Model Free Optimisation in Risk Management

Sergey Shahverdyan

Lincoln College

University of Oxford

A thesis submitted for the degree of

Doctor of Philosophy

Trinity 2014

I express my deepest gratitude to my thesis Supervisor, Dr. Raphael Hauser for his guidance, support and motivation. His invaluable suggestions and encouragement during my work made it possible.

I am also grateful to the organisations that sponsored my work: EPSRC through grant EP/H02686X/1, Luys Foundation, Armenia and the Lord Crewe Scholarship, Lincoln College, Oxford.

I thank my family for their continuous support.

Following the financial crisis of 2008, the need for more robust techniques to quantify the capital charge for risk management has become a pressing problem. Under Basel II/III, banks are allowed to calculate the capital charge using internally developed models subject to regulatory approval. An interesting problem for the regulator is to compare the resulting figures against the required capital under worst case scenarios. The existing literature on the latter problem, which is based on the marginal problem, assumes that no a-priori information is known about the dependencies of contributing risks. These problems are linear optimisation problems over a constrained set of probability measures, discretisation of which leads to large scale LPs. But this approach is very conservative and cannot be implemented robustly in practice, due to the scarcity of historical data. In our approach, we take a less conservative strategy by incorporating dependence information contained in the data in a form that still leads to LPs, an important feature of such problems due to their high dimensionality. Conceptually, our model is the discretisation of an infinite dimensional linear optimisation problem over a set of probability measures. For some specific cases we can prove strong duality, opening up the approach of discretising the dual instead of the primal. This approach is preferable, as it yields better numerical results. In this work we also apply our model to model-free path-dependent option pricing. Use of delayed column generation techniques allows us to solve problems several orders of magnitude larger than via the standard simplex algorithm. For high-dimensional LPs we also implement Nesterov's smoothing technique to solve the problems.

Contents

Chapter 1

Introduction

In recent years considerable interest has developed in the academic and finance communities in model-free finance, both in risk management and in option pricing.

Under the Basel II and later the Basel III regulatory frameworks the financial institutions are allowed to use their own internal methods to calculate the regulatory capital. However, the financial crisis of 2008 raised doubts concerning the modelling aspects of the financial risk. The well known models, though well calibrated to the empirical and the market data, proved to be unreliable. This led to regulators seeking more robust risk management techniques so that banks would hold enough capital to be able to sustain extreme situations, in order to avoid financial catastrophes. The regulator's aim is therefore to find the worst case distribution function for the aggregate loss consistent with the collected data. Banks, on the other hand, want to use a model according to which the calculated capital charge is as small as possible, so as to maximise their profitability.

Under the above-mentioned Basel II and Basel III regulatory frameworks, the capital charge is defined as the Value-at-Risk (VaR) at some probability level (which varies for different types of risks). Though the VaR has been both criticised (see, e.g., [47]) and favoured over CVAR (Expected Shortfall) risk measure (see, e.g., [3]), it still remains a key metric for the calculation of regulatory capital. Let L be a random variable representing the aggregate loss. Then VaR calculated at a probability level α is defined as

$$\mathrm{VaR}_\alpha(L) = F_L^{-1}(\alpha) = \inf\{x \in \mathbb{R} : F_L(x) > \alpha\},$$

where $F_L(x)$ is the distribution function of L.

In Chapter 3 we introduce optimisation approaches that are consistent with the regulators' pessimistic point of view. Assuming the availability of only partial information on the distribution functions of different risk cells, these approaches take the pessimal compound distribution, i.e. maximise the probability that a given threshold is exceeded, given this partial information on the loss distribution. The marginal and moment problems are typical examples of such a framework. In the case of the marginal problem, for example, it is assumed that only marginal distributions of a joint cumulative distribution function are known. This may not be the obvious approach when data about joint events is available, but in other cases it can be appropriate. Although in a more generalised framework P. Embrechts and G. Puccetti, [23] consider a problem in which more information is given in the form of inequalities on the copula function of the joint distribution function, it is not clear how to find the required lower and upper bounds on the copula function.

One obstacle to the wide spread acceptance of the marginal approach to computing the capital charge amongst banks is that the pessimal distributions look unnatural. The industry therefore criticises this approach as being too pessimistic and the capital charge as being too large.

In this work we aim to find models that are acceptable to both sides, to banks and to the financial regulators. Our approach is a generalisation of both marginal and moment problems. In our model we do not assume exact knowledge of the marginal distributions or of the moments, since their calibration itself contains model uncertainty. We formulate the problem of finding the pessimal distribution in a form that allows us to use the available data about the joint distributions, as well as to take into consideration model parameter estimation errors in calibrating the marginal distributions.

In [31], Hauser, Shahverdyan and Embrechts formulate a linear optimisation problem over an infinite dimensional measure space with an infinite number of constraints. The dual problem is then introduced and weak duality follows trivially. In several important special cases the dual is easier to solve, and allows one to derive upper bounds on the worst-case risk exposure that is consistent with the constraints imposed on the distribution.

The basic idea of the general duality result described in the above mentioned work was developed by Hauser, R. and Embrechts, P.. My main contribution to their original work is presented in Section 4, [31], where the model with bounds on integrals is presented for robust risk aggregation. The delayed column generation algorithm for solving the large scale optimisation problems obtained by the model has been developed by me and Hauser, R. In this work, Section 8, we also present application of the above mentioned method to the model-free path-dependent option pricing problem.

We also present an application of Nesterov's algorithm to solve the linear optimisation problem, Section 6.5.
In Chaper 4 we introduce the results of the general duality, [31] and also describe how model-free path-dependent option pricing can be formulated in this framework. This allows us to generalise the model-free option pricing problem again by considering more constraints in the optimisation problem.
In Chapter 5 we present a special case of the general duality problem, in which we consider the optimisation problem not over the set of all signed measures with finite variations, but over the set of measures which have densities belonging to $L^p(\mathbb{R}^n)$ for some $p > 1$, where $L^p(\mathbb{R}^n)$ is the space of functions over $\mathbb{R}^n$ for which the p-th power of the absolute value is Lebesgue integrable on $\mathbb{R}^n$. We prove that in this particular case strong duality holds, hence solving the dual problem will not only give an upper bound but also will provide an exact solution to the original problem. We also generalise a strong duality result of semi-infinite optimisation problems. Since our model described in Section 4.4 is a particular case of a semi-infinite programming problem, we need strong duality. Shapiro [66] proves strong duality for continuous functions, whereas we generalise this to piecewise continuous functions.
A particular application of the general duality framework of [31] is the model with constraints by integrals of piecewise concave functions described in Section 4.4. We also prove strong duality for this model by generalising the strong duality result of semi-infinite optimisation problems, presented in [66].
Although the optimisation problem discussed in Section 4.4 is an infinite dimensional optimisation problem, in Chapter 6 we describe how the dual of these problems can be relaxed to finite dimensional linear programming problems. The resulting LPs are very large scale and impractical to solve numerically for high dimensions with standard methods such as the simplex or interior point methods. In Sections 6.1.1 and 6.5 we introduce techniques to solve these LPs. Of particular interest is the application of the delayed column generation technique for the simplex method to our model (described in Section 6.1.1), which allows us to increase the dimension of the problem considerably.
Chapter 7 describes applications of the model with constraints by integrals of piecewise constant functions in risk management. We explain in detail how to construct the constraints in our model in order to make good use of available historical data. In Chapter 8 we discuss application of our model in model-free path-dependent option pricing. In the standard approach to the problem of model-free derivative pricing, one assumes to have marginal distributions for different time horizons, which are provided

by the market values of vanilla options, such as European call options with different strikes. In practice, we do not have European call option market prices for every strike, hence one has to make an interpolation to obtain the whole curve. This itself introduces some numerical errors into the problem. In this work we do not assume the availability of marginal distribution functions, but use only market-provided option prices. By doing so, we obtain an optimisation problem which has an infinite number of constraints, hence we need to make a finite discretisation. Discretised problems become particular cases of our model with piecewise linear test functions discussed in Section 4.4. Moreover, we see that for some derivatives, e.g. barrier options, we can choose a discretisation which produces an exact solution to the original infinite dimensional problem, if we assume that the optimal solutions of the discretised problems converge to the optimal value of the infinite dimensional problem.

In Chapter 9 we present three algorithms for solving the optimisation problem described in Section 4.4. This discussion is largely theoretical and is offered as a motivation for further research.

Chapter 2

Mathematical Background

2.1 Copula Functions

A copula is used as a general way of formulating a multivariate distribution in a way that represents various general types of dependencies. The approach to formulating a multivariate distribution using a copula is based on the idea that a simple transformation can be made of each marginal variable in such a way that each transformed marginal variable has a uniform distribution. Once this is done, the dependence structure can be expressed as a multivariate distribution that has uniform marginals, a so called copula. The actual multivariate distribution is obtained as the pullback of the copula. Copula functions facilitate a bottom-up approach to multivariate model building.

Definition 2.1.1 (Copula) *An n-dimensional copula is a distribution function on* $[0, 1]^n$ *with standard uniform marginal distributions.*

The following three properties must thus hold.

1. C is non-decreasing in each argument.

2. $C(1, ..., 1, u_i, 1, ..., 1) = u_i$ for all $u_i \in [0, 1], i = 1, ..., n$.

3. C is n-increasing, i.e. for all $a = (a_1, ...a_n), b = (b_1, ..., b_n) \in [0, 1]^n$ with $a_i \leq b_i, i = 1, ..., n$, we have

$$\sum_{j_1=1}^{2} ... \sum_{j_n=1}^{2} (-1)^{j_1+...+j_n} C(u_{1j_1}, ..., u_{nj_n}) \geq 0, \tag{2.1}$$

   where $u_{i1} = a_i, u_{i2} = b_i$ for all $i = 1, ..., n$.

The third property in the above definition is the so-called rectangular inequality, which ensures that if the random vector $(U_1, \dots, U_d)$ has distribution function C, then $P(a_1 \le U_1 \le b_1, \dots, a_d \le U_d \le b_d)$ is non-negative. It is a special case of the inclusion-exclusion formula, as is easy to see in the two-dimensional case.
The following theorem states the two-way connection between copulas and multivariate distribution functions: each multivariate distribution function defines a copula, and given a copula function and the marginal distributions, one can reconstruct the multivariate distribution function.

Theorem 2.1.2 (Sklar 1959, [50] p. 18) *Let F be a joint distribution function with marginal distributions $F_1, ..., F_n$. Then there exists a copula $C : [0,1]^n \to [0,1]$ such that for all $x_1, ..., x_n$ in $\bar{\mathbb{R}} = [-\infty, +\infty]$*

$$F(x_1, ..., x_n) = C(F_1(x_1), ..., F_n(x_n)). \tag{2.2}$$

If the marginal distributions are continuous, then C is unique. Conversely, if C is a copula and $F_1, ..., F_n$ are univariate distribution functions, then the function F defined in (2.2) is a joint distribution function with marginal distributions $F_1, ..., F_n$.

In the following we define the copula for a distribution with continuous marginal distributions.

Definition 2.1.3 (Copula of F, Definition 5.4, [47]) *If the random vector X has joint distribution function F with continuous marginal distributions $F_1, ..., F_n$, then the copula of F (or X) is the distribution function C of $(F_1(X_1), ..., F_n(X_n))$.*

Here we present a theorem that provides bounds for every copula. These bounds are called Fréchet bounds.

Theorem 2.1.4 (Fréchet bounds, Theorem 5.7, [47]) *For every copula $C(u_1, ..., u_n)$ we have*

$$\max\left\{\sum_{i=1}^{n} u_i + 1 - n, 0\right\} \le C(u) \le \min\{u_1, ..., u_n\} \tag{2.3}$$

While the right hand side of (2.3) is always a copula (representing the copula of comonotonic random variables, i.e. perfectly positively dependent), the left hand side of the equation (we denote this function by W) is a copula only for $n \le 2$, which represents countermonotonic random variables.

2.2 Extreme Value Theory

In this section we introduce Extreme Value Theory. The section is based on two main types of modelling extreme values: the block maxima model and the threshold exceedances model, which will be discussed in detail below.

2.2.1 Maxima

Consider i.i.d. random variables $(X_i)_{i\in\mathbb{N}}$ representing financial losses. Denote their distribution function by F. By the Central Limit Theory, the normalised sums of i.i.d. random variables with finite variance converge in distribution to the standard normal distribution. A similar situation occurs in the case for normalised maxima $M_n = \max(X_1, ..., X_n)$, for which the only limiting distribution can be in the GEV family.

Definition 2.2.1 (GEV distribution, Definition 7.1, [47]) *The distribution function of the (standard) generalised extreme value (GEV) distribution is given by*

$$H_\xi = \begin{cases} \exp(-(1+\xi x)^{-1/\xi}), & \xi \neq 0, \\ \exp(-e^{-x}), & \xi = 0, \end{cases}$$

where $1 + \xi x > 0$. A three-parameter family is obtained by defining $H_{\xi,\mu,\sigma} := H_\xi((x - \mu)/\sigma)$ for a location parameter $\mu \in \mathbb{R}$ and a scale parameter $\sigma > 0$.

The parameter ξ is known as the *shape parameter* of the GEV distribution. When $\xi > 0$, the GEV is known as the Fréchet distribution, when $\xi = 0$ it is known as the Gumbel distribution, and when $\xi < 0$ as the Weibull distribution. Note that the parametrisation is continuous in ξ and that for the Weibull distribution one can observe a finite endpoint, which is not true for the other two cases.
Suppose that M_n converges to some probability distribution function under some normalisation, i.e there exist some c_n, d_n such that

$$\lim_{n\to\infty} P((M_n - c_n)/d_n \leq x) = \lim_{n\to\infty} F^n(x) = H(x) \tag{2.4}$$

for some non-degenerate distribution function $H(x)$. We have the following theorem.

Definition 2.2.2 (Maximum domain of attraction, Definition 7.2, [47]) *If* (2.4) *holds for some nondegenerate distribution function H, then F is said to be in the maximum domain of attraction of H. We use notation $F \in MDA(H)$ to describe this situation.*

Theorem 2.2.3 (FisherTippett, Gnedenko, Theorem 7.3, [47]) *If $F \in MDA(H)$ for some non-degenerate distribution function H then H must be a GEV distribution.*

If this convergence takes place, the parameter ξ is defined uniquely, but the scaling and location parameters can vary depending on the choice of normalising parameters, [47], and also these parameters can be chosen such that the limiting distribution is in standard form.

2.2.2 Maximum Domains of Attraction

In practice most of the probability distribution functions belong to maximum domain of attraction for some value of ξ. Here we consider the Fréchet and the Gumbell distributions since distributions of the Weinbull typ eare rarely used in financial modelling.

- The Fréchet case.

  Definition 2.2.4 (Slowly varying and regularly varying functions) *(i) A positive, Lebesgue-measurable function L on $(0, \infty)$ is slowly varying at ∞ if*

  $$\lim_{x \to \infty} \frac{L(tx)}{L(x)} = 1, \quad \forall t > 0, \tag{2.5}$$

  (ii) A positive, Lebesgue-measurable function h on $(0, \infty)$ is regularly varying at ∞ with index $\rho \in \mathbb{R}$ if

  $$\lim_{x \to \infty} \frac{h(tx)}{h(x)} = t^{\rho}, \quad \forall t > 0. \tag{2.6}$$

  Theorem 2.2.5 (Fréchet MDA, Gnedenko, Theorem 7.8, [47]) *For $\xi > 0$,*

  $$F \in MDA(H_\xi) \Leftrightarrow \bar{F}(x) = x^{-1/\xi} L(x) \tag{2.7}$$

  for some function L slowly varying at ∞.

  Fréchet type distributions are used in financial applications because they introduce heavy-tailed distributions with infinite higher moments. Examples include Pareto, inverse gamma and Student-t distributions.

- The Gumbel Case.
  Normal and lognormal distributions are typical cases of these distributions. All the distribution functions that belong to this class have finite moments of any positive order.

2.2.3 Threshold Exceedances

One of the practical weaknesses of the block maxima approach is that most of the available data are not used. In this section we introduce the threshold exceedance method, in which one uses the data that exceeds some particular level.

2.2.4 Generalised Pareto Distribution

Definition 2.2.6 (GPD, Definition 7.16, [47]) *The distribution function of the GPD is given by*

$$G_{\xi,\beta} = \begin{cases} \exp(-(1+\xi x/\beta)^{-1/\xi}), & \xi \neq 0, \\ \exp(-e^{-x/\beta}), & \xi = 0, \end{cases}$$

where $\beta > 0$, and $x \geq 0$ when $\xi \geq 0$ and $0 \leq x \leq -\beta/\xi$ when $\xi < 0$. The parameters ξ and β are the shape and scale parameters respectively.

Definition 2.2.7 (Excess distribution over threshold u, Definition 7.17, [47]) *Let X be a random variable with distribution function F. The excess distribution over the threshold u has distribution function*

$$F_u(x) = P(X - u \leq x | X > u) = \frac{F(x+u) - F(u)}{1 - F(u)} \tag{2.8}$$

for $0 \leq x < x_F - u$, where $x_F \leq \infty$ is the right endpoint of F.

The following theorem allows one to model the heavy tails of a probability distribution function by extremal events.

Theorem 2.2.8 (Pickands Balkema de Haan, Theorem 7.20, [47]) *One can find a (positive-measurable function) $\beta(u) = \beta + \xi u$ such that*

$$\lim_{u \to x_F} \sup_{0 \leq x < x_F - u} \left| F_u(x) - G_{\xi,\beta(u)}(x) \right| = 0, \tag{2.9}$$

if and only if $F \in MDA(H_\xi)$, where $\xi \in \mathbb{R}$.

This theorem implies convergence in distribution of F_u to $G_{\xi,\beta(u)}$ when the threshold u goes to infinity. One can thus approximate the distribution function by choosing a suitable threshold u, followed by two steps:

- Use an empirical distribution to model the losses below the threshold.

- use the GDP to model the losses above the threshold.

The approximation will thus be as follows,

$$EVT[F,u](x) = \begin{cases} F(x) & x < u, \\ G_{\xi,\beta(u)}(x-u)(1-F(u)) + F(u) & x \geq u \end{cases} \tag{2.10}$$

It is important to note that the convergence in distribution does not lead to convergence in mean or convergence in quantiles: Makarov [46] gives a wide class of distributions when EVT does not lead to uniform quantile convergence and further, when it does not lead to mean convergence. We will describe these distributions below.

Definition 2.2.9 *The EVT approximation leads to uniform relative quantile (URQ) convergence if, for any $\epsilon > 0$, there exists a threshold u_0 such that for any $u \geq u_0$ and any quantile q,*

$$(1-\epsilon)Q(EVT[F,u],q) \leq Q(F,q) \leq (1+\epsilon)Q(EVT[F,u],q),$$

where by $Q(F,q)$ we denote the quantile of F at level q.

Since for any cumulative distribution function G we have

$$mean(G) = \int_0^1 Quantile(G,q)\,\mathrm{d}\,q,$$

it follows that URQ convergence implies convergence in mean.

Theorem 2.2.10 (Proposition 3.4, [46]) *For a regularly varying distribution function F, the EVT approximation leads to URQ convergence only if*

$$0 < \lim_{x\to\infty} \bar{F}(x)x^{1/\epsilon} < \infty, \tag{2.11}$$

where $\bar{F}(x) = 1 - F(x)$.

Now we introduce two examples of EVT approximation, the first one without URQ convergence and the second one without mean convergence from the referred paper. Consider a distribution function

$$F_{a,b}(x) = 1 - \frac{\log^a(x)}{x^b}, \tag{2.12}$$

where $b > 0$ and $a \neq 0$.

Lemma 2.2.11 (Lemma 5.1, [46]) *The following holds:*

1. *$F_{a,b}$ is regularly varying with parameter $1/b$.*

2. *EVT approximation does not lead to URQ convergence for $F_{a,b}$.*

For the second example Makarov constructs the following cumulative distribution function,

$$F_{1,-2}(x) = 1 - \frac{1}{x \log^2(x)}, x \geq e. \tag{2.13}$$

This distribution has finite mean, but EVT approximates it with a distribution with an infinite mean and does not approximate its mean, shortfalls or high quantiles. For more details and proofs of the results see [46].

2.3 The g-and-h, GB2 and GCD Distributions

These three four-parameter distributions are used in risk management as alternatives to EVT. Here we briefly introduce each of them.
A random variable X is said to have g-and-h distribution if

$$X = a + b\frac{\exp(gY) - 1}{g} \exp(hY^2/2), \tag{2.14}$$

where Y is a standard normal random variable and (a, b, g, h) are parameters. In [20] it is shown that for this distribution, convergence of the excess distribution to the GPD is very slow.
The density function of GB2 (the generalised Beta distribution of the second kind) is defined as

$$f(x) = \frac{|a|x^{ap-1}}{b^{ap}B(p,q)[1 + (x/b)^a]^{p+q}}, \quad (x > 0), \tag{2.15}$$

where $B(p, q)$ is the Beta function and (a, b, p, q) are the parameters.
Finally, the density function of Generalized Champernowne distribution (GCD) is defined as

$$f(x) = \frac{\alpha(x + c)^{\alpha-1}((M + c)^\alpha - c^\alpha)}{((x + c)^\alpha + (M + C)^\alpha - 2c^\alpha)^2}, \quad (x \geq 0), \tag{2.16}$$

Applications of the first two distributions in operational risk management are studied in [21] and the GCD is studied in [29].

Chapter 3

The Regulator's Approach

In Introduction we discussed the differences in how financial institutions and their regulators differ in their approaches in calculating the capital charge required for risk management purposes. The main problem for the banks is to find a model which represents the existing data plausibly. Once they have such a model, they compute the Value-at-Risk of the compound distribution at the specified level.
From the regulators' point of view, however, finding one model is not sufficient. Due to the lack of sufficient data on extreme events from the industry, particular model will be able to cover all of the possible scenarios. Real life experience shows that unprecedented extremes lead to the collapse of large financial institutions, as we witnessed for example with Lehman Brothers in the 2008 financial crisis. The above described approaches to computing the capital charge of a portfolio of risks strongly depend on historical data which may inadequately reflect future extreme events. Notably, the dependence structure may be totally wrong. Regulators thus wish to convince themselves that the capital charge is set at a level that provides protection under a more robust set of assumptions about future losses.
In the most general framework regulators need to find the solution to the following optimisation problem :

$$\begin{aligned} &\sup_{\mathcal{F} \in \mathscr{M}_{\mathfrak{F}}} \int_{\mathbb{R}^n} h(x) d\mathcal{F}(x) \\ &\text{s.t. } \mathcal{F} \text{ is consistent with the observed data} \end{aligned} \tag{3.1}$$

for some measurable function $h : \mathbb{R}^n \to \mathbb{R}$. $\mathscr{M}_{\mathfrak{F}}$ is the set of all probability measures, and where "s.t. $\mathcal{F}$ is consistent with the observed data" is defined in some meaningful way. Thus they pessimise the risk over a set of possible distributions.
Problem (3.1) is an infinite-dimensional optimisation problem, and in most practical

cases the number of constraints is also infinite. In the following sections we will introduce some results for problem (3.1) for some particular cases - the marginal problem, where it is assumed that we have knowledge of the marginal distributions of $\mathcal{F}$; the moment problem, where a few of the moments of $\mathcal{F}$ are known; and the marginal problem with bounds on the copula, where some information is known about the copula function of joint distributions in the form of lower and upper bounds, in addition to the marginal distributions.

The above-mentioned marginal and moment problems are very conservative in risk management, since in practice we have much more information for joint events than just marginal distributions or moments. Although we discuss also a generalisation of the marginal problem in which we assume some dependency structure, it is not clear how to use information about joint events by imposing lower and upper bounds on the copula function.

In later sections we introduce more generalised optimisation problems that allow us to resolve this drawback.

There are no known analytical solutions to the above mentioned problems in general form. If we try to discretize the joint and marginal distributions, we end up with linear programming problems, where the number of constraints will be so large that no efficient algorithms are known. This is primarily due to the high dimensionality of the problem. For example, for operational risk the dimensionality is 56 (though it can be grouped into 8 business lines).

Another approach to problem (3.1) is duality theory. From standard linear programming techniques we know that any LP problem

$$\begin{aligned} \max_{x} \quad & c^T x \\ \text{s.t.} \quad & Ax \leq b, \\ & x \geq 0 \end{aligned} \tag{3.2}$$

is associated with a dual problem

$$\begin{aligned} \min_{y} \quad & b^T y \\ \text{s.t.} \quad & A^T y \geq c, \\ & y \geq 0, \end{aligned} \tag{3.3}$$

where $c \in \mathbb{R}^n, b \in \mathbb{R}^m$ and A is a matrix of size $m \times n$.

The weak duality theorem states that for any feasible x for (3.2) and for any feasible y for (3.3),

$$c^T x \leq b^T y, \tag{3.4}$$

while the strong duality theorem states that if any of the problems has a finite optimal solution then so does the other, and optimal values of both problems coincide, i.e.,

$$c^T x^* = b^T y^*, \tag{3.5}$$

where x^* and y^* are optimal solutions to the problems (3.2) and (3.3) respectively. Strong duality is a very useful tool when it is difficult to find the optimal solution to the primal problem, since finding any (3.3)-feasible solution will provide us with an upper bound for (3.2) and, by improving these dual feasible solutions, we can get close to the optimal solution with any accuracy. It is important to note in this context that for the purpose of calculating the capital charge, only the optimal objective value $c^T x^*$ is required, and the optimising values x^* of the decision variable play a secondary role.

In this chapter we introduce the marginal and moment problems. Both problems are so far impossible to solve analytically or numerically, and in both cases it turns out to be easier to solve or to find a good feasible solution for their dual formulations. For the marginal problem, strong duality was proved by Ramachadran and Rüschendorf in [60]. Embrechts and Puccetti [24] constructed explicit dual solutions to get an upper bound on the primal problem.

For the moment problem, strong duality was proved by Shapiro [66]. The problem of finding an optimal dual solution can be solved via semidefinite programming problem, see [5]. The latter work does not solve the moment problem exactly, but produces increasingly stronger, asymptotically exact upper bounds.

3.1 The Marginal Problem

In this section we consider the following problem, discussed by Ramachadran and Rüschendorf [60]:

$$\sup_{\mathcal{F} \in \mathcal{M}_{F_1,\dots,F_n}} \int_{\mathbb{R}^n} h(x) d\mathcal{F}(x), \tag{3.6}$$

where $h : \mathbb{R}^n \to \mathbb{R}$ is some measurable function and $M_{F_1,\dots,F_n}$ is the set of probability measures on $\mathbb{R}^n$ whose marginals have cumulative distribution functions $F_1, ..., F_n$. This problem is of particular interest in the risk management framework because setting $h(x) = 1_{\{\psi(x) \geq s\}}$, the problem becomes that of finding the supremum of the probability of the event that a function ψ of a random vector X exceeds some threshold s, given the marginal distribution functions of X. When $\psi(x) = \sum_{i=1}^n x_i$, problem

(3.6) corresponds to a subproblem in maximising the VaR_α via Algorithm 7.3.3 discussed in Chapter 7. Similarly, by choosing $\psi(x) = \sum_{i=1}^n x_i 1_{\{\sum_{i=1}^n x_i \geq s\}}$, problem (3.6) allows us to maximise the CVaR at a given quantile α.

Note that in the case of Value-at-Risk risk measure, the primal problem (3.6) is equivalent to

$$M_\psi(s) := \sup\left\{\mathbb{P}\left[\psi(X_1, ..., X_n) \geq s\right] : X_i \sim F_i, 1 \leq i \leq n\right\}, \quad s \in \mathbb{R}. \tag{3.7}$$

One way to solve problem (3.6) numerically is to discretize the marginal distributions and solve a linear programming problem, but the number of decision variables quickly becomes too large to be handled by known algorithms.

In [60] a dual to problem (3.6) is proposed as follows:

$$\begin{aligned} \inf_{h_1,\ldots,h_n \in L^1(\mathbb{R})} \quad & \sum_{i=1}^n \int_{\mathbb{R}} h_i(z) dF_i(z) \\ \text{s.t} \quad & \sum_{i=1}^n h_i(x_i) \geq h(x), \quad (x \in \mathbb{R}^n). \end{aligned} \tag{3.8}$$

Denote V_1 and V_2 the optimal solutions to problems (3.6) and (3.8) respectively. The strong duality result proved in [60] provides the equality $V_1 = V_2$. Hence, solving the dual problem will give us a solution to the primal problem.

There has been considerable development in solutions of the marginal problem, see [25, 59, 58, 68]. In [59] authors derive sharp bounds for the case of homogenous marginals, i.e. when $F_1 = \ldots, F_n$. For more general case, i.e. not-identically distributed risks, a powerful numerical algorithm called rearrangement algorithm has been developed in [58] and later modified for more efficiency in [59]. With the rearrangment algorithm, the marginal problem can be numerically solved for up to several hundred dimensions.

Duality provides us with a useful method for finding upper bounds on V_1: by the weak duality we have $V_1 \leq V_2$, so that $\sum_{i=1}^n \int_{\mathbb{R}} h_i(z) dF_i(z)$ provides us an upper bound on V_1 for every (3.8)-feasible $h_1, ..., h_n$. In some cases it is easier to find a good (3.8)-feasible solution than good primal solutions, see [24]. In a risk management framework, upper bounds on the maximum loss that a financial institution can suffer at a certain confidence level may be all that is required.

3.2 The Marginal Problem with Bounds on Copulas

In problem (3.6) we considered the case where only marginal distributions are known. In this section, which is based on [23], we discuss the problem where the dependence structure is known to some extent. Dependence is modelled by using copula functions and restrictions are expressed by imposing lower and upper bounds on copulas.
As before, suppose we have random variables $X_1, ..., X_n$ on some probability space $(\Omega, \mathcal{F}, \mathbb{P})$, and that their distribution functions $F_1, ..., F_n$ are known. Consider problem (3.7). Let C be the copula of this random vector, denote by μ_C the corresponding probability measure on $\mathbb{R}^n$ and define

$$\sigma_{C,\psi}(F_1, ..., F_n)(s) := \mu_C[\psi(X) < s] = \int_{\psi<s} dC(F_1(x_1), ..., F_n(X_n)), \tag{3.9}$$

$$\tau_{C,\psi}(F_1, ..., F_n)(s) := \sup_{x_1,...,x_{n-1}\in\mathbb{R}} C(F_1(x_1), ..., F_{n-1}(x_{n-1}), F_n^-(\hat{\psi}_{x_{-n}}(s))), \tag{3.10}$$

where $\hat{\psi}_{x_{-n}}(s) := \sup\{x_n \in \mathbb{R} : \psi(x_{-n}, x_n) < s\}$ for fixed $x_{-n} \in \mathbb{R}^{n-1}$ and for $x = (x_1, ..., x_n) \in \mathbb{R}^n$ we denote $x_{-i} := (x_1, ..., x_{i-1}, x_{i+1}, ..., x_n)$. For a random variable Z we denote it's left-continuous version of probability distribution function by $F^-(z)$, i.e. $F^-(z) := \mathbb{P}[Z < z]$.
Problem (3.7) is equivalent to

$$m_\psi(s) := 1 - M_\psi(s) = \inf\{\sigma_{C,\psi}(F_1, ..., F_n)(s) : C \in \mathcal{C}_n\}, \tag{3.11}$$

where $\mathcal{C}_n$ denotes the set of all n-dimensional copulas. Moreover, since copula functions represent the dependency of random variables, having some lower bound on it can be regarded as having some partial information about the dependency of random variables. We say $C_1 \geq C_2$ if $C_1(u) \geq C_2(u)$ for all $u \in [0,1]^n$. Let us denote Fréchet lower and upper bounds by W and M respectively.

Theorem 3.2.1 (Theorem 5, [22]) *Let $X = (X_1, ..., X_n)$ be a random vector in $\mathbb{R}^n$ $(n > 1)$ having marginal distribution functions $F_1, ..., F_n$ and copula C. Assume that there exists a copula C_L such that $C \geq C_L$. If $\psi : \mathbb{R}^n \to \mathbb{R}$ is non-decreasing in each coordinate, then for every real s we have*

$$\sigma_{C,\psi}(F_1, ..., F_n)(s) \geq \tau_{C_L,\psi}(F_1, ..., F_n)(s). \tag{3.12}$$

So we also have

$$VaR_\alpha(\psi(X_1, ..., X_n)) \leq \tau_{C_L,\psi}(F_1, ..., F_n)^{-1}(\alpha), \tag{3.13}$$

for every $\alpha \in [0,1]$.

