE scheme is to approximate $\nu(t)|\nu(s)$ by a quadratic function for $t > s$. From Definition 2.2.2, we know that the variance process $\nu(t)|\nu(s)$ is propositional to a non-central chi-squared distribution. With high non-centrality parameters, the variance process $\nu(t)$ is far from the origin and can be approximated by a quadratic function of a Gaussian variable (Patnaik, 1949)

$$\nu(t) \approx \nu_1(t) = a\left(b + Z_\nu\right)^2, \; Z_\nu \sim \mathcal{N}(0,1) \text{ and } a, b \in \mathbb{R}. \tag{5.4}$$

The constants a and b depend on the time step size and the parameters of $\nu(t)$, which can be estimated using the moment matching technique.

Proposition 5.1.1. *(Andersen, 2008) Given the mean $\bar{m}$ and the variance s_ν^2 of the variance process $\nu(t)|\nu(s)$ as in Definition 2.2.2, then the constants a and b of $\nu_1(t)$ in Eq.(5.4) are*

$$a = \frac{\bar{m}}{1+b^2} \tag{5.5a}$$

$$b^2 = 2\varphi^{-1} - 1 + \sqrt{2\varphi^{-1}}\sqrt{2\varphi^{-1} - 1}. \tag{5.5b}$$

with $\varphi = \frac{s_\nu^2}{\bar{m}^2} \leq 2$.

Proof. Since $\nu_1(t) = a(b + Z_\nu)^2$ with $Z_\nu \sim \mathcal{N}(0,1)$, it follows that $\nu_1(t)$ is proportional to a non-central chi-squared distribution with degree of freedom 1 and non-centrality parameter b^2, i.e., $\nu_1(t) \sim a\mathcal{X}^2(1, b^2)$(Andersen, 2008). The mean and variance of $\mathcal{X}^2(1, b^2)$ are $(1 + b^2)$ and $2(1 + b^2)$, respectively. Thus

$$\mathbb{E}[\nu_1(t) \mid \mathcal{F}(0)] = a(1 + b^2) \text{ and } \mathrm{Var}[\nu_1(t) \mid \mathcal{F}(0)] = 2a^2(1 + 2b^2).$$

We approximate $\nu(t)$ by $\nu_1(t)$, it means the mean and variance should be equivalent

$$\bar{m} = a(1 + b^2) \text{ and } s_\nu^2 = 2a^2(1 + 2b^2).$$

It gives us $a = \frac{\bar{m}}{1+b^2}$ and $a^2 = \frac{s_\nu^2}{2(1+2b^2)}$, thus

$$\frac{\bar{m}^2}{(1 + b^2)^2} = \frac{s_\nu^2}{2(1 + 2b^2)} \Longrightarrow b^4 + 2\left(1 - \frac{2\bar{m}^2}{s_\nu^2}\right) b^2 + 1 - \frac{2\bar{m}^2}{s_\nu^2} = 0.$$

Let $z = b^2$ and $\varphi = \frac{s_\nu^2}{\bar{m}^2}$, then we need to solve the following quadratic equation

$$z^2 + 2(1 - 2\varphi^{-1})z + (1 - 2\varphi^{-1}) = 0.$$

The discriminant of this equation is $D = 4(1 - 2\varphi^{-1})^2 - 4(1 - 2\varphi^{-1})$ and the solution exists ony if $D \geq 0$. This implies $\varphi \leq 2$. By solving this equation with the constraint $z = b^2 \geq 0$, we obtain

$$z = b^2 = 2\varphi^{-1} - 1 + \sqrt{2\varphi^{-1}}\sqrt{2\varphi^{-1} - 1}.$$

□

The above mentioned approximation becomes inaccurate when the probability mass of $\nu(t)$ accumulates around the origin since the moment matching method for finding a and b fails to work (Andersen, 2008). Therefore, the second algorithm is suggested, where the density of $\nu(t)$ is approximated by the exponential function

$$f_{\nu(t)}(x) \approx f_{\nu_2(t)}(x) = \tilde{c}\delta(0) + \beta(1 - \tilde{c})\mathrm{e}^{-\beta x}, \ x \geq 0 \tag{5.6}$$

with $\delta(0)$ the Dirac delta function and the constants $\tilde{c} \in [0, 1]$ and $0 \leq \beta \in \mathbb{R}$ to be determined. The corresponding cumulative distribution is then

$$F_{\nu(t)}(x) \approx F_{\nu_2(t)}(x) = \tilde{c} + (1 - \tilde{c})(1 - \mathrm{e}^{-\beta x}), \ x \geq 0. \tag{5.7}$$

Note that Eq.(5.7) is invertible, therefore $\nu_2(t)$ can be directly sampled from $\nu_2(t) = F^{-1}_{\nu_2(t)}(u)$ with $u \sim \mathcal{U}(0, 1)$ and

$$F^{-1}_{\nu_2(t)}(u) = \begin{cases} 0, & 0 \leq u \leq \tilde{c} \\ \frac{1}{\beta}\ln\left(\frac{1-\tilde{c}}{1-u}\right), & \tilde{c} \leq u \leq 1 \end{cases}. \tag{5.8}$$

Proposition 5.1.2. *(Andersen, 2008) Given the mean $\bar{m}$ and the variance s_ν^2 of the variance process $\nu(t)|\nu(s)$ as in Definition 2.2.2 and $\varphi = \frac{s_\nu^2}{\bar{m}^2}$, then the constants $\tilde{c}$ and β of $f_{\nu_2(t)}$ in Eq.(5.6) are determined as*

$$\tilde{c} = \frac{\varphi - 1}{\varphi + 1} \tag{5.9a}$$

$$\beta = \frac{2}{\bar{m}(\varphi + 1)}, \tag{5.9b}$$

with conditions $\tilde{c} \in [0, 1]$ and $\beta \geq 0$.

Proof. The mean and the variance of $v_2(t)$ are (Oosterlee and Grzelak, 2020)

$$\mathbb{E}[\nu_2(t) \mid \mathcal{F}(0)] = \frac{1-\tilde{c}}{\beta} \text{ and } \text{Var}[\nu_2(t) \mid \mathcal{F}(0)] = \frac{1-\tilde{c}^2}{\beta^2}.$$

By applying the moment matching method, the following yields

$$\beta = \frac{1-\tilde{c}}{\bar{m}} \text{ and } \beta^2 = \frac{1-\tilde{c}^2}{s_\nu^2},$$

which implies

$$(1-\tilde{c})^2 s_\nu^2 = \bar{m}^2(1-\tilde{c}^2) \xrightarrow{\varphi=\frac{s_\nu^2}{\bar{m}^2}} (1+\varphi)\tilde{c}^2 - 2\varphi\tilde{c} + \varphi - 1 = 0.$$

Under the condition $\tilde{c} \in [0,1]$, the solution of this quadratic equation is

$$\tilde{c} = \frac{\varphi-1}{\varphi+1} \text{ and } \beta = \frac{1-\frac{\varphi-1}{\varphi+1}}{\bar{m}} = \frac{2}{\bar{m}(\varphi+1)},$$

with $\varphi \geq 1$. The constraint $\varphi \geq 1$ is essential as we want $\tilde{c}$ be nonnegative. □

The quadratic scheme in Eq.(5.4) and the exponential scheme in Eq.(5.8) form the QE scheme of the CIR simulation. The remaining is the switching rule between these two algorithms. The first algorithm is well defined for $\varphi \leq 2$ while the second is for $\varphi \geq 1$. It implies that both procedures are applicable for $1 \leq \varphi \leq 2$. We can select a critical value $\varphi_c \in [1,2]$ such that $\nu(t)|\nu(s)$ is approximated by Eq.(5.4) if $\varphi \leq \varphi_c$, otherwise it is approximated by Eq.(5.8). According to Andersen (2008), the choice of the critical level only has small effects on the simulation scheme. We will use $\varphi_c = 1.5$ in this thesis.

Below is the summary of the QE scheme.

Algorithm 2 The Quadratic Exponential(QE) scheme of CIR simulation

1: Given a critical level $\varphi_c \in [1,2]$ and the variance ν_i at the current time step t_i with $i = 0, ..., N-1$. The step size is $h = \frac{T}{N}$.
2: Compute $\bar{m}$ and s_ν^2 using Eq.(2.16) and Eq.(2.17), respectively.
3: **if** $\varphi \leq \varphi_c$ **then**
4: Compute a and b using Eq.(5.5).
5: Draw $Z_\nu \sim \mathcal{N}(0,1)$ and use eq.(5.4) to generate ν_{i+1}.
6: **else**
7: Compute $\tilde{c}$ and β using Eq.(5.9).
8: Draw $u \sim \mathcal{U}(0,1)$ and use Eq.(5.8) to generate ν_{i+1}.
9: **end if**

The remaining is the term $\int_i^{i+1} \nu(s)\mathrm{d}s$. We define it as Andersen (2008) suggests

$$\int_i^{i+1} \nu(s)\mathrm{d}s = [\gamma_1\nu_i + \gamma_2\nu_{i+1}]h, \tag{5.10}$$

for some constants γ_1 and γ_2. In this thesis, we will use the simplest case where $\gamma_1 = 1$ and $\gamma_2 = 0$. There are other options, such as central discretization, where $\gamma_1 = \gamma_2 = \frac{1}{2}$. In Dufresne (2001), the moment-matching method is utilized for finding those two constants.

5.1.2 The simulation of Heston model

Recall

$$X_{i+1} = X_i + (r-q)h + \frac{\rho}{\gamma}\left[\nu_{i+1} - \nu_i - \kappa\bar{\nu}h\right] + \left(\frac{\kappa\rho}{\gamma} - \frac{1}{2}\right)\int_i^{i+1}\nu(s)\mathrm{d}s + \sqrt{1-\rho^2}\int_i^{i+1}\sqrt{\nu_i}\mathrm{d}\widetilde{W}^X(s),$$

where the ν_{i+1}, ν_i are simulated using the QE scheme. As $\int_i^{i+1}\nu(s)\mathrm{d}s = \nu_i h$ and $\int_i^{i+1}\sqrt{\nu_i}d\mathrm{d}\widetilde{W}^X(s)$ is normal distributed with zero mean and variance $\nu_i h$, the equation above can be rearranged as

$$X_{i+1} = X_i + (r - q - \frac{1}{2}\nu_i)h + \frac{\rho}{\gamma}\left[\nu_{i+1} - \nu_i - \kappa h(\bar{\nu} - \nu_i)\right] + \sqrt{1-\rho^2}\sqrt{\nu_i h}Z^X, \tag{5.11}$$

where $Z^X \sim \mathcal{N}(0,1)$. The log price $(X_{i+1}|X_i, \nu_{i+1}, \nu_i)$ is normal distributed.

