



RESEARCH INTERNSHIP

Shape-Constrained Risk Measures

Confidentiality Notice
Non confidential report

Author :
BECHIR TRABELSI
Promotion ENIT-TA 2021

Field of Study APPLIED MATHEMATICS, MACHINE LEARNING
Supervised by:
ZOLTÁN SZABÓ

ENSTA Paris Tutor:
FRANCESCO RUSSO

Academic year 2019/2020
Internship from May 18, 2020 to August 21, 2020

Centre de Mathématiques Appliquées, École Polytechnique
Route de Saclay, 91120 Palaiseau France

Abstract

Shape constraints (such as non-negativity, monotonicity or convexity) arise in a large number of financial and economic applications. Statistical estimators benefiting from domain-specific knowledge encoded via shape priors typically show improved performance in the finite-sample regime and help interpretability. Investigating the efficiency of modern data science techniques in this context is of fundamental interest. One of the most important problems of a financial institution is to estimate its potential profit/loss and the associated risks, which can be phrased under the umbrella of shape-constrained regression. The focus of the internship is to model and estimate risk measures relying on rich function classes (so-called reproducing kernel Hilbert spaces, RKHSs) associated to kernel functions, under shape conditions. We demonstrate the efficiency of the RKHS framework in the estimation of risk measures (conditional quantile and expectile) of profit-and-loss of options where the underlying stocks evolve according to d -dimensional geometric Brownian motion dynamics.

Résumé

Les contraintes de forme (telles que la non-négativité, la monotonie ou la convexité) se posent dans un grand nombre d'applications financières et économiques. Les estimateurs statistiques qui bénéficient de connaissances spécifiques au domaine, codées par le biais de prières de forme, présentent généralement de meilleures performances dans le régime d'échantillon fini et facilitent l'interprétation. L'étude de l'efficacité des techniques modernes de la science des données dans ce contexte est d'un intérêt fondamental. L'un des problèmes les plus importants d'une institution financière est d'estimer son profit/perte potentiel et les risques associés, qui peuvent être formulés sous la forme d'une régression sous contrainte de forme. L'objectif du stage est de modéliser et d'estimer les mesures de risque en s'appuyant sur des classes de fonctions riches (appelées espaces de Hilbert à noyau reproduisant, RKHS) associées aux fonctions du noyau, dans des conditions de forme. Nous démontrons l'efficacité du cadre RKHS dans l'estimation des mesures de risque (quantile conditionnel et expectile) des pertes et profits des options où les actions sous-jacentes évoluent selon une dynamique de mouvement brownien géométrique à dimension d .

Acknowledgements

This work benefited from the support of the Chair Stress Test, RISK Management and Financial Steering, led by the French École Polytechnique and its Foundation and sponsored by BNP Paribas. I would like to also thank Mr. Zoltán Szabó my internship supervisor, Mr. Emmanuel Gobet Scientific leader of the chaire Stress Test as well as Mr. Francesco Russo my ENSTA Paris tutor for their valuable advices and guidance throughout this internship.

Contents

1 Definitions	9
2 Problem Formulation	11
3 Optimization	15
4 Numerical Experiments	23
1 Joint Quantile Regression	25
1.1 One-Dimensional Data	25
1.2 2-Dimensional Data	31
2 Joint Expectile Regression	32
2.1 One-dimensional Data	34
2.2 2-Dimensional Data	36
5 Conclusions	39

List of Figures

2.1	Pinball loss and expectile loss	13
4.1	JQR task ($d = 1$) – estimated quantile functions vs sample size	27
4.2	JQR task ($d = 1$) – computational time vs sample size	28
4.3	JQR task ($d = 1$) – cross-validation scores vs interest rate	29
4.4	JQR task ($d = 1$) – estimated quantile functions vs portfolio size	30
4.5	Eigenspectrum: Laplacian and min kernel	31
4.6	P&L values ($d = 2$)	32
4.7	JQR task ($d \in \{1, 2\}$) – computational time	33
4.8	JQR task ($d = 2$) – estimated quantile surfaces	33
4.9	Data vs interest rate	35
4.10	EJR task ($d = 1$) – estimated expectile functions	36
4.11	Computational time: expectile vs quantile regression ($d = 2$)	37
4.12	EJR task ($d = 2$) – estimated expectile surfaces	38

List of Tables

4.1	JQR task ($d = 1$) – CV scores vs interest rate	26
4.2	JQR task ($d = 1$) – P&L vs portfolio size	28
4.3	JQR task ($d = 1$) – CV scores vs portfolio size	28
4.4	JQR task ($d \in \{1, 2\}$) – CV scores vs dimension	32
4.5	EJR task ($d = 1$) – CV scores vs interest rate	34
4.6	EJR task ($d = 1$) – CV scores vs portfolio size	35
4.7	EJR task ($d = 2$) – CV scores vs interest rate	37

Introduction

Shape constraints are ubiquitous in data science and statistics. They are typically expressed in terms of constraints on the derivatives of functions; examples include non-negativity, monotonicity, convexity. For example, in biology, monotone regression techniques have been applied to identify genome interactions (Luss et al., 2012), in statistics, the conditional quantile function is increasing with respect to the quantile level, European and American call option prices are convex and monotone in the underlying stock price and increasing in volatility.

Despite its wide applicability, constructing *shape-constrained estimators* is a rather challenging problem. Available methods often impose shape constraints

- in a soft way at at finite many points or as a regularizer (Delecroix et al., 1996; Blundell et al., 2012; Aybat and Wang, 2014; Wu et al., 2015; Takeuchi et al., 2006; Sangnier et al., 2016; Chen and Samworth, 2016; Agrell, 2019; Mazumder et al., 2019; Koppel et al., 2019; Han et al., 2019; Yagi et al., 2020),
- using constraint-specific constructions such as quadratic reparameterization (Flaxman et al., 2017), applying positive semi-definite quadratic forms (Bagnell and Farahmand, 2015), integrated exponential functions (Wu and Sickles, 2018), or
- they highly restrict the modelling class for instance to polynomials (Hall, 2018) or polynomial splines (Turlach, 2005; Papp and Alizadeh, 2014; Pya and Wood, 2015; Wu and Sickles, 2018; Meyer, 2018; Koppel et al., 2019). The underlying algebraic structure of polynomials and splines is what makes possible shape-constrained estimation; unfortunately these methods do not give rise to direct extension to richer families of functions such as reproducing kernel Hilbert spaces (RKHS).

In this internship our primary interest is to impose shape constraints in the class of RKHS functions and to investigate the efficiency of the resulting kernel machines in financial context for risk estimation.

This task is among the most fundamental problems of a financial institution: *risk measurement* is of utmost importance to trading in the derivatives industry and insufficient risk analysis can misprice derivatives, while incorrect or poor risk mea-

surement can significantly underestimate particular risk types e.g. market risk, credit risk (see Mitra and Ji 2010). These risk measures include for instance Value at Risk (VaR), the Entropic Value at Risk (EVaR), the superhedging price, the entropic risk measure, the Tail value at risk (TVaR), the Drawdown or the expected shortfall. In this internship we focus on two particular risk measures: quantiles and expectiles. They are among the most-widely applied measures and due their monotonicity properties they also provide an ideal testbed for imposing hard-shape constraints in the context of *RKHS modelling*.

RKHSs are quite flexible function families: for instance under mild conditions (universality) they can approximate several function classes arbitrarily well (Steinwart, 2001; Micchelli et al., 2006; Carmeli et al., 2010; Sriperumbudur et al., 2011; Simon-Gabriel and Schölkopf, 2018); examples of function families which can be approximated well include the space of bounded continuous functions.

From mathematical perspective, in this work we impose *hard* linear shape requirements on derivatives, i.e. our constraints take the form

$$0 \leq b + Df(\mathbf{x}) \quad \forall \mathbf{x} \in K, \quad (1)$$

with bias $b \in \mathbb{R}$ and differential operator $D = \sum_j \gamma_j \partial^{\mathbf{r}_j}$ where $\partial^{\mathbf{r}} f(\mathbf{x}) = \frac{\partial^{\sum_{j=1}^d r_j} f(\mathbf{x})}{\partial x_1^{r_1} \dots \partial x_d^{r_d}}$ and $K \subset \mathbb{R}^d$ is a compact set. The primary challenge is that the set K has non-finite cardinality, hence the pointwise inequality (1) has to be guaranteed at *infinite* many points. To tackle this task we leverage the method proposed by Aubin-Frankowski and Szabó (2020) where it was shown that the infinite number of affine constraints (1) can be tightened into a finite number of SOC constraints

$$\eta \|f\|_{\mathcal{F}_k} \leq b + Df(\mathbf{x}_m) \quad \forall m \in \{1, \dots, M\} \quad (2)$$

for a suitable choice of $\eta > 0$ and $\{\mathbf{x}_m\}_{m \in [M]} \subseteq K$.

The document is structured as follows. Chapter 1 is about notations, followed by problem formulation in Chapter 2. Optimization details are discussed in Chapter 3. Numerical experiments is the focus of Chapter 4. Conclusions are drawn in Chapter 5.

Chapter 1

Definitions

In this section we introduce a few notations and definitions used throughout the manuscript.

Let $\mathbb{N}^* = \{1, 2, \dots\}$ denote the set of positive integers; $[m] = \{1, \dots, m\}$ for $m \in \mathbb{N}^*$. The Euclidean norm of a vector $\mathbf{v} \in \mathbb{R}^d$ is denoted by $\|\mathbf{v}\|_2 = \sqrt{\sum_{i \in [d]} v_i^2}$. The unit ball in \mathbb{R}^d is $B = \{\mathbf{v} \in \mathbb{R}^d : \|\mathbf{v}\|_2 \leq 1\}$. The transpose of a matrix \mathbf{M} is denoted by \mathbf{M}^T . For the concatenation of matrix $\mathbf{A}_1 \in \mathbb{R}^{d_1 \times d}, \dots, \mathbf{A}_L \in \mathbb{R}^{d_L \times d}$ we use the shorthand $[\mathbf{A}_1; \dots; \mathbf{A}_L] \in \mathbb{R}^{(\sum_{l \in [L]} d_l) \times d}$.

Definition 1 (Kernel) *Let \mathcal{X} be a non-empty set, throughout the paper it will be an open subset of \mathbb{R}^d . A function $k : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ is called a kernel if there exist a Hilbert space \mathcal{F} and a feature map $\phi : \mathcal{X} \rightarrow \mathcal{F}$ such that*

$$k(\mathbf{x}, \mathbf{x}') = \langle \phi(\mathbf{x}), \phi(\mathbf{x}') \rangle_{\mathcal{F}}, \quad \forall \mathbf{x}, \mathbf{x}' \in \mathcal{X}.$$

While the feature space \mathcal{F} above might not be unique, there always exists a unique so-called reproducing kernel Hilbert space (RKHS) \mathcal{F}_k associated to the kernel k . This space is defined as follows.

Definition 2 (Reproducing Kernel Hilbert Space, reproducing kernel) *Let \mathcal{X} be a non-empty set as above and let \mathcal{F}_k be a Hilbert space of $\mathcal{X} \rightarrow \mathbb{R}$ functions. The space \mathcal{F}_k is called a Reproducing Kernel Hilbert Space (RKHS) with reproducing kernel $k : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ if the following two properties¹ hold:*

1. $k(\cdot, \mathbf{x}) \in \mathcal{F}_k$ for $\forall \mathbf{x} \in \mathcal{X}$, and
2. $\langle f, k(\cdot, \mathbf{x}) \rangle_{\mathcal{F}_k} = f(\mathbf{x})$ for $\forall \mathbf{x} \in \mathcal{X}$, $\forall f \in \mathcal{F}_k$.

1. We use the abbreviation $k(\cdot, \mathbf{x})$ to denote the mapping $\mathbf{y} \in \mathcal{X} \mapsto k(\mathbf{y}, \mathbf{x}) \in \mathbb{R}$ while keeping \mathbf{x} fixed.

The first property describes the basic elements of \mathcal{F}_k ; the second one is referred to as the reproducing property and makes the canonical feature map explicit: $\phi(\mathbf{x}) = k(\cdot, \mathbf{x})$. The two definitions (kernel, reproducing kernel) are equivalent. We will use the shorthand $\|\cdot\|_k$ to denote the norm on the RKHS \mathcal{F}_k . For further details on kernels and RKHSs, the reader is referred to Aronszajn (1950); Berlinet and Thomas-Agnan (2004); Steinwart and Christmann (2008).