Remark 3.2.2 *Though the function* W *(defined in* (2.3)*) is not a copula for* $n > 2$*, the bound* (3.12) *is valid even when* C_L *is replaced by* W*.*

Williamson and Down [69] state that when $n = 2$ and ψ is continuous, there will always be a copula C_α attaining the bound (3.12), i.e., this bound can not be tightened. In [22] the corresponding copula C_α is constructed as follows,

$$C_\alpha(u) := \begin{cases} \max\{\alpha, C_L(u)\} & \text{if } u = (u_1, u_2) \in [\alpha, 1]^2, \\ \min\{u_1, u_2\} & \text{otherwise,} \end{cases} \tag{3.14}$$

where $\alpha = \tau_{C_L,\psi}(F_1, ..., F_n)(s)$, i.e.,

$$\sigma_{C_\alpha,\psi}(F_1, ..., F_n)(s) = \alpha. \tag{3.15}$$

Hence, for $n > 2$ the function $C_\alpha(u)$ is not a copula.

3.3 The Moment Problem

In this section we analyse the problem of finding optimal bounds of a probability of a random vector having its moments up to some order k.
As in the case of the marginal problem, when we try to solve the moment problem numerically, the number of unknowns is very large and in practice it is impossible to solve the problem. For this problem Bertsimas and Popescu [5] also use an approach based on strong duality results, and they introduce a technique to solve the dual problem numerically (in semidefinite programming context) with an efficient algorithm for some particular cases.
Let $\kappa = (k_1, ..., k_n) \in \mathbb{Z}_+^n$ where $\mathbb{Z}_+$ is the set of non negative integers. Denote $\sigma_\kappa = \sigma_{k_1,...,k_n}$ and $J_k = \{\kappa = (k_1, ..., k_n) | k_1 + ... + k_n \leq k, k_j \in \mathbb{Z}_+, j = 1, ..., n\}$.

Definition 3.3.1 *A sequence* $\bar{\sigma} = (\sigma_\kappa, \kappa \in J_k)$ *is a feasible* (n, k, Ω)*-moment vector if there exists a random vector* $X = (X_1, ..., X_n)^T$ *defined on* $\Omega \subseteq \mathbb{R}^n$ *endowed with its Borel sigma algebra of events, whose moments are given by* $\bar{\sigma}$*, i.e.* $\sigma_\kappa = E[X_1^{k_1} ... X_n^{k_n}]$ *for all* $\kappa \in J_k$*. We say that any such random variable* X *has a* $\bar{\sigma}$*-feasible distribution and denote this as* $X \sim \bar{\sigma}$*.*

Denote $\mathcal{M} = \mathcal{M}(n, k, \Omega)$ the set of feasible (n, k, Ω)-moment vectors.

Definition 3.3.2 *We say that* γ *is a tight upper bound on* $P(X \in S)$ *if* $\gamma = \sup_{X \sim \bar{\sigma}} P(X \in S)$*.*

The (n, k, Ω)-bound problem is to find $\sup_{X \sim \bar{\sigma}} P(X \in S)$ for a given $\bar{\sigma} = (\sigma_\kappa, \kappa \in J_k)$ and a measurable set S.

Results in this section are from the paper by Bertsimas and Popescu [5] in which they propose a dual problem to the optimization problem that we want to solve and also investigate in detail the univariate case ($n = 1$) for $\Omega = \mathbb{R}, \mathbb{R}_+$. In these cases the problem can be solved by semidefinite optimization methods. For more general (n, k, Ω)-moment problems, for semialgebraic Ω and S, they propose a sequence of increasingly stronger, asymptotically exact upper bounds by solving semidefinite optimization problems.They also show that the $(n, k, \mathbb{R}^n_+)$ and $(n, k, \mathbb{R}^n)$-bound problems are NP-hard for $k \geq 2$ and $k \geq 4$ respectively.

Denote $\bar{z}^\kappa = z_1^{k_1} ... z_n^{k_n}$, where $z = (z_1 ... z_n)^T$. The (n, k, Ω)-upper bound problem can be formulated as the following optimization problem over measures μ:

$$\begin{aligned} Z_P = \max_{\mu} & \int_S 1 d\mu \\ \text{s.t.} & \int_\Omega \bar{z}^\kappa d\mu = \sigma_\kappa \quad \forall \kappa \in J_k. \end{aligned} \tag{3.16}$$

If problem (3.16) is feasible, then $\bar{\sigma}$ is a feasible moment sequence and any feasible distribution μ is a $\bar{\sigma}$-feasible distribution.

The dual problem of (3.16) is

$$\begin{aligned} Z_D = \min_{y \in \mathbb{R}^{J_k}} \quad & \sum_{\kappa \in J_k} y_\kappa \sigma_\kappa \\ \text{s.t. } g(\bar{z}) = \quad & \sum_{\kappa \in J_k} y_\kappa \bar{z}^\kappa \geq 1 \quad \forall \bar{z} \in S, \\ g(\bar{z}) = \quad & \sum_{\kappa \in J_k} y_\kappa \bar{z}^\kappa \geq 0 \quad \forall \bar{z} \in \Omega, \end{aligned} \tag{3.17}$$

Note that in the primal problem we optimise in infinite dimensional space, but the number of constraints is finite. In contrast, in the dual problem we optimise over a finite-dimensional space $\mathbb{R}^{J_k}$ but the number of constraints is infinite. Later we will see that the constraint of the dual problem for $n = 1$ can be formulated as a linear matrix inequality, allowing us to implement efficient methods known from semidefinite programming to solve the problem.

Theorem 3.3.3 (Weak duality, Theorem 2.1, [5])

$$Z_P \leq Z_D. \tag{3.18}$$

Hence by solving the dual problem we obtain an upper bound for primal problem.

Theorem 3.3.4 (Strong duality, Theorem 2.2, [5]) *If the moment vector $\bar{\sigma}$ is an interior point of the set $\mathcal{M}$ of feasible moment vectors, then $Z_P = Z_D$.*

In [39] Isii proves that in the univariate case, if $\bar{\sigma}$ is a boundary point of $\mathcal{M}$, then exactly one $\bar{\sigma}$-feasible distribution exists. Kemperman [41] proves that for almost every $\bar{\sigma}$ (with respect to the Lebesgue measure) such that the dual is finite, the dual solution is unique.

In the following subsection we introduce a technique to formulate problem (3.17) in the semidefinite optimization framework when $n = 1$.

3.3.1 The $(1, k, \Omega)$-Bound Problem

Consider a (n, k, Ω)-bound problem for $n = 1$, i.e. given a random variable X on $\Omega \subseteq \mathbb{R}$ and its first k moments $M_1, ..., M_k$, we want to find tight upper and lower bounds on $P(X \in S)$. In this section we will introduce how Bertsimas and Popescu [5] formulate this problem as a semidefinite programming problem.
The dual problem (3.17) becomes

$$\begin{aligned} \min_{y \in \mathbb{R}^k} \quad & \sum_{r=0}^{k} y_r M_r \\ \text{s.t.} \quad & \sum_{r=0}^{k} y_r x^r \geq 1 \quad \forall x \in S, \\ & \sum_{r=0}^{k} y_r x^r \geq 0 \quad \forall x \in \Omega. \end{aligned} \tag{3.19}$$

Proposition 3.1, [5] shows how to express the polynomial inequalities in (3.19) as semidefinite programming constraints when S and Ω are intervals on the real line.

Theorem 3.3.5 (Theorem 3.2, [5]) *Let X be a random variable on Ω and $M_1, ..., M_k$ its first k moments. Then the following holds true:*

1. *If $\Omega = \mathbb{R}_+$, the tight upper bound on $P(X \geq a)$ is given as the solution of the*

following semidefinite optimization problem

$$
\begin{array}{ll}
\min\limits_{X,Z\succeq 0} & \displaystyle\sum_{r=0}^{k} y_r M_r \\
s.t. & \displaystyle\sum_{i,j:i+j=2l-1} x_{ij} = 0, \quad l = 1, ..., k, \\
& \displaystyle\sum_{r=1}^{k} y_r a^r + (y_0 - 1) = x_{00}, \\
& \displaystyle\sum_{r=l}^{k} y_r \begin{pmatrix} r \\ l \end{pmatrix} a^{r-l} = \sum_{i,j:i+j=2l} x_{ij}, \quad l = 1, ..., k, \\
& \displaystyle\sum_{i,j:i+j=2l-1} z_{ij} = 0, \quad l = 1, ..., l, \\
& \displaystyle\sum_{r=0}^{l} y_r \begin{pmatrix} k-r \\ k-l \end{pmatrix} a^r = \sum_{i,j:i+j=2l} z_{ij}, \quad l = 0, ..., k,
\end{array}
$$

If $\Omega = \mathbb{R}$ *then the last equation in* (3.20) *should be replaced by*

$$
(-1)^l \sum_{r=l}^{k} y_r \begin{pmatrix} r \\ l \end{pmatrix} a^{r-l} = \sum_{i,j:i+j=2l} z_{ij}, \quad l = 0, ..., k,
$$

2. *Similar results are derived for* $\Omega = \mathbb{R}_+, \mathbb{R}$ *and* $P(a \leq X \leq b)$*, see [5] for all results and a proof of the theorem.*

3.3.2 The (n, k, Ω)-Bound Problem

For general (n, k, Ω)-bound problems, Bertsimas and Popescu [5] propose a sequence of increasingly stronger, asymptotically exact upper bounds by solving semidefinite optimization problems when the sets S and Ω are given as intersections of inequalities involving polynomials. Although the method is guaranteed to find the exact solution, the size of the semidefinite programming problems in the sequence is not bounded by a polynomial in n even for fixed k. This is natural because in the same paper the authors prove that the $(n, k, \mathbb{R}_+^n)$ and $(n, k, \mathbb{R}^n)$-bound problems are NP-hard for $k \geq 2$ and $k \geq 4$ respectively.

Chapter 4

A General Duality Theorem for Optimisation Problems over Measure Spaces

In Sections 3.1 and 3.2 we were dealing with infinite dimensional optimisation problems with infinitely many constraints. In this chapter we introduce a general duality theory proposed by Hauser, Shahverdyan and Embrechts, [31], which covers moment and marginal problems from the previous chapter as special cases and also gives a new duality formulation for the marginal problem with bounds on copulas. In Section 4.5 we formulate the problem of model-free path-dependent option pricing in our general duality result and construct the dual. We also consider the model with constraints by integrals of piecewise concave functions, which serves us as a main tool in our numerical applications in risk management and model-free option pricing.

In [31] authors propose the following problem with its dual formulation.

Let $(\Phi, \mathfrak{F})$, $(\Gamma, \mathfrak{G})$ and $(\Sigma, \mathfrak{S})$ be complete measure spaces, and let $A : \Gamma \times \Phi \to \mathbb{R}$, $a : \Gamma \to \mathbb{R}$, $B : \Sigma \times \Phi \to \mathbb{R}$, $b : \Sigma \to \mathbb{R}$, and $c : \Phi \to \mathbb{R}$ be bounded measurable functions on these spaces and the corresponding product spaces. Let $\mathscr{M}_{\mathfrak{F}}$, $\mathscr{M}_{\mathfrak{G}}$ and $\mathscr{M}_{\mathfrak{S}}$ be the set of signed measures with finite variation on $(\Phi, \mathfrak{F})$, $(\Gamma, \mathfrak{G})$ and $(\Sigma, \mathfrak{S})$ respectively. We now consider the following pair of optimisation problems over $\mathscr{M}_{\mathfrak{F}}$ and $\mathscr{M}_{\mathfrak{G}} \times \mathscr{M}_{\mathfrak{S}}$ respectively, which authors show to be duals of each other,

$$
\begin{aligned}
\text{(P)} \quad \sup_{\mathcal{F} \in \mathscr{M}_{\mathfrak{F}}} & \int_{\Phi} c(x) \, \mathrm{d}\, \mathcal{F}(x) \\
\text{s.t.} & \int_{\Phi} A(y, x) \, \mathrm{d}\, \mathcal{F}(x) \le a(y), \quad (y \in \Gamma), \\
& \int_{\Phi} B(z, x) \, \mathrm{d}\, \mathcal{F}(x) = b(z), \quad (z \in \Sigma), \\
& \mathcal{F} \ge 0,
\end{aligned}
$$

and

$$
\begin{aligned}
\text{(D)} \qquad \inf_{(\mathcal{G},\mathcal{S})\in\mathscr{M}_{\mathfrak{G}}\times\mathscr{M}_{\mathfrak{S}}} \quad & \int_\Gamma a(y)\,\mathrm{d}\,\mathcal{G}(y) + \int_\Sigma b(z)\,\mathrm{d}\,\mathcal{S}(z), \\
\text{s.t.} \quad & \int_\Gamma A(y,x)\,\mathrm{d}\,\mathcal{G}(y) + \int_\Sigma B(z,x)\,\mathrm{d}\,\mathcal{S}(z) \geq c(x), \quad (x\in\Phi), \\
& \mathcal{G} \geq 0.
\end{aligned}
$$

Theorem 4.0.6 (Weak Duality) *For every (P)-feasible measure $\mathcal{F}$ and every (D)-feasible pair $(\mathcal{G},\mathcal{S})$ we have*

$$\int_\Phi c(x)\,\mathrm{d}\,\mathcal{F}(x) \leq \int_\Gamma a(y)\,\mathrm{d}\,\mathcal{G}(y) + \int_\Sigma b(z)\,\mathrm{d}\,\mathcal{S}(z).$$

Proof. Using Fubini's Theorem, we have

$$
\begin{aligned}
\int_\Phi c(x)\,\mathrm{d}\,\mathcal{F}(x) &\leq \int_{\Gamma\times\Phi} A(y,x)\,\mathrm{d}(\mathcal{G}\times\mathcal{F})(y,x) + \int_{\Sigma\times\Phi} B(z,x)\,\mathrm{d}(\mathcal{S}\times\mathcal{F})(z,x) \\
&\leq \int_\Gamma a(y)\,\mathrm{d}\,\mathcal{G}(y) + \int_\Sigma b(z)\,\mathrm{d}\,\mathcal{S}(z).
\end{aligned}
$$

□

We are interested in finding conditions on measures that imply strong duality between the primal and dual problems. Such particular cases are discussed in the next chapter, where we assume measures to have densities that belong to $L^p(\mathbb{R}^n)$ for some $p > 1$. First we will describe how to formulate the problems from the previous chapter in this framework and we will see that we get exactly the same duality results as we had in Chapter 3.

4.1 The Marginal Problem

Consider the problem discussed in Section 3.1

$$\sup_{\mathcal{F}\in\mathscr{M}_{F_1,\dots,F_n}} \int_{\mathbb{R}^n} h(x)\,\mathrm{d}\,\mathcal{F}(x), \tag{4.1}$$

where $\mathscr{M}_{F_1,\dots,F_n}$ is the set of probability measures on $\mathbb{R}^n$ whose marginals have the cumulative distribution functions F_i $(i = 1,\dots,n)$.
To formulate problem (4.1) in our framework we set $c(x) = h(x)$, $\Phi = \mathbb{R}^n$, $\Gamma = \emptyset$, $\Sigma = \mathbb{N}_n \times \mathbb{R}$ (where we write $\mathbb{N}_n = \{1,\dots,n\}$), $B(i,z,x) = 1_{\{x:\, x_i\leq z\}}(z,x)$, and

$b_i(z) = F_i(z)$ $(i \in \mathbb{N}_n,\ z \in \mathbb{R})$,

$$\begin{aligned}
&\sup_{\mathcal{F}} \int_{\mathbb{R}^n} h(x) \,\mathrm{d}\,\mathcal{F}(x) \\
&\text{s.t.} \int_{\mathbb{R}} 1_{\{x_i \le z\}}(z, x) \,\mathrm{d}\,\mathcal{F}(x) = F_i(z), \quad (z \in \mathbb{R}, ; i \in \mathbb{N}_n) \\
&\qquad \mathcal{F} \ge 0.
\end{aligned} \tag{4.2}$$

Taking the dual, we find

$$\begin{aligned}
&\inf_{\mathcal{S}_1,\dots,\mathcal{S}_n} \sum_{i=1}^{n} \int_{\mathbb{R}} F_i(z) \,\mathrm{d}\,\mathcal{S}_i(z) \\
&\quad \text{s.t.} \sum_{i=1}^{n} \int_{\mathbb{R}} 1_{\{x_i \le z\}}(z, x) \,\mathrm{d}\,\mathcal{S}_i(z) \ge h(x), \quad (x \in \mathbb{R}^n).
\end{aligned} \tag{4.3}$$

The signed measures $\mathcal{S}_i$ being of finite variation, the functions $S_i(z) = \mathcal{S}((-\infty, z])$ and the limits $s_i = \lim_{z \to \infty} S_i(z) = \mathcal{S}_i((-\infty, +\infty))$ are well defined and finite. Furthermore, using $\lim_{z \to -\infty} F_i(z) = 0$ and $\lim_{z \to +\infty} F_i(z) = 1$, we have

$$\begin{aligned}
\sum_{i=1}^{n} \int_{\mathbb{R}} F_i(z) \,\mathrm{d}\,\mathcal{S}(z) &= \sum_{i=1}^{n} \left(F_i(z) S_i(z)|_{-\infty}^{+\infty} - \int_{\mathbb{R}} S_i(z) \,\mathrm{d}\,F_i(z) \right) \\
&= \sum_{i=1}^{n} s_i - \sum_{i=1}^{n} \int_{\mathbb{R}} S_i(z) \,\mathrm{d}\,F_i(z) \\
&= \sum_{i=1}^{n} \int_{\mathbb{R}} (s_i - S_i(z)) \,\mathrm{d}\,F_i(z),
\end{aligned}$$

and likewise,

$$\sum_{i=1}^{n} \int_{\mathbb{R}} 1_{\{x_i \le z\}}(z, x) \,\mathrm{d}\,\mathcal{S}_i(z) = \sum_{i=1}^{n} \int_{x_i}^{+\infty} 1 \,\mathrm{d}\,\mathcal{S}_i(z) = \sum_{i=1}^{n} (s_i - S_i(x_i)).$$

Writing $h_i(z) = s_i - S_i(z)$, (4.3) is therefore equivalent to

$$\begin{aligned}
&\inf_{h_1,\dots,h_n} \sum_{i=1}^{n} \int_{\mathbb{R}} h_i(z) \,\mathrm{d}\,F_i(z) \\
&\quad \text{s.t.} \sum_{i=0}^{n} h_i(x_i) \ge h(x), \quad (x \in \mathbb{R}^n).
\end{aligned}$$

This is the dual identified in Section 3.1.

4.2 The Marginal Problems with Bounds on Copulas

Now consider the problem from Section 3.2,

$$\begin{aligned} \sup_{\mu \in \mathscr{M}_{F_1,\ldots,F_n}} \int_{\mathbb{R}^n} h(x)\,\mathrm{d}\,\mu(x), \\ \text{s.t. } \mathcal{C}_{\mathrm{lo}} \leq \mathcal{C}_{\mathcal{F}} \leq \mathcal{C}_{\mathrm{up}}, \end{aligned} \tag{4.4}$$

Again formulating this in our framework we get

$$\begin{aligned} &\sup_{\mathcal{F}} \int_{\mathbb{R}^n} h(x)\,\mathrm{d}\,\mathcal{F}(x) \\ &\text{s.t. } \int_{\mathbb{R}^n} 1_{\{x \leq (F_1^{-1}(u_1),\ldots,F_n^{-1}(u_n))\}}(u,x)\,\mathrm{d}\,\mathcal{F}(x) \leq \mathcal{C}_{\mathrm{up}}(u), \quad (u \in [0.1]^n), \\ &\quad \int_{\mathbb{R}^n} -1_{\{x \leq (F_1^{-1}(u_1),\ldots,F_n^{-1}(u_n))\}}(u,x)\,\mathrm{d}\,\mathcal{F}(x) \leq -\mathcal{C}_{\mathrm{lo}}(u), \quad (u \in [0.1]^n), \\ &\quad \int_{\mathbb{R}^n} 1_{\{x_i \leq z\}}(z,x)\,\mathrm{d}\,\mathcal{F}(x) = F_i(z), \quad (i \in \mathbb{N}_n,\ z \in \mathbb{R}), \\ &\quad \mathcal{F} \geq 0. \end{aligned} \tag{4.5}$$

The dual of this problem is given by

$$\begin{aligned} \inf_{\mathcal{G}_{\mathrm{up}},\mathcal{G}_{\mathrm{lo}},\mathcal{S}_1,\ldots\mathcal{S}_n} &\int_{[0,1]^n} \mathcal{C}_{\mathrm{up}}(u)\,\mathrm{d}\,\mathcal{G}_{\mathrm{up}}(u) - \int_{[0,1]^n} \mathcal{C}_{\mathrm{lo}}(u)\,\mathrm{d}\,\mathcal{G}_{\mathrm{lo}}(u) + \sum_{i=1}^n \int_{\mathbb{R}} F_i(z)\,\mathrm{d}\,\mathcal{S}_i(z) \\ \text{s.t. } &\int_{[0,1]^n} 1_{\{x \leq (F_1^{-1}(u_1),\ldots,F_n^{-1}(u_n))\}}(u,x)\,\mathrm{d}\,\mathcal{G}_{\mathrm{up}}(u) \\ &\quad - \int_{[0,1]^n} 1_{\{x \leq (F_1^{-1}(u_1),\ldots,F_n^{-1}(u_n))\}}(u,x)\,\mathrm{d}\,\mathcal{G}_{\mathrm{lo}}(u) \\ &\quad + \sum_{i=1}^n \int_{\mathbb{R}} 1_{\{x_i \leq z\}}\,\mathrm{d}\,\mathcal{S}_i(z) \geq h(x), \quad (x \in \mathbb{R}^n), \\ &\mathcal{G}_{\mathrm{lo}}, \mathcal{G}_{\mathrm{up}} \geq 0. \end{aligned} \tag{4.6}$$

Using the notation s_i, S_i introduced in the previous section, this problem can be written as

$$\begin{aligned} \inf_{\mathcal{G}_{\mathrm{up}},\mathcal{G}_{\mathrm{lo}},\mathcal{S}_1,\ldots\mathcal{S}_n} &\int_{[0,1]^n} \mathcal{C}_{\mathrm{up}}(u)\,\mathrm{d}\,\mathcal{G}_{\mathrm{up}}(u) - \int_{[0,1]^n} \mathcal{C}_{\mathrm{lo}}(u)\,\mathrm{d}\,\mathcal{G}_{\mathrm{lo}}(u) + \sum_{i=1}^n \int_{\mathbb{R}} (s_i - S_i(z))\,\mathrm{d}\,F_i(z) \\ \text{s.t. } &\mathcal{G}_{\mathrm{up}}(\mathscr{B}(x)) - \mathcal{G}_{\mathrm{lo}}(\mathscr{B}(x)) + \sum_{i=1}^n (s_i - S_i(x_i)) \geq h(x), \quad (x \in \mathbb{R}^n), \\ &\mathcal{G}_{\mathrm{up}}, \mathcal{G}_{\mathrm{lo}} \geq 0, \end{aligned}$$

where $\mathscr{B}(x) = \{u \in [0,1]^n : u \geq (F_1(x_1), \ldots, F_n(x_n))\}$.
Due to the high dimensionality of the space of variables and constraints both in the primal and the dual, the copula constrained marginal problem is difficult to solve numerically, even for very coarse discrete approximations.
Note that although the primal problem was formulated and discussed in [23], the dual formulation (4.6) is a new result and is a direct implementation of our general duality theory.

4.3 The Moment Problem

Consider the moment problem from Section 3.3

$$\begin{aligned} &\sup_X \mathrm{P}[r(X) \geq 0] \\ &\quad \mathrm{E}_\mu[X_1^{k_1} \ldots X_n^{k_n}] = b_k, \quad (k \in J), \\ &\quad X \text{ a random vector taking values in } \mathbb{R}^n, \end{aligned} \tag{4.7}$$

where $r : \mathbb{R}^n \to \mathbb{R}$ is a multivariate polynomial and $J \subset \mathbb{N}^n$ is a finite set of multi-indices. In other words, a few of the moments of X are known. By choosing $\Phi = \mathbb{R}^n$, $\Gamma = \emptyset$, $\Sigma = J \cup \{0\}$,

$$\begin{aligned} &B(k,x) = x_1^{k_1} \ldots x_n^{k_n}, \quad b(k) = b_k, \quad (k \in J), \\ &B(0,x) = 1_{\mathbb{R}^n}, \quad b(0) = 1, \end{aligned}$$

and $c(x) = 1_{\{x:\, r(x)\geq 0\}}$, where 1 denotes the indicator function, problem (4.7) becomes a special case of the primal problem (3.16) considered in Chapter 3,

$$\begin{aligned} \sup_{\mathcal{F}} &\int_{\mathbb{R}^n} 1_{\{x:\, r(x)\geq 0\}}(x)\,\mathrm{d}\,\mathcal{F}(x) \\ \text{s.t.} &\int_{\mathbb{R}^n} x_1^{k_1} \ldots x_n^{k_n}\,\mathrm{d}\,\mathcal{F}(x) = b_k, \quad (k \in J), \\ &\int_{\mathbb{R}^n} 1\,\mathrm{d}\,\mathcal{F}(x) = 1, \\ &\mathcal{F} \geq 0. \end{aligned}$$

Our dual

$$\begin{aligned} \inf_{(z,z_0)\in\mathbb{R}^{|J|+1}} &\sum_{k\in J} z_k b_k + z_0 \\ \text{s.t.} &\sum_{k\in J} z_k x_1^{k_1} \ldots x_n^{k_n} + z_0 \geq 1_{\{r(x)\geq 0\}}(x), \quad (x \in \mathbb{R}^n) \end{aligned}$$

is easily seen to be identical with the dual (3.17) identified by Bertsimas and Popescu,

$$\begin{aligned}
\inf_{(z,z_0)\in\mathbb{R}^{|J|+1}} & \sum_{k\in J} z_k b_k + z_0 \\
\text{s.t. } & \forall\, x\in\mathbb{R}^n, r(x)\geq 0 \Rightarrow \sum_{k\in J} z_k x_1^{k_1}\dots x_n^{k_n} + z_0 - 1 \geq 0, \\
& \forall\, x\in\mathbb{R}^n, \sum_{k\in J} z_k x_1^{k_1}\dots x_n^{k_n} + z_0 \geq 0.
\end{aligned}$$

4.4 Constraints by Integrals of Piecewise Concave Functions

In quantitative risk management, distributions are often estimated within a parametric family from the available data. For example, the tails of marginal distributions may be estimated via extreme value theory, or a Gaussian copula may be fitted to the multivariate distribution of all risks under consideration, to model their dependencies. The choice of a parametric family introduces *model uncertainty*, while fitting a distribution from this family via statistical estimation introduces *parameter uncertainty*. In both cases, a more robust alternative would be to study models in which the available data is only used to estimate upper and lower bounds on finitely many integrals of the form

$$\int_\Phi \phi(x)\,\mathrm{d}\,\mathcal{F}(x), \tag{4.8}$$

where $\phi(x)$ is a suitable test function. A suitable way of estimating upper and lower bounds on such integrals from sample data x_i $(i\in\mathbb{N}_k)$ is to estimate confidence bounds via bootstrapping.

4.4.1 General Setup and Duality

Let Φ be decomposed into a partition $\Phi=\bigcup_{i=1}^k \Xi_i$ of polyhedra Ξ_i with nonempty interior, chosen as regions in which a reasonable number of data points are available to estimate integrals of the form (4.8).

Each polyhedron has a primal description in terms of generators,

$$\Xi_i = \operatorname{conv}(q_1^i,\dots,q_{n_i}^i) + \operatorname{cone}(r_1^i,\dots,r_{o_i}^i)$$

where $\operatorname{conv}(q_1^i,\dots,q_{n_i}^i)$ is the polytope with vertices $q_n^i\in\mathbb{R}^n$, and

$$\operatorname{cone}(r_1^i,\dots,r_{o_i}^i) = \left\{\sum_{m=1}^{o_i}\xi_m r_m^i \,:\, \xi_m\geq 0\ (m\in\mathbb{N}_{o_i})\right\}$$

is the polyhedral cone with recession directions $r^i_m \in \mathbb{R}^n$. Each polyhedron also has a dual description in terms of linear inequalities,

$$\Xi_i = \bigcap_{j=1}^{k_i} \left\{x \in \mathbb{R}^n : \langle f^i_j, x\rangle \geq \ell^i_j\right\},$$

for some vectors $f^i_j \in \mathbb{R}^n$ and bounds $\ell^i_j \in \mathbb{R}$. The main case of interest is where Ξ_i is either a finite or infinite box in $\mathbb{R}^n$ with faces parallel to the coordinate axes, or an intersection of such a box with a linear half space, in which case it is easy to pass between the primal and dual descriptions. Note however that the dual description is preferrable, as the description of a box in $\mathbb{R}^n$ requires only $2n$ linear inequalities, while the primal description requires 2^n extreme vertices.

Let us now consider the problem

$$\begin{aligned}
\text{(P)} \quad \sup_{\mathcal{F}\in\mathscr{M}_{\mathfrak{F}}} & \int_\Phi h(x)\,\mathrm{d}\,\mathcal{F}(x) \\
\text{s.t.} & \int_\Phi \phi_s(x)\,\mathrm{d}\,\mathcal{F}(x) \leq a_s, \quad (s = 1,\dots,M), \\
\text{s.t.} & \int_\Phi \psi_t(x)\,\mathrm{d}\,\mathcal{F}(x) = b_t, \quad (t = 1,\dots,N), \\
& \int_\Phi 1\,\mathrm{d}\,\mathcal{F}(x) = 1, \\
& \mathcal{F} \geq 0,
\end{aligned}$$

where the test functions ψ_t are piecewise linear on the partition $\Phi = \bigcup_{i=1}^k \Xi_i$, and where $-h(x)$ and the test functions ϕ_s are piecewise linear on the infinite polyhedra of the partition, and either jointly linear, concave, or convex on the finite polyhedra (i.e., polytopes) of the partition. The dual of (P) is

$$\begin{aligned}
\text{(D)} \quad \inf_{(y,z)\in\mathbb{R}^{M+N+1}} & \sum_{s=1}^M a_s y_s + \sum_{t=1}^N b_t z_t + z_0, \\
\text{s.t.} & \sum_{s=1}^M y_s\phi_s(x) + \sum_{t=1}^N z_t\psi_t(x) + z_0\,1_\Phi(x) - h(x) \geq 0,\ (x \in \Phi), \\
& y \geq 0.
\end{aligned} \tag{4.9}$$

We remark that (P) is a semi-infinite programming problem with infinitely many variables and finitely many constraints, while (D) is a semi-infinite programming problem with finitely many variables and infinitely many constraints. However, the constraint (4.9) of (D) can be rewritten as copositivity requirements over the polyhedra Ξ_i,

$$\sum_{s=1}^{M} y_s \phi_s(x) + \sum_{t=1}^{N} z_t \psi_t(x) + z_0 \, 1_\Phi(x) - h(x) \geq 0, \quad (x \in \Xi_i), \quad (i = 1, \dots, k).$$

Later in Chapter 6 we will see how these copositivity constraints can be handled numerically, often by relaxing all but finitely many constraints. Nesterov's first order method can be adapted to solve the resulting problems, see [51, 52].

4.5 Model-Free Option Pricing

Consider an exotic path-dependent option that depends on the value of a stock price at some discrete times $t_1 < t_2 < \cdots < t_n$. Let $X = (X_1, \dots, X_n)$ be a n-dimensional random variable denoting stock prices at times $(t_1, \dots, t_n)$ respectively and $h(X_1, \dots, X_n)$ be the payoff of the given option. We assume 0 interest rate, since the generalisation to non-zero rates is straightforward.

In practice, in a no-arbitrage framework, one assumes that the stock price follows some specific model, calibrates this model to market data and finds the fair price of an option, i.e., find a risk neutral probability measure Q on $\mathbb{R}^n$ and calculate the price as

$$\mathbb{E}_Q[h] = \int_{\mathbb{R}^n} h(x_1, \dots, x_n) \, \mathrm{d}\, Q(x_1, \dots, x_n). \tag{4.10}$$

Under Q the stock price process $(X_i)_{i=1,\dots,n}$ is assumed to be a discrete martingale.