5.2 The multi-dimensional Heston model

Let us now consider the assets price $\mathbf{S}(t) = (S_1(t), ..., S_d(t)) \in \mathbb{R}^d$ generated using the multi-dimensional Heston model, which is defined on the time interval $[0, T]$. By subdividing the time interval into N steps and denoting t_i as i for all $i = 1, ..., N$, the dynamics of the log price $\mathbf{X}_i = \ln(\mathbf{S}_i)$ is

$$\begin{cases} \mathrm{d}X_{i,n} = \left(r - q_n - \frac{1}{2}\nu_{i,n}\right)\mathrm{d}t + \sqrt{\nu_{i,n}}\mathrm{d}W^X_{i,n} \\ \mathrm{d}\nu_{i,n} = \kappa_n(\bar{\nu}_n - \nu_{i,n})\mathrm{d}t + \gamma_n\sqrt{\nu_{i,n}}\mathrm{d}W^\nu_{i,n} \end{cases}, n = 1, ..., d. \tag{5.12}$$

The correlations of the d-dimensional Heston model can be distinguished into four types:

1. ρ_n: the correlation between $W^X_{i,n}$ and $W^\nu_{i,n}$ just as the one dimensional scenario.
2. ρ_{nl}: the correlation between the log price Brownian motions, i.e. $\mathrm{d}W^X_{i,n}\mathrm{d}W^X_{i,l} = \rho_{nl}\mathrm{d}t$ with $n \neq l$.
3. ρ^ν_{nl}: the correlation between the variance Brownian motions, so $\mathrm{d}W^\nu_{i,n}\mathrm{d}W^\nu_{i,l} = \rho^\nu_{nl}\mathrm{d}t$ with $n \neq l$.
4. $\rho^{X\nu}_{nl}$: the correlation between the n-th asset price and the l-th variance Brownian motions, so $\mathrm{d}W^X_{i,n}\mathrm{d}W^\nu_{i,l} = \rho^{X\nu}_{nl}\mathrm{d}t$, with $n \neq l$.

We only consider the first two types of correlations in further discussion.

Let us define the Brownian motion vector of the Heston model at time step i as

$$\mathrm{d}\mathbf{W}_i = \begin{bmatrix} \mathrm{d}\mathbf{W}^\nu_i \\ \mathrm{d}\mathbf{W}^X_i \end{bmatrix} \in \mathbb{R}^{2d}, \text{ with } \mathrm{d}\mathbf{W}^\nu_i = \begin{bmatrix} \mathrm{d}W^\nu_{i,1} \\ \vdots \\ \mathrm{d}W^\nu_{i,d} \end{bmatrix} \text{ and } \mathrm{d}\mathbf{W}^X = \begin{bmatrix} \mathrm{d}W^X_{i,1} \\ \vdots \\ \mathrm{d}W^X_{i,d} \end{bmatrix}.$$

Then the corresponding correlation matrix $\bar{C} \in \mathbb{R}^{2d\times 2d}$ has the form

$$\bar{C} = \begin{bmatrix} C^\nu & C^{X\nu} \\ C^{\nu X} & C^X \end{bmatrix},$$

where $C^{\nu} = I_d \in \mathbb{R}^{d\times d}$ is the identity matrix since we assume that the variances are independent. C^X denotes the correlation matrix of the underlying assets and $C^{\nu X} = \left(C^{X\nu}\right)^T$ with $C^{X\nu}$ the correlation matrix between the asset price and the variance process. As we are only assuming the first two correlation type, it implies

$$C^X = \begin{bmatrix} 1 & \rho_{12} & \dots & \rho_{1d} \\ \rho_{12} & 1 & \dots & \rho_{2d} \\ \vdots & \vdots & \ddots & \vdots \\ \rho_{1d} & \rho_{2d} & \dots & 1 \end{bmatrix} \text{ and } C^{X\nu} = \begin{bmatrix} \rho_1 & 0 & \dots & 0 \\ 0 & \rho_2 & \dots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & 0 & \rho_d \end{bmatrix}.$$

The Brownian motion vector $\mathrm{d}\mathbf{W}_i$ can be expressed as $\mathrm{d}\mathbf{W}_i = L\mathrm{d}\widetilde{\mathbf{W}}_i$ with $\mathrm{d}\widetilde{\mathbf{W}}_i$ the Brownian motion vector with independent components and L the Cholesky decomposition of $\bar{C} = LL^T$. The lower triangular matrix L can be written in a similar form as the correlation matrix

$$L = \begin{bmatrix} L_1 & 0_d \\ L_2 & \widetilde{L} \end{bmatrix},$$

with $0_d \in \mathbb{R}^{d\times d}$ the zero matrix and all other submatrices have the same dimension as 0_d. Given that L is a lower triangular matrix, the matrix $\widetilde{L} \in \mathbb{R}^{d\times d}$ should be lower triangular as well. Moreover, $L_1 = I_d$ and $L_2 = C^{X\nu}$, see Proposition 2 in Wadman (2010) for the proof. Therefore,

$$\begin{bmatrix} \mathrm{d}\mathbf{W}_i^{\nu} \\ \mathrm{d}\mathbf{W}_i^{X} \end{bmatrix} = \begin{bmatrix} I_d & 0_d \\ C^{X\nu} & \widetilde{L} \end{bmatrix} \begin{bmatrix} \mathrm{d}\widetilde{\mathbf{W}}_i^{\nu} \\ \mathrm{d}\widetilde{\mathbf{W}}_i^{X} \end{bmatrix} = \begin{bmatrix} \mathrm{d}\widetilde{\mathbf{W}}_i^{\nu} \\ C^{X\nu}\mathrm{d}\widetilde{\mathbf{W}}_i^{\nu} + \widetilde{L}\mathrm{d}\widetilde{\mathbf{W}}_i^{X} \end{bmatrix}$$

with independent Brownian motion

$$\mathrm{d}\widetilde{\mathbf{W}}^{\nu} = \begin{bmatrix} \widetilde{W}_{i,1}^{\nu} \\ \vdots \\ \widetilde{W}_{i,d}^{\nu} \end{bmatrix} \text{ and } \mathrm{d}\widetilde{\mathbf{W}}^{X} = \begin{bmatrix} \widetilde{W}_{i,1}^{X} \\ \vdots \\ \widetilde{W}_{i,d}^{X} \end{bmatrix}.$$

Then the dynamic of the log price can be rewritten as:

$$\mathrm{d}X_{i,n} = \left(r - q_n - \frac{1}{2}\nu_{i,n}\right)\mathrm{d}t + \rho_n\sqrt{\nu_{i,n}}\mathrm{d}\widetilde{W}_{i,n}^{X} + \sum_{l=1}^{d}\sqrt{\nu_{i,n}}\widetilde{L}_{n,l}d\widetilde{W}_{i,l}^{X}, \tag{5.13}$$

for an arbitrary $n \in \{1, .., d\}$. Its integral form is

$$X_{i+1,n} = X_{i,n} + \int_i^{i+1}\left(r - q_n - \frac{1}{2}\nu_{i,n}\right)ds + \rho_n \underbrace{\int_i^{i+1}\sqrt{\nu_{i,n}}d\widetilde{W}_n^{\nu}(s)}_{=c_1} + \sum_{l=1}^{d}\widetilde{L}_{n,l}\int_i^{i+1}\sqrt{\nu_{i,n}}d\widetilde{W}_l^{X}(s).$$

The term c_1 can be expressed using Eq.(5.3), thus

$$X_{i+1,n} = X_{i,n} + (r - q_n)h + \frac{\rho_n}{\gamma_n}\left[\nu_{i+1,n} - \nu_{i,n} - \kappa_n\bar{\nu}_n h\right] + \left(\frac{\kappa_n\rho_n}{\gamma_n} - \frac{1}{2}\right)\int_i^{i+1}\nu_n(s)\mathrm{d}s + \sum_{l=1}^{d}\widetilde{L}_{m,j}\int_i^{i+1}\sqrt{\nu_{i,n}}\mathrm{d}\widetilde{W}_l^{X}(s). \tag{5.14}$$

Since $\int_i^{i+1} \nu_n(s)\mathrm{d}s = \nu_{i,n}h$ and $\int_i^{i+1} \sqrt{\nu_{i,n}}\mathrm{d}\widetilde{W}_l^X(s) \sim \mathcal{N}(0, \nu_{i,n}h)$, we obtain a similar expression for $X_{i+1,n}$ as in Eq.(5.11)

$$X_{i+1,n} = \underbrace{X_{i,n} + (r - q_n - \frac{1}{2}\nu_{i,n})h + \frac{\rho_n}{\gamma_n}\left[\nu_{i+1,n} - \nu_{i,n} - \kappa_n h(\bar{\nu}_n - \nu_{i,n})\right]}_{\mu_{i+1,n}} + \sum_{l=1}^{d} \widetilde{L}_{n,l}\sqrt{\nu_{i,n}h}Z_{i,l}^X, \tag{5.15}$$

where $Z_{i,l}^X \overset{iid}{\sim} \mathcal{N}(0,1)$, for $l = 1, ..., d$. Let us define matrix $D_i \in \mathbb{R}^{d\times d}$ as

$$D_i = \begin{bmatrix} \sqrt{\nu_{i,1}} & & \\ & \ddots & \\ & & \sqrt{\nu_{i,d}} \end{bmatrix} \widetilde{L},$$

then Eq.(5.15) can be written in matrix form:

$$\mathbf{X}_{i+1} = \boldsymbol{\mu}_{i+1} + D_i\sqrt{h}\mathbf{Z}_i^X,$$

with $\mathbf{Z}_i^X = \begin{bmatrix} Z_{i,1} & \cdots & Z_{i,d} \end{bmatrix}^T$ and $Z_{i,l} \overset{iid}{\sim} \mathcal{N}(0,1)$. For $\boldsymbol{\mu}_{i+1} = \begin{bmatrix} \mu_{i+1,1} & \cdots & \mu_{i+1,d} \end{bmatrix}^T$, we need the variances $\boldsymbol{\nu}_{i+1}$ and $\boldsymbol{\nu}_{i+1}$ which can be simulated independently using the QE scheme. It is clear that $(\mathbf{X}_{i+1}|\mathbf{X}_i, \boldsymbol{\nu}_{i+1}, \boldsymbol{\nu}_i)$ follows a multivariate normal distribution with mean $\boldsymbol{\mu}_{i+1}$ and variance $h\Sigma_i^H$, where $\Sigma_i^H = D_i D_i^T$.