Definition 3 (Quantile) Let $(X, Y) \in \mathcal{X} \times \mathbb{R}$ be a random variable and $\tau \in (0, 1)$. The τ -quantile of the conditional distribution $Y|X$ is defined as

$$\mu_\tau(\mathbf{x}) := \inf\{z \in \mathbb{R} : \mathbb{P}(Y \leq z | X = \mathbf{x}) = \tau\}.$$

Definition 4 (Second-order cone program) An optimization problem of the form

$$\begin{aligned} & \text{minimize} && \mathbf{f}^T \mathbf{x} \\ & \text{subject to} && \|\mathbf{A}_i \mathbf{x} + \mathbf{b}_i\|_2 \leq \mathbf{c}_i^T \mathbf{x} + d_i, \quad i \in [I] \\ & && \mathbf{F}\mathbf{x} = \mathbf{g}, \end{aligned}$$

is called second-order cone program (SOCP), where $\mathbf{x} \in \mathbb{R}^n$ is the variable of interest, $\mathbf{f} \in \mathbb{R}^n$, $\mathbf{A}_i \in \mathbb{R}^{n_i \times n}$, $\mathbf{b}_i \in \mathbb{R}^{n_i}$, $\mathbf{c}_i \in \mathbb{R}^n$, $d_i \in \mathbb{R}$, $\mathbf{F} \in \mathbb{R}^{p \times n}$ and $\mathbf{g} \in \mathbb{R}^p$ for all $i \in [I]$.

Definition 5 (Geometric Brownian motion) A stochastic process $S_t = [S_t^1; \dots; S_t^d] \in \mathbb{R}^d$ is said to follow a d -dimensional geometric Brownian motion (GBM) if it satisfies the following stochastic differential equation

$$dS_t^i = \mu_i S_t^i dt + \sigma_i S_t^i dW_t^i, \quad \forall i \in [d],$$

where W_t^i -s are correlated Wiener processes, i.e. $\mathbb{E}[(dW_t^i)(dW_t^j)] = \rho_{i,j}dt$, $\mu_i \in \mathbb{R}$ is the percentage drift and $\sigma_i > 0$ is the percentage volatility ($\forall i \in [d]$).

GBM is a widely-used and popular model to describe the evolution of stock prices; it forms the basis for instance of the well-known Black-Scholes option pricing formulas.

Using these notations, in the next chapter we proceed with the problem formulation.

Chapter 2

Problem Formulation

In this chapter we define our task. Let $0 < t_1 < t_2$ be two fixed time points. Assume that we have d stocks evolving in time $S_t \in \mathbb{R}^d$, a profit-and-loss (P&L) function ψ , and a portfolio Θ . Our **goal** is to estimate jointly risk measures of the P&L of the portfolio at time t_2 , given access to the stock prices at time t_1 (S_{t_1}), i.e. to quantify the uncertainty of

$$\psi(S_{t_1}, S_{t_2}, \Theta) | S_{t_1}. \quad (2.1)$$

We consider two risk measures: quantiles and expectiles. Particularly, in

- **joint quantile regression (JQR):** we are given Q quantile levels $0 < \tau_1 < \dots < \tau_Q < 1$ and our aim is to estimate jointly the τ_q -conditional quantiles of $\psi(S_{t_1}, S_{t_2}, \Theta) | S_{t_1}$.
- **joint expectile regression (JER):** we have Q expectile levels $0 < \alpha_1 < \dots < \alpha_Q < 1$ and the goal is to infer jointly the α_q -expectiles of $\psi(S_{t_1}, S_{t_2}, \Theta) | S_{t_1}$.

Throughout this work we assume that S_t is a d -dimensional GBM where the volatility of the j^{th} component is denoted by σ_j , and that the portfolio $\Theta = (\theta_1, \dots, \theta_I)$ consists of I European put or call options. Each element in this portfolio is described by a parameter vector $\theta_i = (c_i, s_i, T_i, K_i, w_i)$ where $c_i \in \{0, 1\}$ indicates whether it is a call ($c_i = 1$) or a put option ($c_i = 0$) on stock $s_i \in \{1, \dots, d\}$ with maturity T_i , strike price K_i and nominal value w_i . Moreover, ψ is chosen to be the difference of the price of the portfolio between time point t_1 and t_2

$$\psi(S_{t_1}, S_{t_2}, \Theta) = p(S_{t_2}, \Theta) - p(S_{t_1}, \Theta). \quad (2.2)$$

The price of the portfolio at t_1 and t_2 are computed via the Black-Scholes option pricing formulas C and P as

$$p(S_{t_1}, \Theta) = \sum_{i \in [I]} w_i [\mathbb{I}_{\{c_i=1\}} C(S_{t_1, s_i}, K_i, T_i, \sigma_{s_i}) + \mathbb{I}_{\{c_i=0\}} P(S_{t_1, s_i}, K_i, T_i, \sigma_{s_i})], \quad (2.3)$$

$$p(S_{t_2}, \Theta) = \sum_{i \in [I]} w_i [\mathbb{I}_{\{c_i=1\}} C(S_{t_2, s_i}, K_i, T_i, \sigma_{s_i}) + \mathbb{I}_{\{c_i=0\}} P(S_{t_2, s_i}, K_i, T_i, \sigma_{s_i})], \quad (2.4)$$

where \mathbb{I} stands for the indicator function. The C and P pricing formulas take the form

$$C(S_{t,s}, K, T, \sigma_s) = \Phi(d_1)S_{t,s} - \Phi(d_2)Ke^{-r(T-t)}, \quad (2.5)$$

$$d_1 = \frac{1}{\sigma_s \sqrt{T-t}} \left[\ln \left(\frac{S_{t,s}}{K} \right) + \left(r + \frac{\sigma_s^2}{2} \right) (T-t) \right], \quad (2.6)$$

$$d_2 = d_1 - \sigma_s \sqrt{T-t}, \quad (2.7)$$

$$P(S_{t,s}, K, T, \sigma_s) = \Phi(-d_2)Ke^{-r(T-t)} - \Phi(-d_1)S_{t,s}, \quad (2.8)$$

where r is the risk-free rate, Φ is the cumulative distribution function of the standard normal distribution and $T-t$ is the time to maturity.

Let $(X, Y) := (S_{t_1}, \psi(S_{t_1}, S_{t_2}, \Theta)) \in \mathcal{X} \times \mathbb{R}$, assume that we have access to N i.i.d. observations $\{(\mathbf{x}_n, y_n)\}_{n \in [N]}$, and we are given a kernel $k : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$. We model the uncertainty estimates of $Y|X$ by $g_q + b_q$ with $g_q \in \mathcal{F}_k$ and $b_q \in \mathbb{R}$, and minimize the loss function

$$\mathcal{L}(\mathbf{g}, \mathbf{b}) = \frac{1}{N} \sum_{q \in [Q]} \sum_{n \in [N]} l_q(y_n - (g_q(\mathbf{x}_n) + b_q)) + \lambda_b \|\mathbf{b}\|_2^2 + \lambda_g \sum_{q=1}^Q \|g_q\|_k^2, \quad (2.9)$$

where $\lambda_b > 0$, $\lambda_g > 0$, $\mathbf{g} = (g_q)_{q \in [Q]} \in (\mathcal{F}_k)^Q$, $\mathbf{b} = (b_q)_{q \in [Q]} \in \mathbb{R}^Q$. The loss function l_q in

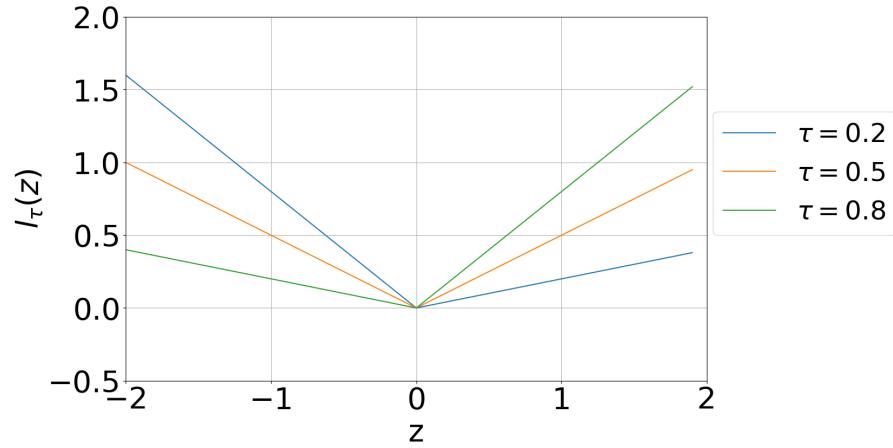
- joint quantile regression: is the pinball loss

$$l_{\tau_q}(z) = \max(\tau_q z, (\tau_q - 1)z), \quad \tau_q \in (0, 1),$$

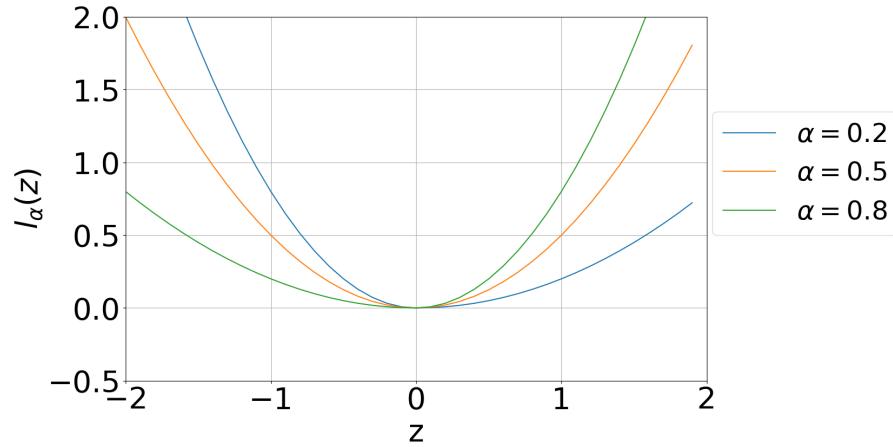
- joint expectile regression: is the expectile loss

$$l_{\alpha_q}(z) = \alpha_q(z)_+^2 + (1 - \alpha_q)(z)_-^2, \quad \alpha_q \in (0, 1).$$

where $(\cdot)_+$ and $(\cdot)_-$ denotes the positive and negative part of its argument, respectively.



(a) Pinball loss for various τ values



(b) Expectile loss for various α values

Figure 2.1: Illustration of the pinball loss and the expectile loss. (a): pinball loss. (b): expectile loss.

For an illustration of the loss functions l_q , see Fig. 2.1.

The estimated conditional quantile and expectile functions have to be monotonically increasing w.r.t. the quantile level τ and the expectile level α respectively, i.e. they have to satisfy the joint shape constraint

$$g_q(\mathbf{x}) + b_q \leq g_{q+1}(\mathbf{x}) + b_{q+1}, \quad \forall \mathbf{x} \in R, \quad \forall q \in [Q-1], \quad (2.10)$$

where $R = \prod_{i \in [d]} [\min_{n \in [N]} (\mathbf{x}_n)_i, \max_{n \in [N]} (\mathbf{x}_n)_i]$ is the smallest rectangle containing the sample points.

In the next chapter we describe the solution of (2.9) under the hard shape constraint (2.10).

Chapter 3

Optimization

This chapter is dedicated to the solution of the optimization problem (2.9) with the hard shape constraint (2.10). The tasks falls under the umbrella of the work developed by Aubin-Frankowski and Szabó (2020) where the goal is to solve hard shape-constrained kernel machines of the form

$$(\bar{\mathbf{g}}, \bar{\mathbf{b}}) = \arg \min_{\substack{\mathbf{g}=(g_q)_{q \in [Q]} \in (\mathcal{F}_k)^Q \\ \mathbf{b}=(b_p)_{p \in [P]} \in \mathcal{B} \\ (\mathbf{g}, \mathbf{b}) \in C}} \mathcal{L}(\mathbf{g}, \mathbf{b}),$$

with an objective function \mathcal{L} , a constraint set C , and a convex constraint set $\mathcal{B} \subset \mathbb{R}^P$ on the biases. The $I \in \mathbb{N}^*$ hard shape requirements take the form

$$C = \{(\mathbf{g}, \mathbf{b}) \mid (\mathbf{b}_0 - \mathbf{U}\mathbf{b})_i \leq D_i(\mathbf{W}\mathbf{g} - \mathbf{g}_0)_i(\mathbf{x}), \forall \mathbf{x} \in K_i, \forall i \in [I]\}. \quad (3.1)$$

The latent interactions within the bias variables $(b_p)_{p \in [P]}$ and functions $(g_q)_{q \in [Q]}$, respectively, are captured through the matrices $\mathbf{U} \in \mathbb{R}^{I \times P}$ and $\mathbf{W} \in \mathbb{R}^{I \times Q}$. The compact sets $K_i \subset \mathcal{X}$ ($i \in [I]$) define the domain where the constraints are imposed.