In this section we aim to find bounds on the prices of the option without assuming any particular model for the stock price process, i.e., find maximum and minimum values over all models that are consistent with the observed market data. Assuming knowledge of market call option prices, one can recover a distribution function $\mathcal{F}_i$ of X_i for every $i = 1, \dots, n$ (see [49] e.g.), which will represent the marginal distributions of a risk neutral equivalent probability measure of X. In this framework, denoting by $\mathcal{M}(\mathcal{F}_1, \dots, \mathcal{F}_n)$ the set of martingale measures with marginals $\mathcal{F}_1, \dots, \mathcal{F}_n$, the problem becomes

$$\inf_{\mathcal{F} \in \mathcal{M}(\mathcal{F}_1, \dots, \mathcal{F}_n)} \int_{\mathbb{R}^n} h(x_1, \dots, x_n) \, \mathrm{d}\, \mathcal{F}(x_1, \dots, x_n). \tag{4.11}$$

Note that this is the marginal problem from Section 3.1, with the difference that the decision variable $\mathcal{F}$ is not from all the probability measures with given marginals but from a set of probability measures which are martingales. Now we will formulate the

condition of being a martingale in our framework.
$\mathcal{F}$ is a martingale measure if and only if for every $k = 1, \dots, n-1$ and every Borel measurable set $\mathcal{B} \subseteq \mathbb{R}^k$

$$\int_{\mathbb{R}^n} 1_{\mathcal{B}}(x_1, \dots, x_k)(x_{k+1} - x_k)\,\mathrm{d}\,\mathcal{F}(x_1, \dots, x_n) = 0, \tag{4.12}$$

which is equivalent to ([2], Lemma 2.3): for every $z, y \in \mathbb{R}^k$ such that $z \leq y$

$$\int_{\mathbb{R}^n} 1_{z \leq x|_k \leq y}(x_1, \dots, x_k)(x_{k+1} - x_k)\,\mathrm{d}\,\mathcal{F}(x_1, \dots, x_n) = 0, \tag{4.13}$$

where by $x|_k$ we denote the first k coordinates of x and the comparison of the vectors is component-wise.
Now we are ready to formulate the problem (8.2) in the new framework.
We aim to find the dual problem of the following optimisation problem.

$$\begin{aligned}
&\inf_{\mathcal{F}} \int_{\mathbb{R}^n} \phi(x)\,\mathrm{d}\,\mathcal{F}(x) \\
&\text{s.t.} \int_{\mathbb{R}^n} 1_{\{x_i \leq t\}}(t, x)\,\mathrm{d}\,\mathcal{F}(x) = F_i(t), \quad (t \in \mathbb{R}, i \in \mathbb{N}_n) \\
&\qquad \int_{\mathbb{R}^n} 1_{z|_k \leq x|_k \leq y|_k}(z, y, x)(x_{k+1} - x_k)\,\mathrm{d}\,\mathcal{F}(x) = 0, \quad (y, z \in \mathbb{R}^n, k < n) \\
&\qquad \mathcal{F} \geq 0.
\end{aligned}$$

Denote $\Phi = \mathbb{R}^n$ and $\Sigma = \{1, \dots, 2n-1\} \times \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^n$.
We construct the function $b(k, t, z, y) = F_k(t)$ if $k \leq n$, and $b(k, t, z, y) = 0$ for $k > n$. Further, for $k \leq n$ we have $B(k, t, z, y, x) = 1_{x_k \leq t}$. For $k = 1, \dots, n-1$ we have

$$B(k+n, t, z, y, x) = 1_{z|_k \leq x|_k \leq y|_k}(x_{k+1} - x_k)$$

Now we construct the dual problem.
As an objective function we have

$$\int_{\Sigma} b(k, t, z, y)\,\mathrm{d}\,\mathcal{S}(k, t, z, y) = \sum_{k=1}^{n} \int_{\mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^n} F_k(t)\,\mathrm{d}\,\bar{\mathcal{S}}_k(t, z, y).$$

For $k = 1, \dots, n$ define the measures $\mathcal{S}_k$ on Borel sigma-algebra $\mathcal{B}(\mathbb{R})$ such that $\mathcal{S}_k(D) = \bar{\mathcal{S}}_k(D \times \mathbb{R}^n \times \mathbb{R}^n)$ for every $D \in \mathcal{B}(\mathbb{R})$. As for the marginal problem, the signed measures $\mathcal{S}_k$ being of finite variation, the functions $S_k(t) = \mathcal{S}((-\infty, t])$ and the limits $s_k = \lim_{t \to \infty} S_k(t) = \mathcal{S}((-\infty, +\infty))$ are well defined and finite. Furthermore, using $\lim_{t\to-\infty} F_k(t) = 0$ and $\lim_{t\to+\infty} F_k(t) = 1$, we have

$$\begin{aligned}\sum_{k=1}^{n}\int_{\mathbb{R}} F_k(t)\,\mathrm{d}\,\mathcal{S}_k(t) &= \sum_{k=1}^{n}\left(F_k(t)S_k(t)|_{-\infty}^{+\infty} - \int_{\mathbb{R}} S_k(t)\,\mathrm{d}\,F_k(t)\right)\\ &= \sum_{k=1}^{n} s_k - \sum_{k=1}^{n}\int_{\mathbb{R}} S_k(t)\,\mathrm{d}\,F_k(t)\\ &= \sum_{k=1}^{n}\int_{\mathbb{R}} (s_k - S_k(t))\,\mathrm{d}\,F_k(t),\end{aligned}$$

and likewise for constraints. Writing $h_k(t) = s_k - S_k(t)$, $(k = 1,\ldots,n)$, the objective function becomes

$$\sum_{k=1}^{n}\int_{\mathbb{R}} h_k(t)\,\mathrm{d}\,F_k(t),$$

and the left hand side of the constraints become

$$\begin{aligned}\int_{\Sigma} B(k,t,z,y,x)\,\mathrm{d}\,\mathcal{S}(k,t,z,y) \qquad &= \sum_{k=1}^{n}\int_{\mathbb{R}\times\mathbb{R}^n\times\mathbb{R}^n} 1_{x_k\le t}\,\mathrm{d}\,\bar{\mathcal{S}}_k(t,z,y)\\ &+\sum_{k=1}^{n-1}\int_{\mathbb{R}\times\mathbb{R}^n\times\mathbb{R}^n} 1_{z|_k\le x|_k\le y|_k}(x_{k+1}-x_k)\,\mathrm{d}\,\bar{\mathcal{S}}_{n+k}(t,z,y)\\ &= \sum_{k=1}^{n} h_k(x_k) + \sum_{k=1}^{n-1} g_k(x_1,\ldots,x_k)(x_{k+1}-x_k),\end{aligned}$$

where $g_k(x_1,\ldots,x_k) = \int_{\mathbb{R}^n\times\mathbb{R}^n} 1_{z|_k\le x|_k\le y|_k}(x_{k+1}-x_k)\,\mathrm{d}\,\mathcal{S}_{k+n}(z,y)$, $k = 1,\ldots,n-1$. Measures $\mathcal{S}_{n+k}$, $k = 1,\ldots,n-1$ are defined such that $\mathcal{S}_{k+n}(D) = \bar{\mathcal{S}}_{k+n}(\mathbb{R}\times D)$ for every $D\in\mathcal{B}(\mathbb{R}^n\times\mathbb{R}^n)$.

We obviously have that the functions g_k, $k = 1,\ldots,n-1$ are bounded. So we get the following dual formulation of our primal problem.

$$\begin{aligned}&\sup_{h_1,\ldots,h_n}\ \sum_{k=1}^{n}\int_{\mathbb{R}} h_k(\tau)\,\mathrm{d}\,F_k(\tau)\\ &\text{s.t.}\ \sum_{k=0}^{n} h_k(x_k) + \sum_{k=1}^{n-1} g_k(x_1,\ldots,x_k)(x_{k+1}-x_k) \le h(x), \quad (x\in\mathbb{R}^n).\end{aligned} \tag{4.14}$$

Since the decision variables $g_1,\ldots,g_{n-1}$ have a zero coefficient in the objective function, we get that the dual problem (4.14) is equivalent to

$$\sup\left\{\sum_{k=1}^{n}\mathbb{E}_{F_k}\left[h_k\right] : \exists g_1,\ldots g_{n-1} \text{ s.t. } \Psi_{(h_k),(g_j)} \le \phi\right\}, \tag{4.15}$$

where

$$\Psi_{(h_k),(g_j)}(x_1, \ldots, x_n) = \sum_{k=0}^{n} h_k(x_k) + \sum_{k=1}^{n-1} g_k(x_1, \ldots, x_k)(x_{k+1} - x_k)$$

and $\mathbb{E}_{F_k}(h_k(t)) = \int_{\mathbb{R}} h(t) \, \mathrm{d}\, F_k(t)$.
Note that (4.15) is exactly the same dual as presented in Theorem 8.1.1.

Chapter 5

Strong Duality

In the previous chapter we considered a general duality theory for linear problems. It is easy to prove a weak duality result, as was done in Theorem 4.0.6. In many applications, numerical solutions of the dual problems are much easier to find. This makes it important to have strong duality between primal and dual problems, so that by solving the dual problem we also find the optimal value of the primal problem.
In the previous chapter we also discussed some particular cases of the duality results that have already been investigated by researchers and where strong duality is proved under some conditions.
The duality theory of linear infinite dimensional optimisation problems has been under research for a long time. [42] studies the problem when the integration functions are continuous functions and the optimisation is over the space of bounded regular Borel measures on compact Hausdorff spaces. In the later work strong duality is proven and the authors also prove that the optimal solution is reached at some external point of the feasible region. [40] presents a numerical approach to the same optimisation problem over L^1 spaces, in the case when the objective function in the integration is non-negative. [27] studies the problem over metric spaces when the integrand functions are all non-negative. To our knowledge, strong duality between the primal and the dual problems has not been studied for more general cases, except of [66], which will be discussed in this chapter.
In this chapter we will introduce two more particular cases where there is no duality gap between the pairs of problems. The first case is where the optimisation is over measures that have L^p density functions. The second case is for semi-infinite optimisation problems and is an extension of a similar problem studied in [66], where integrand functions are considered to be continuous. We extend this result to piecewise continuous integrand functions, which is of our interest since we use these problems for our

model discussed in Section 4.4.

Our work is based on Shapiro's paper on conic linear optimisation problems, [66], so first we introduce these results, and then formulate our problems and complete their proofs.

Note that we do not provide proofs of the theorems discussed in the next subsection; refer to [66] for more details.

5.1 General Results for Conic Optimisation Problems

As mentioned above, in this section we introduce results on conic linear optimisation problems from [66].

Consider a conic linear optimisation problem of the following form

$$\min_{f \in C} \langle c, f \rangle \quad \text{subject to} \quad \mathcal{A}f + h \in K, \tag{5.1}$$

where X and Y are linear spaces, $C \subset X$, $K \subset Y$ are convex cones, $h \in Y$ and $\mathcal{A} : X \to Y$ is a linear map. Assume that X and Y are paired with some linear spaces X' and Y' respectively, so bilinear forms $\langle \cdot, \cdot, \rangle : X' \times X \to \mathbb{R}$ and $\langle \cdot, \cdot, \rangle : Y' \times Y \to \mathbb{R}$ are defined. We call the problem (5.1) the primal problem.

Shapiro's results are based on conjugate duality first introduced by Rockefellar [62], [63].

Define the positive dual cone of C as

$$C^* := \{f^* \in X^* : \langle f^*, f \rangle \geq 0, \quad \forall f \in C\}, \tag{5.2}$$

and similarly for the cone K

$$K^* := \{g^* \in Y^* : \langle g^*, g \rangle \geq 0, \quad \forall g \in K\}, \tag{5.3}$$

We also need an assumption for X' so that the adjoint mapping of A exists.

Assumption 5.1.1 *For any $g^* \in Y'$ there exists a unique $f^* \in X'$ such that $\langle g^*, \mathcal{A}f \rangle = \langle f^*, f \rangle$ for all $f \in X$.*

Based on this assumption we can define the adjoint mapping $\mathcal{A}^* : Y' \to X'$ by the equation

$$\langle g^*, \mathcal{A}f \rangle = \langle \mathcal{A}^* g^*, f \rangle, \quad \forall f \in X. \tag{5.4}$$

Now consider the Lagrangian function of the primal problem (5.1)

$$L(f, g^*) := \langle c, f \rangle + \langle g^*, Af + h \rangle \tag{5.5}$$

and the following optimisation problem

$$\min_{f \in C} \left\{ \psi(f) := \max_{g^* \in -K^*} \{L(f, g^*)\} \right\}. \tag{5.6}$$

By changing the min and max operators we get the dual Lagrangian problem

$$\max_{g^* \in -K^*} \left\{ \phi(g^*) := \min_{f \in C} \{L(f, g^*)\} \right\} \tag{5.7}$$

which is equivalent to the following optimisation problem

$$\max_{g^* \in -K^*} \langle g^*, h \rangle \quad \text{subject to} \quad \mathcal{A}^* g^* + c \in C^*, \tag{5.8}$$

which we call the dual problem.
The aim of this section is to find conditions under which $Val(D) = Val(P)$, where by $Val(D)$ and $Val(P)$ we denote the objective values of the dual (5.8) and primal (5.1) problems respectively.
Note that the dual problem is also a conic linear problem, and that it is easy to show the weak duality, i.e., $Val(D) \leq Val(P)$.
Further, we associate with the primal problem the optimal value function

$$v(g) := \inf \{\langle c, f \rangle : f \in C, \mathcal{A}f + g \in K\}. \tag{5.9}$$

We define $v(g)$ to be $+\infty$ if the set $\{f \in C : \mathcal{A}f + g \in K\}$ is empty. We have $Val(P) = v(h)$.
From [63] we know that the extended optimal value function $v(g)$ is convex and positively homogeneous of degree 1, i.e., $\forall t > 0$ and $g \in Y$ $v(tg) = tv(g)$.
The conjugate of $v(g)$ is defined as

$$v^*(g^*) := \sup_{g \in Y} \{\langle g^*, g \rangle - v(g)\}. \tag{5.10}$$

Evaluating the formulae above we get

$$\begin{aligned}
v^*(g^*) &= \sup \{\langle g^*, g \rangle - \langle c, f \rangle : (f, g^*) \in X \times Y^*, f \in C, \mathcal{A}f + g \in K\} \\
&= \sup_{f \in C} \sup_{\mathcal{A}f + g \in K} \{\langle g^*, g \rangle - \langle c, f \rangle\} \\
&= \sup_{f \in C} \sup_{g \in K} \{\langle g^*, g - \mathcal{A}f \rangle - \langle c, f \rangle\} \\
&= \sup_{f \in C} \sup_{g \in K} \{\langle g^*, g \rangle - \langle \mathcal{A}^* g^* + c, f \rangle\}
\end{aligned}$$

It is easy to show that $v^*(g^*)$ is the indicator function of the feasible set of the dual problem, since from $g^* \in -K^*$ and $\mathcal{A}^*g^* + c \in C^*$ it follows that $v^*(g^*) = 0$, and $v^*(g^*) = +\infty$ otherwise. So we can write the dual problem as

$$\max_{g^* \in Y^*} \left\{ \langle g^*, h \rangle - v^*(g^*) \right\}. \tag{5.11}$$

Taking the biconjugate of $v(y)$

$$v^{**}(g) := \sup_{g^* \in Y^*} \left\{ \langle g^*, g \rangle - v^*(g^*) \right\}, \tag{5.12}$$

we see that $Val(D) = v^{**}(h)$, hence we get, that if $v(h) = v^{**}(h)$, then there is no duality gap between Lagrangian primal and dual problems.
Now we aim to find conditions such that $v(h) = v^{**}(h)$.
We have described the main approach to the proof of the strong duality in this framework, and now, without going into details, we will introduce the results given by Shapiro. More interested reader can refer to [66] for more details.
We make an assumption which will be considered to hold throughout this section.

Assumption 5.1.2 *The spaces Y and Y' are paired locally convex topological vector spaces.*

Denote by $\operatorname{lsc} v$ the lower semicontinous hull of the function v, i.e.

$$\operatorname{lsc} v(g) = \min \left\{ v(g), \liminf_{z \to g} v(z) \right\}, \tag{5.13}$$

and by $\operatorname{cl} v$ the closure of the function v:

$$\operatorname{cl} v(\cdot) := \begin{cases} \operatorname{lsc} v(\cdot), & \text{if } \operatorname{lsc} v(g) > -\infty \text{ for all } g \in Y, \\ -\infty, & \text{if } \operatorname{lsc} v(g) = -\infty \text{ for at least one } g \in Y. \end{cases} \tag{5.14}$$

We say that the problem (P) is sub-consistent if $\operatorname{lsc} v(h) < +\infty$ (if problem (P) is consistent, i.e. its feasible set is nonempty, then it is also sub-consistent). Moreover, the Fenchel-Moreau theorem implies that $v^{**} = \operatorname{cl} v$. Taking into account the fact that if $\operatorname{lsc} v(h) < +\infty$ then $\operatorname{cl} v(h) = \operatorname{lsc} v(h)$, we get the following proposition:

Proposition 5.1.3 (Proposition 2.2, [66]) *The following holds:*

1. *$Val(D) = \operatorname{cl} v(h)$.*

2. *If (P) is sub-consistent, then $Val(D) = \operatorname{lsc} v(h)$.*

The above proposition shows that if (P) is sub-consistent then strong duality holds, i.e. $Val(D) = Val(P)$ iff $v(h)$ is lower semicontinous at $g = h$. But it may be difficult to verify the semicontinuity directly, so we seek more tractable conditions in the subsequent analysis.
Define the sub-differential of the function v at a point g (where v is finite) as

$$\partial v(g) := \{g^* \in Y' : v(z) - v(g) \geq \langle g^*, z - g\rangle, \quad \forall z \in Y\}\,. \tag{5.15}$$

We say that v is sub-differentiable at a point g if $v(g)$ is finite and $\partial v(g)$ is nonempty. Further, we know that if v is sub-differentiable at $g = h$, then $v^{**} = v(h)$, and conversely, if $v(b)$ is finite and $v^{**} = v(h)$, then $\partial v(h) = \partial v^{**}(h)$, [63].
We now get the following proposition.

Proposition 5.1.4 (Proposition 2.5, [66]) *If the optimal value function $v(g)$ is sub-differentiable at the point $g = h$, then $Val(P) = Val(D)$ and the set of optimal solutions of (D) is $\partial v(h)$. Conversely, if $Val(P) = Val(D)$ and is finite, then $Sol(D) = \partial v(h)$. By $Sol(D)$ we denote the set of solutions of the problem (D).*

However, checking the sub-differentiability for the optimal value function may still be difficult.
Consider the set

$$M := \{(g, \alpha) \in Y \times \mathbb{R} : g = k - \mathcal{A}f, \alpha \geq \langle c, f\rangle, f \in C, k \in K\}\,. \tag{5.16}$$

It is easy to show that the optimal value of the problem (P) is equal to the optimal value of the following problem

$$\begin{gathered}\min \alpha \\ \text{s.t. } (h, \alpha) \in M.\end{gathered}$$

Proposition 5.1.5 (Proposition 2.6, [66]) *Suppose that $Val(P)$ is finite and the cone M is closed in the product topology of $Y \times \mathbb{R}$. Then $Val(P) = Val(D)$ and the primal problem (P) has an optimal solution.*

From convex analysis we know that if Y is a Banach space and Y^* is its standard dual, then $\partial v(h)$ is closed and bounded in the dual topology of Y^* ([38], p. 84). We get the following proposition.

Proposition 5.1.6 (Proposition 2.7, [66]) *If the optimal value function $v(g)$ is continuous at $g = h$ and if Y is a Banach space paired with its standard dual Y^*, then the set of optimal solutions of (D) is bounded in the dual norm topology of Y^*.*

From [61] we know that if X and Y are Banach spaces equipped with strong topologies, the cones C and K are closed and $\langle c, \cdot \rangle$ and $\mathcal{A} : X \to Y$ are continuous, then $v(g)$ is continuous at $g = h$ if and only if

$$h \in \operatorname{int}(\operatorname{dom} v). \tag{5.17}$$

Since $\operatorname{dom} v = -\mathcal{A}(C) + K$, we can write the condition (5.17) as

$$-h \in \operatorname{int}(\mathcal{A}(C) - K). \tag{5.18}$$

Hence we get the final component of the framework we need to prove our strong duality.

Proposition 5.1.7 (Proposition 2.9, [66]) *Suppose that X and Y are Banach spaces, the cones C and K are closed, $\langle c, \cdot \rangle$ and $\mathcal{A} : X \to Y$ are continuous, and that condition* (5.18) *holds. Then $Val(P) = Val(D)$ and $Sol(D)$ is nonempty and bounded.*

If the cone K has a non-empty interior, then condition (5.18) is equivalent to ([6], Proposition 2.106)

$$\exists \bar{f} \in C \text{ such that } \mathcal{A}\bar{f} + h \in \operatorname{int}(K). \tag{5.19}$$

If the later condition holds, then it is said that the generalized Slater condition is satisfied for problem (5.1) ([66], p. 9).
In many applications we have equality type constraints for optimisation problems of the form (5.1). In this case the cone K has obviously a single element 0, and hence, the interior is empty.
If the constraint in the problem (5.1) is given by the equality

$$\mathcal{A}f + h = 0,$$

then the regularity condition (5.18) is equivalent to ([6], section 2.3.4)

$$\mathcal{A}(X) = Y,$$
$$\exists \bar{f} \in \operatorname{int}(C) \text{ s. t. } \mathcal{A}\bar{f} + h = 0.$$

Having introduced the mathematical framework for general conic linear optimisation problems, we are now ready to use these results to get strong duality results for two particular cases of our general duality theory. First we discuss a case in which we optimise on measures with L^p densities and then strong duality for the model discussed in Section 4.4.

5.2 Strong Duality for Measures with L^p Densities

Consider a special case of the primal problem (P) from Chapter 4, where the optimisation is over measures that have a density function that belongs to $L^p(\Phi)$, $p > 1$, where $L^p(\Phi)$ is the space of functions over Φ for which the p-th power of the absolute value is Lebesgue integrable on Φ. In this section we introduce some conditions under which the strong duality holds for the primal and the dual problems. Although here we only consider the $p > 1$ case, we do expect the strong duality to hold for $p = 1$ case, though for some stricter regulatory conditions.
Let $\Phi \subseteq \mathbb{R}^n$, $\Gamma \subseteq \mathbb{R}^m$ and $\Sigma \subseteq \mathbb{R}^k$.

$$
\begin{aligned}
(P^L) \sup_{f \in L^p(\Phi)} & \int_\Phi c(x) f(x) dx \\
& \int_\Phi A(y,x) f(x) dx \leq a(y), \text{ a.e. } y \in \Gamma \\
& \int_\Phi B(z,x) f(x) dx = b(z), \text{ a.e. } z \in \Sigma \\
& f \geq 0.
\end{aligned}
$$

Let $c \in L^q(\Phi)$ and $a \in L^p(\Psi)$ for some $1 < p < \infty$ and n, m, k.
Assume $A : \Gamma \times \Phi \to \mathbb{R}$ and $B : \Sigma \times \Phi \to \mathbb{R}$ are such that

$$
\begin{aligned}
A(y, \cdot) &\in L^q(\Phi) \\
A(\cdot, x) &\in L^p(\Gamma) \\
B(z, \cdot) &\in L^q(\Phi) \\
B(\cdot, x) &\in L^p(\Sigma)
\end{aligned}
$$

$\forall x \in \Phi, \forall y \in \Gamma$ and $\forall z \in \Sigma$.
Hence the functions $\tau_A(x) = \|A(\cdot, x)\|_p$, $\tau_B(x) = \|B(\cdot, x)\|_p$ and $\rho_A(y) = \|A(y, \cdot)\|_q$, $\rho_B(z) = \|B(z, \cdot)\|_q$ are well defined.
We also make the following assumption.

Assumption 5.2.1 *Functions A and B are such, that*

$$
\begin{aligned}
& \int_\Phi A(y,x) f(x) dx \in L^p(\Gamma), \ \text{a.e. } y \in \Gamma \\
& \int_\Phi B(z,x) f(x) dx \in L^p(\Sigma), \ \text{a.e. } z \in \Sigma
\end{aligned}
$$

Later we will prove a lemma which, under some conditions, guarantees that the above assumptions are true, but for now we consider them as given.
Denote

$$\begin{aligned} X &:= L^p(\Phi), \\ C &:= L^p_+(\Phi) \\ Y &:= L^p(\Gamma) \times L^p(\Sigma), \\ K &:= L^p_-(\Gamma) \times \{0\}\,. \end{aligned}$$

In this case, since $1 < p < +\infty$ we have

$$\begin{aligned} X^* &:= L^q(\Phi), \\ C^* &:= L^q_+(\Phi) \\ Y^* &:= L^q(\Gamma) \times L^q(\Sigma), \\ K^* &:= L^q_-(\Gamma) \times L^q(\Sigma). \end{aligned}$$

and $X^{**} = X, \quad Y^{**} = Y, \quad C^{**} = C, \quad K^{**} = K.$
With these notations we can write the problem (P^L) as

$$\min_{f \in C} \langle -c, f \rangle \quad \text{subject to} \quad \mathcal{A}f + h \in K, \tag{5.20}$$

where the linear operator $\mathcal{A} : X \to Y$ is defined as

$$\mathcal{A}f(y,z) = \begin{pmatrix} \int_\Phi A(y,x)f(x)dx \\ \int_\Phi B(z,x)f(x)dx \end{pmatrix}$$

and h is defined as

$$h(y,z) = \begin{pmatrix} -a(y) \\ -b(z) \end{pmatrix}.$$

Construct the Lagrangian of the problem (P^L)

$$L(f, \lambda^*) = -\langle c, f \rangle + \langle \lambda^*, \mathcal{A}f + h \rangle, \tag{5.21}$$

where $\lambda^* \in Y$, i.e. it has the following form

$$\lambda^* = \begin{pmatrix} g^* \\ s^* \end{pmatrix}, \quad g^* \in L^q(\Gamma), \quad s^* \in L^q(\Sigma).$$

The Lagrangian function can thus be written as

$$\begin{aligned} L(f, g^*, s^*) &= -\int_\Phi c(x)f(x)\,\mathrm{d}\,x \\ &+ \int_\Gamma \left(\int_\Phi A(y,x)f(x)\,\mathrm{d}\,x - a(y) \right) g^*(y)\,\mathrm{d}\,y \\ &+ \int_\Sigma \left(\int_\Phi B(z,x)f(x)\,\mathrm{d}\,x - b(z) \right) s^*(z)\,\mathrm{d}\,z. \end{aligned} \tag{5.22}$$

The Lagrangian primal problem is

$$\min_{f\in C}\sup_{\lambda^*\in -K^*} L(f,\lambda^*) \tag{5.23}$$

Interchanging the min and max operators, we obtain the dual Lagrangian problem

$$\sup_{\lambda^*\in -K^*}\min_{f\in C} L(f,\lambda^*). \tag{5.24}$$

In order to evaluate (5.24) we change the order of integration in (5.22). We have

$$L(f,g^*,s^*) = -\langle a,g^*\rangle - \langle b,s^*\rangle + \langle \phi, f\rangle, \tag{5.25}$$

where

$$\phi(x) = \int_\Gamma A(y,x)g^*(y)\,\mathrm{d}\,y + \int_\Sigma B(z,x)s^*(z)\,\mathrm{d}\,z - c(x) \tag{5.26}$$

Thus,

$$\min_{f\in C} L(f,g^*,s^*) = \begin{cases} -\langle a,g^*\rangle - \langle b,s^*\rangle, & \text{if } \phi(x)\geq 0, (\text{ a.e. } x\in\phi), \\ -\infty, & \text{otherwise} \end{cases} \tag{5.27}$$

This leads to the equivalence of the Lagrangian dual problem to

$$\begin{aligned} (D^L) \quad \inf_{g\in L^q(\Gamma), s\in L^q(\Sigma)} & \int_\Gamma a(y)g(y)dy + \int_\Sigma b(z)s(z)\,\mathrm{d}\,z \\ & \int_\Gamma A(y,x)g(y)dy + \int_\Sigma B(z,x)s(z)\,\mathrm{d}\,z \geq c(x), \text{ a.e. } x\in\Phi \\ & g\geq 0. \end{aligned}$$

Definition 5.2.2 *We say that Slater condition holds for the problem* (P^L) *if* $\exists \bar{f}\in L^p(\Phi)$ *such that*

$$\begin{aligned} \int_\Phi A(y,x)\bar{f}(x)dx < a(y), &\quad a.e. \quad y\in\Gamma \\ \int_\Phi B(z,x)\bar{f}(x)dx = b(z), &\quad a.e. \quad z\in\Sigma \end{aligned}$$

and the function B *is such that* $\mathcal{B}(X) = L^p(\Sigma)$*, where*

$$\mathcal{B}(f) := \int_\Phi B(z,x)f(x)dx.$$

We are now ready to use the results from the previous section to prove strong duality in our case. Note that all the assumption in Section 5.1 are satisfied, hence we get the following theorem.