5.2.1 KRR method under Heston

The NKRR can be applied to American option pricing under the multi-dimensional Heton model without any modifications. The estimation of the continuation value using LKRR needs to be changed since the covariance matrix is now dependent on the CIR process. The CIR process can be simulated independently under our assumption about the correlations. Therefore, Lemma 3.2.1 still applies to the Heston model as $\left(\mathbf{X}_{i+1} - \mathbf{X}_{i+!}^{(m)}\right)|\mathbf{X}_i, \boldsymbol{\nu}_{i+1}, \boldsymbol{\nu}_i \sim \mathcal{N}(\boldsymbol{\mu}_{i+1} - \mathbf{X}_{i+1}^{(m)}, h\Sigma_i^H)$. The expression of the continuation value using LKRR is similar to Eq.(3.12)

$$\begin{aligned}\hat{Q}_i(\mathbf{X}_i) &= D_{i,i+1}\sum_{m=1}^{M} \hat{\alpha}_{i+1,m}\mathbb{E}\left[\mathcal{K}(\mathbf{X}_{i+1}, \mathbf{X}_{i+1}^{(m)})|\mathbf{X}_i, \nu_{i+1}, \boldsymbol{\nu}_i\right] \\ &= \frac{D_{i,i+1}}{|\frac{2}{C}h\Sigma_i^H{}_i + I_d|^{\frac{1}{2}}}\sum_{m=1}^{M} \hat{\alpha}_{i+1,m}e^{-\frac{1}{C}\left(\boldsymbol{\mu}_{i+1} - \mathbf{X}_{i+1}^{(m)}\right)^T(\frac{2}{C}h\Sigma_i^H + I_d)^{-1}\left(\boldsymbol{\mu}_{i+1} - \mathbf{X}_{i+1}^{(m)}\right)}.\end{aligned} \tag{5.16}$$

Chapter 6

Experiments: The Heston model

This section will examine the pricing accuracy and efficiency of KRR-based approaches for American option pricing under the Heston model. We begin with one-dimensional cases in which the Feller condition is satisfied or violated. The results will then be evaluated against the Premia benchmarks. No reliable benchmarks exist for the American option pricing under the high-dimensional Heston model, not even in Premia. Recall that the benchmarks of Premia are acquired utilizing the primal-dual method. According to Andersen and Broadie (2004), this approach is applicable to any process dynamics. Therefore, we will elaborate on this approach and implement it ourselves. To ensure the correctness of our implementation, we will compare our results for the pricing of American options under the GBM model with the benchmarks of Premia. Finally, we will investigate the accuracy of the KRR-based methods under the Heston model using our results of Primal-dual as benchmarks. The hyperparameters P and $lambda$ used in previous tests, i.e., $P = 100$ and $lambda = 1$, will also be used in the experiments in this chapter.

6.1 One dimensional Heston

This section presents two experiments of the American put option under one dimensional Heston model, where the Feller condition is satisfied or violated. Recall the Feller condition $2\kappa\bar{\nu} \geq \gamma^2$. The two sets of parameters used are study cases from Fang and Oosterlee (2011).

6.1.1 Heston with satisfied Feller condition

The Heston model parameters that satisfy the Feller condition are stated in Table 6.1.

S_0	K_s	r	q	ν_0	$\bar{\nu}$	γ	κ	ρ	T	N	M
10	10	10%	0%	0.0625	0.16	0.9	5	0.1	0.25	50	10,000

Table 6.1: Parameters of 1d Heston with satisfied Feller condition.

The LSM/LSMP methods utilize the same basis functions as the experiments before. The Gaussian kernel parameters are manually found using the grid search approach as explained in subsection 3.2.4. The corresponding Gaussian kernel parameters are $C_{nkrr} = 0.95$ and $C_{lkrr} = 50$.

The pricing results of those four methods obtained from 10 independent iterations are shown in Table 6.2. The Premia benchmark is estimated using Premia, where the primal-dual method of Andersen and Broadie (2004) is used with the same setting as experiments before. We observe

that the KRR-based approach provides better pricing results than the LSMC-based methods. Among NKRR and LKRR, the pricing accuracy of NKRR is better than the LKRR method. It indicates that KRR-based approaches are applicable under the one-dimensional Heston model and NKRR is the most accurate.

Premia	LSM	Error(%)	LSMP	Error(%)	NKRR	Error(%)	LKRR	Error(%)
0.5278	0.5214	1.200	0.5228	0.946	0.5287	0.161	0.5309	0.588

Table 6.2: Pricing accuracy of 1d Heston with satisfied Feller condition.

6.1.2 Heston with violated Feller condition

Let us consider the following parameters where the Feller condition is violated.

S_0	K_s	r	q	ν_0	$\bar{\nu}$	γ	κ	ρ	T	P	N	M
100	100	3%	5%	0.0348	0.0348	0.39	1.15	-0.64	1	100	10	$10,000$

Table 6.3: Parameters of 1d Heston with violated Feller condition.

The Gaussian kernel parameters are $C_{nkrr} = 450$ and $C_{lkrr} = 380$ are found using grid search. Then the pricing accuracy is as follows

Premia	LSM	Error(%)	LSMP	Error(%)	NKRR	Error(%)	LKRR	Error(%)
7.2109	6.8057	5.619	7.1822	0.398	7.2251	0.197	7.2373	0.366

Table 6.4: Pricing accuracy of 1d Heston with violated Feller condition.

The LSM is inaccurate under this set of parameters. The pricing accuracy can be improved by choosing another set of basis functions, as we can conclude from the pricing result of LMSP. The same conclusion can be drawn regarding the KRR-based methods as the previous experiment: the KRR-based methods are better than LSMC. The accuracy of LSMP is similar to LKRR, but the latter performs slightly better. NKRR is the best among all four approaches, as it has the lowest pricing error.

6.2 Multi-dimensional Heston

Since no reliable price reference exists for the valuation of high-dimensional American options under the Heston model (not in the literature or in Premia), we must design our test cases carefully. Recall all the benchmarks from Premia that we used for the experiments are actually results of the primal-dual method proposed by Andersen and Broadie (2004). We will first discuss more details about this primal-dual approach in this section. The pricing results under the GBM model acquired using self-implemented primal-dual and the pricing outcomes obtained using Premia will then be compared to assess our implementation's accuracy. Finally, we will use self-implement primal-dual benchmarks to examine the price accuracy of the KRR-based approaches.

6.2.1 The primal-dual simulation algorithm

The primal-dual approach can be interpreted as computing the option values from the perspective of the option holder and the seller. The primal problem can be seen as determining the best exercise strategy to maximize the payoff, which is the optimal stopping problem we previously derived. Instead, the seller needs to estimate the minimum amount required to cover the holder's payout. Thus, the solution to those two problems provides the lower and upper bounds where the actual option prices belong.

Let $\mathbf{S}_t \in \mathbb{R}^d$ denotes the asset price with $t \in [0, T]$ and the set of possible exercise dates is defined as $\mathcal{T} = \{t_1, ..., t_N\}$ with step size $h = \frac{T}{N}$. The primal problem is the same as the optimal stopping problem derived before

$$\text{Primal: } V_0 = \sup_{\tau \in \mathcal{T}} \mathbb{E}[D_{0,\tau} h_\tau(\mathbf{S}_\tau) | \mathcal{F}_0],$$

where $D_{0,t} = \frac{M_0}{M_t}$ and $M_t = \mathrm{e}^{rt}$.

The main idea of the dual approach is to minimize over a class of martingales. Let H denotes the set of all adapted martingales π such that $\sup_{t \in \mathcal{T}} |\pi_t| < \infty$ and $\pi_0 = 0$. Then for any $\pi \in H$

$$\begin{aligned} V_0 &= \sup_{\tau \in \mathcal{T}} \mathbb{E}\left[D_{0,\tau} h_\tau + \pi_\tau - \pi_\tau | \mathcal{F}_0\right] = \pi_0 + \sup_{\tau \in \mathcal{T}} \mathbb{E}\left[D_{0,\tau} h_\tau - \pi_\tau | \mathcal{F}_0\right] \\ &\leq \pi_0 + \mathbb{E}\left[\max_{t \in \mathcal{T}} \left(D_{0,t} h_t - \pi_t\right) | \mathcal{F}_0\right]. \end{aligned} \tag{6.1}$$

The second equality is due to the martingale property of π_t and the optional sampling theorem. The inequality follows from Jensen's inequality. This inequality still holds when taking the infimum over the set H

$$V_0 \leq \inf_{\pi \in H} \left(\pi_0 + \mathbb{E}\left[\max_{t \in \mathcal{T}} \left(D_{0,t} h_t - \pi_t\right) | \mathcal{F}_0\right]\right). \tag{6.2}$$

as Eq.(6.1) holds for any arbitrary martingale.

Recall the option value at t_i is

$$V_i(\mathbf{S}_i) = \max\{h_i(\mathbf{S}_i), \mathbb{E}[D_{i,i+1} V_{i+1} | \mathcal{F}_i]\},$$

for $i = 0, ..., N-1$. By multiplying the option value with the discounted factor $D_{0,i}$, the following yields

$$D_{0,i} V_i = \max\{D_{0,i} h_i, \mathbb{E}[D_{0,i+1} V_{i+1} | \mathcal{F}_i]\}.$$

Andersen and Broadie (2004) claims the process $D_{0,t} V_t$ is the smallest supermartingale that dominates the discounted payoff process $D_{0,t} h_t$. Then the corresponding Doob-Mayer decomposition of this supermartingale is given by

$$D_{0,t} V_t = \widetilde{M}_t - A_t,$$

where $\widetilde{M}_t$ is the martingale and A_t an increasing process with $A_0 = 0$.