In the case of the considered joint quantile regression (JQR) and joint expectile regression (JER) tasks, one can get the required shape constraints (2.10) by choosing D_i as the identity operator, $P = Q, I = Q - 1, \mathbf{b}_0 = \mathbf{0}, \mathbf{g}_0 = \mathbf{0}, \mathcal{B} = \mathbb{R}^I, K_i = K$ ($i \in [I]$) the smallest rectangle containing our points and

$$\mathbf{U} = \mathbf{W} = \begin{bmatrix} -1 & 1 & 0 & 0 \\ 0 & -1 & 1 & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & -1 & 1 \end{bmatrix}.$$

In the mentioned paper in order to introduce the second-order cone tightening of the constraints, the authors first consider the discretization of the I constraints using M points $\{\tilde{\mathbf{x}}_m\}_{m \in [M]} \subseteq K$. This would lead to the following relaxation of the

optimization problem:

$$\begin{aligned} \min_{\mathbf{g} \in (\mathcal{F}_k)^Q, \mathbf{b} \in \mathbb{R}^Q} \quad & \mathcal{L}(\mathbf{g}, \mathbf{b}) \\ \text{s.t.} \quad & -(\mathbf{Ub})_i \leq \min_{m \in [M]} (\mathbf{Wg})_i(\tilde{\mathbf{x}}_m), \forall i \in [I]. \end{aligned}$$

The proposed SOC-constrained tightening can be thought of as adding extra, appropriately chosen, η -buffer to the left hand side of the constraints:

$$-(\mathbf{Ub})_i + \eta \|(\mathbf{Wg})_i\|_k \leq \min_{m \in [M]} (\mathbf{Wg})_i(\tilde{\mathbf{x}}_m), \forall i \in [I]. \quad (3.2)$$

Constraint (3.2) is determined by points $\{\tilde{\mathbf{x}}_m\}_{m \in [M]}$ which are chosen to form a δ -net of the compact set K for a $\delta > 0$ and by a fixed $\eta \in \mathbb{R}_+$ coefficient defined as

$$\eta = \sup_{\mathbf{u} \in B, m \in [M]} \|k(\tilde{\mathbf{x}}_m, \cdot) - k(\tilde{\mathbf{x}}_m + \delta \mathbf{u}, \cdot)\|_k.$$

Theorem (proven by Aubin-Frankowski and Szabó (2020)): Let $\mathbf{g}_\eta = (g_{\eta,q})_{q \in [Q]}$. Then

- (i) Tightening: any (\mathbf{g}, \mathbf{b}) satisfying (3.2) also satisfies (3.1) in case of the JQR and JER tasks detailed above.
- (ii) Representer theorem: For all $q \in [Q]$, there exist real coefficients $\tilde{a}_{m,q}, a_{n,q} \in \mathbb{R}$ such that

$$g_{\eta,q} = \sum_{m \in [M]} \tilde{a}_{m,q} k(\tilde{\mathbf{x}}_m, \cdot) + \sum_{n \in [N]} a_{n,q} k(\mathbf{x}_n, \cdot).$$

The importance of (ii) is that it allows to find a finite-dimensional SOC-constrained problem. To arrive at this finite-dimensional task, below we calculate the explicit expression of the $(\mathbf{Wg})_i(\tilde{\mathbf{x}}_m)$, $\|(\mathbf{Wg})_i\|_k^2$, $g_q(\mathbf{x}_n)$ and $\|g_q\|_k$ terms in the objective (2.9) and the constraint (3.2).

Particularly, let

$$\hat{\mathbf{x}}_i = \begin{cases} \mathbf{x}_i, & \text{if } i \in [N] \\ \tilde{\mathbf{x}}_{i-N}, & \text{if } i \in \{N+1, \dots, N+M\}, \end{cases}$$

and let us introduce the notation $\tilde{\mathbf{A}} = (\tilde{a}_{m,q})_{m \in [M], q \in [Q]} \in \mathbb{R}^{M \times Q}$, $\mathbf{A} = (a_{i,q})_{i \in [N], q \in [Q]} \in \mathbb{R}^{N \times Q}$, and the Gram matrix \mathbf{G} of $\{k(\hat{\mathbf{x}}_j, \cdot)\}_{j \in [M+N]}$, i.e. $\mathbf{G} = [G_{i,j}] = [k(\hat{\mathbf{x}}_i, \hat{\mathbf{x}}_j)]$. We now express (3.2) as a finite-dimensional optimization problem over $\tilde{\mathbf{A}}$, \mathbf{A} and \mathbf{b} . We consider the terms one-by-one.

- Term $(\mathbf{Wg})_i(\tilde{\mathbf{x}}_m)$:

$$\begin{aligned}
 (\mathbf{Wg})_i(\tilde{\mathbf{x}}_m) &= \sum_{q \in [Q]} W_{i,q} g_q(\tilde{\mathbf{x}}_m) \\
 &= \sum_{q \in [Q]} W_{i,q} \left[\sum_{j \in [M]} \tilde{a}_{j,q} k(\tilde{\mathbf{x}}_j, \tilde{\mathbf{x}}_m) + \sum_{n \in [N]} a_{n,q} k(\mathbf{x}_n, \tilde{\mathbf{x}}_m) \right] \\
 &= \sum_{q \in [Q]} \sum_{j \in [M]} W_{i,q} \tilde{a}_{j,q} k(\tilde{\mathbf{x}}_j, \tilde{\mathbf{x}}_m) + \sum_{q \in [Q]} \sum_{n \in [N]} W_{i,q} a_{n,q} k(\mathbf{x}_n, \tilde{\mathbf{x}}_m) \\
 &= \sum_{n \in [N]} k(\mathbf{x}_n, \tilde{\mathbf{x}}_m) \sum_{q \in [Q]} W_{i,q} a_{n,q} + \sum_{j \in [M]} k(\tilde{\mathbf{x}}_j, \tilde{\mathbf{x}}_m) \sum_{q \in [Q]} W_{i,q} \tilde{a}_{j,q} \\
 &= \sum_{n \in [N]} k(\mathbf{x}_n, \tilde{\mathbf{x}}_m) \sum_{q \in [Q]} W_{i,q} a_{n,q} + \sum_{j=N+1}^{M+N} k(\tilde{\mathbf{x}}_{j-N}, \tilde{\mathbf{x}}_m) \sum_{q \in [Q]} W_{i,q} \tilde{a}_{j-N,q} \\
 &= \sum_{l \in [N]} k(\hat{\mathbf{x}}_l, \tilde{\mathbf{x}}_m) \left[\sum_{q \in [Q]} W_{i,q} a_{l,q} \mathbb{I}_{\{l \leq N\}} + \sum_{q \in [Q]} W_{i,q} \tilde{a}_{l,q} \mathbb{I}_{\{l > N\}} \right] \\
 &\quad + \sum_{l=N+1}^{N+M} k(\hat{\mathbf{x}}_l, \tilde{\mathbf{x}}_m) \left[\sum_{q \in [Q]} W_{i,q} a_{l,q} \mathbb{I}_{\{l \leq N\}} + \sum_{q \in [Q]} W_{i,q} \tilde{a}_{l-N,q} \mathbb{I}_{\{l > N\}} \right] \\
 &= \sum_{l \in [M+N]} k(\tilde{\mathbf{x}}_m, \hat{\mathbf{x}}_l) \left([\mathbf{A}; \tilde{\mathbf{A}}] \mathbf{W}^T \right)_{l,i} \\
 &= \left(\mathbf{G} [\mathbf{A}; \tilde{\mathbf{A}}] \mathbf{W}^T \right)_{N+m,i} \\
 &= \left(\mathbf{G} [\mathbf{A}; \tilde{\mathbf{A}}] \mathbf{W}^T \mathbf{e}_i \right)_{N+m},
 \end{aligned}$$

where $\mathbf{e}_i \in \mathbb{R}^I$ are the canonical basis vectors.

- **Term $\|(\mathbf{Wg})_i\|_k^2$:** By using the linearity of the inner product and the reproducing property of kernels we get that

$$\begin{aligned}
 \|(\mathbf{Wg})_i\|_k^2 &= \left\langle \sum_{q \in [Q]} \sum_{j \in [M]} W_{i,q} \tilde{a}_{j,q} k(\tilde{\mathbf{x}}_j, \cdot) + \sum_{q \in [Q]} \sum_{n \in [N]} W_{i,q} a_{n,q} k(\tilde{\mathbf{x}}_n, \cdot), \right. \\
 &\quad \left. \sum_{q \in [Q]} \sum_{j \in [M]} W_{i,q} \tilde{a}_{j,q} k(\tilde{\mathbf{x}}_j, \cdot) + \sum_{q \in [Q]} \sum_{n \in [N]} W_{i,q} a_{n,q} k(\tilde{\mathbf{x}}_n, \cdot) \right\rangle_k \\
 &= \left\langle \sum_{q \in [Q]} \sum_{j \in [M]} W_{i,q} \tilde{a}_{j,q} k(\tilde{\mathbf{x}}_j, \cdot), \sum_{q \in [Q]} \sum_{j \in [M]} W_{i,q} \tilde{a}_{j,q} k(\tilde{\mathbf{x}}_j, \cdot) \right\rangle_k \quad (3.3) \\
 &\quad + \left\langle \sum_{q \in [Q]} \sum_{n \in [N]} W_{i,q} a_{n,q} k(\mathbf{x}_n, \cdot), \sum_{q \in [Q]} \sum_{n \in [N]} W_{i,q} a_{n,q} k(\mathbf{x}_n, \cdot) \right\rangle_k \\
 &\quad + 2 \left\langle \sum_{q \in [Q]} \sum_{j \in [M]} W_{i,q} \tilde{a}_{j,q} k(\tilde{\mathbf{x}}_j, \cdot), \sum_{q \in [Q]} \sum_{n \in [N]} W_{i,q} a_{n,q} k(\mathbf{x}_n, \cdot) \right\rangle_k \\
 &= \sum_{q \in [Q]} \sum_{p \in [Q]} \sum_{j \in [M]} \sum_{l \in [M]} \langle W_{i,q} \tilde{a}_{j,q} k(\tilde{\mathbf{x}}_j, \cdot), W_{i,p} \tilde{a}_{l,p} k(\tilde{\mathbf{x}}_l, \cdot) \rangle_k \\
 &\quad + \sum_{q \in [Q]} \sum_{p \in [Q]} \sum_{n \in [N]} \sum_{m \in [N]} \langle W_{i,q} a_{n,q} k(\mathbf{x}_n, \cdot), W_{i,p} a_{m,p} k(\mathbf{x}_m, \cdot) \rangle_k \\
 &\quad + 2 \sum_{q \in [Q]} \sum_{p \in [Q]} \sum_{n \in [N]} \sum_{j \in [M]} \langle W_{i,q} \tilde{a}_{j,q} k(\tilde{\mathbf{x}}_j, \cdot), W_{i,p} a_{n,p} k(\mathbf{x}_n, \cdot) \rangle_k \\
 &= \sum_{q \in [Q]} \sum_{p \in [Q]} \sum_{j \in [M]} \sum_{l \in [M]} W_{i,q} W_{i,p} \tilde{a}_{j,q} \tilde{a}_{l,p} k(\tilde{\mathbf{x}}_j, \tilde{\mathbf{x}}_l) \\
 &\quad + \sum_{q \in [Q]} \sum_{p \in [Q]} \sum_{n \in [N]} \sum_{m \in [N]} W_{i,q} W_{i,p} a_{n,q} a_{m,p} k(\mathbf{x}_n, \mathbf{x}_m) \\
 &\quad + 2 \sum_{q \in [Q]} \sum_{p \in [Q]} \sum_{n \in [N]} \sum_{j \in [M]} W_{i,q} W_{i,p} \tilde{a}_{j,q} a_{n,p} k(\tilde{\mathbf{x}}_j, \mathbf{x}_n). \quad (3.4)
 \end{aligned}$$

Now let $\mathbf{h}_i = [\mathbf{A}; \tilde{\mathbf{A}}] \mathbf{W}^T \mathbf{e}_i$. By using the symmetry of \mathbf{G} and hence that of $\mathbf{G}^{\frac{1}{2}}$, we arrive at

$$\begin{aligned}
 \left\| \mathbf{G}^{\frac{1}{2}} \mathbf{h}_i \right\|_2^2 &= \left\langle \mathbf{G}^{\frac{1}{2}} \mathbf{h}_i, \mathbf{G}^{\frac{1}{2}} \mathbf{h}_i \right\rangle \\
 &= \langle \mathbf{G} \mathbf{h}_i, \mathbf{h}_i \rangle \\
 &= \mathbf{h}_i^T \mathbf{G} \mathbf{h}_i = \sum_{l \in [M+N]} (\mathbf{h}_i)_l (\mathbf{G} \mathbf{h}_i)_l.
 \end{aligned}$$