Theorem 5.2.3 (Strong Duality for L^p Problems) *Let the linear operators $\mathcal{A}$ and $\langle c, \cdot\rangle$ be continuous. Suppose that Assumption (5.2.1) holds. Then, if the Slater condition holds for problem (P^L), we have $Val(P^L) = Val(D^L)$ and $Sol(D^L)$ is bounded.*

Now we prove a lemma which ensures that Assumption 5.2.1 holds and also ensures the required continuity of the linear operators under some conditions:

Lemma 5.2.4 *Suppose $A : \Gamma \times \Phi \to \mathbb{R}$ is such that $A(y, \cdot) \in L^q(\Phi)$ a.e. y and $A(\cdot, x) \in L^p(\Gamma)$ a.e. x and for some $1 \le p, q \le \infty$ such, that $1/p + 1/q = 1$. Further suppose that the functions $\tau(x) := \|A(\cdot, x)\|_p$ and $\rho(y) := \|A(y, \cdot)\|_q$ are in $L^q(\Phi)$ and $L^p(\Gamma)$ respectively, and finally, that $\rho(y)$ is uniformly bounded, i.e. $\exists M$ such that $\rho(y) \le M$ a.e. $y \in \Gamma$.*
Then the linear operator $\mathcal{A}$ defined below is continuous, and its image is in $L^p(\Gamma)$:

$$\mathcal{A}(f) := \int_\Phi A(y, x) f(x) dx. \tag{5.28}$$

Proof. We have

$$\begin{aligned}
& \int_\Gamma \left| \int_\Phi A(y,x) f(x)\,\mathrm{d}\,x \right|^p \mathrm{d}\,y \\
\le \quad & \int_\Gamma \left(\int_\Phi |A(y,x)|\,|f(x)|\,\mathrm{d}\,x \right)^p \mathrm{d}\,y & (5.29)\\
\le \quad & \left(\int_\Phi \left(\int_\Gamma |A(y,x)|^p\,|f(x)|^p\,\mathrm{d}\,y \right)^{1/p} \mathrm{d}\,x \right)^p & (5.30)\\
= \quad & \left(\int_\Phi |f(x)| \left(\int_\Gamma |A(y,x)|^p\,\mathrm{d}\,y \right)^{1/p} \mathrm{d}\,x \right)^p \\
= \quad & \left(\int_\Phi |f(x)|\,\tau(x)\,\mathrm{d}\,x \right)^p \le \infty,
\end{aligned}$$

where (5.30) follows from (5.29) by Minkowski's inequality.
Now we prove continuity of $\mathcal{A}$.
For any $\epsilon \ge 0$ take $\delta = \epsilon/M$. Then for any two functions $f_1, f_2 \in L^p(\Phi)$ such that $\|f_1 - f_2\|_p \le \delta$ we have

$$\begin{aligned}
& \left\| \int_\Phi A(y,x) f_1(x)\,\mathrm{d}\,x - \int_{\mathbb{R}^n} A(y,x) f_2(x)\,\mathrm{d}\,x \right\|_p = \\
& \left\| \int_\Phi A(y,x)\,(f_1(x) - f_2(x))\,\mathrm{d}\,x \right\|_p \le \\
& \|f_1 - f_2\|_p\,\rho(y) \le \epsilon.
\end{aligned}$$

The last step follows from Hölder's inequality. □

5.3 Strong Duality for Semi-Infinite Problems

In this section we discuss the strong duality of semi-infinite programming problems, where the integrand functions are piecewise continuous functions.
Consider the following optimisation problem over spaces of measures:

$$\begin{aligned}
\text{(P)} \qquad & \sup_{\mathcal{F} \in \mathscr{M}_{\mathfrak{F}}} \int_{\Phi} h(x) \, \mathrm{d}\,\mathcal{F}(x) \\
& \text{s.t.} \int_{\Phi} \phi_s(x) \, \mathrm{d}\,\mathcal{F}(x) \le a_s, \quad (s = 1, \ldots, M), \\
& \text{s.t.} \int_{\Phi} \psi_t(x) \, \mathrm{d}\,\mathcal{F}(x) = b_t, \quad (t = 1, \ldots, N), \\
& \quad \mathcal{F} \ge 0,
\end{aligned}$$

and its dual problem

$$\begin{aligned}
\text{(D)} \qquad & \inf_{(y,z) \in \mathbb{R}^{M+N+1}} \sum_{s=1}^{M} a_s y_s + \sum_{t=1}^{N} b_t z_t, \\
& \text{s.t.} \sum_{s=1}^{M} y_s \phi_s(x) + \sum_{t=1}^{N} z_t \psi_t(x) - h(x) \ge 0, \quad (x \in \Phi), \qquad (5.31) \\
& y \ge 0,
\end{aligned}$$

where $(\Phi, \mathfrak{F})$ is a complete measure space. Let $\mathscr{M}_{\mathfrak{F}}$ be the set of signed measures with finite variation on $(\Phi, \mathfrak{F})$.
Shapiro, [66], proves that strong duality holds (i.e., $Val(P) = Val(D)$) when Φ is compact and the functions $h(x)$, $\phi_s(x)$ and $\psi_t(x)$ are continuous. In this section we extend this result to the case where these functions are piecewise continuous on the partitioning of Φ into boxes $\Phi = \bigcup_{i=1}^{k} \mathbb{B}_i$, $\cap_{i=1}^{k} \mathbb{B}_i = \emptyset$. Each box $\mathbb{B}_i \subset \mathbb{R}^n$ has the following form :

$$\mathbb{B}_i = \left\{ x \in \mathbb{R}^n : l_j^i \le x_j < u_j^i \right\}.$$

Suppose that each of the functions $h(x)$, $\phi_s(x)$ and $\psi_t(x)$ is continuous on $\mathbb{B}_k$, $\forall k \le K$.
We take a similar approach as in [66].
Note that each box in $\mathbb{R}^n$ can be linearly transformed into a unit box in $\mathbb{R}^n$, so that

with the new transformed variables the optimisation problem becomes

$$
\begin{aligned}
\text{(P')} \quad & \sup_{\mathcal{F}\in\mathscr{M}_{\mathfrak{F}}} \sum_{i=1}^{K} \int_{\mathbb{B}} h^i(x)\,\mathrm{d}\,\mathcal{F}^i(x) \\
& \text{s.t.} \ \sum_{i=1}^{K} \int_{\mathbb{B}} \phi_s^i(x)\,\mathrm{d}\,\mathcal{F}_i(x) \le a_s, \quad (s=1,\ldots,M), \\
& \text{s.t.} \ \sum_{i=1}^{K} \int_{\mathbb{B}} \psi_t^i(x)\,\mathrm{d}\,\mathcal{F}_i(x) = b_t, \quad (t=1,\ldots,N), \\
& \mathcal{F}_i \ge 0, \quad i=1,\ldots,K
\end{aligned}
$$

with the new dual (which, obviously, is equivalent to the original dual problem)

$$
\begin{aligned}
\text{(D')} \quad & \inf_{(y,z)\in\mathbb{R}^{M+N}} \sum_{s=1}^{M} a_s y_s + \sum_{t=1}^{N} b_t z_t \\
& \text{s.t.} \ \sum_{s=1}^{M} y_s \phi_s^i(x) + \sum_{t=1}^{N} z_t \psi_t^i(x) - h^i(x) \ge 0, \quad (x\in\mathbb{B}, \quad i=1,\ldots,K), \\
& y \ge 0,
\end{aligned} \tag{5.32}
$$

where $\phi_s^i(x) = \phi_s|_{\mathbb{B}_i}(T_i^-(x))$ and $T_i : \mathbb{B}_i \to \mathbb{B}$ is the linear map of the transformation (T_i^- being it's inverse map).
Let $X := \mathbb{R}^{M+N}$ and $Y := \times_{i=1}^{K}\mathcal{C}(\mathbb{B})$, where $\mathcal{C}(\mathbb{B})$ is the set of continuous functions on $\mathbb{B}$. Then the problem (D') can be written as

$$
\begin{aligned}
& \inf_{(y,z)\in\mathbb{R}_+^M\times\mathbb{R}^N} \sum_{s=1}^{M} a_s y_s + \sum_{t=1}^{N} b_t z_t \\
& \text{s.t.} \ \mathcal{A}\cdot(y,z) + b \in \mathcal{K},
\end{aligned} \tag{5.33}
$$

where $\mathcal{A} : X \to Y$ is defined as

$$
\mathcal{A}\cdot(y,z) = \left(\tau_1(y,z,x),\ldots,\tau_K(y,z,x)\right), \tag{5.34}
$$

$\tau_i(y,z,x) = \sum_{s=1}^{M} y_s\phi_s^i(x) + \sum_{t=1}^{N} z_t\psi_t^i(x)$ and b is defined as vector

$$
b = \left(h^i(x),\ldots,h^K(x)\right). \tag{5.35}
$$

For cone $\mathcal{K}$ we have $\mathcal{K} = \times_{i=1}^{K}\mathcal{C}_+(\mathbb{B})$, where $\mathcal{C}_+(\mathbb{B})$ is the set of non-negative continuous functions on $\mathbb{B}$.
The dual space Y^* of Y is $\times_{i=1}^{K}\mathcal{M}$, where $\mathcal{M}$ is the set of finite signed Borel measures

on $\mathbb{B}$. By equipping Y and Y^* with strong and weak topologies and defining the scalar product between Y and Y^* as

$$\langle \phi, \mu \rangle := \sum_{i=1}^{K} \int_{\mathbb{B}} \phi_i(x) \,\mathrm{d}\, \mu_i(x), \tag{5.36}$$

we obtain a pair of locally convex topological vector spaces.
It is easy to see that the Lagrangian dual of problem (5.33) coincides with problem (P'). The following proposition is a direct result of Proposition 5.1.7.

Proposition 5.3.1 *Suppose that the optimal value of the problem* (D) *is finite. If there exists* $(\bar{y}, \bar{z}) \in \mathbb{R}^M \times \mathbb{R}^N$ *such that*

$$\sum_{s=1}^{M} \bar{y}_s \phi_s(x) + \sum_{t=1}^{N} \bar{z}_t \psi_t(x) - h(x) > 0, \quad \forall x \in \Phi, \tag{5.37}$$

then $Val(P) = Val(D)$ *and the set of optimal solution of the problem* (P) *is bounded.*

The Slater condition described in (5.37) is of particular interest for us since we use the model with piecewise linear test functions in our application for robust risk aggregation and for option pricing in later chapters. Given a particular problem, one can confirm the existence of such $(\bar{y}, \bar{z}) \in \mathbb{R}^M \times \mathbb{R}^N$ by solving the optimisation problem with a objective integrand $h + \epsilon$ for some small ϵ. This has been the case for both the risk management and the option pricing problems.

Chapter 6

Numerical Methods for Solving the Model

Recall the model introduced in Section 4.4. Let Φ be decomposed into a partition $\Phi = \bigcup_{i=1}^{k} \Xi_i$ of polyhedra Ξ_i with nonempty interior. Each polyhedron has a primal description in terms of generators

$$\Xi_i = \mathrm{conv}(q_1^i, \ldots, q_{n_i}^i) + \mathrm{cone}(r_1^i, \ldots, r_{o_i}^i)$$

where $\mathrm{conv}(q_1^i, \ldots, q_{n_i}^i)$ is the polytope with vertices $q_n^i \in \mathbb{R}^n$, and

$$\mathrm{cone}(r_1^i, \ldots, r_{o_i}^i) = \left\{ \sum_{m=1}^{o_i} \xi_m r_m^i : \xi_m \geq 0 \ (m \in \mathbb{N}_{o_i}) \right\}$$

is the polyhedral cone with recession directions $r_m^i \in \mathbb{R}^n$. Each polyhedron also has a dual description in terms of linear inequalities,

$$\Xi_i = \bigcap_{j=1}^{k_i} \left\{ x \in \mathbb{R}^n : \langle f_j^i, x \rangle \geq \ell_j^i \right\},$$

for some vectors $f_j^i \in \mathbb{R}^n$ and bounds $\ell_j^i \in \mathbb{R}$. Consider the problem

$$\begin{aligned}
\sup_{\mathcal{F} \in \mathscr{M}_{\mathfrak{F}}} \ & \int_{\Phi} h(x) \, \mathrm{d}\,\mathcal{F}(x) \\
\text{s.t.} \ & \int_{\Phi} \phi_s(x) \, \mathrm{d}\,\mathcal{F}(x) \leq a_s, \quad (s = 1, \ldots, M), \\
\text{s.t.} \ & \int_{\Phi} \psi_t(x) \, \mathrm{d}\,\mathcal{F}(x) = b_t, \quad (t = 1, \ldots, N), \\
& \int_{\Phi} 1 \, \mathrm{d}\,\mathcal{F}(x) = 1, \\
& \mathcal{F} \geq 0,
\end{aligned} \tag{6.1}$$

where the test functions ψ_t are piecewise linear on the partition $\Phi = \bigcup_{i=1}^k \Xi_i$, and where $-h(x)$ and the test functions ϕ_s are piecewise linear on the infinite polyhedra of the partition, and either jointly linear, concave, or convex on the finite polyhedra (i.e., polytopes) of the partition. As described in Section 4.4, the dual problem to (6.1) is

$$\begin{aligned} \inf_{(y,z)\in\mathbb{R}^{M+N+1}} & \sum_{s=1}^{M} a_s y_s + \sum_{t=1}^{N} b_t z_t + z_0, \\ \text{s.t.} & \sum_{s=1}^{M} y_s \phi_s(x) + \sum_{t=1}^{N} z_t \psi_t(x) + z_0\, 1_\Phi(x) - h(x) \geq 0, \\ & \quad (x \in \Xi_i), \quad (i = 1, \dots, k). \\ & y \geq 0. \end{aligned} \tag{6.2}$$

Note that the constraint in the dual problem (6.2) is an infinite dimensional constraint. In this section we present how these constraints can be relaxed to finitely many constraints for different types of integrand functions ϕ_s, ψ_t and h, $s = 1, \dots, M$, $t = 1, \dots, N$.

6.1 Piecewise Linear Test Functions

The first case we discuss is when $\phi_s|_{\Xi_i}$ and $h|_{\Xi_i}$ are jointly linear. Since we furthermore assumed that the functions $\psi_t|_{\Xi_i}$ are linear, there exist vectors $v_s^i \in \mathbb{R}^n$, $w_t^i \in \mathbb{R}^n$, $g^i \in \mathbb{R}^n$ and constants $c_s^i \in \mathbb{R}$, $d_t^i \in \mathbb{R}$ and $e^i \in \mathbb{R}$ such that

$$\begin{aligned} \phi_s|_{\Xi_i}(x) &= \langle v_s^i, x\rangle + c_s^i, \\ \psi_t|_{\Xi_i}(x) &= \langle w_t^i, x\rangle + d_t^i, \\ h|_{\Xi_i}(x) &= \langle g^i, x\rangle + e^i. \end{aligned}$$

The copositivity condition

$$\sum_{s=1}^{M} y_s \phi_s(x) + \sum_{t=1}^{N} z_t \psi_t(x) + z_0\, 1_\Phi(x) - h(x) \geq 0, \quad (x \in \Xi_i)$$

can then be written as

$$\begin{aligned} & \langle f_j^i, x\rangle \geq \ell_j^i, \quad (j = 1, \dots, k_i) \Longrightarrow \\ & \qquad \left\langle \sum_{s=1}^{M} y_s v_s^i + \sum_{t=1}^{N} z_t w_t^i - g^i \;,\; x \right\rangle \geq e^i - \sum_{s=1}^{M} y_s c_s^i - \sum_{t=1}^{N} z_t d_t^i - z_0. \end{aligned}$$

By Farkas' Lemma, this is equivalent to the constraints

$$\begin{aligned} \sum_{s=1}^{M} y_s v_s^i + \sum_{t=1}^{N} z_t w_t^i - g^i &= \sum_{j=1}^{k_i} \lambda_j^i f_j^i, \\ e^i - \sum_{s=1}^{M} y_s c_s^i - \sum_{t=1}^{N} z_t d_t^i - z_0 &\leq \sum_{j=1}^{k_i} \lambda_j^i \ell_j^i, \\ \lambda_j^i &\geq 0, \quad (j = 1, \dots, k_i), \end{aligned} \tag{6.3}$$

where λ_j^i are additional auxiliary decision variables.

Thus, if all test functions are linear on all polyhedral pieces Ξ_i, then the dual (6.3) can be solved as a linear programming problem with $M + N + 1 + \sum_{i=1}^{k} k_i$ variables and $K(n+1)$ linear constraints, plus bound constraints on y and the λ_j^i. K is the number of polyhedra in the model. More generally, if some but not all polyhedra correspond to jointly linear test function pieces, then jointly linear pieces can be treated as discussed above, while other pieces can be treated as discussed below.

Note that although in the general setup the number of variables λ_i^j increases exponentially, the constraint matrix is very sparse and optimisation techniques on sparse linear problems can be efficient for this problem.

As will be described later in Chapters 7 and 8, in all our applications we have polyhedrons of the following form

$$\Xi_i = \bigcap_{j=1}^{k_i} \left\{ x \in \mathbb{R}^n : \underline{\xi}_j^i \leq x_j \leq \bar{\xi}_j^i \right\},$$

i.e. Ξ_i is a cube in $\mathbb{R}^n$. In this case $k_i = 2n$. For $j = 1, \dots, n$, f_{2j-1}^i is a vector of zeros with an only 1 at the coordinate j and $f_{2j}^i = -f_{2j-1}^i$. For $j = 1, \dots, n$, $l_{2j-1}^i = \underline{\xi}_j^i$ and $l_{2j}^i = -\bar{\xi}_j^i$.

Hence we get the following constraints for the polyhedron Ξ_i

$$\left(\sum_{s=1}^{M} y_s v_s^i + \sum_{t=1}^{N} z_t w_t^i - g^i \right)_j = \lambda_{2j-1}^i - \lambda_{2j}^i, \quad j = 1, \dots, n \tag{6.4}$$

$$e^i - \sum_{s=1}^{M} y_s c_s^i - \sum_{t=1}^{N} z_t d_t^i - z_0 \leq \sum_{j=1}^{n} \left(\lambda_{2j-1}^i \underline{\xi}_j^i - \lambda_{2j}^i \bar{\xi}_j^i \right) \tag{6.5}$$

$$\lambda_j^i \geq 0, \quad (j = 1, \dots, 2n),$$

where the notation in the left hand side of (6.4) is used to indicate the jth coordinate of the vector. Denoting $\tau_j^i := \left(\sum_{s=1}^{M} y_s v_s^i + \sum_{t=1}^{N} z_t w_t^i - g^i \right)_j$ we have

$$\lambda_{2j}^i = \lambda_{2j-1}^i - \tau_j^i.$$

Now the right hand side of (6.5) reads as

$$\sum_{j=1}^{n}\left(\lambda^i_{2j-1}\underline{\xi}^i_j-(\lambda^i_{2j-1}-\tau^i_j)\bar{\xi}^i_j\right) \tag{6.6}$$

$$=\sum_{j=1}^{n}\left(\lambda^i_{2j-1}(\underline{\xi}^i_j-\bar{\xi}^i_j)+\tau^i_j\bar{\xi}^i_j\right) \tag{6.7}$$

$$=\sum_{j=1}^{n}\tau^i_j\bar{\xi}^i_j+\sum_{j=1}^{n}\lambda^i_{2j-1}(\underline{\xi}^i_j-\bar{\xi}^i_j). \tag{6.8}$$

So we get the following constraint for Ξ_i

$$\begin{aligned}
&e^i-\sum_{s=1}^{M}y_sc^i_s-\sum_{t=1}^{N}z_td^i_t-z_0\\
&\le\sum_{j=1}^{n}\left(\sum_{s=1}^{M}y_sv^i_s+\sum_{t=1}^{N}z_tw^i_t-g^i\right)_j\bar{\xi}^i_j+\sum_{j=1}^{n}\lambda^i_{2j-1}(\underline{\xi}^i_j-\bar{\xi}^i_j)\\
&=\sum_{s=1}^{M}y_s\sum_{j=1}^{n}v^i_{sj}\bar{\xi}^i_j+\sum_{t=1}^{N}z_t\sum_{j=1}^{n}w^i_{tj}\bar{\xi}^i_j-\sum_{j=1}^{n}g^i_j\bar{\xi}^i_j+\sum_{j=1}^{n}\lambda^i_{2j-1}(\underline{\xi}^i_j-\bar{\xi}^i_j).
\end{aligned}$$

We get

$$\begin{aligned}
&e^i-\sum_{s=1}^{M}y_s(c^i_s+\sum_{j=1}^{n}v^i_{sj}\bar{\xi}^i_j)-\sum_{t=1}^{N}z_t(d^i_t+\sum_{j=1}^{n}w^i_{tj}\bar{\xi}^i_j)-z_0\\
&\le-\sum_{j=1}^{n}g^i_j\bar{\xi}^i_j+\sum_{j=1}^{n}\lambda^i_{2j-1}(\underline{\xi}^i_j-\bar{\xi}^i_j).
\end{aligned}$$

Now we can eliminate the decision variables λ^i_{2j}, $j=1,\dots,n$ by changing constraint (6.4) to

$$\lambda^i_{2j-1}\ge\left(\sum_{s=1}^{M}y_sv^i_s+\sum_{t=1}^{N}z_tw^i_t-g^i\right)_j,$$

which is implied by the positivity constraints $\lambda^i_{2j}\ge 0$, $j=1,\dots,n$.
Hence we get the following optimisation problem

$$\begin{aligned}
\inf_{(y,z,\lambda)\in\mathbb{R}^{M+N+nK+1}}\ &\sum_{s=1}^{M}a_sy_s+\sum_{t=1}^{N}b_tz_t+z_0,\\
\text{s.t. }&\sum_{s=1}^{M}y_sv^i_s+\sum_{t=1}^{N}z_tw^i_t-\lambda^i\le g^i,\quad (x\in\Xi_i),\quad (i=1,\dots,K).\\
&-\sum_{s=1}^{M}y_s\bar{v}^i_s-\sum_{t=1}^{N}z_t\bar{w}^i_t-z_0-\sum_{j=1}^{n}\lambda^i_j(\underline{\xi}^i_j-\bar{\xi}^i_j)\le\bar{g}^i,\\
&y\ge 0,\quad \lambda^i\ge 0,\quad i=1,\dots,K,
\end{aligned} \tag{6.9}$$

where for $i = 1, \ldots, K$

$$\bar{v}_s^i := c_s^i + \sum_{j=1}^{n} v_{sj}^i \bar{\xi}_j^i, \quad s = 1, \ldots, M$$

$$\bar{w}_t^i := d_t^i + \sum_{j=1}^{n} w_{tj}^i \bar{\xi}_j^i, \quad t = 1, \ldots, N$$

$$\bar{g}^i := -e^i - \sum_{j=1}^{n} g_j^i \bar{\xi}_j^i.$$

6.1.1 The Delayed Column Generation Method

In the previous section we considered piecewise linear integrand functions and illustrated that the infinite dimensional optimisation problem (6.2) is equivalent to the standard linear programming problem (6.9). In many applications, particularly in risk management (Chapter 7), we use piecewise constant test functions in our model. For these particular examples, the delayed column generation algorithm for the simplex method is very useful. In this section we will describe the algorithm in detail.

Suppose the test functions that we consider are piecewise constant. In this setup we have $v_s^i = 0$, $w_t^i = 0$ and $g^i = 0$, for $i = 1, \ldots, K$, $s = 1, \ldots, M$, $t = 1, \ldots, N$. Then the dual optimisation problem (6.9) becomes

$$\begin{aligned} \inf_{(y,z,\lambda) \in \mathbb{R}^{M+N+nK+1}} & \sum_{s=1}^{M} a_s y_s + \sum_{t=1}^{N} b_t z_t + z_0, \\ \text{s.t.} \quad & -\sum_{s=1}^{M} y_s c_s^i - \sum_{t=1}^{N} z_t d_t^i - z_0 \leq -e^i - \sum_{j=1}^{n} \lambda_j^i (\underline{\xi}_j^i - \bar{\xi}_j^i), \\ & y \geq 0, \quad \lambda^i \geq 0, \quad i = 1, \ldots, K, \end{aligned} \tag{6.10}$$

Since $\underline{\xi}_j^i < \bar{\xi}_j^i$ and the decision variables λ_j^i have zero coefficient in the objective function, we need to make the right hand side of the constraint of (6.10) as large as possible, hence the optimal values of λ_j^i are zeros. The optimisation problem becomes

$$\begin{aligned} \inf_{(y,z) \in \mathbb{R}^{M+N+1}} & \sum_{s=1}^{M} a_s y_s + \sum_{t=1}^{N} b_t z_t + z_0, \\ \text{s.t.} \quad & -\sum_{s=1}^{M} y_s c_s^i - \sum_{t=1}^{N} z_t d_t^i - z_0 \leq -e^i, \quad (i = 1, \ldots, K) \\ & y \geq 0. \end{aligned} \tag{6.11}$$

Writing the dual to the problem (6.11) we get

$$
\begin{aligned}
&\sup_{x\in\mathbb{R}^K} \sum_{i=1}^{K} x_i e^i, \\
&\text{s.t. } \sum_{i=1}^{K} x_i c_s^i \le a_s, \quad (s = 1, \dots, M) \\
&\sum_{i=1}^{K} x_i d_t^i = b_t, \quad (t = 1, \dots, N) \\
&\sum_{i=1}^{K} x_i = 1, \\
&x \ge 0.
\end{aligned}
\tag{6.12}
$$

Note that the optimisation problem above has K decision variables, which, in many applications, is very large (Chapters 7 and 8). For this reason we implement the delayed column generation algorithm for simplex methods. The idea behind this method is that one doesn't need to construct the whole constraint matrix and start simplex algorithm iterations, but can add columns to it during the run.

First we will briefly describe the simplex algorithm and then the delayed column generation methodology. For a more detailed discussion of the simplex method and the delayed column generation algorithm, see [12].

We consider the linear optimisation problem to be in the standard form for the sake of simplicity in describing the algorithm. In the end of this section we will comment on how to bring the original problem to the standard form (Remark 6.1.3).

Suppose we aim to solve the following linear optimisation problem

$$
\begin{aligned}
&\max_x \langle c, x\rangle \\
&\text{s.t. } Ax \le b,
\end{aligned}
$$

where $c \in \mathbb{R}^n$, $b \in \mathbb{R}^m$ and A is a $m \times n$ matrix. Since we assumed to have the problem in standard form, $b > 0$. This means that $x = 0$ is a feasible solution. The following algorithm describes the simplex method

Algorithm 6.1.1 (The Simplex Method)

1. *Initialise* $C^* = 0$.

2. *While* $\exists i$ *s.t.* $c_i > 0$ *do*

- *Find a pivoting column i^*, i.e. find i s.t. $c_i > 0$. There are different methods of choosing a pivoting column, for example $i^* = \min\{i : c_i > 0\}$ (this guarantees final number of iterations) or $i^* = \arg\max_i\{c_i\}$.*
- *Find the pivoting row, i.e. find $j^* := \arg\min_j\{\frac{b_j}{A_{ji^*}} : A_{ji^*} > 0\}$. If the set $\{b_j/A_{ji^*} : A_{ji^*} > 0\}$ is empty, then the problem is unbounded.*
- *Update the optimal value $C^* = C^* + c_{i^*} * \frac{b_{j*}}{A_{j^*i^*}}$.*
- *Divide b_{j^*} and the row j^* of A on A_{ji^*}.*
- *For each row $j \neq j^*$ assign $A_j := A_j - A_{ji^*}A_{j^*}$.*
- *Assign $c := c - c_{i^*}A_{j^*}$.*

3. *Return C^*.*

Note that in the standard simplex method one doesn't need to keep all the columns of the constraint matrix all the time. In particular, we can add new columns to the matrix when there are no more positive entries in the objective. This method is known as the delayed column generation algorithm and is described in the following algorithm.

Algorithm 6.1.2 (The Delayed Column Generation Algorithm)

1. *Initialise a set $C^* := 0$, $V := \emptyset$ (set of columns to consider later). Also set $\bar{A} = \emptyset$ and $k := 1$. Initialise a Boolean parameter $columnsAdded :=$ true.*
2. *While $k \leq n$ or ($V \neq \emptyset$ and $columnsAdded$)*
   (a) *If $k \leq n$ then set $l := k$ and increment k. Otherwise, if $V \neq \emptyset$ and and $k = n+1$ set $l := V_0$ and remove l from V.*
   (b) *Find the column $\bar{A}_l$ and $\bar{c}_l$ that correspond to $\bar{A}$. These are calculated from the column A_l and c_l given the current tableau $\bar{A}$ and $\bar{c}$.*
   (c) *If $\bar{c}_l \leq 0$, add the index l to the set V, set $columnsAdded :=$ true and go to point (a). Otherwise, add $\bar{A}_l$ to $\bar{A}$ and $\bar{c}_l$ to $\bar{c}$.*
   (d) *Run simplex iterations on the problem*

$$\max_x \langle \bar{c}, x\rangle$$
$$s.t.\ \bar{A}x \leq \bar{b},$$

   with the standard simplex method and add its optimal value to C^*. $\bar{b}$ is b modified with respect to $\bar{A}$.

3. Return C^*.

Note that the columns of the constraint matrix correspond to the boxes of model, i.e. every time we want to add a column we consider a new box. In the following experiments, computations were performed on a 2.5 GHz Macintosh computer with 4Gb DDR memory.
We compare the performance of the standard and delayed column generation simplex methods on the marginal problem described in Section 4.1. Tables 6.1, 6.2 and 6.3 illustrate the performance of the DCG algorithm compared to the standard method. One can observe huge memory advantage in the DCG method, though we gain in calculation time as well. As matrix size we report the number of columns in the constraint matrix.
Note that for discretisation size 8 and dimensions higher than 6 the standard method requires large memory and hence starts using the HDD drive. The later makes the performance very poor and impractical, so we don't report any numbers for the standard method. In Table 6.4 we report numbers for dimensions from 2 to 9 for the marginal problem (8 discretisation points) calculated with the delayed column generation algorithm. The memory advantage is huge, especially in high dimensions.

Remark 6.1.3 *Recall that before describing the simplex method we assumed to have the linear programming problem in standard form, i.e. $b > 0$. This ensures that 0 is a feasible solution. In general setup, we need to find an initial feasible tableau. In order to get an initial feasible solution and a tableau we solve the following problem*

$$\begin{aligned}&\max_x -x_0\\ &s.t.\ A^i x - x_0 \leq b_i, \quad i = 1, \dots, m,\end{aligned}$$

where A^i is the i row of the constraint matrix. Pivot the last column $n+1$ at the first iteration and the row i with the smallest negative value b_i and continue the standard simplex iterations. If the optimal solution $x_0^* = 0$, then we have a feasible initial tableau. Otherwise, the original problem has no feasible solution. This procedure is the phase 1 of the two-phased simplex method. In the second stage we solve the original problem with the feasible dictionary obtained at phase 1. See [12] for more details.

	Standard		DCG		Relative Difference	
Dimension	Matrix Size	Time	Matrix Size	Time	Matrix Size	Time
2	80	0.006	28	0.01	0.3500	1.0900
3	536	0.045	48	0.03	0.0900	0.7530
4	4128	0.861	129	0.30	0.0310	0.3500
5	32808	14.031	292	3.13	0.0090	0.2230
6	262192	159.596	469	21.07	0.0020	0.1320

Table 6.1: Comparison of DCG algorithm with standard simplex method for different dimensions and 8 discretisation points.

	Standard		DCG		Relative Difference	
Dimension	Matrix Size	Time	Matrix Size	Time	Matrix Size	Time
2	288	0.014619	52	0.02	0.1810	1.3970
3	4144	0.256135	73	0.40	0.0180	1.5760
4	65600	11.9915	264	8.99	0.0040	0.7500
5	1048656	444.168	601	124.98	0.0010	0.2810

Table 6.2: Comparison of DCG algorithm with standard simplex method for different dimensions and 16 discretisation points.

	Standard		DCG		Relative Difference	
Dimension	Matrix Size	Time	Matrix Size	Time	Matrix Size	Time
2	1088	0.055871	100	0.13	0.0920	2.2940
3	32864	3.22363	142	5.49	0.0040	1.7020
4	1048704	299.166	544	177.81	0.0010	0.5940

Table 6.3: Comparison of DCG algorithm with standard simplex method for different dimensions and 32 discretisation points.

Dimension	DCG Matrix Size	Standard Matrix Size	Time (seconds)
2	28	64	0.01
3	50	512	0.03
4	146	4,096	0.30
5	342	32,768	3.13
6	465	262,144	21.07
7	1023	2,097,152	211.38
8	941	16,777,216	1895.56
9	3574	134,217,728	21323.50

Table 6.4: DCG Algorithm for different dimensions and 8 discretisation points for the marginal problem.

6.1.2 The Delayed Column Generation Algorithm for the Marginal Problem

Here we describe how the delayed column generation algorithm can be developed further to solve the marginal problem. Recall step (b) of the Algorithm 6.1.2. Given the current tableau $\bar{A}$ and the objective $\bar{c}$ we need to find new columns $\bar{A}_l$ and $\bar{c}_l$, which correspond to the current tableau as though they had been included in the constraint matrix from the start of the algorithm but never used as pivot columns. It is straightforward to see that the new entry of the objective $\bar{c}_l$ has this form

$$\bar{c}_l = c_l + \sum_{j=1}^{m} \bar{c}_{i_j} A_{lj},$$

for some $i_1, \ldots, i_m$, which are known at the beginning of the algorithm (these are usually the indices of the slack variables if we had only inequalities in the constraint). Without loss of generality we can suppose $i_j = j$ so we get

$$\bar{c}_l = c_l + \sum_{j=1}^{m} \bar{c}_j A_{lj}, \tag{6.13}$$

For the general problems we have to potentially consider all the unchecked boxes before we find one with $\bar{c}_l > 0$. In our implementation we took the overly conservative approach of scanning all the boxes whose corresponding columns were not included in the tableau and continue taking simplex pivots. Checking the non-included boxes consumes the majority of the computation time when most of the boxes do not produce a new column to add. To speed up the algorithm further, we need an efficient way to find a box which will produce a positive objective entry. This can be done for the marginal problem.

Suppose we want to solve the marginal problem in n dimensions and d discretisation points. Then we have nd test functions. In this setup each box is uniquely represented with n indices $i_1, \ldots, i_n$ such that $1 \le i_j \le d$, $j = 1, \ldots, n$. The column of the constraint matrix representing this box will consist of n blocks with d entries each. Block j will contain all 0s except of the entry i_j (which is 1). Hence, for the additional objective entry we have

$$\bar{c}_l = c_l + \sum_{j=1}^{m} \bar{c}^i_{i_j}, \tag{6.14}$$

where $\bar{c}^i_{i_j} = \bar{c}_{(i-1)d+i_j}$ and

$$c_l = 1_{\sum_{i=1}^{n} \bar{\xi}^i_{i_j} > S}, \tag{6.15}$$

where $\bar{\xi}^i_{i_j}$ is the discretisation point of the box i_j of the dimension i as defined earlier in this Section. S is the risk threshold.
Now we need to solve the following integer programming problem every time we need to find an additional column to add to the current tableau:

$$\max_{1\le i_1,\dots i_n\le d} 1_{\sum_{i=1}^n \bar{\xi}^i_{i_j}>S} + \sum_{j=1}^m \bar{c}^i_{i_j} \tag{6.16}$$

Note that for the boxes where the objective function is 0 the problem becomes

$$\max_{1\le i_1,\dots i_n\le d} \sum_{j=1}^m \bar{c}^i_{i_j}, \tag{6.17}$$

which is straightforward to solve by taking the maximum value from each block $\bar{c}^i$, $i = 1, \dots, n$. Hence, even if solving the problem (6.16) is time-expensive, with this method we don't have to go through all the boxes that have objective value 0. This saves a considerable amount of computation time, especially if we need to find VaR for very large thresholds (i.e. the number of boxes that have non-zero objective is a small fraction of the overall number of boxes). Moreover, it can be seen from Table 6.4 that upon increasing the dimension of the problem by 1, the computation time grows by a factor roughly equal to the discretisation size. This suggests that the algorithm spends most of the time considering all the boxes and not on the iterations of the simplex method itself. Therefore, solving the subproblem (6.16) instead, even if only approximately, offers a considerable time saving potential.
Note that this method can also be applied to more general cases of our model, i.e. when we have additional constraints to the marginal problem. For example, if we have a constraint that at some box the probability is fixed at some level or a constraint on tail probabilities of joint events (as described in Section 7.2), then we will have additional indicator function over a rectangular in (6.16), which itself does not lead to more numerical complexity.