Equality holds in Eq.(6.1) if we take $\pi_t = \widetilde{M}_t$, with $\widetilde{M}_t = D_{0,t} V_t + A_t$ and $\widetilde{M}_0 = V_0$. Since substituting this expression into Eq.(6.1), we obtain

$$V_0 \leq V_0 + \mathbb{E}\left[\max_{t \in \mathcal{T}} \left(D_{0,t} h_t - D_{0,t} V_t - A_t\right) | \mathcal{F}_0\right] \leq V_0.$$

The second inequality holds as $D_{0,t}V_t$ dominates $D_{0,t}h_t$, i.e., $D_{0,t}V_t \geq D_{0,t}h_t$ and A_t is an increasing process. Then the dual problem can be formulated as

$$\text{Dual: } V_0 = \inf_{\pi \in H} \left(\pi_0 + \mathbb{E}\left[\max_{t \in \mathcal{T}} \left(D_{0,\tau} h_\tau - \pi_t \right) | \mathcal{F}_0 \right] \right). \tag{6.3}$$

Nevertheless, finding the optimal martingale is difficult as we generally do not know how the martingale involves over time. Andersen and Broadie (2004) suggest to construct π_t as the martingale part of the approximation of $D_{0,t}V_t$. The lower bound price process $\hat{V}_t$ can be used to approximate V_t. Then $D_{0,t}V_t$ can be approximated by $L_t = D_{0,t}\hat{V}_t$. We aim to compute the martingale part of the following price process

$$\{ L_i = \mathbb{E}\left[D_{0,\tau_i} h_{\tau_i} \mid \mathcal{F}_i \right], i = 1, \ldots, N \},$$

where τ_i are the exercise policy of the lower bound price process.

Glasserman (2010) proposed a simplified version of Andersen and Broadie (2004) where the martingale can be constructed as follows

$$\begin{cases} \hat{\pi}_0 = L_0 & \text{(6.4a)} \\ \hat{\pi}_i = \hat{\pi}_{i-1} + L_i - \mathbb{E}\left[L_i \mid \mathcal{F}_{i-1} \right] \quad \text{for } i = 1, \ldots, N. & \text{(6.4b)} \end{cases}$$

We can verify that the obtained process $\hat{\pi}_t$ indeed satisfies the martingale property:

$$\begin{aligned} \mathbb{E}\left[\hat{\pi}_i \mid \mathcal{F}_{i-1} \right] &= \mathbb{E}\left[\hat{\pi}_{i-1} + L_i - \mathbb{E}\left[L_i \mid \mathcal{F}_{i-1} \right] \mid \mathcal{F}_{i-1} \right] \\ &= \hat{\pi}_{i-1} + \mathbb{E}\left[L_i \mid \mathcal{F}_{i-1} \right] - \mathbb{E}\left[L_i \mid \mathcal{F}_{i-1} \right] \\ &= \hat{\pi}_{i-1} \end{aligned}$$

for $i = 1, ..., N$.

Let us denote the difference as $\Delta_i = L_i - \mathbb{E}\left[L_i \mid \mathcal{F}_{i-1} \right]$. Note

$$L_i = \mathbb{E}\left[D_{0,\tau_i} h_{\tau_i} \mid \mathcal{F}_i \right] = \begin{cases} D_{0,i} h_i, & \text{if } \; h_i \geq Q_i \\ \mathbb{E}\left[L_{i+1} \mid \mathcal{F}_i \right], & \text{else} \end{cases}.$$

Since the continuation value $Q_N = 0$ which implies $h_N \geq Q_N$, the only term that we need to estimate is $\mathbb{E}\left[L_i \mid \mathcal{F}_{i-1} \right]$ for $i = 1, ..., N$. The algorithm for finding the By using the tower property,

$$\mathbb{E}\left[L_i \mid \mathcal{F}_{i-1} \right] = \mathbb{E}\left[\mathbb{E}\left[D_{0,\tau_i} h_{\tau_i} \mid \mathcal{F}_i \right] \mid \mathcal{F}_{i-1} \right] = \mathbb{E}\left[D_{0,\tau_i} h_{\tau_i} \mid \mathcal{F}_{i-1} \right].$$

The term $\mathbb{E}\left[D_{0,\tau_i} h_{\tau_i} \mid \mathcal{F}_{i-1} \right]$ will be simulated using nested Monte Carlo simulation and the difference can be computed using

$$\Delta_i = \begin{cases} D_{0,i} h_i - \mathbb{E}\left[D_{0,\tau_i} h_{\tau_i} \mid \mathcal{F}_{i-1} \right], & \text{if } \; h_i \geq Q_i \\ \mathbb{E}\left[D_{0,\tau_{i+1}} h_{\tau_{i+1}} \mid \mathcal{F}_i \right] - \mathbb{E}\left[D_{0,\tau_i} h_{\tau_i} \mid \mathcal{F}_{i-1} \right], & \text{else} \end{cases}. \tag{6.5}$$

.

Once the martingale $\hat{\pi}_t$ is constructed, the duality gap between the lower and upper bound can be calculated as

$$G_0 = \mathbb{E}\left[\max_{t \in \mathcal{T}} \left(D_{0,t} h_t - \hat{\pi}_t \right) | \mathcal{F}_0 \right].$$

The corresponding upper bound $U_0 = L_0 + G_0$.

Computing the upper bound

In order to construct the martingale, the only conditional expectation that we have to estimate is $\mathbb{E}\left[D_{0,\tau_{k+1}} h_{\tau_{k+1}} \mid \mathcal{F}_k\right]$, for $k = 0, ..., N-1$. Andersen and Broadie (2004) proposes to estimate it using M_{in} nested Monte Carlo simulations within an outer Monte Carlo simulation.

Suppose we generate M_{out} sample paths for computing the lower bound price process. In addition, we will estimate the optimal exercise strategy and the corresponding continuation values Q_i at each timestep $i \in \{1, ..., N\}$ using the LSMC method. Then the upper bound computation proceeds by repeating the following steps M_{out} times

1. Let $\mathbf{S}_0, ..., \mathbf{S}_N$ with $\mathbf{S}_j \in \mathbf{R}^d$ be an arbitrary simulated outer path, for $j = 0, ..., N$.
2. At each timestep $k = 0, ..., N-1$
   - Simulate N_{in} subpaths $\hat{\mathbf{S}}_k, ..., \hat{\mathbf{S}}_N$ with $\hat{\mathbf{S}}_k = \mathbf{S}_k$ as the initial value.
   - Use the LSMC method for computing the discounted payoff $D_{0,\tau_{k+1}} h_{\tau_{k+1}}\left(\hat{\mathbf{S}}_{\tau_{k+1}}\right)$ of each inner simulation and use the average of the discounted payoff as an estimate of $\mathrm{E}\left[D_{0,\tau_{k+1}} h_{\tau_{k+1}}\left(\mathbf{S}_{\tau_{k+1}}\right) \mid \mathcal{F}_k\right]$.
   - Store the estimated conditional expectation in matrix $E \in \mathbb{R}^{M_{out} \times N}$.
3. Use matrix E and the pre-determined continuation values to computes Δ_i as in Eq.(6.5). Then calculate the martingale $\hat{\pi}_i$ using Eq.(6.4).
4. Determine the duality gap $G = \max_{t \in \mathcal{T}} (D_{0,t} h_t - \hat{\pi}_t)$.

The output of this procedure is a vector of duality gaps $\mathbf{G} \in \mathbf{R}^{M_{out}}$. We use the average as an estimate for the duality gap G_0, and the upper bound is calculated by $U_0 = L_0 + G_0$, where L_0 is the low bound price computed using the outer simulation paths.

Implementation of primal-dual algorithm

Andersen and Broadie (2004) suggests to compute the lower bound $\hat{L}_0$ and the duality gap $\hat{G}_0$ using independent simulations. Furthermore, the point estimate offers a more accurate price estimate than either the lower limit or higher bound by themselves(Andersen and Broadie, 2004). The summary of the implementation can be found in Algorithm 3.

Algorithm 3 The primal-dual algorithm

Given M_1, M_2, M_{out} and M_{in} as the numbers of Monte Carlo simulations. The number of exercise opportunities is N.

1: Simulate M_1 sample paths and use the LSMC method to estimate the regression coefficients and store them for later usage.

2: Simulate M_2 sample paths and use the estimated regression coefficients to compute the lower bound estimate $\hat{L}_0$.

3: Use M_{out} outer simulations and M_{in} inner simulations for computing the duality gap $\hat{G}_0$ as explained before.

4: Compute the point estimate $\hat{V}_0 = \hat{L}_0 + \frac{1}{2}\hat{G}_0$ and the corresponding upper bound $\hat{U}_0 = \hat{L}_0 + \hat{G}_0$.

The first two steps are done to remove the forecasting bias that may occur in the Monte Carlo pricing of Bermudan options (Fries, 2005). Longstaff and Schwartz (2001) propose only using the ITM paths for the calculation of the option price. We will include all paths for the

LSMC method since the estimated regression coefficients influence the computation of the upper and lower bounds.

6.2.2 Accuracy of the self-implemented primal-dual algorithm

In this subsection, we will analyze the accuracy of the self-implemented primal-dual algorithm using the GBM model with the same parameters as in section 4.1. Recall the used parameters are:

$S_{0,v}$	K_s	σ_v	r	q_v	ρ_{vl}	T	N
100	100	0.2	5%	10%	0	3	3

Table 6.5: Parameters for GBM.

The results of Premia are based on $M = 5 \times 10^4$ simulations and use LSMC with Hermite polynomial from degree $0 - 3$ as basis functions for computing the lower bound. Moreover, $N_{out} = 500$ and $N_{in} = 100$ for the duality approach. To keep the same setting as Premis, we set $M_1 = M_2 = 5 \times 10^4$ for the self-implemented primal-dual algorithm. The pricing result of the self-implemented primal-dual approach using 60 independent iterations is presented in Table 6.6. The values between the brackets are the standard deviations. The error rate is the relative error between Premia and the obtained point estimate.

d	Premia	Lower	Upper	Gap	Estimate	Error(%)
5	25.306	25.1196(0.0792)	25.3550(0.0816)	0.2354(0.1215)	25.2373(0.0526)	0.2716
10	37.698	37.1744(0.0786)	37.4550(0.1071)	0.2807(0.1261)	37.3147(0.0696)	1.0168
15	45.569	44.9750(0.0946)	45.1869(0.1142)	0.2120(0.1479)	45.0810(0.0743)	1.0710
20	51.443	50.5994(0.1095)	50.8231(0.1112)	0.2237(0.1438)	50.7112(0.0837)	1.4225
30	59.775	58.6089(0.0995)	58.8226(0.1219)	0.2137(0.1652)	58.7158(0.0745)	1.7720
40	65.525	64.3290(0.1057)	64.5465(0.1155)	0.2175(0.1495)	64.4378(0.0817)	1.6592
60	73.900	72.4333(0.1067)	72.6201(0.0999)	0.1868(0.1504)	72.5267(0.0709)	1.8583
80	79.908	78.1792(0.1004)	78.3752(0.1088)	0.1961(0.1393)	78.2772(0.0781)	2.0409
100	84.501	82.6925(0.1350)	82.8653(0.1185)	0.1728(0.1802)	82.7789(0.0895)	2.0379

Table 6.6: Comparison self-implemented Primal-dual with Premia.