Let us split $\mathbf{h}_i^T(\mathbf{G}\mathbf{h}_i)$ into two terms as

$$\mathbf{h}_i^T(\mathbf{G}\mathbf{h}_i) = \sum_{l \in [N]} (\mathbf{h}_i)_l (\mathbf{G}\mathbf{h}_i)_l + \sum_{s \in [M]} (\mathbf{h}_i)_{s+N} (\mathbf{G}\mathbf{h}_i)_{s+N}$$

and use that

$$(\mathbf{G}\mathbf{h}_i)_l = \sum_{q \in [Q]} \sum_{n \in [N]} a_{n,q} W_{i,q} k(\hat{\mathbf{x}}_l, \mathbf{x}_n) + \sum_{q \in [Q]} \sum_{j \in [M]} \tilde{a}_{j,q} W_{i,q} k(\hat{\mathbf{x}}_l, \tilde{\mathbf{x}}_j),$$

to rewrite the terms individually. Particularly, we get that

$$\begin{aligned} \sum_{l \in [N]} (\mathbf{h}_i)_l (\mathbf{G}\mathbf{h}_i)_l &= \sum_{l \in [N]} \sum_{q \in [Q]} a_{l,q} W_{i,q} \left[\sum_{n \in [N]} \sum_{p \in [Q]} a_{n,p} W_{i,p} k(\mathbf{x}_l, \mathbf{x}_n) \right. \\ &\quad \left. + \sum_{j \in [M]} \sum_{p \in [Q]} \tilde{a}_{j,p} W_{i,p} k(\mathbf{x}_l, \tilde{\mathbf{x}}_j) \right] \\ &= \sum_{q \in [Q]} \sum_{p \in [Q]} \sum_{l \in [N]} \sum_{n \in [N]} a_{l,q} a_{n,p} W_{i,q} W_{i,p} k(\mathbf{x}_l, \mathbf{x}_n) \\ &\quad + \sum_{q \in [Q]} \sum_{p \in [Q]} \sum_{l \in [N]} \sum_{j \in [M]} a_{l,q} \tilde{a}_{j,p} W_{i,q} W_{i,p} k(\mathbf{x}_l, \tilde{\mathbf{x}}_j), \end{aligned}$$

and

$$\begin{aligned} \sum_{s \in [M]} (\mathbf{h}_i)_{s+N} (\mathbf{G}\mathbf{h}_i)_{s+N} &= \sum_{s \in [M]} \sum_{q \in [Q]} \tilde{a}_{s,q} W_{i,q} \left[\sum_{n \in [N]} \sum_{p \in [Q]} a_{n,p} W_{i,p} k(\tilde{\mathbf{x}}_s, \mathbf{x}_n) \right. \\ &\quad \left. + \sum_{j \in [M]} \sum_{p \in [Q]} \tilde{a}_{j,p} W_{i,p} k(\tilde{\mathbf{x}}_s, \tilde{\mathbf{x}}_j) \right] \\ &= \sum_{q \in [Q]} \sum_{p \in [Q]} \sum_{n \in [N]} \sum_{s \in [M]} a_{n,p} \tilde{a}_{s,q} W_{i,q} W_{i,p} k(\tilde{\mathbf{x}}_s, \mathbf{x}_n) \\ &\quad + \sum_{q \in [Q]} \sum_{p \in [Q]} \sum_{s \in [M]} \sum_{j \in [M]} \tilde{a}_{s,q} \tilde{a}_{j,p} W_{i,q} W_{i,p} k(\tilde{\mathbf{x}}_s, \tilde{\mathbf{x}}_j). \end{aligned}$$

These expressions imply that

$$\begin{aligned} \mathbf{h}_i^T(\mathbf{G}\mathbf{h}_i) &= \sum_{q \in [Q]} \sum_{p \in [Q]} \sum_{l \in [N]} \sum_{n \in [N]} a_{l,q} a_{n,p} W_{i,q} W_{i,p} k(\mathbf{x}_l, \mathbf{x}_n) \\ &\quad + \sum_{q \in [Q]} \sum_{p \in [Q]} \sum_{s \in [M]} \sum_{j \in [M]} \tilde{a}_{s,q} \tilde{a}_{j,p} W_{i,q} W_{i,p} k(\tilde{\mathbf{x}}_s, \tilde{\mathbf{x}}_j) \\ &\quad + 2 \sum_{q \in [Q]} \sum_{p \in [Q]} \sum_{n \in [N]} \sum_{s \in [M]} a_{n,p} \tilde{a}_{s,q} W_{i,q} W_{i,p} k(\tilde{\mathbf{x}}_s, \mathbf{x}_n). \end{aligned}$$

We got the same expression as in (3.4), thus

$$\|(\mathbf{W}\mathbf{g})_i\|_k^2 = \left\| \mathbf{G}^{\frac{1}{2}} \mathbf{h}_i \right\|_2^2.$$

- **Term $g_q(\mathbf{x}_n)$:** Now let $\mathbf{G}_N = [G_{i,j}]_{i \in [N], j \in [N+M]}$ and $\hat{\mathbf{A}} = [\mathbf{A}; \tilde{\mathbf{A}}]$. Using these notations and the representer theorem, we have

$$\begin{aligned} (\mathbf{G}_N \hat{\mathbf{A}})_{i,j} &= \sum_{l \in [M+N]} (\mathbf{G}_N)_{i,l} \hat{A}_{l,j} \\ &= \sum_{l \in [N]} (\mathbf{G}_N)_{i,l} \hat{A}_{l,j} + \sum_{l \in [M]} (\mathbf{G}_N)_{i,l+N} \hat{A}_{l+N,j} \\ &= \sum_{l \in [N]} a_{l,j} k(\mathbf{x}_i, \mathbf{x}_l) + \sum_{l \in [M]} \tilde{a}_{l,j} k(\tilde{\mathbf{x}}_l, \mathbf{x}_i) \\ &= g_j(\mathbf{x}_i). \end{aligned}$$

This means that

$$g_q(\mathbf{x}_n) = \left(\mathbf{G}_N [\mathbf{A}, \tilde{\mathbf{A}}] \right)_{n,q}.$$

- **Term $\|g_q\|_k$:** We will now proceed with the reformulation of $\|g_q\|_k$.

$$\begin{aligned} \|g_q\|_k^2 &= \langle g_q, g_q \rangle_k \\ &= \left\langle \sum_{j \in [M]} \tilde{a}_{j,q} k(\tilde{\mathbf{x}}_j, \cdot) + \sum_{n \in [N]} a_{n,q} k(\mathbf{x}_n, \cdot), \sum_{j \in [M]} \tilde{a}_{j,q} k(\tilde{\mathbf{x}}_j, \cdot) + \sum_{n \in [N]} a_{n,q} k(\mathbf{x}_n, \cdot) \right\rangle_k \\ &= \sum_{m \in [M]} \sum_{j \in [M]} \tilde{a}_{j,q} \tilde{a}_{m,q} k(\tilde{\mathbf{x}}_j, \tilde{\mathbf{x}}_m) + \sum_{n \in [N]} \sum_{i \in [N]} a_{n,q} a_{i,q} k(\mathbf{x}_n, \mathbf{x}_i) \\ &\quad + 2 \sum_{n \in [N]} \sum_{m \in [M]} \tilde{a}_{m,q} a_{n,q} k(\tilde{\mathbf{x}}_m, \mathbf{x}_n) \\ &= \left[[\mathbf{A}; \tilde{\mathbf{A}}] \tilde{\mathbf{e}}_q \right]^T \mathbf{G} \left[[\mathbf{A}; \tilde{\mathbf{A}}] \tilde{\mathbf{e}}_q \right] \\ &= \langle \mathbf{G} \tilde{\mathbf{h}}_q, \tilde{\mathbf{h}}_q \rangle_2 \\ &= \left\| \mathbf{G}^{\frac{1}{2}} \tilde{\mathbf{h}}_q \right\|_2, \end{aligned}$$

where $\tilde{\mathbf{e}}_q \in \mathbb{R}^Q$ are the canonical basis vectors and $\tilde{\mathbf{h}}_q = \left[[\mathbf{A}; \tilde{\mathbf{A}}] \tilde{\mathbf{e}}_q \right]$.

Using the obtained expression for $(\mathbf{W}\mathbf{g})_i(\tilde{\mathbf{x}}_m)$, $\|(\mathbf{W}\mathbf{g})_i\|_k^2$, $g_q(\mathbf{x}_n)$ and $\|g_q\|_k$, the finite-dimensional SOC-constrained problem one has to solve takes the form

$$\begin{aligned} & \min_{\substack{\mathbf{A} \in \mathbb{R}^{N \times Q}, \mathbf{b} \in \mathbb{R}^Q \\ \tilde{\mathbf{A}} \in \mathbb{R}^{M \times Q}}} \frac{1}{N} \sum_{q \in [Q]} \sum_{n \in [N]} l_q \left(y_n - \left((\mathbf{G}_N [\mathbf{A}, \tilde{\mathbf{A}}])_{n,q} + b_q \right) \right) \\ & + \lambda_b \|\mathbf{b}\|_2^2 + \lambda_g \sum_{q=1}^Q \left\| \mathbf{G}^{\frac{1}{2}} [\mathbf{A}; \tilde{\mathbf{A}}] \tilde{\mathbf{e}}_q \right\|_2^2, \end{aligned}$$

subject to

$$-(\mathbf{U}\mathbf{b})_i + \eta \left\| \mathbf{G}^{\frac{1}{2}} \mathbf{h}_i \right\|_2 \leq \min_{m \in [M]} (\mathbf{G}\mathbf{h}_i)_{N+m}, \quad \forall i \in [I], \quad \forall q \in [Q].$$

Using the reproducing property of kernels, we can calculate a simplified expression for the η coefficient

$$\begin{aligned} \eta &= \sup_{\mathbf{u} \in B, m \in [M]} \|k(\tilde{\mathbf{x}}_m, \cdot) - k(\tilde{\mathbf{x}}_m + \delta\mathbf{u}, \cdot)\|_k \\ &= \sup_{\mathbf{u} \in B, m \in [M]} [\langle k(\tilde{\mathbf{x}}_m, \cdot), k(\tilde{\mathbf{x}}_m, \cdot) \rangle_k + \langle k(\tilde{\mathbf{x}}_m + \delta\mathbf{u}, \cdot), k(\tilde{\mathbf{x}}_m + \delta\mathbf{u}, \cdot) \rangle_k - 2 \langle k(\tilde{\mathbf{x}}_m + \delta\mathbf{u}, \cdot), k(\tilde{\mathbf{x}}_m, \cdot) \rangle_k]^{\frac{1}{2}} \\ &= \sup_{\mathbf{u} \in B, m \in [M]} [k(\tilde{\mathbf{x}}_m, \tilde{\mathbf{x}}_m) + k(\tilde{\mathbf{x}}_m + \delta\mathbf{u}, \tilde{\mathbf{x}}_m + \delta\mathbf{u}) - 2k(\tilde{\mathbf{x}}_m + \delta\mathbf{u}, \tilde{\mathbf{x}}_m)]^{\frac{1}{2}}. \end{aligned}$$

Assuming that the kernel is such that $k(\mathbf{x}, \mathbf{x}) = 1 \forall \mathbf{x}$, then the value of η reduces to

$$\eta = \sup_{\mathbf{u} \in B, m \in [M]} \sqrt{2} [1 - k(\tilde{\mathbf{x}}_m + \delta\mathbf{u}, \tilde{\mathbf{x}}_m)]^{\frac{1}{2}}.$$

Specifically, taking the

- Gaussian kernel $k(\mathbf{x}, \mathbf{x}') = e^{-\frac{\|\mathbf{x}-\mathbf{x}'\|_2^2}{\gamma^2}}$, we have

$$\eta = \sup_{\mathbf{u} \in B, m \in [M]} \sqrt{2} \left[1 - e^{-\frac{\|\delta\mathbf{u}\|_2^2}{\gamma^2}} \right]^{\frac{1}{2}} = \sqrt{2} \left[1 - e^{-\frac{\|\delta\|_2^2}{\gamma^2}} \right]^{\frac{1}{2}},$$

and it is sufficient to solve the optimization problem

$$\begin{aligned} & \min_{\substack{\mathbf{A} \in \mathbb{R}^{N \times Q}, \mathbf{b} \in \mathbb{R}^Q \\ \tilde{\mathbf{A}} \in \mathbb{R}^{M \times Q}}} \frac{1}{N} \sum_{q \in [Q]} \sum_{n \in [N]} l_q \left(y_n - \left((\mathbf{G}_N [\mathbf{A}, \tilde{\mathbf{A}}])_{n,q} + b_q \right) \right) + \lambda_b \|\mathbf{b}\|_2^2 \\ & + \lambda_g \sum_{q=1}^Q \left\| \mathbf{G}^{\frac{1}{2}} [\mathbf{A}; \tilde{\mathbf{A}}] \tilde{\mathbf{e}}_q \right\|_2^2, \end{aligned}$$

subject to the constraint

$$-(\mathbf{U}\mathbf{b})_i + \sqrt{2} \left[1 - e^{-\frac{\|\delta\|_2^2}{\gamma^2}} \right]^{\frac{1}{2}} \left\| \mathbf{G}^{\frac{1}{2}} \mathbf{h}_i \right\|_2 \leq \min_{m \in [M]} (\mathbf{G}\mathbf{h}_i)_{N+m}, \quad \forall i \in [I], \quad \forall q \in [Q].$$