6.2 Piecewise Convex Test Functions

When $\phi_s|_{\Xi_i}$ and $-h|_{\Xi_i}$ are jointly convex, then $\varphi_{y,z}(x)$ is convex. The copositivity constraint

$$\sum_{s=1}^M y_s\phi_s(x) + \sum_{t=1}^N z_t\psi_t(x) + z_0\, 1_\Phi(x) - h(x) \ge 0, \quad (x \in \Xi_i)$$

can then be written as

$$\langle f_j^i, x\rangle \geq \ell_j^i, \quad (j = 1, \ldots, k_i) \Longrightarrow \varphi_{y,z}(x) \geq 0,$$

and by Farkas' Theorem (Theorem 2.1, [57]), this condition is equivalent to

$$\begin{aligned} \varphi_{y,z}(x) + \sum_{j=1}^{k_i} \lambda_j^i \left(\ell_j^i - \langle f_j^i, x\rangle\right) \geq 0, \quad (x \in \mathbb{R}^n), \\ \lambda_j^i \geq 0, \quad (j = 1, \ldots, k_i), \end{aligned} \tag{6.18}$$

where λ_j^i are once again auxiliary decision variables. While (6.18) does not reduce to finitely many constraints, the validity of this condition can be checked numerically by globally minimizing the convex function $\varphi_{y,z}(x) + \sum_{j=1}^{k_i} \lambda_j^i \left(\ell_j^i - \langle f_j^i, x\rangle\right)$. The constraint (6.18) can then be enforced explicitly if a line-search method is used to solve the dual (D).

6.3 Piecewise Concave Test Functions

When $\phi_s|_{\Xi_i}$ and $-h|_{\Xi_i}$ are jointly concave but not linear, then $\varphi_{y,z}(x)$ is concave and $\Xi_i = \operatorname{conv}(q_1^i, \ldots, q_{n_i}^i)$ is a polytope. The copositivity constraint

$$\sum_{s=1}^{M} y_s \phi_s(x) + \sum_{t=1}^{N} z_t \psi_t(x) + z_0\, 1_\Phi(x) - h(x) \geq 0, \quad (x \in \Xi_i) \tag{6.19}$$

can then be written as

$$\varphi_{y,z}(q_j^i) \geq 0, \quad (j = 1, \ldots, n_i).$$

Thus, (6.19) can be replaced by n_i linear inequality constraints on the decision variables y_s and z_t.

6.4 Piecewise Polynomial Test Functions

Another case that can be treated via finitely many constraints is when $\phi_s|_{\Xi_i}$, $\psi_t|_{\Xi_i}$ and $h|_{\Xi_i}$ are jointly polynomial. The approach of Lasserre [43] and Parrilo [54] can be applied to turn the copositivity constraint

$$\langle f_j^i, x\rangle \geq \ell_j^i, \quad (j = 1, \ldots, k_i) \Longrightarrow \varphi_{y,z}(x) \geq 0$$

into finitely many linear matrix inequalities. However, this approach is generally limited to low-dimensional applications.

6.5 Nesterov's Smoothing Technique for Solving High-Dimensional Linear Programming Problems

In the previous chapters we came across large scale optimisation problems. Although for piecewise constant test functions the delayed column generation algorithm works efficiently, for the piecewise concave case we have to use simplex or interior point methods to solve these problems. As we discussed, the size of the linear programming problems increases exponentially with the dimension of the original problems.
Let n be the dimension of the linear programming problem that we want to solve. Although interior point methods usually make a small number of iterations, they require $O(n^2)$ memory and $O(n^3)$ operations on each iteration, which makes them practically useless on large scales.
For general non-smooth optimisation problems, the number of iterations required to achieve a tolerance level ϵ is of order $\frac{1}{\epsilon^2}$, whereas for smooth problems there are well known schemes with complexity $\frac{1}{\sqrt{(\epsilon)}}$.
In [52] Nesterov describes an algorithm for non-smooth optimisation for class of functions which takes $\frac{1}{\epsilon}$ iterations. Note that this number can be significantly larger compared to interior point methods, but on each iteration the algorithm makes only $O(n\log(n))$ operations and requires $O(n)$ computer memory. Thus, for large scale problems, the large number of iterations but the small number of operations in them outperforms interior point methods.
In this chapter we first describe Nesterov's algorithm, then show how we can use it for our problems.

6.5.1 Smooth Approximation of Non-Smooth Functions

In this section we describe a method for smooth approximation of a non-smooth function which has a specific form.
Suppose we aim to solve the following problem

$$\min_x \{f(x) : x \in Q_1\}, \tag{6.20}$$

where Q_1 is a bounded closed convex set in a finite-dimensional real vector space E_1 and $f(x)$ is a continuous convex function on Q_1. f is not necessarily differentiable.
By smoothening we mean to find a function $\bar{f}$ such that for any prior chosen $\epsilon > 0$

$$\bar{f}(x) \leq f(x) \leq \bar{f}(x) + \epsilon, \quad \forall x. \tag{6.21}$$

If we are able to find such function $\bar{f}$, then, by minimising $\bar{f}$ (we aim to construct an approximation which is easier to optimise), we will get an approximate minimiser for the original function f.

Suppose f has the following structure

$$f(x) = \hat{f}(x) + \max_u \left\{ \langle Ax, u \rangle - \hat{\phi}(u) : u \in Q_2 \right\}, \tag{6.22}$$

where $\hat{f}(x)$ is a differentiable function, $\bar{\phi}(u)$ is a continuous convex function on Q_2 and $A : E_1 \to E_2^*$ is a linear operator.

Let $d_2(u)$ be a prox-function of the set Q_2, i.e. $d_2(u)$ is continuous and strongly convex on Q_2 with some convexity parameter $\sigma > 0$. Let

$$u_0 = \arg\min_u \{d_2(u) : u \in Q_2\}. \tag{6.23}$$

Without loss of generality we can assume that $d_2(u_0) = 0$. Hence we have

$$d_2(u) \geq \frac{1}{2}\sigma_2 \|u - u_0\|^2. \tag{6.24}$$

For a given parameter $\mu > 0$ consider the following function

$$f_\mu(x) = \max_u \left\{ \langle Ax, u \rangle - \hat{\phi}(u) - \mu d_2(u) : u \in Q_2 \right\}. \tag{6.25}$$

Denote $u_\mu(x)$ the optimal solution of (6.25), which is unique due to strong convexity of $d_2(u)$.

Theorem 6.5.1 (Theorem 1, [52]) *The function $f_\mu(x)$ is well defined and continuously differentiable at any $x \in E_1$. Moreover, this function is convex and its gradient*

$$\bigtriangledown f_\mu(x) = A^* u_\mu(x) \tag{6.26}$$

is Lipschitz continuous with constant

$$L_\mu = \frac{1}{\mu\sigma_2}\|A\|^2. \tag{6.27}$$

Proof of the theorem can be found in [52].

Let $D_2 := \max_u \{d_2(u) : u \in Q_2\}$. Then, for every $x \in E_1$ we have

$$f_\mu(x) \leq f_0(x) \leq f_\mu(x) + \mu D_2, \tag{6.28}$$

hence

$$\hat{f}(x) + f_\mu(x) \leq f(x) \leq \hat{f}(x) + f_\mu(x) + \mu D_2, \tag{6.29}$$

which means $\hat{f}(x) + f_\mu(x)$ is a smooth approximation of our original objective function $f(x)$.

In the following sections we will describe an optimal method for optimising a differentiable function and then implement it for our problems of interest.

6.5.2 An Optimal Scheme for Optimisation of Differentiable Functions

Let E be a finite-dimensional real vector space and $Q \subseteq E$ be a closed convex set. Suppose $f(x)$ is a differentiable function on Q with Lipschitz continuous gradient

$$\|f(y) - f(x)\| \leq L\|y - x\|, \quad \forall x, y \in Q. \tag{6.30}$$

From Lipschitz continuity it follows that

$$f(y) \leq f(x) + \langle \triangledown f(x), y - x \rangle + \frac{L}{2}\|y - x\|^2. \tag{6.31}$$

Let $T_Q(x)$ be the optimal solution of the following problem

$$\min_y \left\{ \langle \triangledown f(x), y - x \rangle + \frac{L}{2}\|y - x\|^2 : y \in Q \right\}. \tag{6.32}$$

If the problem doesn't have a unique optimal solution (when the norm is not strictly convex function), we take as $T_Q(x)$ any of the optimal solutions.
For each $x \in Q$ we have

$$f(T_Q(x)) \leq f(x) + \min_y \left\{ \langle \triangledown f(x), y - x \rangle + \frac{L}{2}\|y - x\|^2 : y \in Q \right\}. \tag{6.33}$$

As in the previous section, we suppose to have a prox-function of the set Q with convexity parameter σ.
In order to solve the optimization problem

$$\min_x \{f(x) : x \in Q\}, \tag{6.34}$$

we recursively construct two sequences of points $\{x_k\}_{k\geq 0} \subset Q$ and $\{y_k\}_{k\geq 0} \subset Q$ such that

$$A_k \leq \min_x \left\{ \frac{L}{\sigma} d(x) + \sum_{i=0}^{k} \alpha_i \left[f(x_i) + \langle \triangledown f(x_i), x - x_i \rangle \right] : x \in Q \right\}, \tag{6.35}$$

where $\{\alpha_i\}_{i\geq 0}$ are positive and $A_k = \sum_{i=0}^{k} \alpha_i$.
Denote

$$z_k = \arg\min_x \left\{ \frac{L}{\sigma} d(x) + \sum_{i=0}^{k} \alpha_i \left[f(x_i) + \langle \triangledown f(x_i), x - x_i \rangle \right] : x \in Q \right\}. \tag{6.36}$$

Lemma 1 from [52] states that if $\alpha_0 \in (0, 1]$ and $\alpha_k^2 \leq A_k, k > 0$, then, constructing points y_k and x_k as

$$\begin{aligned} x_k &= \frac{\alpha_k}{A_k} z_{k-1} + (1 - \frac{\alpha_k}{A_k}) y_{k-1} \\ y_k &= T_Q(x_k), \end{aligned}$$

we are guaranteed to satisfy the inequality 6.35. This can be proved by induction.
One way to satisfy the condition $\alpha_k^2 \leq A_k$ is to choose $\alpha_k = \frac{k+1}{2}$, since

$$\alpha_k^2 = \frac{(k+1)^2}{4} < \frac{(k+1)(k+2)}{4} = A_k. \tag{6.37}$$

Now consider the following algorithm

Algorithm 6.5.2 (Minimisation of a Smooth Function)

- *Set* $k = 0$ *and a starting point* x_0.

- *While not converged*

  1. *Compute* $f(x_k)$ *and* $\bigtriangledown f(x_k)$.
  2. *Find* $y_k = T_Q(x_k)$.
  3. *Find* $z_k = \arg\min_x \left\{ \frac{L}{\sigma} d(x) + \sum_{i=0}^{k} \alpha_i \left[f(x_i) + \langle \bigtriangledown f(x_i), x - x_i \rangle \right] : x \in Q \right\}$.
  4. *Set* $x_{k+1} = \frac{2}{k} z_k + \frac{k+1}{k+3} y_k$.
  5. *Increase* k *by* 1.

Theorem 6.5.3 (Theorem 2, [52]) *Sequence* $\{y_k\}_{k \geq 0}$ *generated by the Algorithm 6.5.2 satisfy*

$$f(y_k) - f(x^*) \leq \frac{4Ld(x^*)}{\sigma(k+1)(k+2)}, \tag{6.38}$$

where x^* *is the optimal solution of the original problem (6.34).*

The later inequality shows that the number of required iterations of the algorithm to achieve ϵ precision is

$$N \sim \sqrt{\frac{L}{\epsilon}}. \tag{6.39}$$

This rate of convergence is optimal since it represents the lower complexity bound for this problem.
We will analyse the algorithm step-by-step in the following section, in which it is implemented for linear programming problems resulting within our models.

6.5.3 Application to Linear Programming Problems

In this section we will discuss a special type of linear programming problem which can be solved with Nesterov's smoothing technique. Particularly, we are interested in problems where optimisation is required on a simplex, i.e. on the convex and closed set $\Delta_n := \{x \in \mathbb{R}^n : \sum_{i=1}^n x_i = 1, x_i \geq 0, i = 1, \dots, n\}$. This type of problem arises for optimisation problems over probability measures.
Suppose we have the following linear optimisation problem

$$\begin{aligned} \min_{x\in\Delta_n} \quad & \langle c, x\rangle \\ \text{s.t.} \quad & Ax \leq b, \end{aligned} \tag{6.40}$$

where $b \in \mathbb{R}^m$ for some m and A is a linear map $A : \mathbb{R}^n \to \mathbb{R}^m$. Note that in the case where we have equality constraints, they can be transformed into inequality constraints by adding additional variables. In the remainder of the section we assume this problem has an optimal solution.
The constraint (6.40) is equivalent to

$$\langle a_j, x\rangle - b_j \leq 0, \quad j = 1, \dots, m, \tag{6.41}$$

where b_j is the j-th element of b and a_j is the j-th row vector of the matrix A.
Consider the following function

$$f(t; x) = \max\left\{\langle c, x\rangle - t; \langle a_j, x\rangle - b_j, j = 1, \dots, m\right\}, \quad t \in \mathbb{R}^1, x \in \Delta_n. \tag{6.42}$$

Denote

$$f^*(t) = \min_{x\in\Delta_n} f(t; x). \tag{6.43}$$

Since the original LP problem has a solution, the problem above also has an optimal solution.

Lemma 6.5.4 *Let t^* be an optimal value of problem (6.40). Then*

$$\begin{aligned} f^*(t) &\leq 0, \text{ for all } t \geq t^*, \\ f^*(t) &> 0, \text{ for all } t < t^*. \end{aligned}$$

Proof. Let x^* be a solution to (6.40) (if the optimal solution of the problem is not unique, then we take any of them). If $t \geq t^*$, then

$$\begin{aligned} f^*(t) &\leq f(t; x^*) = \max\left\{\langle c, x^*\rangle - t; \langle a_j, x^*\rangle - b_j, j = 1, \dots, m\right\} \\ &\leq \max\left\{t^* - t; \langle a_j, x^*\rangle - b_j, j = 1, \dots, m\right\} \leq 0. \end{aligned}$$

Further, suppose $t < t^*$ and $f^*(t) \leq 0$. Then, there exists $y \in \Delta_n$ such that

$$\langle c, y\rangle \leq t < t^*, \quad \langle a_j, y\rangle - b_j \leq 0, \quad j = 1, \ldots, m,$$

hence, t^* cannot be an optimal value of (6.40). $\square$

Lemma 6.5.4 implies that the smallest root of the function $f^*(t)$ will be the optimal value of the original linear programming problem. Although there are many algorithms for finding the root of the function $f^*(t)$, in many applications we require only the information of whether the optimal value is larger or smaller than a given threshold. For this purpose, solving the problem (6.43) for a fixed threshold is the only task required, since the sign of the optimal value will provide the required information.

Following the discussions above, we are interested in solving the following optimisation problem

$$\min_{x \in \Delta_n} \max \left\{ \langle c, x\rangle - t; \langle a_j, x\rangle - b_j, j = 1, \ldots, m \right\}, \tag{6.44}$$

for a given $t \in \mathbb{R}$.

Note that if we add the vector c as a new row vector into matrix A, and the scalar t as a new entry into b (at the same position where c was inserted into A), then the problem becomes

$$\min_{x \in \Delta_n} \max_{0 \leq j \leq m} \left\{ \langle \bar{a}_j, x\rangle - \bar{b}_j \right\}, \tag{6.45}$$

where $\bar{a}_0 = c, \bar{a}_j = a_j, j = 1, \ldots m$ and $\bar{b}_0 = t, \bar{b}_j = b_j, j = 1, \ldots, m$.

Note that the problem $\max_{0 \leq j \leq m} \left\{ \langle \bar{a}_j, x\rangle - \bar{b}_j \right\}$ is equivalent to

$$\max_{u \in \Delta_{m+1}} \left\{ \bar{A}x, u\rangle - \langle \bar{b}, u\rangle \right\}. \tag{6.46}$$

So that we get the following optimisation problem

$$\min_{x \in \Delta_n} \max_{u \in \Delta_{m+1}} \left\{ \bar{A}x, u\rangle - \langle \bar{b}, u\rangle \right\}. \tag{6.47}$$

Note that the problem (6.48) has a structure (6.22), so we implement the smoothing technique described above.

After smoothing with parameter μ we obtain the following problem,

$$\min_{x \in \Delta_n} \max_{u \in \Delta_{m+1}} \left\{ \bar{A}x, u\rangle - \langle \bar{b}, u\rangle - \mu d_2(u) \right\}, \tag{6.48}$$

where as a prox-function for the set Δ_{m+1} we take

$$d_2(u) = \frac{1}{2}\|u - u_c\|^2, \tag{6.49}$$

with prox-center $u_c = \left(\frac{1}{m+1}, \dots, \frac{1}{m+1}\right) \in \Delta_{m+1}$.
Similarly, we take $x_c = \left(\frac{1}{n}, \dots, \frac{1}{n}\right) \in \Delta_n$ and

$$d_1(x) = \frac{1}{2}\|x - x_c\|^2. \tag{6.50}$$

So we have $\sigma_1 = \sigma_2 = 1$, $D_1 = 1 - \frac{1}{n}, D_2 = 1 - \frac{1}{m+1}$.
Since in our case $\hat{f}(x) = 0$, we have

$$L_\mu = \frac{1}{\mu}\|\bar{A}\|^2. \tag{6.51}$$

Theorem 6.5.5 *For a given $\epsilon > 0$, number of iterations in Algorithm 6.5.2 with smoothness parameter $\mu = \frac{\epsilon}{2D_2}$ to generate an ϵ-solution does not exceed*

$$N = 4\|\bar{A}\|\sqrt{D_1 D_2}\frac{1}{\epsilon}. \tag{6.52}$$

Proof. Proof follows directly from Theorem 3, [52]. □

6.5.4 Numerical Results

In this section we implement Nesterov's algorithm to compare it with interior point methods.
As was described in the beginning of this chapter, interior point methods perform only a very small number of iterations compared to Nesterov's algorithm, but the advantage of Nesterov's algorithm lies in the complexity of its iterations.
Suppose we are given a linear optimisation problem:

$$(LP) \quad \min_{x \in \Delta^n} \quad \langle c, x \rangle \tag{6.53}$$
$$\text{s.t.} \quad Ax \le b, \tag{6.54}$$

where $c \in \mathbb{R}^n$, $b \in \mathbb{R}^m$ and A is a $n \times m$ matrix. As before, we denote by Δ_n the simplex in $\mathbb{R}^n$. For a given p, we are interested in comparing $Val(LP)$ with p.
Note that, for high-dimensional problems, Nesterov's algorithm has two significant advantages:

1. **Memory:** Interior point methods use Hessian matrix in their calculations, which requires $O(n^2)$ memory. Compared to this, Nesterov's algorithm works with a gradient and hence requires only $O(n)$ memory. When the required memory is larger than the computer's primary memory (RAM), reading and writing from a secondary memory (such as hard drives) may take more computational time than a floating point operation.

2. **Number of operations per iterations:** As described at the beginning of this chapter, Interior point methods perform $O((n+m)^3)$ floating point operations per iteration. Compared to this, Nesterov's algorithm for linear optimisation problems of the form (LP) performs only $O(n \log n)$ operations (sorting of an array as described in Appendix A).

To illustrate the advantage of the Nesterov's method mentioned above, we solve the model-free option pricing problem for plain vanilla variance swap option described in Chapter 8.4.2 for dimension 4 and strike step 4 (for the notations of the dimension and the strike step see Section 8). The size of the resulting optimisation problem is 1152×160000 with matrix norm $4.33e+05$. For this particular problem, the SDPT3 solver fails to solve the problem, whereas the Nesterov's method, described earlier, solves it in $190,000$ iterations and 4 hours and 37 minutes.
Similar situation is observed for dimension 5 and strike step 8. The dimension of the optimisation problem is 1586×248832 and the norm of the matrix is $7.04e+05$. Nesterov's method solves the optimisation problem in $360,000$ iterations and 12 hours and 46 minutes.

Chapter 7

Applications in Risk Management

In this chapter we describe how we can use the framework proposed in Section 4.4 in risk management. We first present a general approach to the problem and then, in the beginning of Section 7.3 we describe in detail how our experiments has been set up.

Let $X := (X_1, \dots, X_n)$ be a random variable representing the loss vector.

Recall our model from Section 4.4:

$$\begin{aligned} \sup_{\mathcal{F} \in \mathscr{M}_{\mathfrak{F}}} & \int_\Phi h(x) \, \mathrm{d}\,\mathcal{F}(x) \\ \text{s.t.} & \int_\Phi \phi_s(x) \, \mathrm{d}\,\mathcal{F}(x) \le a_s, \quad (s = 1, \dots, M), \\ \text{s.t.} & \int_\Phi \psi_t(x) \, \mathrm{d}\,\mathcal{F}(x) = b_t, \quad (t = 1, \dots, N), \\ & \int_\Phi 1 \, \mathrm{d}\,\mathcal{F}(x) = 1, \\ & \mathcal{F} \ge 0. \end{aligned} \tag{7.1}$$

Similarly to the problem discussed in Chapter 3, we are interested in finding worst case distributions for different risk measures (i.e. distributions that maximise these risk measures). For example, for a given threshold S, when the risk measure is Value at Risk, we take

$$h(x) = 1_{\left\{\sum_{i=1}^n x_i \ge S\right\}} \tag{7.2}$$

and when the risk measure is Expected Shortfall, i.e., CVaR, we take

$$h(x) = 1_{\left\{\sum_{i=1}^n x_i \ge S\right\}} \sum_{i=1}^n x_i. \tag{7.3}$$

Note that in numerical applications we can restrict ourselves to the case of a finite domain $\Phi = [L_1, U_1] \times [L_2, U_2] \times [L_n, U_n]$. We do this for the sake of simplicity in

implementing the numerical models, but the model works for infinite domains as well (as discussed in Section 4.4). A particular way of constructing the domain will be presented later in this chapter.
In this case, when Φ is finite, it will be a union of polytopes $\Phi = (\bigcup_{i=1}^{K} \Pi_i)$. Each polytope Π_i is the convex hull of finitely many points $p_m^i \in \mathbb{R}^n$ (the vertices of Π_i)

$$\Pi_i = \mathrm{conv}(p_1^i, \ldots, p_{m_i}^i) = \left\{ \sum_{j=1}^{m_i} \xi_j p_j^i : \xi_j \geq 0 \ (j \in \mathbb{N}_{m_i}), \ \sum_{j=1}^{m_i} \xi_j = 1 \right\}.$$

In our numerical applications we take the polytopes to be boxes in R^n. Figure 7.1

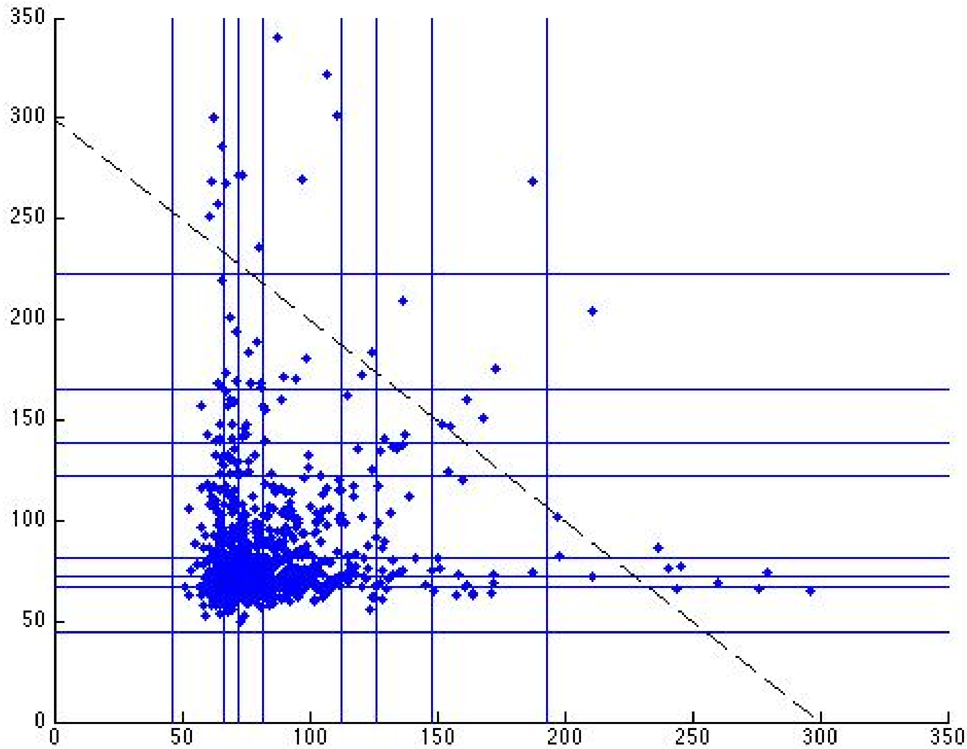

Figure 7.1: Simulated Losses for two Risk Cells.

demonstrates a typical domain Φ, its partitioning into boxes (each a rectangular cell) and the threshold line. For a given data set as pictured in the figure, we are interested in finding the maximum probability mass that the region above the threshold line can have.
Note that the objective function is not continuous on the boxes that are crossed by the threshold line (Figure 7.1), since it is 1 above the threshold and 0 below it. One

way of overcoming this issue is of course to divide the box into two smaller polytopes (triangles in the two-dimensional case). But in our numerical implementation we take another approach.
Suppose a box B is divided into two parts B_1 and B_2, and

$$\tau(x) = \begin{cases} \tau_1(x), & \text{if } x \in B_1 \\ \tau_2(x), & \text{if } x \in B_2. \end{cases} \tag{7.4}$$

In such a case we approximate the function $\tau(x)$ on the whole box $B = B_1 \cup B_2$ by the function

$$\bar{\tau}(x) = w_1\tau_1(x) + w_2\tau_2(x), \text{ for } x \in B, \tag{7.5}$$

where $w_i = \frac{V(B_i)}{V(B)}$. We denote the volume of a polytope Π by $V(\Pi)$.
Note that the same procedure can be applied if a box is divided into more than one polytope. Also note that, although this approach leads to an approximate solution of the optimisation problem, when the number of boxes increases, the error decreases and we can get closer to the optimal value with any accuracy. This procedure also renders the optimal value of problem (7.1) continuous with respect to the threshold.
In the following sections we will describe different methods for constructing constraint functions and analyse numerical results. We will provide different ways to impose restrictions on the joint distribution function, and one can choose to implement only part of the constraints or add new constraints with regard to his/her additional information about losses or his/her bias to a more or less conservative approach.
Throughout this chapter we use the bootstrapping method to get bounds on different quantities given the historical realisation of the random losses. Here we describe the bootstrapping method.
Suppose for a given empirical data $\bar{x} := (\bar{x}^1, \bar{x}^2, \ldots, \bar{x}^m)$, $\bar{x}^i \in \mathbb{R}^n$, we want to find bounds on some function $g : \mathbb{R}^m \times \mathbb{R}^n \to \mathbb{R}$ that depends on $\bar{x}$. Given an integer numbers K and t we do the following :

1. For $i = 1$ to K do
   - Generate t integer random numbers $j_1, \ldots, j_t$, $1 \leq j_l \leq m$, $l = 1, \ldots, t$.
   - Construct new data $\bar{y}_i := \{\bar{x}^{j_1}, \ldots, \bar{x}^{j_t}\}$.
   - Calculate $g_i = g(\bar{y}_i)$.
2. Return $\bar{g}_m$ and $\bar{g}_\sigma$, the mean and the standard deviation of $g_1, \ldots, g_K$ respectively.

Now we can choose the bounds on the function g for our original empirical data $\bar{x}$ to be $[\bar{g}_m - \alpha\bar{g}_\sigma, \bar{g}_m + \alpha\bar{g}_\sigma]$ for some $\alpha > 0$.
Note that the parameters t and K can be chosen depending on one's preference on how robust the model should be. In particular, one can chose these parameters such that, for a given α, the empirical probability

$$\bar{P}(g(\bar{x}) \in [\bar{g}_m - \alpha\bar{g}_\sigma, \bar{g}_m + \alpha\bar{g}_\sigma])$$

is around p for a specific probability p.
As it is described later in this chapter, while estimating bounds on the tail indices for marginal distributions calibrated with Extreme Value Theory, the function g represents the tail index calculated with Hill's method on empirical data. In the case of joint moment constraints, again described below, the function g is the empirical probability of joint loses exceeding a given threshold.

7.1 Marginal Constraints

One of the commonly occurring problems in risk management is the insufficient amount of historical data, which makes it very difficult to model joint distribution. In the previous chapters we considered problems (such as marginal problem), in which the knowledge of marginal distributions is assumed to be given. In this section we make the model more robust by including the uncertainty of the distribution function in the tails into our model.
In the following two parts of this section we will first introduce the estimation of the marginal distributions and then the construction of the constraints for our model. We use Extreme Value Theory (Section 2.2) to model the distributions, with some modifications.

7.1.1 Modelling Marginal Distributions

Suppose we are given a set of observed data points $\{\bar{z}_i\}_{i=1,\dots,M}$, $\bar{z}_i \in \mathbb{R}$ and want to find its cumulative distribution function F_z.
Recall from Section 2.2 that Extreme Value Theory can be used to model F_z:

$$F_z(z) = EVT[\bar{F}_z, u](z) = \begin{cases} \bar{F}_z(z) & z < u, \\ G_{\xi,\beta(u)}(z-u)(1-\bar{F}_z(u)) + \bar{F}_z(u) & z \geq u, \end{cases} \tag{7.6}$$

where u is the threshold, $G_{\xi,\beta(u)}$ is a Generalised Pareto Distribution, and $\bar{F}_z$ is the empirical distribution function, i.e.

$$\bar{F}_z(z) = \frac{1}{M}\sum_{i=1}^{M} 1_{\bar{z}^i<=z} \tag{7.7}$$

In this work we implement the Hill's estimator, i.e. instead of $G_{\xi,\beta(u)}(z-u)$ we use $H_{\xi,u}(z)$ (Section 7.2.4, [47]), where

$$H_{\xi,u}(z) = \int_u^z f_{\xi,u}(t)\,\mathrm{d}\,t = \int_u^z \frac{u^{\frac{1}{\xi}}}{\xi} t^{-(1+\frac{1}{\xi})}\,\mathrm{d}\,t = 1 - \left(\frac{u}{z}\right)^{\frac{1}{\xi}}, \qquad z \geq u, \tag{7.8}$$

and $H_{\xi,u}(z) = 0$, for $z < u$.
Given the empirical data, the parameter ξ can be estimated by the Hill's method, Section 7.2.4, [47]. As we know, the estimation of the tail parameter ξ is unstable, see e.g. [48, 53]. In order to add more robustness in the model, we don't restrict ourselves in assuming a fixed tail parameter, rather, we find lower and upper bounds on this parameter, with some confidence interval (the use of this will be described later). To do this we apply the bootstrapping method to find the lower and upper bounds of the tail index $\underline{\xi} := \xi_m - 2\sigma_\xi$ and $\bar{\xi} = \xi_m + 2\sigma_\xi$. In this particular case, the function g in the bootstrapping method is the function that returns the Hill's estimator given the empirical data.
In the following sections we will show how these bounds can be used to increase robustness in our model.

7.1.2 Construction of Marginal Constraints

In this subsection we describe in detail how to construct marginal constraints for our model.
We assume that we have already found marginal distribution functions $F_1(x_1), \ldots, F_n(x_n)$ and constructed the finite domain $\Phi = [L_1, U_1] \times [L_2, U_2] \times [L_n, U_n]$. We choose an integer number N and divide each of the segments $[L_i, U_i]$ into N small segments $[L_i =: x_i^0, x_i^1) \cup [x_i^1, x_i^2) \cup \cdots \cup [x_i^{N-1}, x_i^N := U_i]$ (note that we could choose different N for each dimension, but we chose to have a fixed number for simplicity). Now we implement the marginal constraints for this construction of boxes.
If one is confident that the estimated marginal distribution is good enough (depending on one's sensitivity to errors), then for each dimension $i = 1, \ldots, n$ one has N

marginal constraints of the form $\int_\Phi \psi_i^j(x)\,\mathrm{d}\,F(x) = b_i^j$, $j = 1,\ldots,N$, where

$$\psi_i^j(x) = \begin{cases} 1, & \text{if } x_i \in [x_i^{j-1}, x_i^j) \\ 0, & \text{otherwise} \end{cases} \tag{7.9}$$

and

$$b_i^j = F_i(x_i^j) - F_i(x_i^{j-1}). \tag{7.10}$$

Thus, there are nN marginal constraints overall.
We assume $F_i(x_i^N) = 1$ since we have a finite domain.
Now we go back again to the two dimensional case (Figure 7.1) for simplicity of graphical illustration. For each row and for each column we will have a constraint, for which the constraint functions are indicator functions that take value one when the point is in the respective row/column.
More in detail discussion how we construct the marginal constraints in our numerical experiments will be presented in the beginning of Section 7.3.