As we can conclude from this table, the price estimates using the self-implemented primal-dual simulation algorithm are generally lower than the benchmarks. Moreover, the relative error with respect to Premia increases in dimension. The reason may be the loose lower bounds we get as the dimension increases. Premia's documentation does not clarify whether they have used variance reduction or other improvement methods. In conclusion, our implementation provides biased pricing results. As the pricing errors are above 2% for $d = 80$ and $d = 100$, we will exclude those two dimensions to examine the pricing accuracy of NKRR and LKRR under the Heston model.

6.2.3 Multi-Heston with satisfied Feller condition

To investigate the pricing accuracy of the KRR-based approaches under the Heston model, we will use the results of the self-implemented primal-dual as the benchmark prices. Since those benchmark prices are biased, as concluded in the previous subsection, we will only verify whether NKRR and LKRR are applicable for high American option pricing under the Heston model. It is hard to conclude which method is the best with biased benchmarks. For low dimensions $d \leq 15$, we can conclude the best method among all four methods as the pricing results in the previous subsection are acceptable.

We keep the same settings as much as feasible from the GBM numerical experiments with the hoop to reduce the computational time for finding the appropriate Gaussian kernel parameters C_{nkrr} and C_{lkrr}. Those of GBM may be used as initial values for the grid search. Table 6.7 lists the set of parameters that satisfy the Feller criterion.

S_{0_v}	K_s	σ_v	r_v	q_v	$\nu_{0,v}$	$\bar{\nu}_v$	γ_v	κ_v	$\rho_v^{\nu S}$	ρ_v^S	T	N	M
100	100	0.2	5%	10%	0.03	0.04	0.1	1	0.1	0	3	3	10000

Table 6.7: Parameters for Heston with satisfied Feller condition.

Pricing accuracy and computational time

Recall the number of bundles $P = 100$ and the ridge parameter $\lambda = 1$ for the KRR-based methods. Moreover, the LSMC methods in future experiments utilize the same basis functions as previous experiments: polynomial up to degree 2 denoted as LSM, and LSMP has payoff as an additional basis function.

Although the assessment of $d = 80, 100$ regarding pricing accuracy is pointless due to the significant bias in our benchmarks, we can still examine those dimensions in terms of computing time. We observe similar results as for the other two models. For $d \leq 40$, the running time of LKRR is the highest. As dimension increases further, KRR-based methods are more efficient than LSMC methods. NKRR performs the best in running time and is slightly influenced by the dimensions, indicated in Figure 6.1 and Table 6.8.

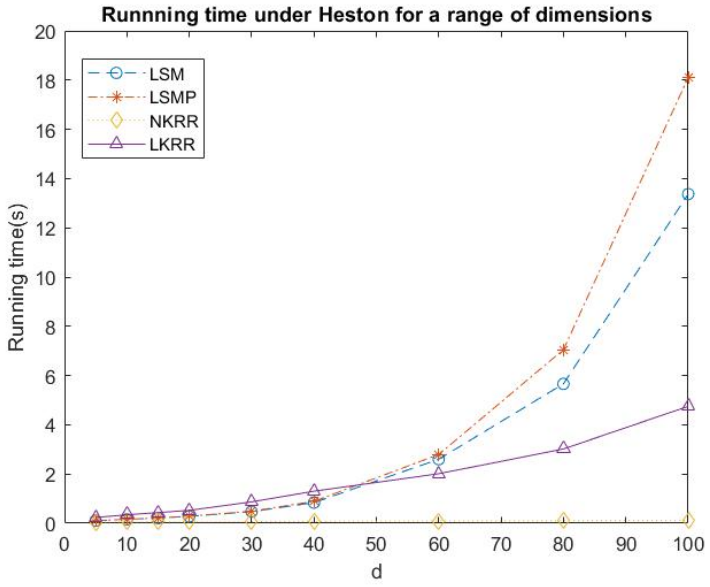


Figure 6.1: Computational time under Heston.

d	5	10	15	20	30	40	60	80	100
LSM	0.1060	0.1607	0.2096	0.2812	0.4710	0.8393	2.6016	5.6627	13.3707
LSMP	0.1106	0.1598	0.2177	0.2878	0.4904	0.8925	2.7940	7.0514	18.1013
NKRR	0.0446	0.0534	0.0586	0.0572	0.0673	0.0756	0.0928	0.1044	0.1236
LKRR	0.2351	0.3448	0.4346	0.5246	0.8696	1.3023	2.0124	3.0179	4.7500

Table 6.8: Heston: Running time in (s) over dimensions.

We use the self-implemented primal-dual algorithm as the benchmark for the pricing results, shown under the column Benchmark in the Table 6.9. The benchmark is obtained using 10 independent iterations. The pricing results of those four methods are based on 60 independent iterations. In between the brackets are the standard deviations.

d	Benchmark	LSM	Error(%)	LSMP	Error(%)	NKRR	Error(%)	LKRR	Error(%)
5	21.9697 (0.0776)	21.9784(0.1739)	0.0396	21.9985(0.1786)	0.1311	21.9864(0.1854)	0.0761	21.9523(0.1745)	0.0789
10	32.9357 (0.0691)	33.0254(0.2674)	0.2723	33.0835(0.2693)	0.4487	33.0080(0.2476)	0.2196	33.0519(0.2344)	0.3527
15	40.1964 (0.0576)	40.2637(0.1746)	0.1674	40.3570(0.1772)	0.3995	40.2171(0.1754)	0.0515	40.3298(0.1921)	0.3319
20	45.4648 (0.0948)	45.5425(0.1975)	0.1709	45.6637(0.1878)	0.4373	45.4855(0.1805)	0.0454	45.6207(0.1901)	0.3427
30	53.0932 (0.0859)	53.3156(0.2310)	0.4189	53.4910(0.2245)	0.7493	53.2420(0.2172)	0.2804	53.3598(0.2161)	0.5022
40	58.4968 (0.0483)	58.9207(0.2240)	0.7246	59.0955(0.2224)	1.0234	58.7787(0.2406)	0.4818	58.8385(0.2316)	0.5840
60	66.2300 (0.0693)	67.0923(0.2226)	1.3019	67.2161(0.2288)	1.4888	66.8229(0.2352)	0.8952	66.7076(0.2274)	0.7210

Table 6.9: Pricing results under Heston with $C_{nkrr} = 1.5 \cdot 10^5$ and $C_{lkrr} = 100$.

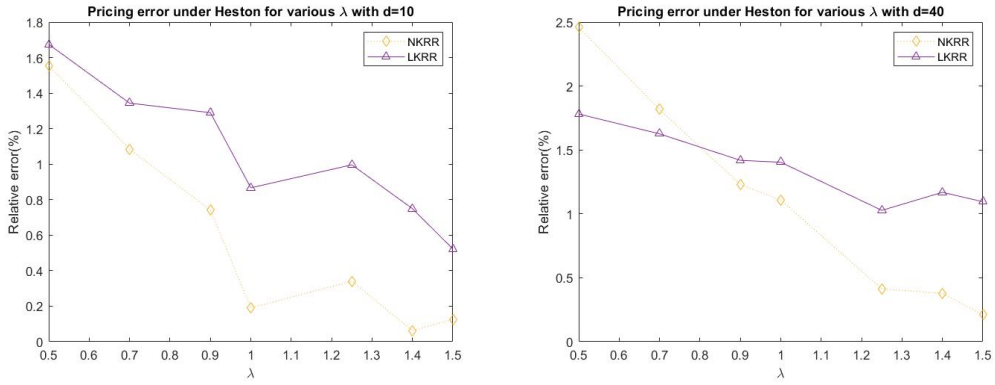
As expected, the pricing errors of all methods are increasing in dimensions as the benchmarks become more biased than lower dimensions. From the accuracy test in subsection 6.2.2, we can only guarantee the correctness of the relative error of those four methods for $d = 5$. The KRR-based methods produce accurate results in this dimension. Moreover, we have concluded that the result of the self-implemented algorithm is generally lower than the Premia results. Therefore, the benchmarks should be greater than what we stated in the table above, which might lead to a smaller relative error for all four methods, especially when $d \geq 30$. The NKRR method is the best for the lower dimensions ($d \leq 15$) among those four methods if we only look at the current pricing results and computational costs.

The pricing results of KRR-based methods are similar to those of LSMC-based methods, as the Gaussian kernel parameters are calibrated accordingly. It is done on purpose, as the benchmarks are biased. Those similar results indicate that KRR-based approaches can provide accurate pricing results once appropriate hyperparameters are found. Considering the calibration process of the hyperparameters, the NKRR is a better choice than LKRR since the running time of NKRR is only slightly affected by the dimensions. Additionally, KRR-based algorithms require less running time in higher dimensions compared to LSMC. When the calibration and pricing running time is shorter than LSMC, KRR-based methods perform better as they can provide similar results with cheaper computational costs. The C_{nkrr} and C_{lkrr} used in this chapter are manually found utilizing grid search. A more efficient way for finding the Gaussian kernels should be designed. In conclusion, LKRR is robust under varying Gaussian kernel and ridge parameters, while NKRR is less robust for the latter one, especially with high dimensions.