- Laplacian kernel $k(\mathbf{x}, \mathbf{x}') = e^{-\frac{\|\mathbf{x}-\mathbf{x}'\|_2}{\gamma^2}}$, one gets

$$\eta = \sup_{\mathbf{u} \in B, m \in [M]} \sqrt{2} \left[1 - e^{-\frac{\|\delta \mathbf{u}\|_2}{\gamma^2}} \right]^{\frac{1}{2}} = \sqrt{2} \left[1 - e^{-\frac{\|\delta\|_2}{\gamma^2}} \right]^{\frac{1}{2}},$$

and the optimization problem to solve is

$$\begin{aligned} & \min_{\substack{\mathbf{A} \in \mathbb{R}^{N \times Q}, \mathbf{b} \in \mathbb{R}^Q \\ \tilde{\mathbf{A}} \in \mathbb{R}^{M \times Q}}} \frac{1}{N} \sum_{q \in [Q]} \sum_{n \in [N]} l_q \left(y_n - \left(\left(\mathbf{G}_N [\mathbf{A}, \tilde{\mathbf{A}}] \right)_{n,q} + b_q \right) \right) + \lambda_b \|\mathbf{b}\|_2^2 \\ & \quad + \lambda_g \sum_{q=1}^Q \left\| \mathbf{G}^{\frac{1}{2}} [\mathbf{A}; \tilde{\mathbf{A}}] \tilde{\mathbf{e}}_q \right\|_2^2, \end{aligned}$$

subject to the constraint

$$-(\mathbf{U}\mathbf{b})_i + \sqrt{2} \left[1 - e^{-\frac{\|\delta\|_2}{\gamma^2}} \right]^{\frac{1}{2}} \left\| \mathbf{G}^{\frac{1}{2}} \mathbf{h}_i \right\|_2 \leq \min_{m \in [M]} (\mathbf{G}\mathbf{h}_i)_{N+m}, \quad \forall i \in [I], \forall q \in [Q].$$

In the next chapter, we demonstrate the applicability of these tightened SOC-based solutions in risk estimation.

Chapter 4

Numerical Experiments

In this chapter we demonstrate the efficiency of the SOC-based hard-shape constrained kernel machine framework in the context of risk estimation. In Section 1 we focus on joint quantile regression (JQR). Section 2 is about joint expectile regression (JER).

The **goal** of the numerical experiments is various-fold:

1. to investigate the estimation difficulty of different risk measures (quantile, expectile),
2. to explore the scalability of the proposed approach as a function of the sample size N and dimension d ,
3. to understand the stability of the estimators using Monte-Carlo experiments,
4. to study the feasibility of the estimation as a function of various parameters of the portfolio and the market (interest rate r , size of the portfolio I),
5. to illustrate the efficiency of the estimation applying different hypothesis classes \mathcal{F}_k encoded by the choice of the kernel k .

In our experiments

- we investigated the efficiency of both the Gaussian kernel and the Laplacian kernel. Their bandwidth γ was chosen to be the median of the Euclidean distance between the \mathbf{x}_n points. According to our experiences this was a reliable choice; it sets the value of γ according to the scale of the data.
- We replaced the quadratic penalties (see (2.9)) with the equivalent $\sqrt{\sum_{q \in [Q]} \|f_q\|_k^2} \leq \tilde{\lambda}_g, \|\mathbf{b}\|_2 \leq \tilde{\lambda}_b$ forms. This is the formulation suggested by the used CVX solver.

- We used the CVXPY framework on Python 3.7 with the MOSEK solver. The experiments took from seconds to a few minutes to run on a Ryzen 5-CPU 20GB-RAM laptop.
- We applied 2-fold cross-validation (CV) to choose the optimal hyperparameters $\tilde{\lambda}_g$ and $\tilde{\lambda}_b$. The cross-validation was carried out on the domains

$$\begin{aligned}\tilde{\lambda}_g &\in \left\{4.5^i, i \in [7]\right\}, \\ \tilde{\lambda}_b &\in \left\{10^{\log i} \times \max_{n \in [N]} |y_n|, i \in \{e, 3, 4.5, 6, e^2\}\right\}.\end{aligned}$$

Notice that the studied values of $\tilde{\lambda}_b$ incorporate the scale of the output data ($\max_{n \in [N]} |y_n|$), which we found to be beneficial in practice.

- The time was normalized to the interval $[0, 1]$ corresponding to one year. The time points t_1 and t_2 were chosen to be 10 and 20 days.
- For JQR we learned jointly five quantile functions ($\tau_q \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$), for EJR we estimated jointly 5 expectile functions ($\alpha_q \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$).
- The GBM parameters were chosen as follows. In
 - dimension one ($d = 1$): volatility $\sigma_1 = 0.2$, drift $\mu_1 = 0.01$, initialization $S_0 = 440$.
 - dimension two ($d = 2$): correlation $\rho_{12} = 0.3$, volatilities $\sigma_1 = 0.2$, $\sigma_2 = 0.25$, drifts $\mu_1 = 0.01$, $\mu_2 = -0.03$, initialization $S_0 = [440; 460]$.
- The portfolio parameters were as follows. In case of
 - $d = 1, I = 1$:

$$\Theta = \left(\left(1, 1, \frac{1}{12}, 455, 1 \right) \right).$$

– $d = 1, I = 3$:

$$\Theta = \left(\left(1, 1, \frac{1}{12}, 455, 1 \right), \left(0, 1, \frac{1}{12}, 455, 0.9 \right), \left(0, 1, \frac{2}{12}, 435, -1.3 \right) \right).$$

– $d = 1, I = 10$:

$$\Theta = \left(\left(1, 1, \frac{1}{12}, 455, 1 \right), \left(0, 1, \frac{1}{12}, 455, 0.9 \right), \left(0, 1, \frac{2}{12}, 435, -1.3 \right), \right. \\ \left(1, 1, \frac{2}{12}, 435, -1.3 \right), \left(1, 1, \frac{3}{12}, 405, 0.5 \right), \left(0, 1, \frac{3}{12}, 405, 0.5 \right), \\ \left(1, 1, \frac{5}{12}, 370, -1.2 \right), \left(0, 1, \frac{5}{12}, 370, -1.3 \right), \left(1, 1, \frac{2}{12}, 475, 0.72 \right), \\ \left. \left(0, 1, \frac{2}{12}, 475, 0.73 \right) \right).$$

– $d = 2, I = 2$:

$$\Theta = \left(\left(1, 1, \frac{1}{12}, 455, 1 \right), \left(1, 2, \frac{1}{12}, 475, 0.5 \right) \right).$$

1. Joint Quantile Regression

This section is about joint quantile regression. In Section 1.1, the dimension is kept to be one ($d = 1$). The dimension was chosen to be two ($d = 2$) in Section 1.2.

1.1 One-Dimensional Data ($d = 1$)

To investigate the efficiency of the SOC approach in the JQR task in case of $d = 1$ we designed the following experiments.

- **Experiment 1** (effect of N): In this experiment we studied the effect of the sample size N on the estimated quantile functions. We fixed the interest rate to be $r = 5\%$, the portfolio was simple ($I = 1$), and used the Laplacian kernel. The estimated quantile functions are illustrated in Fig. 4.1 with the associated computational times in Fig. 4.2.
- **Experiment 2** (interest rate r , kernel k): Here we studied how various interest rates ($r \in \{5\%, 10\%, 15\%, 25\%\}$) and the choice of the kernel (k : Gaussian or Laplacian) affect the estimation, while keeping the portfolio simple ($I = 1$) and the sample size fixed $N = 250$. In order to assess the stability of the estimation the experiment (cross-validation) was repeated 10 times. The mean and standard deviation of the resulting scores are summarized in Table 4.1. The underlying cross-validation surfaces are illustrated in Fig. 4.3.
- **Experiment 3** (portfolio size I , kernel k): In the third experiment, the complexity of the portfolio parameterized by its size $I \in \{1, 3, 10\}$ was varied while allowing both kernels (k : Gaussian or Laplacian). The summary statistics of

Interest rate r	mean \pm std GK	mean \pm std LK	Interval GK	Interval LK
5%	18.96 ± 2.24	13.30 ± 0.61	[12.88, 20.54]	[12.68, 14.13]
10%	19.83 ± 1.61	17.50 ± 1.74	[16.53, 22.40]	[15.46, 19.53]
15%	20.21 ± 2.10	19.52 ± 1.69	[16.23, 23.75]	[18.10, 22.38]
25%	21.65 ± 1.27	23.82 ± 1.71	[20.12, 22.65]	[21.28, 26.05]

Table 4.1: JQR task ($d = 1$) - summary statistics of the optimal CV scores for different interest rates using the Gaussian kernel (GK) and the Laplacian kernel (LK). 1st column: interest rate. 2nd column: mean \pm std for the Gaussian kernel. 3rd column: mean \pm std for the Laplacian kernel. 4th column: CV score interval [min, max] for the Gaussian kernel. 5th column: CV score interval [min, max] for the Laplacian kernel.

the resulting profit-and-loss values are given in Table 4.2, the summary statistics of the CV scores are available in Table 4.3. The estimated quantile curves are illustrated for the two extremes ($I = 1, I = 10$) in Fig. 4.4, for both kernels with interest rate $r = 1.5\%$ and sample size $N = 200$.

Our experiences can be summarized as follows.

- **Experiment-1** (Fig. 4.1, Fig. 4.2): One can observe (Fig. 4.1) that the estimated quantile functions tend to become smoother as the sample size N is increasing. The increasing sample size comes at a higher computational time though as it can be read out from Fig. 4.2: it follows an approximately quadratic ascent.
- **Experiment-2** (Table 4.1, Fig. 4.3): As it can be observed in Table 4.1, the CV scores are increasing as the interest rate does so. Moreover, for the more realistic $r \leq 15\%$ interest rate range the Laplacian kernel achieves more accurate estimation. Fig. 4.3 shows that the CV surfaces have a U-shaped behaviour: this indicates that a sufficient domain of the hyperparameters was investigated and that the hyperparameters can be selected properly.
- **Experiment-3** (Table 4.2, Table 4.3, Fig. 4.4): Table 4.2 shows that the dispersion of the data is increasing as the number of elements in the portfolio grows; this also leads to less accurate estimation in terms of the CV scores (Table 4.3). Similarly, to Experiment-2 our results indicate that the Laplacian kernel seems to be more favourable. Though the estimated quantile functions (Fig. 4.4) associated to both kernels satisfy the imposed hard shape requirements, the Laplacian kernel is visually also preferred: it captures the scattered pattern of the data more accurately.

Remark (min kernel): We note that in addition to the Gaussian and the Laplacian kernel, we investigated the min kernel which underlies the Brownian motion. It is