Remark 7.1.1 *Taking into account the instability of the EVT, one can make the model more robust by using the lower and upper bounds on the tail index described in Section 7.1.1.*
Suppose for a given marginal distribution function $F(z)$ we want to find $P(Z \in [l,h])$, where $u \le l < h$ and u is the threshold in (7.6). For a given tail parameter ξ we can find this probability analytically:

$$P_{\xi,u}(Z \in [l,h]) = \int_l^h f_{\xi,u}(t)\,\mathrm{d}\,t = \int_l^h \frac{u^{\frac{1}{\xi}}}{\xi} t^{-(1+\frac{1}{\xi})}\,\mathrm{d}\,t = \left(\frac{u}{l}\right)^{\frac{1}{\xi}} - \left(\frac{u}{h}\right)^{\frac{1}{\xi}}, \tag{7.11}$$

which is an increasing function with respect to ξ. Hence, when we have lower and upper bounds on the tail parameter $\underline{\xi}, \bar{\xi}$ we can write the constraint in the following form :

$$P_{\underline{\xi},u}(Z \in [l,h]) \le P(Z \in [l,h]) \le P_{\bar{\xi},u}(Z \in [l,h]), \tag{7.12}$$

i.e. we have two inequality constraints instead of one equality constraint.

7.2 Constraints for Joint Probabilities

The next type of constraint that we implement for our model is for joint probabilities. As with the previous constraints, we assign bounds to probabilities rather than fixed

values.
The second type of constraint is to assign joint probability bounds in the tails of the data. Let $i_1, \ldots, i_k \in \{1, \ldots, n\}$ be the set of indices for which we aim to implement joint probability constraints. For every $l = 1, \ldots, m-1$ we calculate bounds on probabilities obtained by empirical data, i.e. we aim to find bounds on the following integrals :

$$\int_{\mathbb{R}^n} 1_{\substack{x_{i_j} \geq \xi_{i_j}, \\ j=1,\ldots,k}} \, \mathrm{d}\, \mathcal{F}(x). \tag{7.13}$$

In order to get bounds on the integral (7.13) we again use the method of bootstrapping.

7.3 Numerical Results

In this section we perform numerical experiments on simulated data for different dimensions. For a given dimension we present the results of the marginal problem described in Section 3.1 and for our model with additional constraints, i.e. marginal constraints and the constraints for joint tail probabilities. We call the latter model MPCF (model with piecewise constant test functions).
We will thus produce one graph for each of the problems (for a fixed number of discretisation points) showing the sparsity pattern of the worst case distributions, in order to see the effect of the joint moment constraint of the model.
In this section we implement our model for three different types of simulated data. In the first two cases the marginal distributions are chosen as g-and-h distributions. These distributions are used as marginal distributions for operational losses in practice. In the last case we use one factor Gaussian Copula model. In the first part we introduce results for simulated data which has larger correlations for large losses and small correlations (in absolute value) for small losses. Motivation in constructing the input data in this way is presented at the beginning of the next subsection. In this subsection we will also present sparsity patterns of the worst case distribution obtained under out model or the marginal problem.
In our numerical experiments for finding the worst case probabilities of exceeding a given threshold S we construct a model described below. As before, let n be the dimension of the loss vector and N be the discretisation size. Suppose we are given empirical loss data $\bar{x} := (\bar{x}^1, \bar{x}^2, \ldots, \bar{x}^m)$, $\bar{x}^i \in \mathbb{R}^n$, $i = 1, \ldots, m$.

- Calibrate the marginal distributions. For the heavy-tailed distributions, we use the Extreme Value Theory, as described in 7.1.1, i.e. we find the thresholds u_i

and estimated mains and standard deviations of tail indices ξ^i_m and σ^i_m respectively for $i = 1, \ldots, n$. For the Gaussian Copula case the marginal distributions are standard normal distribution functions.

- Construct the domain $\Phi = [L_1, U_1] \times [L_2, U_2] \times [L_n, U_n]$ of the problem and it's discretisation. For the heavy-tailed distributions, the lower bounds of the domain $L_1, \ldots, L_n$ are chosen as $0.9 \min_{j=1,\ldots,m} \bar{x}^j_i$. Note that optimal solution of the optimisation problem is not sensitive to the choice of the lower bounds as long as they are smaller than the minimal value of the realised losses. For the upper bounds, we set $U_i = F_i^{-1}(1-q)$ for a given q, where F_i is the calibrated marginal distribution for dimension i. In our experiments we used $q = 0.0001$. Since we suppose the have finite domain and set $F_i(U_i) = 1$, $i = 1, \ldots, n$, the particular choice of q also does not matter as long as it is significantly smaller than the worst case probability we aim to find. For the Gaussian Copula case, we set $L_i = F_i^{-1}(q/2), U_i = F_i^{-1}(1-q/2)$.
  The discretisation points $\left\{L_i =: x^0_i < x^1_i < x^2_i < \cdots < x^N_i := U_i\right\}$ for a dimension i are calculated as
  $$x^j_i = F_i^{-1}\left(\frac{j}{N}\right), \quad j = 1, \ldots, N-1.$$

- Construct the marginal constraints. For heavy-tailed distributions, approximated by Extreme Value Theory, for each dimension i we create
  $k_i := \min\left\{j : x^j_i \geq u_i\right\} - 1$ equalities
  $$\int_{\mathbb{R}^n} 1_{x^j_i \leq x_i \leq x^{j+1}_i} \,\mathrm{d}\,\mathcal{F}(x) = \frac{1}{N},$$
  $$i = 1, \ldots, n; \quad j = 0, \ldots, k_i - 1$$
  and $2(N - k_i)$ inequalities
  $$\underline{p}^j_i \leq \int_{\mathbb{R}^n} 1_{x^j_i \leq x_i \leq x^{j+1}_i} \,\mathrm{d}\,\mathcal{F}(x) \leq \bar{p}^j_i,$$
  $$i = 1, \ldots, n; \quad j = k_i, \ldots N-1,$$
  where
  $$\begin{aligned} \underline{p}^j_i &:= H_{\underline{\xi}^i, u_i}(x^{j+1}_i) - H_{\underline{\xi}^i, u_i}(x^j_i) \\ \bar{p}^j_i &:= H_{\bar{\xi}^i, u_i}(x^{j+1}_i) - H_{\bar{\xi}^i, u_i}(x^j_i). \end{aligned}$$

$H_{\xi,u}$ is defined as in (7.8) and $\underline{\xi}^i := \xi^i_m - 2\sigma^i_m$, $\bar{\xi}^i := \xi^i_m + 2\sigma^i_m$.
For the Gaussian Copula case we have N equalities for each dimension i

$$\int_{\mathbb{R}^n} 1_{x_i^j \le x_i \le x_i^{j+1}} \,\mathrm{d}\,\mathcal{F}(x) = \frac{1}{N},$$
$$j = 0, \ldots, N-1.$$

- Construct the joint probability constraints for all pairs of losses (X_i, X_j), $1 \le i < j \le n$ as described in Section 7.2. For each pair (i, j) of losses we first estimate the probabilities $P(X_i \ge x_i^k, X_j \ge x_j^k)$, $k = N/2, \ldots, N-1$ with bootstrapping method described in the beginning of this section (we start with $k = N/2$ in our experiments, but the particular choice of this number also can vary since we are not interested in the lower part of the domain). Let p^k_{ij} and σ^k_{ij} be estimated probability and it's standard deviation. Then, we have the following inequalities

$$p^k_{ij} - 3\sigma^k_{ij} \le \int_{\mathbb{R}^n} 1_{x_i \ge x_i^k, x_j \ge x_j^k} \,\mathrm{d}\,\mathcal{F}(x) \le p^k_{ij} + 3\sigma^k_{ij},$$

  $k = N/2, \ldots, N-1$.

- Construct the joint probability constraints for the full vector of losses. This is similar to the previous step and the resulting inequalities are as the following

$$p^k_v - 3\sigma^k_v \le \int_{\mathbb{R}^n} 1_{x_1 \ge x_1^k, \ldots, x_n \ge x_n^k} \,\mathrm{d}\,\mathcal{F}(x) \le p^k_v + 3\sigma^k_v,$$

  where p^k_v and σ^k_v are estimated means and the standard deviations (again by bootstrapping method) of the probabilities $P(X_1 \ge x_1^k, \ldots, X_n \ge x_n^k)$, $k = N/2, \ldots, N-1$.

7.3.1 'Onion'-Shaped Distributions

The financial crisis of 2008 showed that the widely used Gaussian Copula models for financial losses do no capture the realised losses. Also for heavy-tailed distributions, that can produce rare but extreme losses, the correlation between them becomes stronger as the losses grow larger. For example, this is observed in operational losses and can be explained by the chain of losses that one business cell can trigger in an unprecedented event. A similar argument holds for credit default events as were observed during the financial crisis of 2008. Taking this into account, in this subsection we consider synthetically constructed data which has an 'onion'-shape, i.e. it has a

small correlation for small losses, and a large correlation for big ones.
We say that a random variable X has g-and-h distribution if

$$X = a + b\frac{\exp(gZ) - 1}{g}\exp(hZ^2/2), \tag{7.14}$$

where Z is a standard normal random variable.
To simulate the n-dimensional loss vector we do the following:

- Simulate g-and-h distributed Y, which we call the common factor.
- Simulate n independent g-and-h distributed $Y_1, \ldots, Y_n$.
- Set $X_i = F(Y)Y + (1 - F(Y))Y_i$, where $F : \mathbb{R} \to [0, 1]$ is an increasing function.

We simulate losses of size 1000 for different dimensions. The parameters of the g-and-h distributions are $A = 60, B = 10, h = 0.2$ and $g = 1.2$. The input data used in the numerical experiments in this subsection is presented in Appendix B.
Given these simulated losses, we calibrate the EVT distribution to this data. A typical calibrated tail index mean and standard deviations are

$$\xi_m = 0.39,$$
$$\sigma_\xi = 0.055.$$

Note that the estimated parameters are inline with the numerical estimations in [47], Section 7.2.
To illustrate an example of means and the standard deviations (StdDev) of joint tail probabilities, consider the two-dimensional case. The domain of the discretised problem is $[46.3, 1472.0] \times [44.4, 1715.8]$. In Table 7.1 we present the average values and the standard deviations (p_{12}^k and σ_{12}^k as defined above) of the probabilities for joint losses obtained by bootstrapping for two loss variables. Note that the standard deviation as a percentage of the mean increases as we go farther into the tails, as we expected. The discretisation size N is equal 16.
Now we compare the results of the marginal problem described in Section 4.1 and of our model MPCF. The results are presented in Tables 7.2 and 7.3. In separate columns we present the running times and the sizes of the memory used in computations. We can see that the maximum probability of the marginal problem is larger than the probability for MPCF. In Table 7.4 we also report probabilities obtained from Monte Carlo simulations, which, obviously, produce the smallest probabilities.

k	x_1^k	x_2^k	Mean	StdDev	StdDev/Mean (%)
9	77.9	77.4	0.047	0.00649	14%
10	81.7	81.7	0.034	0.00530	16%
11	88.3	89.6	0.036	0.00528	15%
12	96.9	99.4	0.028	0.00440	15%
13	112.6	121.1	0.023	0.00478	20%
14	126.0	138.4	0.011	0.00342	30%
15	147.6	165.0	0.005	0.00265	51%
16	193.4	222.7	0.002	0.00134	67%

Table 7.1: Average values and the standard deviations of the probabilities for joint losses obtained by bootstrapping for two-dimensional loss data simulated as 'onion'-shaped. The discretisation size is 16.

The Marginal Problem			
Dimension	Optimal Value	Memory Size	Time
2	0.0750	28	0.01
3	0.1042	50	0.03
4	0.1292	146	0.30
5	0.1315	342	3.13
6	0.1062	465	21.07
7	0.1216	1023	211.38
8	0.0902	941	1895.56
9	0.1352	3574	21323.50

Table 7.2: Worst case probabilities for the marginal problem for 'onion'-shaped synthetic data for different dimensions. The threshold for dimension k is $150k$.

MPCF			
Dimension	Optimal Value	Memory Size	Time
2	0.0580	72	0.22
3	0.0740	160	0.90
4	0.0919	508	10.01
5	0.0847	960	60.11
6	0.0619	2092	494.64
7	0.0898	4144	5833.77

Table 7.3: Worst case probabilities for MPCF for two-dimensional 'onion'-shaped synthetic data for different dimensions. The threshold for dimension k is $150k$.

Monte Carlo	
Dimension	Optimal Value
2	0.0359
3	0.0311
4	0.0289
5	0.0268
6	0.0274
7	0.0215

Table 7.4: Monte Carlo probabilities of exceeding a given threshold for 'onion'-shaped synthetic data for different dimensions. The threshold for dimension k is $150k$.

Remark 7.3.1 *Note that one of the concerns of practitioners to use the marginal problem for calculating the capital charge required by the regulators is that the joint distributions obtained by marginal problem do not have a realistic form. Figure 7.2 represents sparsity pattern of the marginal problem. One can see that the whole probability of exceeding the threshold is concentrated on the threshold line. Compared to the marginal problem, the model with additional constraints distributes the probability of exceeding the threshold more evenly. The sparsity pattern of an optimal solution to this model is presented in Figure 7.3. Note that the probabilities below the threshold can be redistributed over the rectangular in an evenly way and do not have to be on the diagonal line as presented in Figures 7.2 and 7.3. The discretisation size is* 64 *and the dimension is* 2.

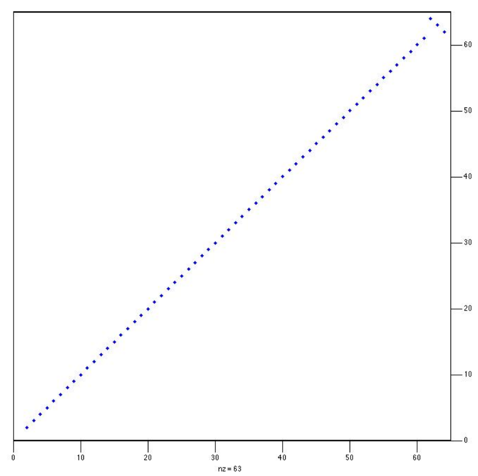

Figure 7.2: Sparsity pattern of the solution of the marginal problem for dimension 2. The discretisation size is 64.

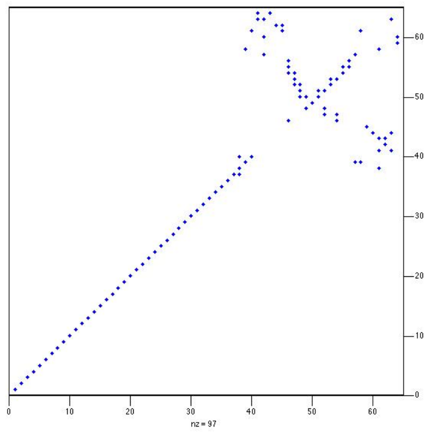

Figure 7.3: Sparsity pattern of the solution of MPCF model for dimension 2. The discretisation size is 64.

7.3.2 Heavy-Tailed Distributions with Constant Correlations

In this subsection we introduce numerical results for data which has been simulated similarly to the previous examples but with constant correlations. The n-dimensional loss vector with a constant correlation ρ is simulated with the following model:

- Simulate g-and-h distributed Y, which we call the common factor.
- Simulate n independent g-and-h distributed $Y_1, \ldots, Y_n$.
- Set $X_i = \sqrt{\rho}Y + \sqrt{1-\rho}Y_i$.

We simulate marginal distributions with g-and-h distribution with parameters $A = 60, B = 10, h = 0.2$ and $g = 1.2$. We use the same correlation for all losses and consider three cases : independent losses, 20% correlation and 40% correlation. We also consider different discretisation sizes, which is denoted by N in the tables below. We also report the probabilities obtained by Monte Carlo algorithm (10000 simulation size).

Tables 7.5-7.8 represent the worst case probabilities for MPCF and the marginal problem for 2 to 5 dimensional loss vectors respectively simulated with a constant correlation 0.

Tables 7.9-7.12 represent the worst case probabilities for MPCF and the marginal problem for 2 to 5 dimensional loss vectors respectively simulated with a constant correlation 20%.

Tables 7.13-7.16 represent the worst case probabilities for MPCF and the marginal problem for 2 to 5 dimensional loss vectors respectively simulated with a constant correlation 40%.

In Table 7.17 we report probabilities for dimension 6 for all 3 correlations $0, 20\%$ and 40%.

S / N	160		200		240		280		320	
	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal
8	0.400	0.400	0.242	0.250	0.097	0.100	0.059	0.075	0.029	0.050
16	0.312	0.314	0.145	0.151	0.085	0.100	0.044	0.075	0.030	0.050
32	0.303	0.314	0.141	0.179	0.084	0.100	0.041	0.075	0.030	0.050
64	0.324	0.335	0.144	0.178	0.085	0.100	0.044	0.075	0.030	0.050
128	0.333	0.347	0.141	0.176	0.084	0.100	0.040	0.075	0.030	0.050
256	0.341	0.353	0.141	0.175	0.084	0.100	0.040	0.075	0.030	0.050
512	0.343	0.354	0.141	0.175	0.084	0.100	0.045	0.075	0.029	0.050
Monte Carlo	0.179		0.086		0.0519		0.0334		0.023	

Table 7.5: Worst case probabilities for MPCF and the marginal problem for a 2-dimensional loss vector simulated with a constant correlation 0. S is the threshold and N is the discretisation size.

S / N	240		300		360		420		480	
	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal
8	0.400	0.400	0.188	0.188	0.091	0.122	0.055	0.088	0.046	0.056
16	0.382	0.403	0.174	0.202	0.094	0.132	0.051	0.088	0.041	0.056
32	0.377	0.424	0.168	0.219	0.090	0.138	0.050	0.075	0.040	0.056
64	0.396	0.450	0.169	0.228	0.090	0.141	0.048	0.075	0.039	0.056
Monte Carlo	0.1861		0.0844		0.0461		0.0293		0.0204	

Table 7.6: Worst case probabilities for MPCF and the marginal problem for a 3-dimensional loss vector simulated with a constant correlation 0. S is the threshold and N is the discretisation size.

S / N	320		400		480		560		640	
	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal
8	0.487	0.488	0.200	0.200	0.114	0.138	0.072	0.088	0.057	0.075
16	0.427	0.454	0.190	0.236	0.096	0.141	0.070	0.088	0.056	0.075
32	0.426	0.482	0.194	0.244	0.094	0.142	0.069	0.088	0.057	0.075
Monte Carlo	0.1914		0.0813		0.0425		0.0268		0.0173	

Table 7.7: Worst case probabilities for MPCF and the marginal problem for a 4-dimensional loss vector simulated with a constant correlation 0. S is the threshold and N is the discretisation size.

7.3.3 One Factor Gaussian Copula Model

In this section we present numerical results for data simulated with one factor Gaussian copula model with a single correlation. For a given correlation ρ, the data $X_1, \ldots, X_n$ is simulated as

- Simulate standard normal distributed Y, which, as before, we call the common factor.
- Simulate n independent standard normal distributed $Y_1, \ldots, Y_n$.
- Set $X_i = \sqrt{\rho}Y + \sqrt{1-\rho}Y_i$.

S / N	400		500		600		700		800	
	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal
8	0.498	0.500	0.250	0.250	0.128	0.135	0.087	0.104	0.072	0.083
16	0.458	0.482	0.199	0.241	0.107	0.136	0.081	0.104	0.068	0.083
Monte Carlo	0.1906		0.0749		0.0387		0.0233		0.016	

Table 7.8: Worst case probabilities for MPCF and the marginal problem for a 5-dimensional loss vector simulated with a constant correlation 0. S is the threshold and N is the discretisation size.

S / N	160		200		240		280		320	
	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal
8	0.858	0.875	0.409	0.425	0.272	0.275	0.125	0.125	0.099	0.100
16	0.998	1.000	0.386	0.426	0.227	0.250	0.143	0.151	0.099	0.100
32	0.965	1.000	0.405	0.458	0.227	0.237	0.147	0.150	0.098	0.100
64	0.944	1.000	0.416	0.478	0.228	0.253	0.148	0.150	0.097	0.100
128	0.941	1.000	0.418	0.488	0.236	0.263	0.147	0.154	0.099	0.100
256	0.938	1.000	0.418	0.490	0.240	0.265	0.148	0.156	0.099	0.100
512	0.940	1.000	0.421	0.492	0.240	0.269	0.147	0.155	0.098	0.100
Monte Carlo	0.7046		0.2534		0.1248		0.0728		0.0473	

Table 7.9: Worst case probabilities for MPCF and the marginal problem for a 2-dimensional loss vector simulated with a constant correlation 20%. S is the threshold and N is the discretisation size.

S / N	240		300		360		420		480	
	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal
8	1.000	1.000	0.584	0.600	0.305	0.313	0.154	0.156	0.108	0.117
16	1.000	1.000	0.480	0.527	0.276	0.288	0.171	0.170	0.117	0.125
32	1.000	1.000	0.519	0.603	0.282	0.309	0.173	0.187	0.115	0.126
64	1.000	1.000	0.545	0.640	0.297	0.328	0.177	0.193	0.114	0.127
Monte Carlo	0.7556		0.2622		0.122		0.068		0.0444	

Table 7.10: Worst case probabilities for MPCF and the marginal problem for a 3-dimensional loss vector simulated with a constant correlation 20%. S is the threshold and N is the discretisation size.

S / N	320		400		480		560		640	
	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal
8	1.000	1.000	0.624	0.641	0.361	0.363	0.169	0.175	0.116	0.117
16	1.000	1.000	0.559	0.629	0.306	0.328	0.181	0.196	0.115	0.125
32	1.000	1.000	0.603	0.685	0.316	0.349	0.186	0.208	0.114	0.127
Monte Carlo	0.7847		0.2697		0.1189		0.0663		0.0413	

Table 7.11: Worst case probabilities for MPCF and the marginal problem for a 4-dimensional loss vector simulated with a constant correlation 20%. S is the threshold and N is the discretisation size.

S / N	400		500		600		700		800	
	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal
8	1.000	1.000	0.662	0.684	0.357	0.367	0.201	0.204	0.130	0.131
16	1.000	1.000	0.619	0.682	0.322	0.349	0.182	0.201	0.125	0.132
Monte Carlo	0.81		0.2705		0.1158		0.0648		0.0388	

Table 7.12: Worst case probabilities for MPCF and the marginal problem for a 5-dimensional loss vector simulated with a constant correlation 20%. S is the threshold and N is the discretisation size.

S / N	160		200		240		280		320	
	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal
8	0.924	0.938	0.398	0.425	0.272	0.288	0.242	0.250	0.114	0.125
16	1.000	1.000	0.445	0.501	0.235	0.276	0.147	0.151	0.115	0.125
32	1.000	1.000	0.478	0.570	0.252	0.281	0.162	0.179	0.111	0.125
64	1.000	1.000	0.476	0.575	0.258	0.298	0.161	0.178	0.110	0.120
128	1.000	1.000	0.479	0.589	0.264	0.307	0.158	0.176	0.109	0.120
256	1.000	1.000	0.481	0.593	0.266	0.311	0.156	0.174	0.108	0.120
512	1.000	1.000	0.480	0.595	0.265	0.312	0.155	0.174	0.105	0.118
Monte Carlo	0.8415		0.3087		0.1482		0.0849		0.056	

Table 7.13: Worst case probabilities for MPCF and the marginal problem for a 2-dimensional loss vector simulated with a constant correlation 40%. S is the threshold and N is the discretisation size.

N \ S	240		300		360		420		480	
	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal
8	1.000	1.000	0.584	0.638	0.362	0.375	0.173	0.188	0.126	0.129
16	1.000	1.000	0.573	0.680	0.305	0.355	0.175	0.189	0.134	0.132
32	1.000	1.000	0.601	0.749	0.316	0.368	0.199	0.220	0.136	0.146
64	1.000	1.000	0.620	0.781	0.333	0.391	0.203	0.227	0.134	0.142
Monte Carlo	0.8822		0.3208		0.1454		0.0807		0.0534	

Table 7.14: Worst case probabilities for MPCF and the marginal problem for a 3-dimensional loss vector simulated with a constant correlation 40%. S is the threshold and N is the discretisation size.

N \ S	320		400		480		560		640	
	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal
8	1.000	1.000	0.658	0.700	0.375	0.400	0.176	0.200	0.136	0.142
16	1.000	1.000	0.645	0.771	0.349	0.391	0.210	0.232	0.137	0.150
32	1.000	1.000	0.695	0.839	0.364	0.417	0.222	0.245	0.140	0.152
Monte Carlo	0.905		0.332		0.1448		0.0798		0.0516	

Table 7.15: Worst case probabilities for MPCF and the marginal problem for a 4-dimensional loss vector simulated with a constant correlation 40%. S is the threshold and N is the discretisation size.

N \ S	400		500		600		700		800	
	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal
8	1.000	1.000	0.694	0.750	0.415	0.446	0.218	0.247	0.140	0.146
16	1.000	1.000	0.714	0.835	0.374	0.414	0.215	0.235	0.139	0.147
Monte Carlo	0.9209		0.334		0.1423		0.0764		0.0486	

Table 7.16: Worst case probabilities for MPCF and the marginal problem for a 5-dimensional loss vector simulated with a constant correlation 40%. S is the threshold and N is the discretisation size.

ρ \ S	600			720			840			960		
	MPCF	Marginal	MC	MPCF	Marginal	MC	MPCF	Marginal	MC	MPCF	Marginal	MC
0	0.256	0.256	0.070	0.133	0.150	0.035	0.099	0.116	0.035	0.083	0.100	0.013
0.2	0.681	0.697	0.272	0.355	0.360	0.113	0.213	0.225	0.113	0.139	0.141	0.037
0.4	0.729	0.793	0.335	0.417	0.444	0.138	0.253	0.264	0.138	0.152	0.163	0.047

Table 7.17: Worst case probabilities for MPCF and the marginal problem for a 6-dimensional loss vector simulated with correlations $0, 20\%$ and 40%. We denote the threshold by S and the correlation by ρ.

Tables 7.18-7.21 represent the worst case probabilities for MPCF and the marginal problem for 2 to 5 dimensional loss vectors respectively simulated with a constant correlation 0.
Tables 7.22-7.25 represent the worst case probabilities for MPCF and the marginal problem for 2 to 5 dimensional loss vectors respectively simulated with a constant correlation 20%.
Tables 7.26-7.29 represent the worst case probabilities for MPCF and the marginal problem for 2 to 5 dimensional loss vectors respectively simulated with a constant correlation 40%.
In Table 7.30 we report probabilities for dimension 6 for all 3 correlation $0, 20\%$ and 40%.

N \ S	**2**		**3**		**4**		**5**		**6**	
	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal
8	0.523	0.525	0.311	0.333	0.161	0.165	0.047	0.096	0.023	0.026
16	0.525	0.559	0.286	0.314	0.143	0.165	0.047	0.072	0.013	0.026
32	0.517	0.553	0.299	0.315	0.142	0.166	0.047	0.072	0.013	0.026
64	0.523	0.537	0.304	0.317	0.141	0.166	0.048	0.072	0.013	0.026
128	0.52	0.53	0.303	0.318	0.139	0.166	0.042	0.068	0.012	0.024
256	0.518	0.526	0.304	0.319	0.136	0.163	0.041	0.068	0.011	0.023
512	0.516	0.524	0.302	0.316	0.133	0.161	0.04	0.067	0.011	0.023
Monte Carlo	7.90E-02		1.70E-02		2.30E-03		2.30E-04		1.50E-05	

Table 7.18: Worst case probabilities for MPCF and the marginal problem for a 2-dimensional loss vector simulated with Gaussian Copula with correlation 0. S is the threshold and N is the discretisation size.

N \ S	**3**		**4.5**		**6**		**7.5**		**9**	
	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal
8	0.453	0.458	0.316	0.346	0.097	0.143	0.044	0.096	0.008	0.021
16	0.44	0.44	0.223	0.271	0.08	0.132	0.022	0.053	0.008	0.018
32	0.434	0.434	0.21	0.257	0.068	0.126	0.02	0.049	0.008	0.016
64	0.437	0.438	0.204	0.251	0.064	0.121	0.016	0.044	0.005	0.014
Monte Carlo	4.20E-02		4.70E-03		2.60E-04		1.20E-05		0.00E+00	

Table 7.19: Worst case probabilities for MPCF and the marginal problem for a 3-dimensional loss vector simulated with Gaussian Copula with correlation 0. S is the threshold and N is the discretisation size.

N \ S	**4**		**6**		**8**		**10**		**12**	
	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal
8	0.44	0.44	0.213	0.277	0.069	0.129	0.029	0.052	0.005	0.018
16	0.396	0.4	0.172	0.234	0.053	0.109	0.015	0.043	0.005	0.014
32	0.373	0.391	0.156	0.216	0.043	0.099	0.012	0.038	0.004	0.009
Monte Carlo	2.30E-02		1.30E-03		4.00E-05		1.00E-06		0.00E+00	

Table 7.20: Worst case probabilities for MPCF and the marginal problem for a 4-dimensional loss vector simulated with Gaussian Copula with correlation 0. S is the threshold and N is the discretisation size.

Remark 7.3.2 *Note that up to now we have been considering the problem of finding maximum probabilities given some constraints. In practice, financial institutions are required to calculate the VaR or CVaR levels given the quantile. Algorithm 7.3.3 describes how to calculate the risk levels for a given quantile. Note that the constraint*

S / N	**5**		**7.5**		**10**		**12.5**		**15**	
	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal
8	0.426	0.426	0.239	0.289	0.057	0.121	0.033	0.065	0.007	0.017
16	0.363	0.378	0.14	0.209	0.042	0.098	0.014	0.038	0.006	0.012
Monte Carlo	1.30E-02		3.70E-04		3.00E-06		0.00E+00		0.00E+00	

Table 7.21: Worst case probabilities for MPCF and the marginal problem for a 5-dimensional loss vector simulated with Gaussian Copula with correlation 0. S is the threshold and N is the discretisation size.

S / N	**2**		**3**		**4**		**5**		**6**	
	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal
8	0.525	0.525	0.318	0.333	0.154	0.165	0.059	0.096	0.026	0.026
16	0.517	0.559	0.276	0.314	0.156	0.165	0.059	0.072	0.016	0.026
32	0.511	0.553	0.293	0.315	0.156	0.166	0.061	0.072	0.016	0.026
64	0.514	0.537	0.304	0.317	0.157	0.166	0.06	0.072	0.017	0.026
128	0.518	0.53	0.304	0.318	0.156	0.166	0.055	0.068	0.014	0.024
256	0.515	0.526	0.309	0.319	0.153	0.163	0.055	0.068	0.014	0.023
512	0.513	0.524	0.307	0.316	0.148	0.161	0.054	0.067	0.013	0.023
Monte Carlo	9.80E-02		2.60E-02		5.10E-03		6.20E-04		6.40E-05	

Table 7.22: Worst case probabilities for MPCF and the marginal problem for a 2-dimensional loss vector simulated with Gaussian Copula with correlation 20%. S is the threshold and N is the discretisation size.

S / N	**3**		**4.5**		**6**		**7.5**		**9**	
	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal
8	0.449	0.458	0.329	0.346	0.123	0.143	0.051	0.096	0.014	0.021
16	0.439	0.44	0.24	0.271	0.097	0.132	0.031	0.053	0.01	0.018
32	0.434	0.434	0.226	0.257	0.09	0.126	0.029	0.049	0.009	0.016
64	0.438	0.438	0.223	0.251	0.086	0.121	0.025	0.044	0.008	0.014
Monte Carlo	7.20E-02		1.40E-02		1.70E-03		1.30E-04		6.00E-06	

Table 7.23: Worst case probabilities for MPCF and the marginal problem for a 3-dimensional loss vector simulated with Gaussian Copula with correlation 20%. S is the threshold and N is the discretisation size.

S / N	**4**		**6**		**8**		**10**		**12**	
	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal
8	0.434	0.44	0.226	0.277	0.09	0.129	0.031	0.052	0.008	0.018
16	0.4	0.4	0.19	0.234	0.072	0.109	0.022	0.043	0.008	0.014
32	0.386	0.391	0.175	0.216	0.062	0.099	0.019	0.038	0.005	0.009
Monte Carlo	5.70E-02		8.90E-03		8.00E-04		4.90E-05		1.00E-06	

Table 7.24: Worst case probabilities for MPCF and the marginal problem for a 4-dimensional loss vector simulated with Gaussian Copula with correlation 20%. S is the threshold and N is the discretisation size.

S / N	**5**		**7.5**		**10**		**12.5**		**15**	
	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal
8	0.424	0.426	0.255	0.289	0.075	0.121	0.037	0.065	0.007	0.017
16	0.376	0.378	0.167	0.209	0.058	0.098	0.018	0.038	0.006	0.012
Monte Carlo	4.80E-02		6.20E-03		4.60E-04		2.00E-05		1.00E-06	

Table 7.25: Worst case probabilities for MPCF and the marginal problem for a 5-dimensional loss vector simulated with Gaussian Copula with correlation 20%. S is the threshold and N is the discretisation size.