The influence of hyperparameters

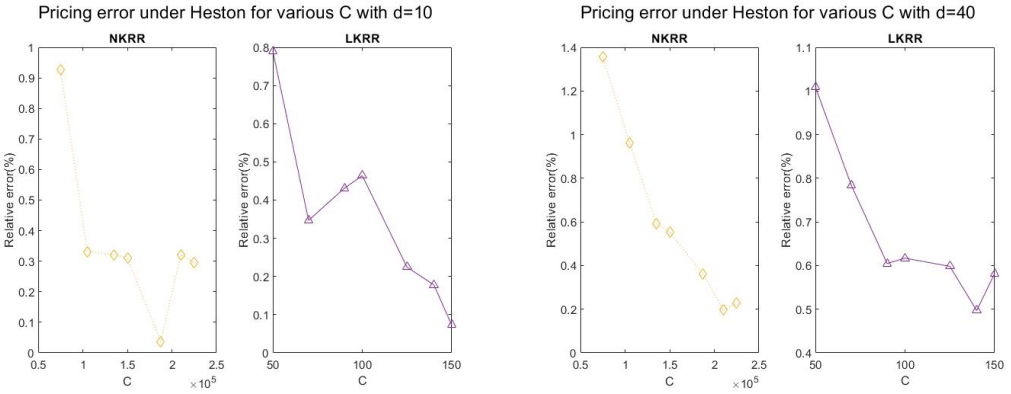
Even though the current pricing errors are biased, we can still investigate the robustness of KRR-based methods. If the error rates are not fluctuating between wide intervals, we can conclude that those two methods are not sensitive to varying hyperparameters. We select $d = 10$ and $d = 40$ as representatives for the low and high dimensions. The KRR hyperparameters C and λ are multiplied with a scale vector $s = [0.5, 0.7, 0.9, 1, 1.25, 1.4, 1.5]$.

In Figure 6.2, the pricing error is plotted against a range of λ for both dimensions. The NKRR approach is more sensitive to a varying ridge parameter than LKRR, especially for high dimensions. Optimal choices for λ exist, but we cannot conclude from the current results as the benchmarks are biased. On the contrary, those two methods are less influenced by the choices of the Gaussian kernel parameters in both dimensions, as shown in Figure 6.3.



(a) $d = 10$ (b) $d = 40$

Figure 6.2: Pricing accuracy under Heston for various λ.



(a) $d = 10$ (b) $d = 40$

Figure 6.3: Pricing accuracy under Heston for various C.

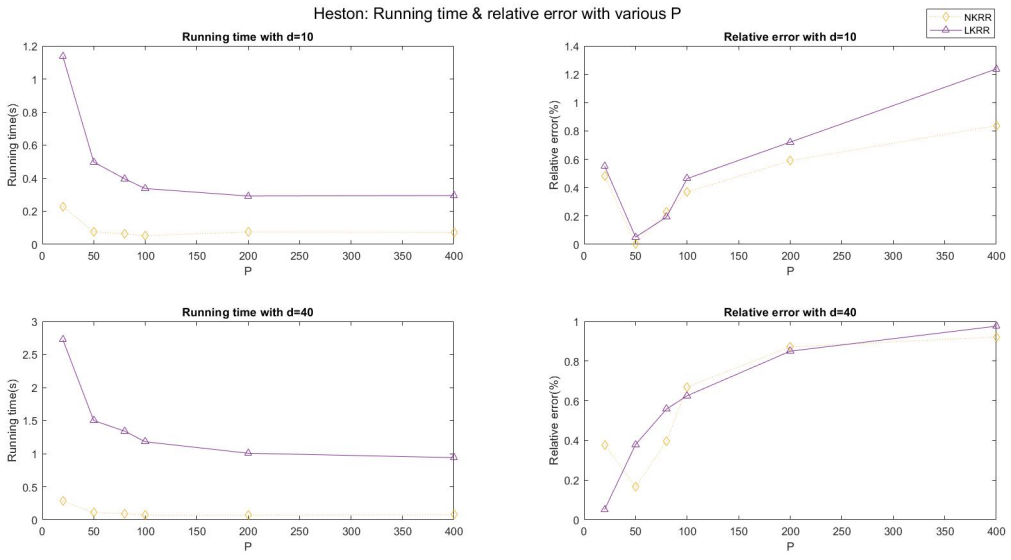


Figure 6.4: Running time and pricing error under Heston with various P.

Figure 6.4 presents the computing time and pricing accuracy against $P = [20, 50, 80, 100, 200, 400]$. We observe similar phenomena for the running time and the error rate as for the other two models. The running time decreases as the P increases, and it is only slightly affected by P for $P \geq 100$. The right sugfigures indicate the existence of optimal P for both dimensions. Moreover, the pricing error increases for $P \geq 100$. As the relative error varies within a small range of values, we can conclude that KRR-based methods are robust, provided P is not too great.

We have also studied pricing accuracy and efficiency using the Heston model with violated Feller. As the results are similar to the one with satisfied Feller condition, we have set pricing results in Appendix D.

Chapter 7

Conclusion and future research

7.1 Conclusion

The main idea of regression-based Monte Carlo methods for American option pricing is to approximate the continuation values using regression. Regarding the data used for the regression, those methods can be categorized into regression now and later. In this thesis, we investigated the KRR-based methods for valuing high-dimensional American options in various asset price models. As the name suggests, those approaches utilize kernel ridge regression, where regression now and later are denoted as NKRR and LKRR. The Gaussian kernel is used as the kernel function. Moreover, the KRR-based method involves the bundling technique, which divides the data into P non-overlapping bundles. We also incorporated the LSMC (using linear regression) with two sets of basis functions in the experiments to evaluate the pricing accuracy and computing efficiency of the KRR-based approaches.

For the asset prices generated by high-dimensional geometric Brownian motion, the KRR-based methods provide more accurate pricing results than the LSMC methods. For dimensions $d < 40$, all methods have similar computational costs, where LKRR is the slowest. Although the computational time of LKRR increases with dimension, it is still computationally more efficient than LSMC. The LSMC suffers from the curse of dimension, as we can conclude from our experiment. Its running time gains exponentially as the dimension increases. The computational time of NKRR does not influence by the dimension, and its pricing accuracy is slightly worse than LKRR.

The KRR-based methods are definitely a better choice than LSMC for pricing high-dimensional American options in Merton jump-diffusion. LSMC provides inaccurate prices as the dimension grows, while the pricing errors of NKKR and LKRR are up to 1.5%. Moreover, NKRR is better than LKRR concerning computational costs.

No reliable results exist for valuing the high-dimensional American option under the Heston model, neither in the literature nor in the third-party software Premia, which we used for the other two models. For the experiments, we used the self-implemented primal-dual pricing results as benchmarks. Nevertheless, those obtained benchmarks are biased. Therefore, we cannot conclude for sure which of the LSMC and KRR-based methods provides the most accurate pricing results. However, we can conclude that the KRR-based techniques are applicable to the American option pricing for the high-dimensional Heston model. Once the proper hyperparameters are found, the KRR-based approaches can also provide comparable pricing results with lower computing costs than the LSMC method in higher dimensions ($d \geq 40$). NKRR outperforms LKRR in terms of computational efficiency as its running time is far below 1 second among all dimensions $d \leq 100$.

Furthermore, the robustness of the KRR-based methods is investigated for those asset price

models. LKRR is robust both under varying Gaussian kernel parameters and ridge parameters, while NKRR is more sensitive for the latter parameter. The optimal number of bundles P seems to depend on the data size and the asset price model, as different optimal P are obtained for the geometric Brownian motion and Merton jump diffusion.

7.2 Future research

The computational time of NKRR is regardless of dimensions and the models, as we concluded from the experiments. Therefore, the NKRR approach is better than LKRR for practical usage. However, the main drawback of the KRR-based methods is the time-consuming hyperparameter tuning process. One of the suggestions for future research is finding a better hyperparameter optimization strategy to reduce the total running time of the KRR-based methods.

In addition, the primal-dual algorithm we implemented provides biased results. In future research, We can include an improvement strategy to reduce the bias and use the KRR-based methods for the primal-dual algorithm.

Bibliography

Alfonsi, A. (2015). *Simulation of the CIR Process*, pages 67–92. Springer International Publishing, Cham.

Andersen, L. (2008). Simple and efficient simulation of the heston stochastic volatility model. *The Journal of Computational Finance*, 11(3):1–42.

Andersen, L. and Broadie, M. (2004). Primal-dual simulation algorithm for pricing multidimensional american options. *Management Science*, 50(9):1222–1234.

Bishop, C. (2007). *Pattern Recognition and Machine Learning (Information Science and Statistics)*.

Broadie, M. and Yamamoto, Y. (2003). Application of the fast gauss transform to option pricing. *Management Science*, 49(8):1071–1088.

Carriere, J. F. (1996). Valuation of the early-exercise price for options using simulations and nonparametric regression. *Mathematics and Economics*, 19(1):19–30.

Cox, J. C., Ingersoll, J. E., and Ross, S. A. (1985). A theory of the term structure of interest rates. *Econometrica*, 53(2):385–407.

Dufresne, D. (2001). The integrated square-root process.

Exterkate, P. (2013). Model selection in kernel ridge regression. *Computational Statistics Data Analysis*, 68:1–16.

Fang, F. and Oosterlee, C. W. (2011). A fourier-based valuation method for bermudan and barrier options under heston's model. *SIAM Journal on Financial Mathematics*, 2(1):439–463.

Feller, W. (1951). Two singular diffusion problems. *Annals of Mathematics*, 54(1):173–182.

Fries, C. (2005). Foresight bias and suboptimality correction in monte-carlo pricing of options with early exercise: Classification, calculation removal. *SSRN Electronic Journal*.

Glasserman, P. (2010). *Monte Carlo Methods in financial engineering*. Springer.

Glasserman, P. and Yu, B. (2004). Simulation for american options: Regression now or regression later? In Niederreiter, H., editor, *Monte Carlo and Quasi-Monte Carlo Methods 2002*, pages 213–226, Berlin, Heidelberg. Springer Berlin Heidelberg.

Han, G.-S., Kim, B.-H., and Lee, J. (2008). Kernel-based monte carlo simulation for american option pricing.

Heston, S. (1993). A closed-form solution for options with stochastic volatility with applications to bond and currency options. *Review of Financial Studies*, 6:327–343.

Hu, W. and Zastawniak, T. (2020). Pricing high-dimensional american options by kernel ridge regression. *Quantitative Finance*, 20(5):851–865.

Jain, S. and Oosterlee, C. W. (2015). The stochastic grid bundling method: Efficient pricing of bermudan options and their greeks. *Applied Mathematics and Computation*, 269:412–431.

Kung, S. Y. (2014). *Kernel Methods and Machine Learning*. Cambridge University Press.

Longstaff, F. A. and Schwartz, E. S. (2001). Valuing american options by simulation: A simple least-squares approach. *Review of Financial Studies*, pages 113–147.