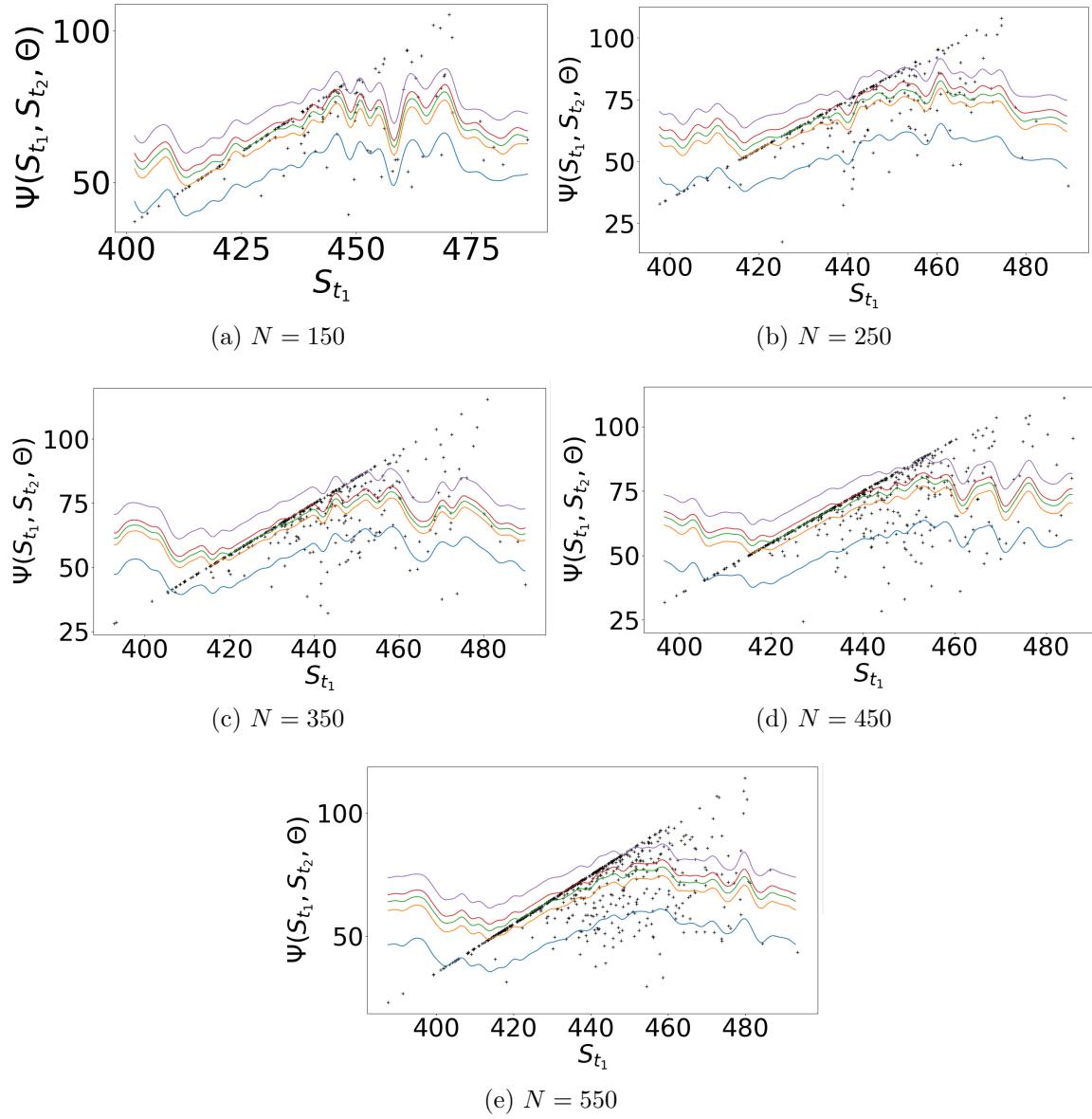
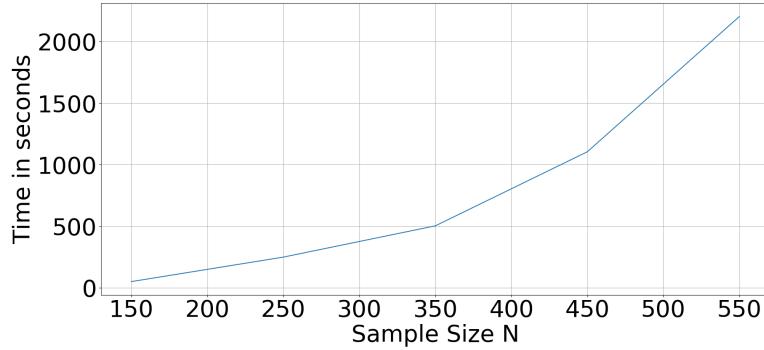


Figure 4.1: JQR task ($d = 1$) – estimated quantile functions for different sample sizes (N) and for the optimal hyperparameters. (a): $N = 150$. (b): $N = 250$. (c): $N = 350$. (d): $N = 450$. (e): $N = 550$.

Figure 4.2: JQR task ($d = 1$) – computational time as a function of the sample size.

I	mean of $\Psi(S_{t_1}, S_{t_2}, \Theta)$	std
1	13.22	± 10.96
3	- 5.73	± 23.15
10	12.82	± 29.35

Table 4.2: JQR task ($d = 1$) – summary statistics of the P&L for different portfolio sizes. 1st column: number of elements in the portfolio. 2nd column: mean. 3rd column: std.

I	mean \pm std GK	mean \pm std LK
1	11.48 ± 1.42	8.97 ± 1.76
3	28.08 ± 0.57	22.86 ± 1.23
10	51.77 ± 0.62	28.90 ± 2.02

Table 4.3: JQR task ($d = 1$) – summary statistics of the CV scores for different portfolio sizes. 1st column: number of elements in the portfolio. 2nd column: mean \pm std for the Gaussian kernel (GK). 3rd column: mean \pm std for the Laplacian kernel (LK).

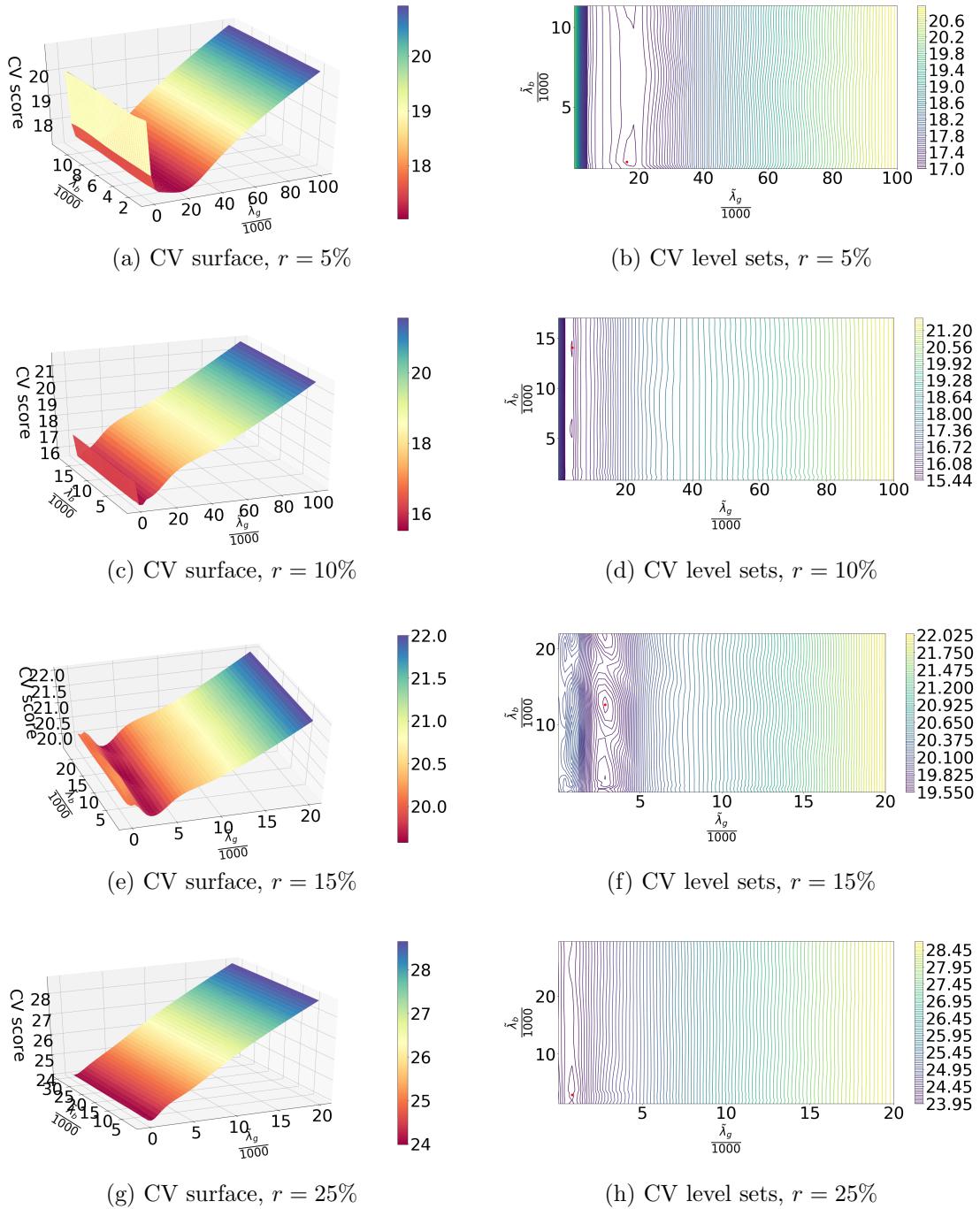


Figure 4.3: JQR task ($d = 1$) – cross-validation scores for various interest rates (r).
 1st column: CV surfaces. 2nd column: CV level sets; the minimum is indicated with a red dot. 1st row: $r = 5\%$. 2nd row: $r = 10\%$. 3rd row: $r = 15\%$. 4th row: $r = 25\%$.

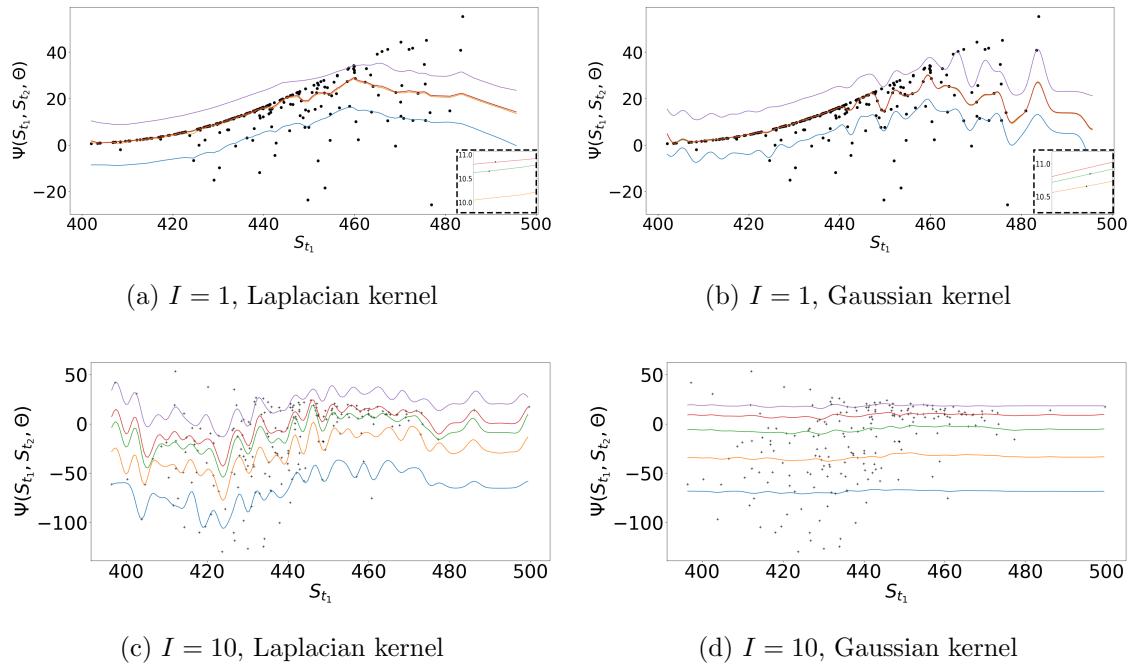


Figure 4.4: JQR task ($d = 1$) – estimated quantile functions for different portfolio sizes. 1st row: $I = 1$. 2nd row: $I = 10$. Left: Laplacian kernel. Right: Gaussian kernel. The colours indicate quantile values (τ). Blue: $\tau = 0.1$. Orange: $\tau = 0.3$. Green: $\tau = 0.5$. Red: $\tau = 0.7$. Purple: $\tau = 0.9$.

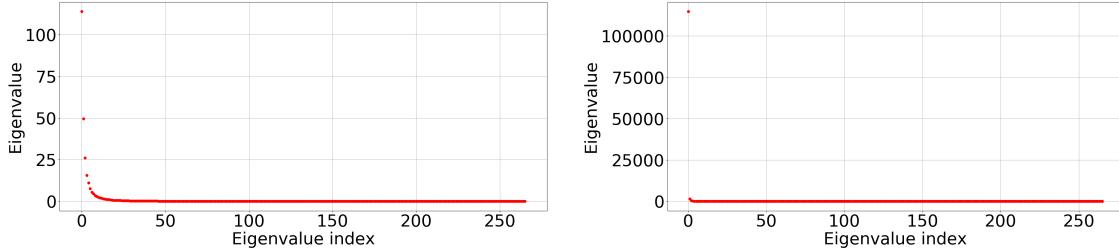


Figure 4.5: Eigenspectrum comparison of the Gram matrix for the Laplacian and the min kernel. Left: Laplacian kernel. Right: min kernel.

defined as

$$k(\mathbf{x}, \mathbf{y}) = \sum_{i \in [d]} \min(x_i, y_i), \quad (\mathbf{x}, \mathbf{y}) \in \mathbb{R}^d. \quad (4.1)$$

We do not recommend using this kernel: according to our numerical experiences, the CVX solver often becomes unstable under this choice. A possible reason of this instability is the highly concentrated spectrum of the Gram matrix \mathbf{G} . This behaviour is illustrated and compared to the Laplacian kernel in Fig. 4.5. In the example the dominant eigenvalue was around 10^5 for the min kernel, while the remaining ones were negligible in the order of 10^{-14} .

1.2 2-Dimensional Data ($d = 2$)

In order to investigate the scalability of the SOC-based approach in terms of dimension in JQR, in this section we considered two stocks ($d = 2$) which were assumed to follow jointly a 2-dimensional GBM. Fig. 4.6 provides an illustration of the data in case of an interest rate $r = 1.5\%$. In accordance with the experiences of Section 1.2, the kernel was fixed to be the Laplacian.