S / N	**2**		**3**		**4**		**5**		**6**	
	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal
8	0.525	0.525	0.32	0.333	0.148	0.165	0.067	0.096	0.026	0.026
16	0.505	0.559	0.27	0.314	0.153	0.165	0.068	0.072	0.024	0.026
32	0.495	0.553	0.288	0.315	0.158	0.166	0.068	0.072	0.022	0.026
64	0.49	0.537	0.301	0.317	0.161	0.166	0.069	0.072	0.022	0.026
128	0.489	0.53	0.302	0.318	0.16	0.166	0.065	0.068	0.019	0.024
256	0.486	0.526	0.304	0.319	0.157	0.163	0.063	0.068	0.019	0.023
512	0.486	0.524	0.304	0.316	0.155	0.161	0.062	0.067	0.018	0.023
Monte Carlo	1.20E-01		3.70E-02		8.60E-03		1.40E-03		1.70E-04	

Table 7.26: Worst case probabilities for MPCF and the marginal problem for a 2-dimensional loss vector simulated with Gaussian Copula with correlation 40%. S is the threshold and N is the discretisation size.

N \ S	3		4.5		6		7.5		9	
	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal
8	0.439	0.458	0.332	0.346	0.14	0.143	0.062	0.096	0.021	0.021
16	0.437	0.44	0.248	0.271	0.11	0.132	0.04	0.053	0.012	0.018
32	0.433	0.434	0.235	0.257	0.103	0.126	0.037	0.049	0.012	0.016
64	0.437	0.438	0.231	0.251	0.1	0.121	0.033	0.044	0.01	0.014
Monte Carlo	9.80E-02		2.70E-02		5.00E-03		6.10E-04		4.60E-05	

Table 7.27: Worst case probabilities for MPCF and the marginal problem for a 3-dimensional loss vector simulated with Gaussian Copula with correlation 40%. S is the threshold and N is the discretisation size.

N \ S	4		6		8		10		12	
	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal
8	0.42	0.44	0.235	0.277	0.114	0.129	0.035	0.052	0.013	0.018
16	0.4	0.4	0.205	0.234	0.085	0.109	0.029	0.043	0.009	0.014
32	0.389	0.391	0.191	0.216	0.076	0.099	0.025	0.038	0.006	0.009
Monte Carlo	8.90E-02		2.20E-02		3.50E-03		3.90E-04		2.00E-05	

Table 7.28: Worst case probabilities for MPCF and the marginal problem for a 4-dimensional loss vector simulated with Gaussian Copula with correlation 40%. S is the threshold and N is the discretisation size.

N \ S	5		7.5		10		12.5		15	
	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal	MPCF	Marginal
8	0.417	0.426	0.266	0.289	0.099	0.121	0.04	0.065	0.011	0.017
16	0.378	0.378	0.182	0.209	0.073	0.098	0.025	0.038	0.007	0.012
Monte Carlo	8.30E-02		1.90E-02		2.80E-03		2.70E-04		1.50E-05	

Table 7.29: Worst case probabilities for MPCF and the marginal problem for a 5-dimensional loss vector simulated with Gaussian Copula with correlation 40%. S is the threshold and N is the discretisation size.

ρ \ S	6			9			12			15		
	MPCF	Marginal	MC	MPCF	Marginal	MC	MPCF	Marginal	MC	MPCF	Marginal	MC
0	0.416	0.416	0.007	0.175	0.26	0	0.051	0.116	0	0.019	0.041	0
0.2	0.412	0.416	0.042	0.206	0.26	0.005	0.069	0.116	0	0.022	0.041	0
0.4	0.404	0.416	0.078	0.219	0.26	0.017	0.09	0.116	0.002	0.03	0.041	0

Table 7.30: Worst case probabilities for MPCF and the marginal problem for a 6-dimensional loss vector simulated with Gaussian Copula with correlations 0, 20% and 40%. We denote the threshold by S and the correlation by ρ.

of the optimisation problem stays the same and only the objective changes, hence we warm-start the optimisation problems from the solutions of the previous iterations. This significantly reduces the running time of the algorithm.

Algorithm 7.3.3 *(Finding Worst Case Scenario Risk Measures)*

1. *Choose* S_d *and* S_u *such that* $S_d < S^* < S_u$.

2. *Do*

   - $S \leftarrow \frac{1}{2}(S_d + S_u)$,

   - *If* $(Val(P(S)) < q)$, *set* $S_u \leftarrow S$, *else* $S_d \leftarrow S$,

   While $(|Val(P(S)) - q| > \epsilon)$.

3. *Return* S.

Chapter 8

Model-Free Bounds on Path-Dependent Options

In Section 4.5 we presented a duality result for calculating model-free bounds on prices of path-dependent options. In the problem setting discussed thus far, it is assumed that we know risk free probability measures of the stock price at different time intervals (which we considered as marginal distributions for our multidimensional random variable). It is well known that these marginal distributions can be extracted from the market data of call/put European option prices. The whole motivation for trying to find the model-free bounds on option prices is to provide an estimate of their values without any additional assumptions about the stock price process (except for the martingale property of discounted stock prices). In Section 4.5 we showed how the problem can be formulated in the framework of our general duality theory. We will now show that the problem of solving the infinite dimensional optimisation problem can be solved via the method presented in Section 4.4. In order to get the distribution function of the stock prices at some time in the future we need to interpolate option prices with respect to the strikes. In this chapter we make the model-free pricing of derivatives more robust by using only the information provided by the market.

The problem of finding model-free bounds on prices of vanilla type options, i.e. where the payoff depends on the value of a stock price at maturity, has been under research for a long time. Some of the earlier works include [16, 30, 64]. Obtaining bounds on European options is studied in [44] for the given mean and the variance of the underlying stock price and in [28], where the knowledge of the first k moments is given. The latter work uses semidefinite optimisation approach to calculate lower and upper bounds. More general cases are considered in [4], where the partial knowledge of the market information includes the first two moments of the stock price and also prices of other European call options. [4] also uses semidefinite optimisation approach to obtain the bounds and also calculates bounds on the variance of the stock given European call option prices for different strikes. The latter also considers the problem in multi dimensions.

In the case of multiple stocks, the options in interest are the basket options, the payoff of which depends on the prices of the basket of stocks. In the case of basket options, the problems of obtaining the lower or the upper bounds differ in complexity. [36] studies the calculation of upper bounds of basket options. The paper considers both cases, where European call option prices for all strikes are provided and when only a finite number of option prices are given. In the first case the cheapest super-replicating strategy includes trading only a single European option for each underlying. In the second case, it consists of trading two options for each underlying stock. Note that although the replicating portfolios contain only one or two call options per stock, in both cases the full knowledge of option prices is required to determine the corresponding strikes. A similar work can be found in [17].

The calculation of lower bounds on basket options is considered, e.g. [35] or [55]. Compared to the super-replicating strategies, the sub-replicating strategies considered in the above mentioned articles consists of long and short positions in call options.

In all above mentioned works it is assumed that the market given European options can be bought and sold at the same price, i.e. with no bid-ask spreads. [56] studies the problem of the upper bound calculation in presence of bid-ask spreads. Moreover, the size of the dual problem (in all the above mentioned works the bounds are obtained via duality) resulting from the infinite-dimensional optimisation problem is linear in the input data size. In [18] the authors also consider basket options and present the so-called 'Feasible discretization method' (Algorithm 2, [18]) for numerical calculation of the upper bounds.

The model-free pricing of financial derivatives is more complicated for path-dependent options, where the payoff depends not only on the stock price at maturity, but also its intermediate values. The latter problem has been under extensive research in recent years for non-American type options, which includes Asian, barrier, lookback, variance swap options and etc. The model-free option pricing of American put options is studied in [65], where the author obtains an upper and lower bounds on an American put option with fixed strike from given American put options with the same maturity, but different strikes. In this work and all the articles mentioned below only non-American type options are considered.

Although there are numerical methods for solving model-free option pricing for generic

path-dependent options, [2, 32], to our knowledge, theoretical results in the literature consider only specific options. In the recent work [33], the authors consider the problem of option pricing where the payoff of the option depends on the terminal value and the maximum of the underlying stock price. They obtain sharp bounds on the distribution of a continuous martingale with fixed marginal distributions at finitely many intermediate times. This particularly allows one to obtain upper bounds on the lookback options. The above mentioned article in general provides good overview of the existing literature and is an extension of an earlier work [9], where the authors consider only one intermediate time. The case of only one intermediate time is also considered in [37] in pricing forward start options.
Robust pricing of barrier options is studied in [14, 13, 10]. Model-free options pricing of Asian options can be found for example in [67]. [19, 15, 7, 34, 11] discuss robust pricing of variance swap options.
Throughout this chapter we assume that the interest rate is zero.

8.1 Duality Theory for Model-Free Bounds on Option Prices

Consider an exotic path-dependent option that depends on the value of a stock price at some discrete times $t_1 < t_2 < \cdots < t_n$. Let $X = (X_1, \dots, X_n)$ be an n-dimensional random variable denoting stock prices at times $(t_1, \dots, t_n)$ respectively and $h(X_1, \dots, X_n)$ be the payoff of the given option.
In practice, in a no-arbitrage framework, one assumes that the stock price follows some specific model, calibrates this model to market data and finds the fair price of an option, i.e. finds a risk neutral probability measure Q on $\mathbb{R}^n$ and calculates the price as

$$\mathbb{E}_Q[h] = \int_{\mathbb{R}^n} h(x_1, \dots, x_n) \, \mathrm{d}\, Q(x_1, \dots, x_n). \tag{8.1}$$

Under Q the stock price process $(X_i)_{i=1,\dots,n}$ is assumed to be a discrete martingale.
In this section we aim to find bounds on the prices of the option without assuming any particular model for the stock price process, i.e. find maximum and minimum values over all models that are consistent with the observed market data. Assuming knowledge of market call option prices, one can recover a distribution function $\mathcal{F}_i$ of X_i for every $i = 1, \dots, n$ (see for example [49]), which will represent the marginal distributions of a risk neutral equivalent probability measure of X. In this framework,

denoting by $\mathcal{M}(\mathcal{F}_1, \dots, \mathcal{F}_n)$ the set of martingale measures with marginals $\mathcal{F}_1, \dots, \mathcal{F}_n$, the problem becomes

$$\inf_{\mathcal{F} \in \mathcal{M}(\mathcal{F}_1, \dots, \mathcal{F}_n)} \int_{\mathbb{R}^n} h(x_1, \dots, x_n) \, \mathrm{d}\, \mathcal{F}(x_1, \dots, x_n). \tag{8.2}$$

We call the above problem the primal problem.

For any $i = 1, \dots, n$ define the set of F_i integrable functions from $\mathbb{R} \to \mathbb{R}$ by $\mathcal{H}_i$ and for any $j = 1, \dots, n-1$ the set of bounded measurable functions from $\mathbb{R}^j \to \mathbb{R}$ by $\mathcal{T}_j$. For any $h_i \in \mathcal{H}_i$, $i = 1, \dots, n$ and $g_j \in \mathcal{T}_j$, $j = 1, \dots, n-1$ consider the function $\Psi_{(h_i),(g_j)}(x_1, \dots, x_n) : \mathbb{R}^n \to \mathbb{R}$ defined by

$$\Psi_{(h_i),(g_j)}(x_1, \dots, x_n) = \sum_{i=1}^{n} h_i(x_i) + \sum_{j=1}^{n-1} g_j(x_1, \dots, x_j)(x_{j+1} - x_j). \tag{8.3}$$

Consider the following optimisation problem:

$$\begin{aligned} &\sup_{(h_i \in \mathcal{H}_i)_1^n, (g_j \in \mathcal{T}_j)_1^{n-1}} \sum_{i=1}^{n} \int_{\mathbb{R}} h_i(x_i) \, \mathrm{d}\, \mathcal{F}_i(x_i) \\ &\text{s.t. } \Psi_{(h_i),(g_j)}(x_1, \dots, x_n) \leq h(x_1, \dots, x_n). \end{aligned} \tag{8.4}$$

We have the following theorem :

Theorem 8.1.1 (Theorem 1, [2]) *Assume $\mathcal{F}_1, \dots, \mathcal{F}_n$ are Borel probability measures on $\mathbb{R}$ so that $\mathcal{M}(\mathcal{F}_1, \dots, \mathcal{F}_n)$ is non-empty. Let $h : \mathbb{R}^n \to (-\infty, \infty]$ be a lower semi-continuous function so that*

$$h(x_1, \dots, x_n) \geq -K \cdot (1 + |x_1| + \dots + |x_n|),$$

on $\mathbb{R}^n$ for some constant K. Then there is no duality gap between primal (8.2) *and dual* (8.4) *problems. Moreover, the primal optimal value is obtained, i.e. there exists a martingale measure $\mathcal{F} \in \mathcal{M}(\mathcal{F}_1, \dots, \mathcal{F}_n)$ such that the optimal value of* (8.2) *is equal to $\mathbb{E}_{\mathcal{F}}[h]$.*

A proof of the theorem can be found in [2].

In order to solve the dual problem numerically, we need to make some discretisation of this infinite dimensional problem. One way of doing so (as is described in [32]) is to decompose the functions $(h_i)_1^n$ and $(g_j)_1^{n-1}$ over a finite dimensional basis. Since the functions $(g_j)_1^{n-1}$ are assumed to be continuous, we can choose a polynomial basis, i.e.

$$g_j(x_1, \dots, x_j) = \sum_k g_j^k e_j^k(x_1, \dots, x_j),$$

where e_j^k are functions of the basis, and g_j^k are the coordinates of g_j in that basis. For functions $(h_i)_1^n$ we can choose call option payoffs as basis

$$h_i(x_i) = \sum_k h_i^k (x_i - k^b)^+.$$

With this approximation, the dual problem reduces to a linear programming problem, so that one can solve them with interior point or simplex methods.
Note that in the cited above papers [32, 2], authors consider given marginal distributions in the primal problem. Moreover, as is described above, in [2] authors decompose the functions $(h_i)_1^n$ and $(g_j)_1^{n-1}$ over a finite dimensional basis, thus making the dual problem a finite linear optimisation problem. In [32] a specific payoff is considered (forward start option) and authors make a good guess for the functions $(g_j)_1^{n-1}$ and use the linearity of the dual problem to check the inequality constraints only in a finite number of points. Compared to these results, in this work we take a different approach to the numerical solution of the problem of arbitrage-free pricing. Our approach is based on the model described in Section 4.4, in which we use only market given European option prices (and not the marginal distributions) and also discretise the martingale constraints in the primal problem. The rest of this section presents our model.

8.2 Application of the General Duality Theory in Model-Free Option Pricing

In practice, we are not given European option prices for every strike, hence we cannot have an exact knowledge of marginal distributions. Moreover, for some time steps the number of traded liquid products can be so small that any interpolation of option prices will contain additional errors. To avoid inducing additional estimation errors in the model and to use only the information given by the market, we replace constraints for marginal distributions by constraints which represent the calculation of vanilla option values.

Taking into consideration the above observations, we construct the following problem:

$$
\begin{aligned}
&\sup_{\mathcal{F}} \int_{\mathbb{R}^n} h(x)\,\mathrm{d}\,\mathcal{F}(x) \\
&\text{s.t.} \int_{\mathbb{R}^n} (x_i - k_i^j)^+\,\mathrm{d}\,\mathcal{F}(x) \leq \bar{C}_i^j, \quad i = 1\dots,n, \quad j = 1,\dots,m_j \\
&\quad\;\; \int_{\mathbb{R}^n} (x_i - k_i^j)^+\,\mathrm{d}\,\mathcal{F}(x) \geq \underline{C}_i^j, \quad i = 1\dots,n, \quad j = 1,\dots,m_j \\
&\quad\;\; \int_{\mathbb{R}^n} \phi_i(x)\,\mathrm{d}\,\mathcal{F}(x) \leq \bar{\delta}_i, \quad i = 1,\dots,N_\phi && (8.5) \\
&\quad\;\; \int_{\mathbb{R}^n} \phi_i(x)\,\mathrm{d}\,\mathcal{F}(x) \geq \underline{\delta}_i, \quad i = 1,\dots,N_\phi && (8.6) \\
&\quad\;\; \int_{\mathbb{R}^n} 1_{z\leq x|_k \leq y}(z,y,x)(x_{k+1}-x_k)\,\mathrm{d}\,\mathcal{F}(x) = 0, \quad (y,z \in \mathbb{R}^k, k<n) && (8.7) \\
&\quad\;\; \int_{\mathbb{R}^n} 1\,\mathrm{d}\,\mathcal{F}(x) = 1, \\
&\quad\;\; \mathcal{F} \geq 0,
\end{aligned}
$$

where $(k_i^j)_{j=1}^{m_i}$ are the strikes of the European call options available in the market, $(\bar{C}_i^j)_{j=1}^{m_i}$ and $(\underline{C}_i^j)_{j=1}^{m_i}$ are their respective market bounds of prices (bid-ask prices) and m_i is the number of such options.

Note that in addition to call or put options, we might observe other options in the market. Let them have payoff functions $\phi_i(x)$ and bid-ask prices $\underline{\delta}_i$ and $\bar{\delta}_i$, $i = 1,\dots,N_\phi$. For each of these additional observed options we added 2 more inequality constraints to the model ((8.5)-(8.6)).

The constraints representing call option prices fit our model (the functions are piecewise linear). To render the martingale constraints compatible with our model, we approximate them by requiring that (8.7) hold for only finitely many boxes in $\mathbb{R}^n$ of our choosing.

It is natural to impose lower and upper bounds on stock price X_i for each t_i, $i = 1,\dots,n$. 0 will represent a natural lower limit, and for the upper limit, we can take a large enough value that the stock price is statistically unlikely to exceed it. We also insure that the strikes are included in the set of points in our discretisation.

Let $0 = d_i^1 < d_i^2 < \cdots < d_i^{n_i}$ be the set of discretisation points for X_i. Denote $\Phi := \{s : d_i^1 \leq x_i \leq d_i^{n_i}, (i = 1,\dots,n)\}$.

The martingale constraint for $k = 1$ is then:

$$\int_{\mathbb{R}^n} 1_{\{x\in\mathbb{R}^n : d_i^{j_i} \leq x_i \leq d_i^{\ell_i}, (i=1,\dots,k)\}}(x)(x_{k+1}-x_k)\,\mathrm{d}\,\mathcal{F}(x) = 0, \quad (1 \leq j_i \leq \ell_i \leq n_i, \forall i).$$

For each k we thus have $M_k := \prod_{i=1}^{k}(n_i - 1)$ equality constraints, which are all represented as integral inequalities or equalities with piecewise linear test functions. We saw that, after discretisation, all of our constraints fit the model described in Section 4.4. Furthermore, if the payoff functions $h(x)$ and $\phi_i(x), \quad i = 1, \dots, N_\phi$ are piecewise linear (put/call or binary barrier options, Asian options, lookback options etc.), then we can solve the dual problem as described in Section 6.1. If the payoff function $h(x)$ is concave, we can find the upper bound with method described in Section 6.3, and if it is convex, we can find the lower bound. For all other cases we have to make approximations, for example, first order Taylor approximation.
Now we can write the dual problem as follows:

$$
\begin{aligned}
&\inf_{\underline{y},\bar{y},\underline{z},\bar{z},t,w} \sum_{i=1}^{n}\sum_{j=1}^{n_i}(\bar{y}_i^j \bar{C}_i^j - \underline{y}_i^j \underline{C}_i^j) + \sum_{i=1}^{M_\phi}(\bar{z}_i\bar{\delta}_i - \underline{z}_i\underline{\delta}_i) + w \\
&\text{s.t.} \ \sum_{i=1}^{n}\sum_{j=1}^{n_i}(\bar{y}_i^j - \underline{y}_i^j)(x_i - K_i^j)^+ \sum_{i=1}^{M_\phi}(\bar{z}_i - \underline{z}_i)\phi_i(s) + \\
&\sum_{k=1}^{n-1}\sum_{\tau \in I_k} 1_{p_\tau \le s|_k \le p_{\tau+1}}(s)(x_{k+1} - x_k) + w \ge h(s), \quad \forall s \in \Phi.
\end{aligned}
\tag{8.8}
$$

The arbitrage-free pricing is usually used not to find lower and upper model-free bounds for options, but rather for finding arbitrage opportunities. Suppose some path-dependent option with payoff $h(x)$ is traded in the market with price h^M. By implementing our model for this option we find no-arbitrage bounds $[L, U]$. If $h^M \notin [L, U]$, then there is an arbitrage in the market (the dual problem also provides with a strategy to make risk-free money in this case, but we will not go into details in this work, interested reader can refer to [2]). Hence, more often we are interested in comparing the optimal value of problem (8.8) with the given market price h^M. In this case Nesterov's smoothening technique can be implemented as described in Section 6.5.

8.3 Payoffs as Piecewise Linear Functions

In this section we describe different payoffs that are piecewise functions. Obviously, call and put options fit to this category.

8.3.1 Barrier Options

For given lower and upper barriers $B_1 < B_2$ (if the option is defined for one barrier only, e.g. down and out, up and in, then we assume that the other barrier is $-\infty$ or ∞ depending on the context), barrier options have the payoff

$$h(x) = H(x) \prod_{i=1}^{n} 1_{B_1 \leq x_i \leq B_2}, \tag{8.9}$$

where $H(x)$ is the final payoff. For digital barrier options we have $H(x) = 1, \quad \forall x \in \mathbb{R}^n$ and for call and put barrier options with strike K we have $H(x) = (x_n - K)^+$ and $H(x) = (K - x_n)^+$ respectively.
It is obvious that these options have piecewise linear payoffs.

8.3.2 Lookback and Asian Options

Lookback options have payoffs that are dependent on the $X_{max} = \max\{X_1, \ldots, X_n\}$ and/or $X_{min} = \min\{X_1, \ldots, X_n\}$. Asian options are defined as having payoff functions at maturity which are dependent on the average value $X_A = \frac{1}{n}\sum_{i=1}^{n} X_i$ of the stock price over the path. Table 8.1 describes payoffs for different lookback and Asian options. The following lemma proves that all payoffs described in Table 8.1 are

Table 8.1: Lookback and Asian Option Pyaoffs

	Call	Put
Lookback Fixed	$\max(X_{max} - K, 0)$	$\max(K - X_{min}, 0)$
Lookback Floating	$X_n - X_{min}$	$X_{max} - X_n$
Asian Fixed	$\max(X_A - K, 0)$	$\max(K - X_A, 0)$
Asian Floating	$X_n - X_A$	$X_A - X_n$

piecewise linear functions on Φ.

Lemma 8.3.1 *Let Φ be a polytope in $\mathbb{R}^n$ and be further divided into finitely many polytopes $\Phi = (\bigcup_{i=1}^{M} \Pi_i)$. Let function $f : \Phi \to \mathbb{R}$ and $g : \Phi \to \mathbb{R}$ be piecewise linear on these polytopes. Then, the functions* $\max\{f(x), g(x)\}$ *and* $\min\{f(x), g(x)\}$ *are also piecewise linear on some finite partitioning of $\Phi = (\bigcup_{i=1}^{\bar{M}} \bar{\Pi}_i)$.*

8.4 Numerical Experiments

In this section we present our numerical results for barrier and variance swap options on simulated market data, i.e. we compute call option prices with Black-Scholes formulae for different time horizons.

First we present how the linear optimisation problem corresponding ti model-free option pricing problem is constructed for a particular barrier option with lower and upper barriers B_1 and B_2 respectively. We suppose that the number of European Call option prices and their strikes are the same for each maturity, though this restriction is not necessary and the construction of the linear programming problem follows the same procedure.

Consider that we are given call option prices (with bid-ask spreads) for n maturities $t_1, \ldots, t_n$ and N strikes $K_1, \ldots, K_N$ strikes. For a maturity t_i and a strike K_j let $\underline{C}_i^j(t_i, K_j)$ and $\bar{C}(t_i, K_j)$ be the bid and ask prices of a European call option with maturity t_i and strike K_j respectively.

As for the risk aggregation problem described in Section 7.3, we first need to construct the domain of the problem and it's discretisation. For the lower bounds of the domain a natural choice is $L_1 = L_2 = \cdots = L_n = 0$. As upper bounds we chose $U_1 = U_2 = \cdots = U_n$ such that the option prices $C(t_i, U_i) < \epsilon$ for $\epsilon = 0.00001$, $i = 1, \ldots, n$.

In this setup, the payoff of the double no-touch digital barrier option is

$$h(x) = 1_{B_1 \leq x_1, \ldots, x_n \leq B_2} \tag{8.10}$$

and the payoff of the double no-touch barrier call option is

$$h(x) = 1_{B_1 \leq x_1, \ldots, x_n \leq B_2} (x_n - K)^+ , \tag{8.11}$$

where K is the strike of the call option.

For each dimension i we have $2N$ inequality constraints for European Call options in the form

$$\begin{aligned} \textstyle\int_{\mathbb{R}^n} (x_i - k_i^j)^+ \,\mathrm{d}\,\mathcal{F}(s) &\leq \bar{C}_i^j(t_i, K_j), \quad j = 1, \ldots, N \\ \textstyle\int_{\mathbb{R}^n} (x_i - k_i^j)^+ \,\mathrm{d}\,\mathcal{F}(s) &\geq \underline{C}_i^j(t_i, K_j), \quad j = 1, \ldots, N \end{aligned}$$

For the martingale constraints, for each $i = 1, \ldots, n-1$ we have

$$\int_{\mathbb{R}^n} 1_{K(\pi_l) \leq x|_k \leq K(\pi_u)}(x)(x_{k+1} - x_k) \,\mathrm{d}\,\mathcal{F}(x) = 0,$$

for each $\pi_l < \pi_u$ k-dimensional vectors of integer numbers from $[0, N+1]$. For a given index vector π we have

$$K(\pi)_i = \begin{cases} K_{\pi_i} & \text{if } 1 \leq \pi_i \leq N \\ 0 & \text{if } \pi_i = 0 \\ U & \text{if } \pi_i = N+1. \end{cases} \quad .$$

The inequality for index vectors is defined as component-wise.
Combining all of the above, we get the following linear optimisation problem

$$\begin{aligned}
\sup_{\mathcal{F}} \quad & \int_{\mathbb{R}^n} 1_{B_1 \leq x_1, \ldots, x_n \leq B_2} \, \mathrm{d}\, \mathcal{F}(x) \\
\text{s.t.} \quad & \int_{\mathbb{R}^n} (x_i - k_i^j)^+ \, \mathrm{d}\, \mathcal{F}(s) \leq \bar{C}_i^j(t_i, K_j), \quad i = 1 \ldots, n, \quad j = 1, \ldots, N \\
& \int_{\mathbb{R}^n} (x_i - k_i^j)^+ \, \mathrm{d}\, \mathcal{F}(x) \geq \underline{C}_i^j(t_i, K_j), \quad i = 1 \ldots, n, \quad j = 1, \ldots, N \\
& \int_{\mathbb{R}^n} 1_{K(\pi_l) \leq x|_k \leq K(\pi_u)}(x)(x_{k+1} - x_k) \, \mathrm{d}\, \mathcal{F}(x) = 0, \quad k = 1, \ldots, N-1, \\
& \qquad\qquad \pi_l < \pi_u \in [0, N+1]^k, \\
& \int_{\mathbb{R}^n} 1 \, \mathrm{d}\, \mathcal{F}(x) = 1, \\
& \mathcal{F} \geq 0.
\end{aligned}$$

In the rest of this section we first introduce results for double no-touch barrier call and digital options and then for the variance swap options.
We use the simulated market data presented in Table 8.2 as input to our model. The prices of the European Call option prices are calculated by Black-Scholes formula:

$$\begin{aligned}
C(S, t; K) &= N(d_1)S - N(d_2)Ke^{-rt}, \\
d_1 &= \frac{1}{\sigma\sqrt{t}} \left[\ln\left(\frac{S}{K}\right) + \left(r + \frac{\sigma^2}{2} t\right) \right], \\
d_2 &= \frac{1}{\sigma\sqrt{t}} \left[\ln\left(\frac{S}{K}\right) + \left(r - \frac{\sigma^2}{2} t\right) \right],
\end{aligned}$$

where S is the spot price and K is the strike. Table 8.2 is calculated with $r = 0$ and $\sigma = 30\%$.

8.4.1 Double No-Touch Barrier Options

In this section we introduce results for double no-touch barrier call and digital options, which have payoffs as described in (8.11). We did two types of numerical experiments.

Strike \ Maturity	3M	4M	6M	8M	9M	1Y	1Y6M	2Y	2Y6M
30	16.00	16.01	16.06	16.15	16.20	16.38	16.82	17.28	17.74
31	15.01	15.03	15.10	15.21	15.27	15.50	15.99	16.50	17.00
32	14.01	14.04	14.14	14.29	14.37	14.63	15.19	15.75	16.29
33	13.03	13.07	13.21	13.38	13.48	13.79	14.41	15.02	15.60
34	12.05	12.11	12.29	12.50	12.62	12.97	13.66	14.32	14.93
35	11.08	11.17	11.40	11.65	11.78	12.18	12.94	13.64	14.29
36	10.13	10.25	10.53	10.83	10.97	11.41	12.24	12.98	13.67
37	9.20	9.36	9.69	10.03	10.20	10.68	11.56	12.35	13.07
38	8.30	8.49	8.89	9.27	9.45	9.98	10.92	11.75	12.49
39	7.43	7.67	8.12	8.54	8.74	9.31	10.30	11.16	11.94
40	6.61	6.88	7.39	7.85	8.07	8.67	9.71	10.61	11.40
41	5.83	6.14	6.70	7.20	7.43	8.06	9.15	10.07	10.89
42	5.10	5.44	6.05	6.58	6.82	7.48	8.61	9.56	10.40
43	4.42	4.80	5.45	6.00	6.25	6.94	8.09	9.07	9.93
44	3.81	4.21	4.88	5.46	5.72	6.42	7.61	8.60	9.47
45	3.25	3.67	4.36	4.95	5.22	5.94	7.14	8.15	9.04
46	2.75	3.17	3.89	4.48	4.75	5.48	6.70	7.73	8.62
47	2.31	2.73	3.45	4.05	4.32	5.06	6.29	7.32	8.23
48	1.92	2.34	3.05	3.65	3.92	4.66	5.90	6.93	7.85
49	1.59	1.99	2.69	3.28	3.55	4.29	5.52	6.57	7.48
50	1.30	1.69	2.36	2.95	3.21	3.94	5.17	6.22	7.13
51	1.06	1.42	2.07	2.64	2.90	3.62	4.84	5.88	6.80
52	0.85	1.19	1.81	2.36	2.61	3.32	4.53	5.57	6.49
53	0.68	0.99	1.58	2.10	2.35	3.04	4.24	5.27	6.18
54	0.54	0.82	1.37	1.87	2.11	2.79	3.96	4.98	5.89
55	0.43	0.68	1.19	1.67	1.90	2.55	3.70	4.71	5.62
56	0.34	0.56	1.02	1.48	1.70	2.33	3.46	4.46	5.35
57	0.26	0.46	0.88	1.31	1.52	2.13	3.23	4.21	5.10
58	0.20	0.37	0.76	1.16	1.36	1.94	3.02	3.98	4.86
59	0.16	0.30	0.65	1.03	1.21	1.77	2.81	3.76	4.63
60	0.12	0.25	0.56	0.91	1.08	1.61	2.63	3.56	4.42
61	0.09	0.20	0.48	0.80	0.96	1.47	2.45	3.36	4.21
62	0.07	0.16	0.41	0.70	0.86	1.34	2.28	3.18	4.01

Table 8.2: European Call option prices for different maturities and strikes, which is used as input to our model. The prices are calculated with Black-Scholes formula. M and Y denote month and year respectively.

First, where we solve the model-free option pricing for the barrier options for fixed barriers but different market data as input. Then we solve the problem for a give set of market data but for different options (different lower and upper barriers).

The market data that is used in a specific problem is specified by 3 parameters: dimension, strike step and the bid-ask spread. The dimension is the number of maturities that we consider prices of call options are given for. In this section the maturities are with 6 months frequencies. The strike step is the parameter specifying the frequency of the strikes. The bid-ask spread is the bid-ask of the option prices as percentage points. Hence, for a given set of parameters n, dK and p, the following set of the options is used as market input:

$$\begin{aligned} \bar{C}_i^j &= C(S, 0.5i; 30 + jdK)(1 + p), \\ \underline{C}_i^j &= C(S, 0.5i; 30 + jdK)(1 - p), \\ i &= 1, \dots, n; \quad j = 0, \dots, 32/dK. \end{aligned}$$

Tables 8.3 and 8.4 represent the model-free upper bounds of a barrier no-touch call option prices for dimensions 2 and 3 respectively for different strike steps and bid-ask spreads. The lower and upper barriers are 38 and 54 respectively. The strike of the barrier option is equal to the spot price 46.