Merton, R. C. (1976). Option pricing when underlying stock returns are discontinuous. *Journal of Financial Economics*, 3(1):125–144.

Oosterlee, C. W. and Grzelak, L. A. (2020). *Mathematical Modeling and Computation in Finance: With exercises and python and MATLAB computer codes*. World Scientific, 1 edition.

Patnaik, P. B. (1949). The non-central χ^2- and f-distributions and their applications. *Biometrika*, 36(1-2):202–232.

Samimi, O. and Mehrdoust, F. (2018). Pricing multi-asset american option under heston stochastic volatility model. *International Journal of Financial Engineering*, 05:1850026.

Tilley, J. (1993). 111 valuing american options in a path simulation model. *Transactions of the Society of Actuaries*, 45.

Tsitsiklis, J. and Van Roy, B. (2001). Regression methods for pricing complex american-style options. *IEEE Transactions on Neural Networks*, 12(4):694–703.

van Wieringen, W. N. (2015). Lecture notes on ridge regression. *arXiv: Methodology*.

Wadman, W. S. (2010). An advanced monte carlo method for the multi-asset heston model. M.s. thesis, Technology university of Delft, Delft, the Netherlands.

Zhang, Y., Duchi, J., and Wainwright, M. (2015). Divide and conquer kernel ridge regression: A distributed algorithm with minimax optimal rates. *The Journal of Machine Learning Research*, 16(1):3299–3340.

Appendices

Appendix A

Proof: Feynmann-Kac theorem

Proof. Let us consider the term $\frac{V(t,S)}{M(t)}$, with $M(t) = \exp(r(t-t_0))$. So $\frac{V(t,S)}{M(t)} = e^{-r(t-t_0)}V(t,S)$. By using the product rule, we know that

$$d\frac{V(t,S)}{M(t)} = e^{-r(t-t_0)}dV + Vd(e^{-r(t-t_0)})$$

Since $dS(t) = \tilde{\mu}(t,S(t))dt + \tilde{\sigma}(t,S(t))dW^{\mathbb{Q}}(t)$ and $(dS(t))^2 = \tilde{\sigma}^2(t,S(t))dt$, we get

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

We know that for the money saving account $d(e^{-r(t-t_0)}) = -re^{-r(t-t_0)}dt$, then

$$d\frac{V(t,S)}{M(t)} = e^{-r(t-t_0)}\left[\left(\frac{\partial V}{\partial t} + \tilde{\mu}(t,S)\frac{\partial V}{\partial S} + \frac{1}{2}\tilde{\sigma}^2(t,S)\frac{\partial^2 V}{\partial S^2}\right)dt + \tilde{\sigma}\frac{\partial V}{\partial S}dW^{\mathbb{Q}}\right] - re^{-r(t-t_0)}Vdt.$$

By multiplying both sides with $e^{r(t-t_0)}$ and using the assumption that was made, we get the following expression

$$\begin{aligned} e^{r(t-t_0)}d\frac{V(t,S)}{M(t)} &= \underbrace{\left(\frac{\partial V}{\partial t} + \tilde{\mu}(t,S)\frac{\partial V}{\partial S} + \frac{1}{2}\tilde{\sigma}^2(t,S)\frac{\partial^2 V}{\partial S^2} - rV\right)}_{=0}dt + \tilde{\sigma}\frac{\partial V}{\partial S}dW^{\mathbb{Q}} \\ &= \tilde{\sigma}\frac{\partial V}{\partial S}dW^{\mathbb{Q}} \end{aligned}$$

As a result, we get

$$\begin{aligned} d\frac{V(t,S)}{M(t)} &= e^{-r(t-t_0)}\tilde{\sigma}\frac{\partial V}{\partial S}dW^{\mathbb{Q}} \\ \Rightarrow \int_{t_0}^{T} d(e^{-r(t-t_0)}V(t,S) &= \int_{t_0}^{T} e^{-r(t-t_0)}\tilde{\sigma}\frac{\partial V}{\partial S}dW^{\mathbb{Q}} \\ e^{-r(T-t_0)}V(T,S(T)) - V(t_0,S) &= \int_{t_0}^{T} e^{-r(t-t_0)}\tilde{\sigma}\frac{\partial V}{\partial S}dW^{\mathbb{Q}} \end{aligned} \tag{A.1}$$

By taking expectation at both sides of above equation with respect to $\mathbb{Q}$-measure, we get the following expression for the option value at t_0:

$$V(t_0,S) = \mathbb{E}^{\mathbb{Q}}\left[e^{-r(T-t_0)}V(T,S(T)|\mathcal{F}(t_0)\right] - \mathbb{E}^{\mathbb{Q}}\left[\underbrace{\int_{t_0}^{T} e^{-r(t-t_0)}\tilde{\sigma}\frac{\partial V}{\partial S}dW^{\mathbb{Q}}}_{I(t)}|\mathcal{F}(t_0)\right]. \tag{A.2}$$

We know that $I(t_0) = 0$ and $I(t)$ is a martingale, since it has no drift term. Therefore, $\mathbb{E}^{\mathbb{Q}}[I(t)|\mathcal{F}(t_0)] = 0$ for all $t \geq t_0$.As a result, we get

$$V(t_0, S) = \mathbb{E}^{\mathbb{Q}} \left[e^{-r(T-t_0)} H(T, S(T)|\mathcal{F}(t_0) \right], \tag{A.3}$$

note that $V(T, S(T)) = H(T, S(T))$. □

Appendix B

Proof: Lemma 3.2.1

The characteristic function of the multivariate normal is needed to prove this lemma. Therefore, we will first derive the characteristic function of the multivariate normal.

Suppose $\mathbf{X} \sim \mathcal{N}(\boldsymbol{\mu}, \Sigma)$, then its density function $f_{\mathbf{X}}(\mathbf{x})$ is

$$f_{\mathbf{X}}(\mathbf{x}) = \frac{1}{(2\pi)^d |\Sigma|^{\frac{1}{2}}} e^{-\frac{1}{2}(\mathbf{x}-\mu)^T \Sigma^{-1}(\mathbf{x}-\mu)}.$$

By definition, the characteristic function of variable $\mathbf{X}$ is given by

$$\begin{aligned}
\phi_{\mathbf{X}}(\mathbf{t}) = \mathbb{E}[e^{i\mathbf{t}^T\mathbf{x}}] &= \int_{\mathbb{R}^d} e^{i\mathbf{t}^T\mathbf{x}} f_{\mathbf{X}}(\mathbf{x}) d\mathbf{x} = \int_{\mathbb{R}^d} e^{i\mathbf{t}^T\mathbf{x}} \frac{1}{(2\pi)^d |\Sigma|^{\frac{1}{2}}} e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu})} d\mathbf{x} \\
&= \frac{1}{(2\pi)^d |\Sigma|^{\frac{1}{2}}} \int_{\mathbb{R}^d} e^{i\mathbf{t}^T\mathbf{x} - \frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu})} d\mathbf{x} \\
&= \frac{1}{(2\pi)^d |\Sigma|^{\frac{1}{2}}} \int_{\mathbb{R}^d} e^{i\mathbf{t}^T(\mathbf{x}-\boldsymbol{\mu}) - \frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu}) + i\mathbf{t}^T\boldsymbol{\mu} + \frac{1}{2}\boldsymbol{t}^T\Sigma\mathbf{t} - \frac{1}{2}\boldsymbol{t}^T\Sigma\mathbf{t}} d\mathbf{x} \\
&= \frac{1}{(2\pi)^d |\Sigma|^{\frac{1}{2}}} \int_{\mathbb{R}^d} e^{i\mathbf{t}^T\Sigma\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu}) - \frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu}) + i\mathbf{t}^T\boldsymbol{\mu} + \frac{1}{2}\boldsymbol{t}^T\Sigma\Sigma^{-1}\Sigma\mathbf{t} - \frac{1}{2}\boldsymbol{t}^T\Sigma\mathbf{t}} d\mathbf{x} \\
&= \frac{1}{(2\pi)^d |\Sigma|^{\frac{1}{2}}} \int_{\mathbb{R}^d} e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}-i\Sigma\mathbf{t})^T\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu}-\Sigma\mathbf{t}) + i\mathbf{t}^T\boldsymbol{\mu} - \frac{1}{2}\mathbf{t}^T\Sigma\mathbf{t}} d\mathbf{x} \\
&= e^{i\mathbf{t}^T\boldsymbol{\mu} - \frac{1}{2}\mathbf{t}^T\Sigma\mathbf{t}} \underbrace{\int_{\mathbb{R}^d} \frac{1}{(2\pi)^d |\Sigma|^{\frac{1}{2}}} e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}-i\Sigma\mathbf{t})^T\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu}-\Sigma\mathbf{t})} d\mathbf{x}}_{=1} = e^{i\mathbf{t}^T\boldsymbol{\mu} - \frac{1}{2}\mathbf{t}^T\Sigma\mathbf{t}}.
\end{aligned} \tag{B.1}$$

Proof. Since $\mathbf{X} \sim \mathcal{N}(\boldsymbol{\mu}, \Sigma)$, we can rewrite it as $\mathbf{X} = \boldsymbol{\mu} + \mathbf{Y}$ with $\mathbf{Y} \sim \mathcal{N}(\mathbf{0}, \Sigma)$. Then

$$\begin{aligned}
\mathbb{E}[\mathrm{e}^{-\frac{1}{C}\mathbf{X}^T\mathbf{X}}] &= \mathbb{E}[e^{-\frac{1}{C}(\boldsymbol{\mu}+\mathbf{Y})^T(\boldsymbol{\mu}+\mathbf{Y})}] = \mathbb{E}[e^{-\frac{1}{C}\left(\boldsymbol{\mu}^T\boldsymbol{\mu} + 2\boldsymbol{\mu}^T\mathbf{Y} + \mathbf{Y}^T\mathbf{Y}\right)}] \\
&= e^{-\frac{1}{C}\boldsymbol{\mu}^T\boldsymbol{\mu}} \mathbb{E}[e^{-\frac{2}{C}\boldsymbol{\mu}^T\mathbf{Y} - \frac{1}{C}\mathbf{Y}^T\mathbf{Y}}].
\end{aligned} \tag{B.2}$$