We performed the following experiment:

- **Experiment** (effect of r and d , stability): The focus was on the accuracy of the estimation as a function of the interest rate r . In order to assess the stability of the approach the experiment was repeated 10 times for each interest rate. The results are summarized and compared to the case of a portfolio with a single stock ($d = 1$) in Table 4.4. The corresponding computational times are plotted in Fig. 4.7. In addition we provide an illustration for the estimated quantile functions in case of interest rate $r = 1.5\%$ in Fig. 4.8, for clarity showing only the surfaces associated to $\tau = 0.1$, $\tau = 0.7$ and $\tau = 0.9$.

Our numerical experiences can be summarized as follows.

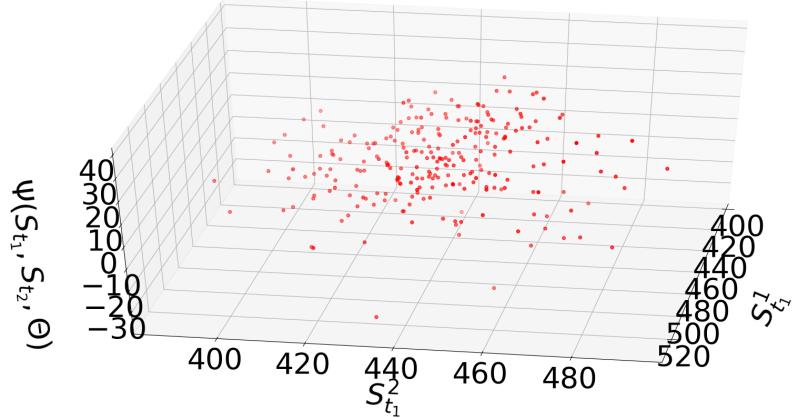


Figure 4.6: Illustration of the profit-and-loss values for $d = 2$. x-axis: price of stock-1 at time t_1 ($S_{t_1}^1$). y-axis: price of stock-2 at time t_1 ($S_{t_1}^2$). z-axis: Profit-and-loss at time t_2 .

Interest rate r	mean±std $d = 2$	mean±std $d = 1$	Interval $d = 2$	Interval $d = 1$
5%	17.43 ± 2.42	13.30 ± 0.61	[$14.71, 21.34$]	[$12.68, 14.13$]
10%	21.41 ± 0.95	17.50 ± 1.74	[$19.95, 22.448$]	[$15.46, 19.53$]
15%	24.30 ± 1.21	19.52 ± 1.69	[$22.43, 25.47$]	[$18.10, 22.38$]

Table 4.4: JQR task – summary statistics of the optimal CV scores: $d = 2$ vs $d = 1$. 1st column: interest rate. 2nd column: mean±std for $d = 2$. 3rd column: mean±std for $d = 1$. 4th column: CV score interval [min, max] for $d = 2$. 5th column: CV score interval [min, max] for $d = 1$.

- **Experiment** (Table 4.4, Fig. 4.7, Fig. 4.8): As we can read it out from Table 4.4 the accuracy of the estimation drops slightly as the dimension is increasing following the intuition that one has to solve a harder problem. The difficulty is also reflected in the computational times (Fig. 4.7): the required time almost doubles as one moves from $d = 1$ to $d = 2$. As Fig. 4.8 shows the estimated quantile functions fulfill the hard shape constraints, and represent well the variations of the profit-and-loss values.

2. Joint Expectile Regression

Having studied the joint quantile regression problem (Section 1), we now move on to joint expectile regression. Section 2.1 is about the one-dimensional case; the two-dimensional setting is the focus of Section 2.2.

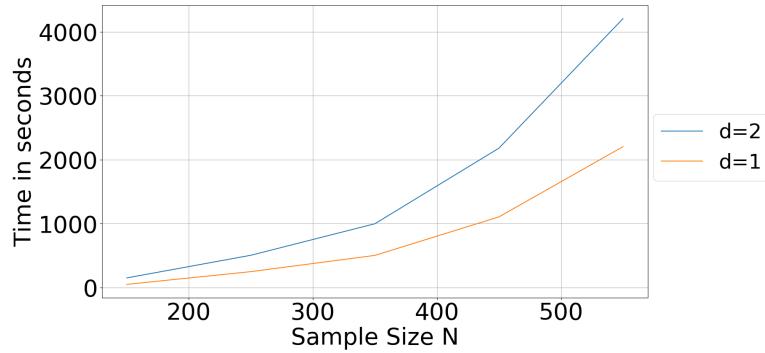


Figure 4.7: JQR task – comparison of the computational time: $d = 2$ vs $d = 1$. The colours indicate dimensions. Blue: $d = 2$. Orange: $d = 1$.

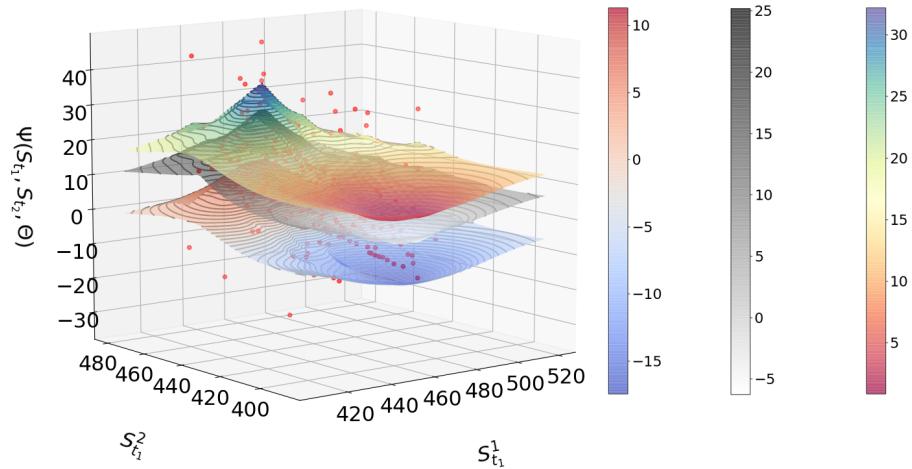


Figure 4.8: JQR task ($d = 2$) – illustration of the estimated quantile surfaces. The colours indicate different quantile levels. From left to right: coolwarm colormap ($\tau = 0.1$), binary colormap ($\tau = 0.7$), spectral colormap ($\tau = 0.9$).

Interest rate r	mean±std GK	mean±std LK
5%	45.42 ± 2.55	38.43 ± 2.07
10%	32.19 ± 0.92	27.05 ± 1.72
15%	27.29 ± 2.50	25.81 ± 2.32

Table 4.5: EJR task ($d = 1$) – summary statistics of the CV scores for different interest rates. 1st column: interest rate, 2nd column: mean±std for the Gaussian kernel (GK), 3rd column: mean±std for the Laplacian kernel (LK). The number of elements in the portfolio is $I = 10$.

2.1 One-dimensional Data ($d = 1$)

To assess the performance of the SOC-based approach in joint expectile regression we designed the following experiments.

- **Experiment-1** (effect of r , stability): The focus of the first experiment is to illustrate the effect of the interest rate r on the quality of the estimation. The number of elements in the portfolio was fixed to be $I = 1$ and $I = 10$. The sample size was $N = 250$. To simultaneously evaluate the stability of the method, we repeated the experiment (cross-validation) 10 times. We allowed the kernel to be both Laplacian and Gaussian. The data is illustrated in Fig. 4.9. The cross-validation scores are summarized in Table 4.5. In Fig. 4.10 we illustrate the estimated expectile functions for $I = 1$, $r = 1.5\%$, with the Laplacian kernel.
- **Experiment-2** (I -dependence, stability): In the second experiment, the complexity of the portfolio expressed via the number of portfolio elements I was investigated. The interest rate was fixed to $r = 1.5\%$, the sample size was $N = 250$. The cross-validation was carried out 10 times to evaluate the stability of the approach. The kernel was the Laplacian (as it was more competitive based on Experiment-1). The obtained results are summarized in Table 4.6.

Our experiences can be summarized as follows:

- **Experiment-1** (Fig. 4.9, Table 4.5, Fig. 4.10): Fig. 4.9 shows that for $I = 10$ growing interest rate gives rise to reduced dispersion of the data (unlike for $I = 1$). This behaviour is reflected in decreasing CV scores (Table 4.5). Similarly to the JQR task, the application of the Laplacian kernel looks favourable; the resulting estimates accurately model the observations while satisfying the hard shape restrictions (see Fig. 4.10).
- **Experiment-2** (Table 4.6): Table 4.6 indicates that as the size of the portfolio grows, the estimation task becomes more challenging and the CV scores grow benignly.

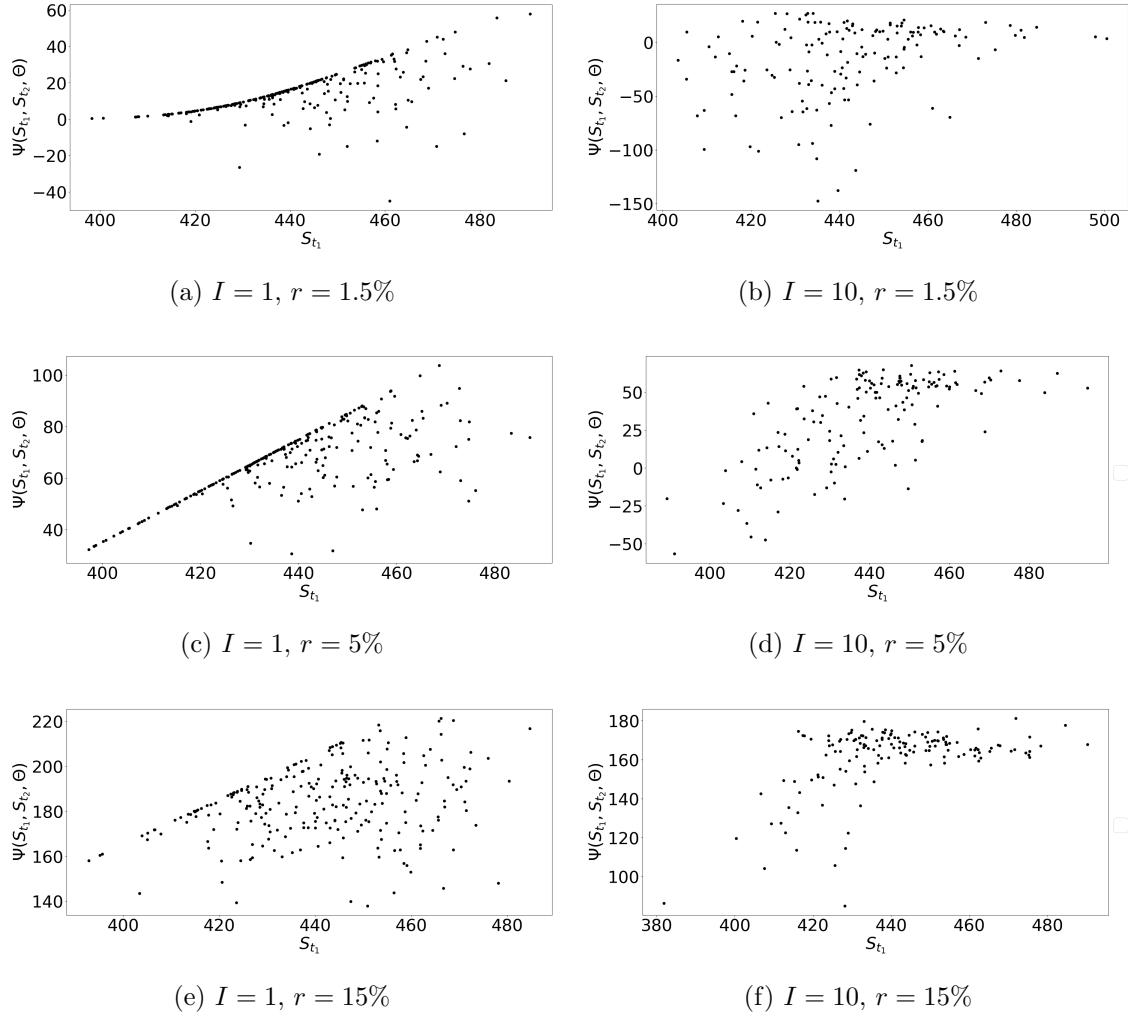


Figure 4.9: Illustration of the data for different interest rates ($r \in \{1.5\%, 5\%, 15\%\}$).
 1st column: simple portfolio ($I = 1$). 2nd column: complex portfolio ($I = 10$). 1st row: $r = 1.5\%$. 2nd row: $r = 5\%$. 3rd row: $r = 15\%$.

I	mean \pm std
1	13.10 ± 1.50
3	17.45 ± 0.34
10	46.35 ± 3.80

Table 4.6: EJR task ($d = 1$) – summary statistics of the CV scores for different portfolio sizes. 1st column: number of elements in the portfolio. 2nd column: mean \pm std.