Since $\mathbf{Y}$ has zero mean and covariance Σ, the expectation term gives us the following

$$\begin{aligned}
\mathbb{E}[e^{-\frac{2}{C}\boldsymbol{\mu}^T\mathbf{Y}-\frac{1}{C}\mathbf{Y}^T\mathbf{Y}}] &= \frac{1}{(2\pi)^d|\Sigma|^{\frac{1}{2}}}\int_{\mathbb{R}^d} e^{-\frac{2}{C}\boldsymbol{\mu}^T\mathbf{z}-\frac{1}{C}\mathbf{z}^T\mathbf{z}}e^{-\frac{1}{2}\mathbf{z}^T\Sigma^{-1}\mathbf{z}}d\mathbf{z} \\
&= \frac{|\frac{2}{C}I+\Sigma^{-1}|^{\frac{1}{2}}}{(2\pi)^d|\Sigma|^{\frac{1}{2}}|\frac{2}{C}I+\Sigma^{-1}|^{\frac{1}{2}}}\int_{\mathbb{R}^d} e^{-\frac{2}{C}\boldsymbol{\mu}^T\mathbf{z}-\frac{1}{2}\mathbf{z}^T(\frac{2}{C}I+\Sigma^{-1})\mathbf{z}}d\mathbf{z} \\
&= \frac{|\frac{2}{C}I+\Sigma^{-1}|^{\frac{1}{2}}}{(2\pi)^d|\frac{2}{C}\Sigma+I|^{\frac{1}{2}}}\int_{\mathbb{R}^d} e^{-\frac{2}{C}\boldsymbol{\mu}^T\mathbf{z}-\frac{1}{2}\mathbf{z}^T(\frac{2}{C}I+\Sigma^{-1})\mathbf{z}}d\mathbf{z} \\
&= \frac{1}{|\frac{2}{C}\Sigma+I|^{\frac{1}{2}}}\int_{\mathbb{R}^d} \frac{|\frac{2}{C}I+\Sigma^{-1}|^{\frac{1}{2}}}{(2\pi)^d}e^{-\frac{2}{C}\boldsymbol{\mu}^T\mathbf{z}-\frac{1}{2}\mathbf{z}^T(\frac{2}{C}I+\Sigma^{-1})\mathbf{z}}d\mathbf{z}. \qquad \text{(B.3)}
\end{aligned}$$

Let $\hat{\Sigma}^{-1} = \frac{2}{C}I+\Sigma^{-1}$, then $|\hat{\Sigma}|^{-\frac{1}{2}} = |\frac{2}{C}I+\Sigma^{-1}|^{\frac{1}{2}}$. It means the expression above can be rewritten as

$$\mathbb{E}[e^{-\frac{2}{C}\boldsymbol{\mu}^T\mathbf{Y}-\frac{1}{C}\mathbf{Y}^T\mathbf{Y}}] = \frac{1}{|\frac{2}{C}\Sigma+I|^{\frac{1}{2}}}\mathbb{E}[e^{-\frac{2}{C}\boldsymbol{\mu}^T\mathbf{Z}}], \qquad \text{(B.4)}$$

with $\mathbf{Z}\sim\mathcal{N}(\mathbf{0},\hat{\Sigma})$. Note $i^2=-1$, therefore

$$\mathbb{E}[e^{-\frac{2}{C}\boldsymbol{\mu}^T\mathbf{Z}}] = \mathbb{E}[e^{i^2\frac{2}{C}\boldsymbol{\mu}^T\mathbf{Z}}] = \mathbb{E}[e^{(\frac{2}{C}i)i\boldsymbol{\mu}^T\mathbf{Z}}] = \phi_{\mathbf{Z}}\left(\frac{2}{c}i\boldsymbol{\mu}\right).$$

By using the result that we have found in Eq.(B.1), we get

$$\mathbb{E}[e^{-\frac{2}{C}\boldsymbol{\mu}^T\mathbf{Z}}] = e^{-\frac{1}{2}\frac{2}{C}i\boldsymbol{\mu}^T(\frac{2}{C}I+\Sigma^{-1})^{-1}\frac{2}{c}i\boldsymbol{\mu}} = e^{\frac{2}{C^2}\boldsymbol{\mu}^T(\frac{2}{C}I+\Sigma^{-1})^{-1}\boldsymbol{\mu}}$$

and

$$\mathbb{E}[e^{-\frac{2}{C}\boldsymbol{\mu}^T\mathbf{Y}-\frac{1}{C}\mathbf{Y}^T\mathbf{Y}}] = \frac{1}{|\frac{2}{C}\Sigma+I|^{\frac{1}{2}}}e^{\frac{2}{C^2}\boldsymbol{\mu}^T(\frac{2}{C}I+\Sigma^{-1})^{-1}\boldsymbol{\mu}}. \qquad \text{(B.5)}$$

If we substitute the above expression in equation (B.2), the following can be obtained:

$$\mathbb{E}[e^{-\frac{1}{C}\mathbf{X}^T\mathbf{X}}] = \frac{1}{|\frac{2}{C}\Sigma+I|^{\frac{1}{2}}}e^{-\frac{1}{C}\boldsymbol{\mu}^T\boldsymbol{\mu}+\frac{2}{C^2}\boldsymbol{\mu}^T(\frac{2}{C}I+\Sigma^{-1})^{-1}\boldsymbol{\mu}} = \frac{1}{|\frac{2}{C}\Sigma+I|^{\frac{1}{2}}}e^{-\frac{2}{C^2}\boldsymbol{\mu}^T\left(\frac{C}{2}I-(\frac{2}{C}I+\Sigma^{-1})^{-1}\right)\boldsymbol{\mu}}. \qquad \text{(B.6)}$$

Since $(\frac{2}{C}\Sigma\Sigma^{-1}+\Sigma^{-1})^{-1} = \Sigma(\frac{2}{C}\Sigma+I)^{-1}$, then

$$\begin{aligned}
-\frac{2}{C^2}\left(\frac{C}{2}I-\left(\frac{2}{C}I+\Sigma^{-1}\right)^{-1}\right) &= -\frac{2}{C^2}\left(\frac{C}{2}I-\Sigma\left(\frac{2}{C}\Sigma+I\right)^{-1}\right) \\
&= -\frac{2}{C^2}\left[\left(\frac{C}{2}\left(\frac{2}{C}\Sigma+I\right)-\Sigma\right)\left(\frac{2}{C}\Sigma+I\right)^{-1}\right] \\
&= -\frac{2}{C^2}\left[\frac{C}{2}\left(\frac{2}{C}\Sigma+I\right)^{-1}\right] = -\frac{1}{C}\left(\frac{2}{C}\Sigma+I\right)^{-1}.
\end{aligned}$$

□

Appendix C

Confidence interval Premia

Table 2. 95% confidence interval of the benchmark prices from Premia.

d	5	10	15	20	30	40	60	80	100
Premia	25.306	37.698	45.569	51.443	59.775	65.525	73.900	79.908	84.501
STD	0.073	0.079	0.087	0.110	0.056	0.091	0.129	0.090	0.097
LCL	25.261	37.649	45.515	51.375	59.740	65.468	73.820	79.852	84.441
UCL	25.351	37.747	45.623	51.511	59.810	65.582	73.980	79.964	84.561

Notes. The benchmark prices are point estimations calculated by the primal–dual method in Andersen and Broadie (2004) from Premia. The primal early-exercise policy is obtained by LSM with 50 000 simulation paths and Hermite polynomials of degree 0–3 as the basis functions. The dual prices are calculated with 500 outer simulation paths and 100 nested simulation paths. STD is the standard deviation of 10 runs, LCL = Premia $-$ 1.96 $*$ STD/$\sqrt{10}$ is the 95% lower confidence limit and UCL = Premia $+$ 1.96 $*$ STD/$\sqrt{10}$ is the 95% upper confidence limit.

Figure C.1: Confidence interval reprinted from (Hu and Zastawniak, 2020).

Appendix D

Heston with violated Feller condition

The set of parameters is listed in the table below, where the Feller condition is violated. The γ_v is changed to $\gamma_v = 0.39$. Moreover, the correlation between the variance process and the asset price is set as a negative value. The number of bundles P and the ridge parameter λ remain unchanged.

S_{0_v}	K_s	σ_v	r_v	q_v	$\nu_{0,v}$	$\bar{\nu}_v$	γ_v	κ_v	$\rho_v^{\nu S}$	ρ_v^S	T	N	M
100	100	0.2	5%	10%	0.03	0.04	0.39	1	-0.1	0	3	3	10000

Table D.1: Parameters for Heston with violated Feller condition.

We only present the pricing results as the computational time does not change with the varying of those two parameters. The benchmark is obtained using 10 independent iterations of the self-implemented primal-dual algorithm. Furthermore, the pricing results are based on 20 independent iterations. We can see that KRR-based methods provide similar pricing errors as in the experiment of Heston with satisfied Feller condition.

d	Benchmark	LSM	Error(%)	LSMP	Error(%)	NKRR	Error(%)	LKRR	Error(%)
5	21.3332 (0.0556)	21.3357(0.2559)	0.0119	21.3525(0.2450)	0.0903	21.4449(0.2746)	0.5236	21.3523(0.2811)	0.0894
10	32.5300 (0.0905)	32.6641(0.2484)	0.4122	32.7393(0.2381)	0.6434	32.8528(0.2442)	0.9923	32.7681(0.2631)	0.7319
15	40.3076 (0.0704)	40.4826(0.2854)	0.4341	40.6002(0.2908)	0.7259	40.6135(0.2691)	0.7590	40.5801(0.2790)	0.6763
20	46.2288 (0.1412)	46.4638(0.2600)	0.5083	46.5907(0.2698)	0.7829	46.5617(0.2459)	0.7202	46.5552(0.2610)	0.7061
30	55.1407 (0.0917)	55.6755(0.2954)	0.9699	55.8740(0.3046)	1.3300	55.7120(0.3103)	1.0361	55.7301(0.2804)	1.0689
40	61.7835 (0.1202)	62.4093(0.3662)	1.0129	62.5733(0.3809)	1.2783	62.3807(0.3395)	0.9665	62.3058(0.3798)	0.8454
60	71.7754 (0.0742)	72.8254(0.2861)	1.4630	72.9079(0.2939)	1.5779	72.6854(0.3051)	1.2679	72.4286(0.2755)	0.9101

Table D.2: Violated Feller: pricing results under Heston with $C_{nkrr} = 1.95 \cdot 10^5$ and $C_{lkrr} = 150$.