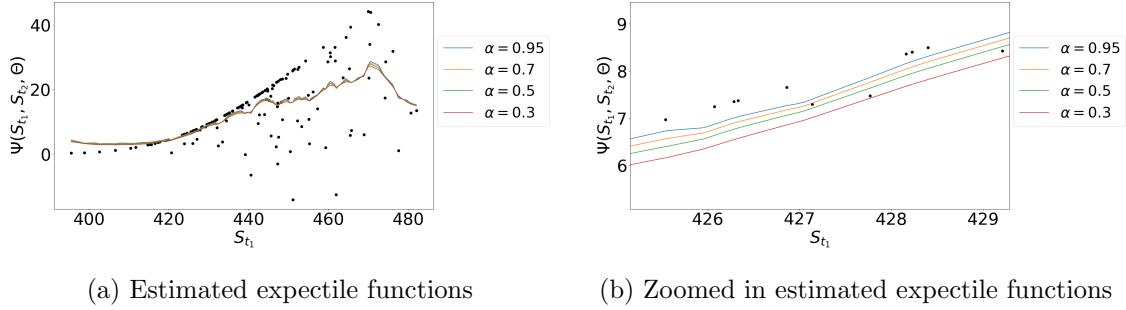


Figure 4.10: EJR task ($d = 1$) – illustration of estimated expectile functions. (a): estimated expectile functions. (b): zoomed in view to show that the estimates respect the shape restrictions, i.e. there are no crossings.

2.2 2-Dimensional Data ($d = 2$)

This section is dedicated to joint expectile regression with 2 stocks ($d = 2$). In accordance with the experiences collected for $d = 1$ (Section 2.1), the kernel was fixed to be the Laplacian.

We performed the following experiments.

- **Experiment-1 (effect of r , stability):** In the first experiment we investigated the effect of interest rate ($r \in \{5\%, 10\%, 15\%\}$) on the quality of the estimation. In the portfolio there was one option for each stock ($I = 2$). To illustrate the performance of the approach, the experiment (cross-validation) was repeated 10 times; the obtained results are available in Table 4.7. The estimated expectile functions are illustrated in Fig. 4.12 for $r = 1.5\%$ and $N = 250$.
- **Experiment-2 (computational times):** JQR and JER use different loss functions. In this experiment we illustrate the running times associated to the two tasks as a function of the sample size N ; the results are summarized in Fig. 4.11. The number of options in the portfolio was $I = 2$ and the interest rate was fixed to $r = 1.5\%$.

Our experiences are as follows.

- **Experiment-1 (Table 4.7, Fig. 4.12):** Table 4.7 indicates that for smaller interest rates the estimation becomes more accurate. As it can be seen in Fig. 4.12, the fitted expectile functions capture the underlying trend of the data and respect the hard shape constraints.
- **Experiment-2 (Fig. 4.11):** The graphs in Fig. 4.11 show that the computational time for joint quantile regression is comparable to that of expectile regression, up to around a factor of two on the considered sample size domain.

Interest rate r	mean \pm std
5%	19.80 \pm 2.10
10%	25.43 \pm 1.81
15%	26.48 \pm 0.59

Table 4.7: EJR task ($d = 2$) – summary statistics of the CV scores for different interest rates. 1st column: interest rates. 2nd column: mean \pm std.

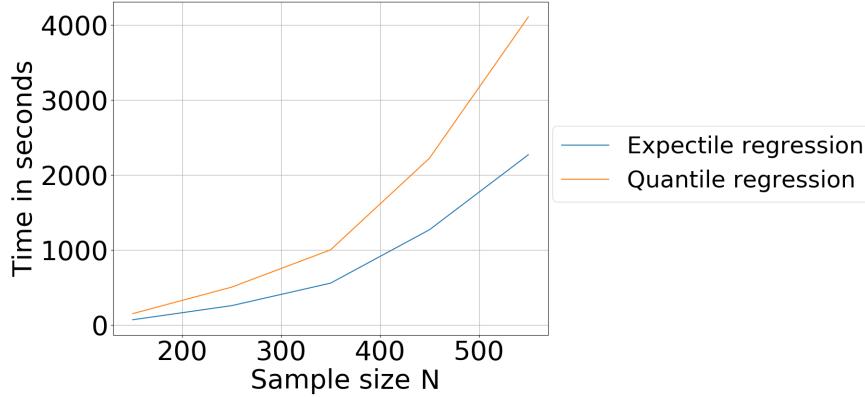


Figure 4.11: Computational time: expectile vs quantile regression ($d = 2$).

These experiments demonstrate the efficiency of the SOC-based approach in risk estimation.

Remark (encoding of the losses): Finally we make the practical note that the pinball and expectile losses have various equivalent formulations. The CVX package performs checking to ensure the convexity of the optimization problem. To be compatible with this functionality one can encode the losses as:

$$\begin{aligned} l_\tau(z) &= \frac{1}{2} |z| + \left(\tau - \frac{1}{2}\right) z, \\ l_\alpha(z) &= \mathbb{I}_{\{z<0\}} (1-\alpha) z^2 + \mathbb{I}_{\{z\geq 0\}} \alpha z^2. \end{aligned}$$

In the pinball loss l_τ it is sufficient to encode $|z|$ as the CVX function `cvxpy.abs()`, in l_α the sign of z ($\mathbb{I}_{\{z<0\}}$) can be coded up by the CVX function `.is_nonpos()`.

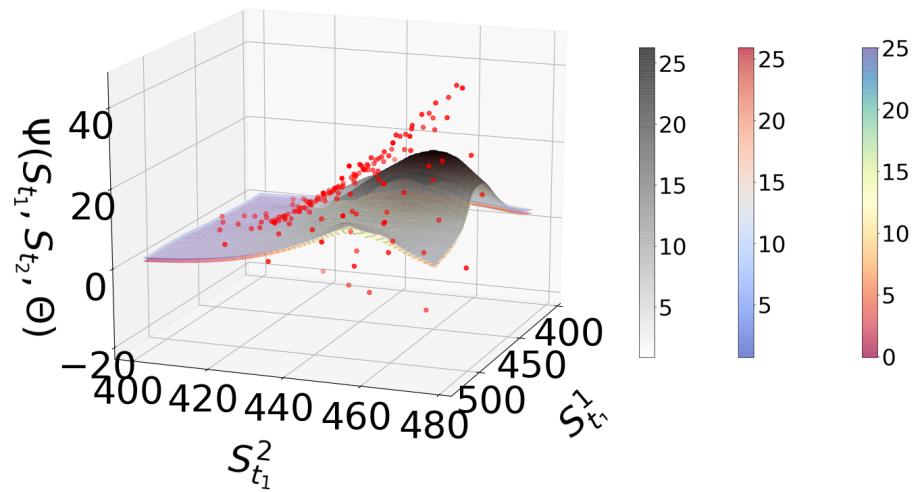


Figure 4.12: EJR task ($d = 2$) – illustration of the estimated expectile surfaces. The colours indicate different expectile levels. From left to right: binary colormap ($\alpha = 0.5$), coolwarm colormap ($\alpha = 0.3$), spectral colormap ($\alpha = 0.1$).

Chapter 5

Conclusions

In this internship we focused on the estimation of risk measures under non-crossing hard shape constraints and using a reproducing kernel Hilbert space as hypothesis class. Particularly, we tackled the tasks of joint quantile and expectile regression. We ran various numerical experiments for the risk estimation of the profit-and-loss of portfolios consisting of call and put options, under different choices of the kernel, sample size, dimension and market behaviours modelled by a geometric Brownian motion. We demonstrated the efficiency of a recent second-order cone-constrained approach which ensures the imposed hard shape constraints and is capable of accurately capturing the patterns of the observation.

In this work we limited ourselves to two risk measures and one shape constraint. Various research directions are left open, including the study of alternative risk measures, shape constraints, different stochastic market model models, higher-dimensional observations, more complex portfolios and real-world data.

Bibliography

- Christian Agrell. Gaussian processes with linear operator inequality constraints. *Journal of Machine Learning Research*, 20:1–36, 2019.
- Nachman Aronszajn. Theory of reproducing kernels. *Transactions of the American Mathematical Society*, 68:337–404, 1950.
- Pierre-Cyril Aubin-Frankowski and Zoltán Szabó. Hard shape-constrained kernel machines. Technical report, 2020. (<http://arxiv.org/abs/2005.12636>).
- Necdet Serhat Aybat and Zi Wang. A parallel method for large scale convex regression problems. In *Conference on Decision and Control (CDC)*, pages 5710–5717, 2014.
- J. Andrew Bagnell and Amir Farahmand. Learning positive functions in a Hilbert space. NIPS Workshop on Optimization, (OPT2015), 2015. (https://www.ri.cmu.edu/pub_files/2015/0/Kernel-SOS.pdf).
- Alain Berlinet and Christine Thomas-Agnan. *Reproducing Kernel Hilbert Spaces in Probability and Statistics*. Kluwer, 2004.
- Richard Blundell, Joel L. Horowitz, and Matthias Parey. Measuring the price responsiveness of gasoline demand: economic shape restrictions and nonparametric demand estimation. *Quantitative Economics*, 3:29–51, 2012.
- Claudio Carmeli, Ernesto De Vito, Alessandro Toigo, and Veronica Umanitá. Vector valued reproducing kernel Hilbert spaces and universality. *Analysis and Applications*, 8:19–61, 2010.
- Yining Chen and Richard J. Samworth. Generalized additive and index models with shape constraints. *Journal of the Royal Statistical Society – Statistical Methodology, Series B*, 78(4):729–754, 2016.
- Michel Delecroix, Michel Simioni, and Christine Thomas-Agnan. Functional estimation under shape constraints. *Journal of Nonparametric Statistics*, 6(1):69–89, 1996.

Seth Flaxman, Yee Whye Teh, and Dino Sejdinovic. Poisson intensity estimation with reproducing kernels. *Electronic Journal of Statistics*, 11(2):5081–5104, 2017.

Georgina Hall. Optimization over nonnegative and convex polynomials with and without semidefinite programming. PhD Thesis, Princeton University, 2018.

Qiyang Han, Tengyao Wang, Sabyasachi Chatterjee, and Richard J. Samworth. Isotonic regression in general dimensions. *Annals of Statistics*, 47(5):2440–2471, 2019.

Alec Koppel, Kaiqing Zhang, Hao Zhu, and Tamer Başar. Projected stochastic primal-dual method for constrained online learning with kernels. *IEEE Transactions on Signal Processing*, 67(10):2528–2542, 2019.

Rahul Mazumder, Arkopal Choudhury, Garud Iyengar, and Bodhisattva Sen. A computational framework for multivariate convex regression and its variants. *Journal of the American Statistical Association*, 114(525):318–331, 2019.

Mary C. Meyer. A framework for estimation and inference in generalized additive models with shape and order restrictions. *Statistical Science*, 33(4):595–614, 2018.

Charles Micchelli, Yuesheng Xu, and Haizhang Zhang. Universal kernels. *Journal of Machine Learning Research*, 7:2651–2667, 2006.

Sovan Mitra and Tong Ji. Risk measures in quantitative finance. *International Journal of Business Continuity and Risk Management*, 1(2), 2010.

Dávid Papp and Farid Alizadeh. Shape-constrained estimation using nonnegative splines. *Journal of Computational and Graphical Statistics*, 23(1):211–231, 2014.

Natalya Pya and Simon N. Wood. Shape constrained additive models. *Statistics and Computing*, 25:543–559, 2015.

Maxime Sangnier, Olivier Fercoq, and Florence d’Alché Buc. Joint quantile regression in vector-valued RKHSs. *Advances in Neural Information Processing Systems (NIPS)*, pages 3693–3701, 2016.

Carl-Johann Simon-Gabriel and Bernhard Schölkopf. Kernel distribution embeddings: Universal kernels, characteristic kernels and kernel metrics on distributions. *Journal of Machine Learning Research*, 19(44):1–29, 2018.

Bharath Sriperumbudur, Kenji Fukumizu, and Gert Lanckriet. Universality, characteristic kernels and RKHS embedding of measures. *Journal of Machine Learning Research*, 12:2389–2410, 2011.

Ingo Steinwart. On the influence of the kernel on the consistency of support vector machines. *Journal of Machine Learning Research*, 6(3):67–93, 2001.

- Ingo Steinwart and Andreas Christmann. *Support Vector Machines*. Springer, 2008.
- Ichiro Takeuchi, Quoc Le, Timothy Sears, and Alexander Smola. Nonparametric quantile estimation. *Journal of Machine Learning Research*, 7:1231–1264, 2006.
- Berwin A. Turlach. Shape constrained smoothing using smoothing splines. *Computational Statistics*, 20:81–104, 2005.
- Jiwen Wu, Mary C. Meyer, and Jean D. Opsomer. Penalized isotonic regression. *Journal of Statistical Planning and Inference*, 161:12–24, 2015.
- Ximing Wu and Robin Sickles. Semiparametric estimation under shape constraints. *Econometrics and Statistics*, 6:74–89, 2018.
- Daisuke Yagi, Yining Chen, Andrew L. Johnson, and Timo Kuosmanen. Shape-constrained kernel-weighted least squares: Estimating production functions for Chilean manufacturing industries. *Journal of Business & Economic Statistics*, 38(1):43–54, 2020.