Tables 8.5 and 8.6 represent the model-free upper bounds of a barrier no-touch digital

option prices for dimensions 2 and 3 respectively for different strike steps and bid-ask spreads. As in the previous table, the lower and upper barriers are 38 and 54 respectively.
Now we introduce results for different barrier options, i.e. for different sets of lower and upper barriers. Bid-ask spreads are considered to be 0. Tables 8.7 and 8.8 represent the model-free upper bounds of barrier call options for strike steps 2 and 4 respectively for different lower and upper barriers. Similarly, Tables 8.9 and 8.10 present results for dimension 3.
Tables 8.11 and 8.12 represent the model-free upper bounds of barrier digital options for strike steps 2 and 4 respectively for different lower and upper barriers. Similarly, Tables 8.13 and 8.14 present results for dimension 3.
Table 8.15 presents results for dimension 4, bid-ask spread 0 and strike step 4. The lower and upper barrier are 40 and 50 respectively.

Strike Step \ Bid-Ask	0.0%	2.5%	5.0%
1	0.018	0.036	0.043
2	0.021	0.038	0.044
4	0.028	0.040	0.047
8	0.043	0.049	0.053

Table 8.3: Upper bounds of the barrier no-touch call option price for dimension 2 and different strike steps and bid-ask spreads. The strike of the call option is equal to the spot price 46.

Strike Step \ Bid-Ask	0.0%	2.5%	5.0%
2	0.035	0.055	0.061
4	0.046	0.061	0.067
8	0.068	0.077	0.082

Table 8.4: Upper bounds of the barrier no-touch call option price for dimension 3 and different strike steps and bid-ask spreads. The strike of the call option is equal to the spot price 46.

After carrying out some numerical experiments, we observe that as long as we have call option prices for the strikes that coincide with the barriers of our options, any finer discretisation than the one described above does not lead to any improvement, i.e. the optimal value of the optimisation problem remains the same. This

Bid-Ask Strike Step	0.0%	2.5%	5.0%
1	0.032	0.044	0.049
2	0.035	0.046	0.051
4	0.044	0.051	0.056
8	0.060	0.068	0.073

Table 8.5: Upper bounds of the barrier no-touch digital option price for dimension 2 and different strike steps and bid-ask spreads.

Bid-Ask Strike Step	0.0%	2.5%	5.0%
2	0.049	0.066	0.071
4	0.063	0.074	0.080
8	0.087	0.097	0.104

Table 8.6: Upper bounds of the barrier no-touch digital option price for dimension 3 and different strike steps and bid-ask spreads.

Upper Barrier Lower Barrier	50	54	58	62
30	0.019	0.030	0.044	0.073
34	0.015	0.026	0.040	0.069
38	0.010	0.021	0.035	0.065
42	0.007	0.017	0.030	0.060
46	0.007	0.016	0.028	0.054

Table 8.7: Upper bounds of the barrier no-touch call option price for dimension 2 and strike step 2 for different lower and upper barriers. The strike of the call option is equal to the spot price 46.

Upper Barrier Lower Barrier	50	54	58	62
30	0.025	0.038	0.053	0.081
34	0.021	0.034	0.049	0.076
38	0.014	0.028	0.043	0.070
42	0.010	0.022	0.036	0.064
46	0.010	0.021	0.034	0.058

Table 8.8: Upper bounds of the barrier no-touch call option price for dimension 2 and strike step 4 for different lower and upper barriers. The strike of the call option is equal to the spot price 46.

Lower Barrier \ Upper Barrier	50	54	58	62
30	0.038	0.052	0.066	0.101
34	0.030	0.045	0.059	0.094
38	0.019	0.035	0.050	0.085
42	0.010	0.024	0.040	0.076
46	0.007	0.017	0.031	0.063

Table 8.9: Upper bounds of the barrier no-touch call option price for dimension 3 and strike step 2 for different lower and upper barriers. The strike of the call option is equal to the spot price 46.

Lower Barrier \ Upper Barrier	50	54	58	62
30	0.050	0.067	0.082	0.116
34	0.040	0.057	0.073	0.106
38	0.027	0.046	0.062	0.095
42	0.015	0.032	0.051	0.083
46	0.010	0.023	0.039	0.070

Table 8.10: Upper bounds of the barrier no-touch call option price for dimension 3 and strike step 4 for different lower and upper barriers. The strike of the call option is equal to the spot price 46.

Lower Barrier \ Upper Barrier	50	54	58	62
30	0.047	0.059	0.071	0.098
34	0.033	0.045	0.057	0.083
38	0.021	0.035	0.047	0.073
42	0.012	0.024	0.038	0.065
46	0.007	0.016	0.028	0.054

Table 8.11: Upper bounds of the barrier no-touch digital option price for dimension 2 and strike step 2 for different lower and upper barriers.

Lower Barrier \ Upper Barrier	50	54	58	62
30	0.056	0.071	0.083	0.108
34	0.042	0.056	0.069	0.093
38	0.029	0.044	0.057	0.081
42	0.016	0.031	0.045	0.069
46	0.010	0.021	0.034	0.058

Table 8.12: Upper bounds of the barrier no-touch digital option price for dimension 2 and strike step 4 for different lower and upper barriers.

Lower Barrier \ Upper Barrier	50	54	58	62
30	0.075	0.092	0.107	0.139
34	0.050	0.067	0.082	0.114
38	0.031	0.049	0.065	0.097
42	0.014	0.030	0.048	0.081

Table 8.13: Upper bounds of the barrier no-touch digital option price for dimension 3 and strike step 2 for different lower and upper barriers.

Lower Barrier \ Upper Barrier	50	54	58	62
30	0.091	0.110	0.126	0.156
34	0.065	0.084	0.101	0.131
38	0.043	0.063	0.080	0.109
42	0.020	0.040	0.059	0.089
46	0.010	0.023	0.039	0.070

Table 8.14: Upper bounds of the barrier no-touch digital option price for dimension 3 and strike step 4 for different lower and upper barriers.

Option Type	Upper Bound
Call	0.055
Digital	0.067

Table 8.15: Upper bounds of the barrier no-touch call and digital option prices for dimension 4 and strike step 4 for lower barrier 40 and upper barrier 50. The strike of the call option is equal to the spot price 46.

observation might be of interest for future research. If the latter statement is proved to be true, this means that with our method we get the best bounds (denote the lower and upper bounds obtained by our method by L and U respectively) for the given market data, i.e. for any other bound $l > L$ and $u < U$ we can construct a martingale measure for stock prices such that under this measure we get prices for the barrier option that are out of these new bounds l and u.

Taking into consideration the above observation, in cases where we don't have barriers as strikes in our market data, we should perform some interpolation at the beginning to find approximate call option prices for the barriers as strikes and then construct our model. This approach is preferable to solving the problem by making a finer approximation.

8.4.2 Variance Swap Options

In this section we introduce numerical results for plain vanilla variance swap options. If $(S_t, t \in [0,T])$ denotes the price of a financial asset, the weighted realised variance is defined as

$$RV_T = \sum_{i=1}^{n} h(S_{t_i}) \left(\log \frac{S_{t_i}}{S_{t_{i-1}}} \right)^2, \tag{8.12}$$

where t_i is a pre-specified sequence of times $0 = t_0 < t_1 < \cdots < t_n = T$ and h is a given weight function. A weighted variance swap is a forward contract in which cash amounts equal to $A \times RV_T$ and $A \times P_T^{RV}$ are exchanged at time T, where A is the dollar value of one variance swap and P_T^{RV} is the variance swap 'price', agreed at time 0. In this section we consider only plain vanilla variance swap $h(s) \equiv 1$. The reader can find more in detail discussion of variance swap options in [19] and the references there.

The problem of finding the price (or lower and upper bounds in model-free context) of a variance swap option becomes the problem of finding 8.12.

Note that the variance swap option has neither a piecewise linear nor a piecewise concave payoff. In this case we make a linear approximation in each of the polytopes. In this section we consider 0 bid-ask spread and as before, the spot price is 46. In Table 8.16 we present numerical results for a variance swap for different dimensions and strike steps. We use a fixed time step of 6 months to produce results, i.e. the prices of European call options with maturities $i \times 6M, i = 1, \ldots, n$ are used as market input, where n is the dimension.

Compared to Table 8.16, where we consider a fixed time steps, in Table 8.17 we present results for variance swap options with a fixed maturity of 1 year. Hence, for dimension 2 the prices of call options with maturities $6M$ and $1Y$ are used. For dimension 3, we use the prices of call options with maturities $4M$, $8M$ and $1Y$. Similarly, for dimension 4 prices of call options with maturities $3M$, $6M$, $9M$ and $1Y$ are used as input. The last column of the table represents the upper bound of the plain vanilla variance swap option that is calculated with the method described in [19]. The ∞ sign in the table means that the payoff of the option is calculated in the limit for $n \to \infty$.

Remark 8.4.1 *In this section we used interior-point method for solving the linear optimisation problems. The missing optimal values in Tables 8.16 and 8.17 are due to the large size of the problems. Though the Nesterov's method described in Section 6.5 can solve some of these problems, a single run of the algorithm only states if the optimal value of the problem is larger or smaller than a given value. One can to use a bisection algorithm to find the upper or lower bounds of the option price.*

dK \ Dimension	**2**		**3**		**4**	
	Lower	Upper	Lower	Upper	Lower	Upper
2	0.0352	0.1151	0.0390	0.0635	-	-
4	0.0354	0.1676	0.0364	0.0729	0.0365	0.2432

Table 8.16: Lower and upper bounds of plain vanilla variance swap option for different dimensions and strike steps with fixed time steps.

dK \ Dimension	**2**		**3**		**4**		∞
	Lower	Upper	Lower	Upper	Lower	Upper	Upper
2	0.0510	0.1297	0.0426	0.1550	-	-	0.199
4	0.0506	0.1399	0.0425	0.1727	0.0383	-	0.279
8	0.0509	0.1697	0.0430	0.2267	0.0397	0.2790	0.498

Table 8.17: Lower and upper bounds of plain vanilla variance swap option for different dimensions and strike steps with fixed maturity of 1 year. The Monte Carlo value of the option is 0.09.

Chapter 9

Additional Algorithms

In this chapter we present three algorithms for solving the optimisation problem described in Section 4.4. Although we haven't implemented these algorithms, they may be of interest for future research and development.

9.1 Models with Rank Minimisation

Recall the marginal problem from Section 3.1:

$$\sup_{\mathcal{F}\in\mathcal{M}_{F_1,\dots,F_n}} \int_{\mathbb{R}^n} h(x)d\mathcal{F}(x), \tag{9.1}$$

One way to solve this problem numerically is to discretize the distribution and solve it as a linear programming problem.
Suppose $h(x) = 1_{\sum_{i=1}^n x_i \geq s}(x)$ for some threshold s and denote

$$\tau_j^i := Pr\big[X_i \in (x_j^i, x_{j+1}^i]\big], j = 1, 2, ..., m_i$$

$$\mu_{(i_1,\dots,i_n)} := Pr\big[X_1 \in (x_{i_1}^1, x_{i_1+1}^1], ..., X_n \in (x_{i_n}^n, x_{i_n+1}^n]\big],$$

so τ is the discretisation of marginal distributions and μ is the discretisation of the joint distribution. The problem of maximising the probability of exceeding the threshold then becomes

$$\begin{array}{ll} \max_\mu & \sum_{\{(i_1,i_n):x_i^1+x_{i_n}^n>s\}} \mu_{i_1,\dots,i_n} \\ \text{s.t.} & \sum_{(i_1,\dots,i_{k-1},i_{k+1},\dots,i_n)} \mu_{(i_1,\dots,i_n)} = \tau_{i_k}^k, \quad i_k = 1, ..., m_k; \quad k = 1, ..., n \\ & \mu_{i_1,\dots,i_n} \geq 0 \end{array} \tag{9.2}$$

The number of variables increases significantly with the increase of the number of variables or the accuracy of discretisation. Even for the four-dimensional case, discretising marginal distributions in 100 grid points makes the number of variables 100 million, which cannot be solved efficiently with known LP solvers. This is where the duality result introduced earlier becomes useful.

By solving LP (9.2) for $n = 2$ and plotting the support of the measure μ we get Figure 9.1. From Figure 9.1 or using MATLAB function rank(), one can see that

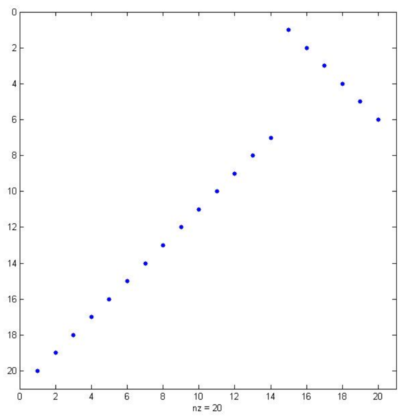

Figure 9.1: Suport of the probability measure in worst case scenario

the matrix has a full rank when probability of exceeding the threshold is maximum. On the other hand, the probability is at a minimum when the two random variables are independent, i.e. the matrix has rank 1. In general, if the rank of the matrix corresponding to the joint distribution is large, then the covariance between them is small. Therefore, by controlling the rank of the matrix corresponding to the joint distribution, we can control the dependence between them, since we want to analyse all of the possible distributions that are consistent with the given partial information, and not only extreme cases calculated by practitioners and regulators. This leads us to investigate the following model:

$$(M1)\quad \begin{array}{rl} \max_{\mu} & risk_s(\mu) - \lambda Rank(\mu) \\ \text{s.t.} & \mu \text{ is consistent with observed data} \end{array}$$

where $risk_s(\mu)$ is the risk measure at level s.

The constraint on μ to be consistent with the observed data can be different; for instance, it can be an equality constraint for marginals or moments, or have the structure discussed in Section 4.4.

The rank minimisation problem arises in many areas such as machine learning, signal processing, statistics and combinatorial optimization, and it is well known to be an NP-hard problem. A common approach to the rank minimisation problem is to minimise the convex envelope, which in this case turns out to be the nuclear norm of the matrix.

Definition 9.1.1 *The convex envelope of $f : \mathcal{C} \to \mathbb{R}$ is the largest convex function g such that $g(x) \le f(x)$ for all $x \in \mathcal{C}$.*

Definition 9.1.2 $||X||_* = \sum_{i=1}^{\min(m,n)} \sigma_i(X)$ *is called nuclear norm of X, where X is an $m \times n$ matrix and $\sigma_i(X)$ are the non-zero singular values of X.*

Theorem 9.1.3 (Theorem 1, [26]) *On the set $\mathcal{S} = \{X \in \mathbb{R}^{m\times n} | ||X||_2 \le 1\}$, the convex envelope of the function $rank(X)$ is $||X||_* = \sum_{i=1}^{\min(m,n)} \sigma_i(X)$, where $||X||_2$ is the operator norm of X, i.e. the largest singular value of X.*

This means that, if we assume that the feasible set is bounded by some number M, i.e. $||X|| \le M$ for all feasible X, then the convex envelope of $Rank(X)$ will be $\frac{1}{M}||X||_*$. Theorem 9.1.3 provides evidence that the minimisation of the nuclear norm can be a good approximation for the matrix rank minimisation problem. In this framework, our problem becomes

$$\begin{array}{rl} \max_\mu & risk_s(\mu) - \lambda||\mu||_* \\ \text{s.t.} & \mu \text{ is consistent with observed data.} \end{array} \tag{9.3}$$

In the following subsection we will introduce some existing approaches for the nuclear norm minimisation problem, following which we will adopt one of these approaches, the semidefinite programming approach, to perform some numerical experiments for our model.

9.1.1 Semidefinite Programming Formulation

In her PhD thesis, Fazel M. [26] provides a method for solving the nuclear norm minimisation problem by formulating the problem in a semidefinite programming context. Here we reproduce the main theorem from her work, which provides this framework.

Theorem 9.1.4 (Lemma 2, [26]) *For $X \in \mathbb{R}^{m\times n}$ and $t \in \mathbb{R}$, we have $||X||_* \le t$ if and only if there exist matrices $Y \in \mathbb{R}^{m\times m}$ and $Z \in \mathbb{R}^{n\times n}$ such that*

$$\begin{bmatrix} Y & X \\ X^T & Z \end{bmatrix} \ge 0, \qquad TrY + TrZ \le 2t. \tag{9.4}$$

Hence our optimization problem can be formulated in the following framework

$$
\begin{array}{rl}
\max_{\mu,Y,Z} & risk_s(\mu) - \lambda(TrY + TrZ) \\
\text{s.t.} & \begin{bmatrix} Y & \mu \\ \mu^T & Z \end{bmatrix} \geq 0, \\
 & \mu \text{ is consistent with observed data.}
\end{array}
$$

Here, however, the number of variables is $(m+n)(m+n+1)/2$, which makes the method impractical, because semidefinite programming solvers such as SeDuMi or SDPT3 cannot solve the problem when m and n are larger then 100.

9.1.2 The Fixed Point and Bregman Iterative Methods

In this section we introduce the Fixed Point and Bregman Iterative methods to solve the rank minimisation problem (for more details see [45])

$$
\begin{array}{rl}
\max_{\mu} & ||X||_* \\
\text{s.t.} & \mathcal{A}(X) = b,
\end{array}
\tag{9.5}
$$

where, as before, $X \in \mathbb{R}^{m\times n}$ and $\mathcal{A} : \mathbb{R}^{m\times n} \to \mathbb{R}^p$ is a linear map, and $b \in \mathbb{R}^p$ is a given vector.
Consider the following optimisation problem:

$$
\min_{X\in\mathbb{R}^{m\times n}} \nu||X||_* + \frac{1}{2}||\mathcal{A}(X) - b||_2^2. \tag{9.6}
$$

Instead of solving problem (9.5), we will solve problem (9.6) for a decreasing sequence of ν, since when $\nu \to 0$ the solution set of (9.6) converges to the solution set of (9.5).

9.1.2.1 The Fixed Point Method

We denote a subdifferential of a function f by ∂f and denote the gradient of $\frac{1}{2}||\mathcal{A}(X) - b||_2^2$ at the point X by $g(X) = \mathcal{A}^*(\mathcal{A}(X) - b)$.
Since the objective function in (9.6) is convex, X^* is the optimal solution to (9.6) if and only if

$$
0 \in \nu\partial||X^*||_* + g(X^*). \tag{9.7}
$$

If the Singular Value Decomposition (SVD) of a matrix $X \in \mathbb{R}^{m\times n}$ is $X = U\Sigma V^T$, where $U \in \mathbb{R}^{m\times r}$, $\Sigma = Diag(\sigma) \in \mathbb{R}^{r\times r}$, $V \in \mathbb{R}^{n\times r}$, then by [1]

$$
\partial||X^*||_* = \{UV^T + W : U^T W = 0, WV = 0, ||W||_2 \leq 1\}. \tag{9.8}
$$

This provides us with the following theorem for the optimality condition:

Theorem 9.1.5 (Theorem 2, [45]) *The matrix $X \in \mathbb{R}^{m\times n}$ with a singular value decomposition $X = U\Sigma V^T$, $U \in \mathbb{R}^{m\times r}$, $\Sigma = Diag(\sigma) \in \mathbb{R}^{r\times r}$, $V \in \mathbb{R}^{n\times r}$, is optimal for the problem* (9.6) *if and only if there exists a matrix $W \in \mathbb{R}^{m\times n}$ such that*

$$\nu(UV^T + W) + G(X) = 0,$$

$$U^T W = 0, WV = 0, ||W||_2 \leq 1.$$

Let

$$Y^* = X^* - \tau g(X^*),$$

then (9.7) is equivalent to

$$0 \in \tau\nu\partial||X^*||_* + X^* - Y^*, \tag{9.9}$$

i.e. X^* is the optimal solution to

$$\min_{X\in\mathbb{R}^{m\times n}} \tau\nu||X||_* + \frac{1}{2}||X - Y*||_F^2. \tag{9.10}$$

In the following definition we will define the matrix shrinkage operator and introduce a theorem which states that the matrix shrinkage operator applied to Y^* provides the optimal solution to (9.10).

Definition 9.1.6 (Matrix Shrinkage Operator) *Assume $X \in \mathbb{R}^{m\times n}$ with SVD $X = UDiag(\sigma)V^T$, $U \in \mathbb{R}^{m\times r}$, $\sigma \in \mathbb{R}^r_+$, $V \in \mathbb{R}^{n\times r}$. For any $\nu > 0$, the matrix shrinkage operator $S_\nu(\cdot)$ is defined as*

$$S_\nu(X) := UDiag(\bar{\sigma})V^T, \text{ with } \bar{\sigma} = s_\nu(\sigma), \tag{9.11}$$

where $s_\nu(\sigma)$ is defined as

$$s_\nu(\sigma) := \bar{\sigma}, \text{ with } \bar{\sigma}_i = \begin{cases} \sigma_i - \nu, & \text{if } \sigma_i - \nu > 0 \\ 0, & \text{otherwise} \end{cases} \tag{9.12}$$

and is called nonnegative vector shrinkage operator.

Theorem 9.1.7 (Theorem 3, [45]) *Assume a matrix $Y \in \mathbb{R}^{m\times n}$ and a scalar $\nu > 0$. $X := S_\nu(Y)$ is then an optimal solution of the problem*

$$\min_{X\in\mathbb{R}^{m\times n}} \tau\nu||X||_* + \frac{1}{2}||X - Y||_F^2. \tag{9.13}$$

The following corollary provides us with a framework for the iterative algorithm for solving the problem (9.6).

Corollary 9.1.8 (Corollary 1, [45]) *X^* is an optimal solution to problem (9.6) if and only if $X^* = S_{\tau\nu}(h(X^*))$, where $h(\cdot) = I(\cdot) - \tau g(\cdot)$.*

This gives us the following theorem:

Theorem 9.1.9 *The sequence X^k generated by the following algorithm:*

$$\begin{cases} Y^k = X^k - \tau g(X^k) \\ X^{k+1} = S_{\tau\nu}(Y^k) \end{cases} \tag{9.14}$$

converges to the optimal solution of (9.6) if $\tau \in (0, 2/\lambda_{max}(A^T A))$, where matrix A is defined so that $\mathcal{A}(X) = Avec(X)$.

Proof of this result is based on the fact that operators $S_{\tau\nu}(\cdot)$ and $I(\cdot) - \tau g(\cdot)$ are nonexpansive, i.e.

$$||S_{\tau\nu}(X_1) - S_{\tau\nu}(X_2)||_F \leq ||X_1 - X_2||_F, \tag{9.15}$$

$$||X_1 - \tau g(X_1) - X_2 + \tau g(X_2)||_F \leq ||X_1 - X_2||_F, \tag{9.16}$$

for $\forall X_1, X_2 \in \mathbb{R}^{m\times n}$.

To solve the original problem (9.5), we need to choose a small ν and implement the above-described algorithm to (9.6). As described in [45], however, by taking a larger ν and decreasing its value in a homotopy we can increase rate of convergence of the algorithm. This is called a fixed point continuation.

Algorithm 9.1.10 (Fixed Point Continuation)

- *Initialise: Given $X_0, \bar{\nu} > 0$. Select $\nu_1 > \nu_2 > ... > n_L = \bar{\nu} > 0$. Set $X = X_0$*
- *For $\nu = \nu_1, \nu_2, ..., \nu_L$, do*
  - *while NOT converged, do*
    - *select $\tau > 0$*
    - *compute $Y = X - \tau\mathcal{A}^*(\mathcal{A}(X) - b)$, and SVD of Y, $Y = UDiag(\sigma)V^T$*
    - *compute $X = UDiag(s_{\nu\tau}(\sigma))V^T$*
  - *end while*
- *end for*

Sequence ν_k can be decreased by $\nu_{k+1} = \eta\nu_k$, and as a stopping criteria in the inner loop we can choose either

$$||U_k V_k^T + g^k \nu||_2 - 1 \leq gtol, \tag{9.17}$$

or

$$\frac{||X^{k+1} - X^k||_F}{\max\{1, ||X^k||_F\}} \leq xtol, \tag{9.18}$$

where $gtol$ and $xtol$ are small positive parameters representing the tolerance level.

9.1.2.2 Bregman's Iterative Algorithm

As mentioned above, the Fixed Point Continuation algorithm (FPC) solves the unconstrained minimisation problem (9.6) for a fixed ν. In order to solve the constrained problem (9.5), we need to implement a homotopy variant of the FPC algorithm for a decreasing values $\nu_k \to 0$. However, by incorporating FPC into Bregman's iterative technique, we can solve (9.5) by solving a limited number of instances (9.6) for different b's. The method is described below.
Given a convex function $J(\cdot)$, the Bregman distance [8] of the point u from the point v is defined as

$$D_J^p(a, v) := J(u) - J(v) - < p, u - v >, \tag{9.19}$$

where $p \in \partial J(v)$ is some subgradient in the subdifferential of J at point v, i.e.

$$p \in \partial J(v) := \{r : J(u) - J(v) \geq < r, u - v >, \forall u\}.$$

The iterative scheme proposed in [45] is the following

$$\begin{gathered} X^{k+1} \leftarrow \min_X D_J^{p^k}(X, X^k) + \tfrac{1}{2}||\mathcal{A}(X) - b||_2^2 \\ p^{k+1} := p^k - \mathcal{A}^*(\mathcal{A}(X^{k+1}) - b), \end{gathered} \tag{9.20}$$

which is equivalent to the following scheme

$$\begin{gathered} b^{k+1} \leftarrow b + (b^k - \mathcal{A}(X^k)) \\ X^{k+1} \leftarrow arg\min_X \nu||X|_* + \tfrac{1}{2}||\mathcal{A}(X) - b^{k+1}||_2^2. \end{gathered} \tag{9.21}$$

So the algorithm is the following

Algorithm 9.1.11 (Bregman's Iterative Algorithm)

- $b^0 \leftarrow 0, X^0 \leftarrow 0,$
- *for* $k = 0, 1, \cdots$ *do*

- $b^{k+1} \leftarrow b + (b^k - \mathcal{A}(X^k))$
- $X^{k+1} \leftarrow arg\min_X \nu||X|_* + \frac{1}{2}||\mathcal{A}(X) - b^{k+1}||_2^2.$

The last step of the algorithm can be solved with FPC.
Note that Fixed Point and Bregman's iterative methods are efficient only for low rank matrices, whereas the two dimensional example of worst case VaR scenarios introduces a full rank matrix. However, this issue can be easily overcome, since in our model we want to maximise the tail events, which leaves room for flexibility at the beginning of the distribution function, and we can construct this part of the distribution to get a low rank submatrix. In the following subsection we introduce some numerical results, following which these arguments will become clearer.

9.1.3 Numerical Experiments

In this section we introduce some numerical experiments for rank minimisation models. Assume we are given a marginal distribution for two risks, which are simulated as a Generalized Champernowne distribution (2.3).
We use semidefinite programming formulation to find the optimal solution for (9.3) for $\lambda \in [0, 7]$. The probability for exceeding the given threshold doesn't increase for $\lambda > 7$ which is the probability when one maximises it without minimising the rank. Figure 9.2 shows the dependency of the maximal probability from parameter λ.
One of the issues between regulators' and practitioners' approaches is that theoretical worst case distributions for the marginal problem are very artificial and are very unlikely to occur in practice. Figure 9.3 shows a two dimensional distribution function for the worst case scenario (log scale), from which one can see the non-smooth behaviour of the function.
But from Figure 9.4 we can see that, when $\lambda = 2$, the maximal probability (0.0048) is sufficiently close to the worst case scenario probability, but has a smoother shape, as shown in Figure 9.4.

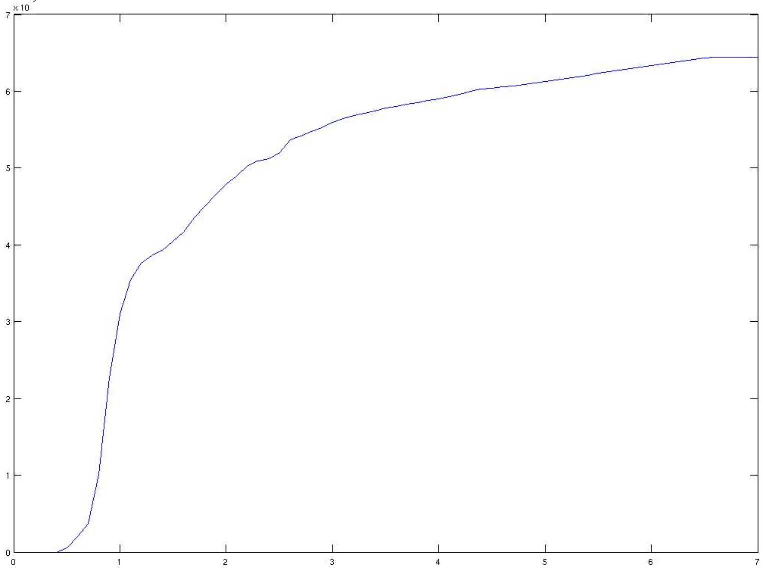

Figure 9.2: Dependence of the maximal probability from λ.

From these experiments we conclude that, although worst case scenarios do not represent a typical distribution in practice, using the rank minimisation technique we can get smoother distribution functions closer to reality which produce similar probabilities of extreme losses. This leads us to a new topic of research for higher dimensional problems.

9.2 Grouping of Partitions

In many practical applications we have cases in which we can choose subsets of the constraint functions ϕ_i and ψ_j (model in Chapter 7), such that the supports of these functions do not intersect (or intersect only on a hyperline). This is the case, for example, with marginal constraints, where we can choose $n + 1$ (where n is the dimension of the problem) subsets, each of the subsets containing constraint functions for the particular dimension. We will also choose one more set for the objective function h.

In this section we will describe an algorithm for making use of this information in a way that avoids a large number of constraints in the resulting linear programming problem.

To render the underlying of the algorithm more readable, we will only treat the illustrative two-dimensional case with equality constraints.

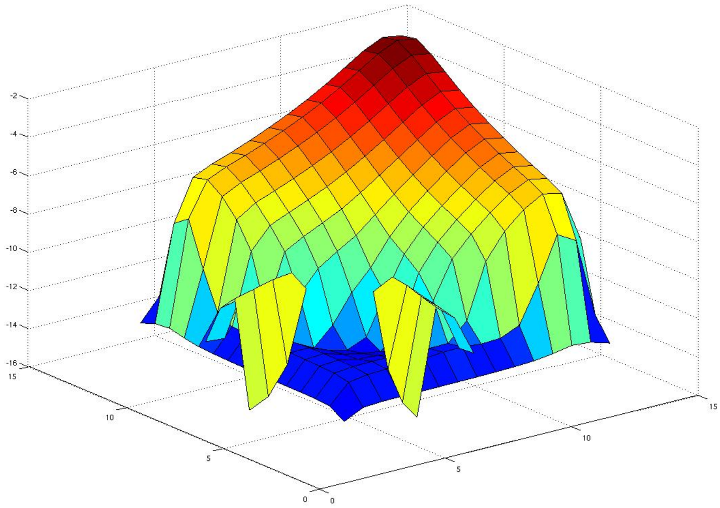

Figure 9.3: Distribution function for the worst case scenario, probability for exceeding the threshold is 0.0064, log scale plot

Assume the partitioning $\{\Pi_i, i = 1, \dots, K\}$, and the objective function h, and the constraints are of such a form that the original problem can be written in the following form:

$$
\begin{aligned}
(P) \qquad \sup_{\mathcal{F} \in \mathscr{M}_{\mathfrak{F}}} & \int_H h(x) \, \mathrm{d}\, \mathcal{F}(x) \\
\text{s.t.} \; & \int_{\Phi_s} \phi_s(x) \, \mathrm{d}\, \mathcal{F}(x) = a_s, \quad (s = 1, \dots, M), \\
& \int_{\Psi_t} \psi_t(x) \, \mathrm{d}\, \mathcal{F}(x) = b_t, \quad (t = 1, \dots, N), \\
& \int_{\Phi} 1 \, \mathrm{d}\, \mathcal{F}(x) = 1, \\
& \mathcal{F} \geq 0,
\end{aligned}
$$

where $\Psi_i \cap \Psi_j = \emptyset, \quad \Phi_i \cap \Phi_j = \emptyset, i \neq j$.
This is true, for example, in the case of marginal problems, where each of the Φ_i will be the column representing the segment at one direction and each of the Ψ_i will be the row representing the other dimension.
In the standard form of our numerical algorithm described in 4.4, we had to deal with all the intersections of all the partitions. We now show how this can be avoided, since having many partitions introduces a large number of constraints in the dual problem.

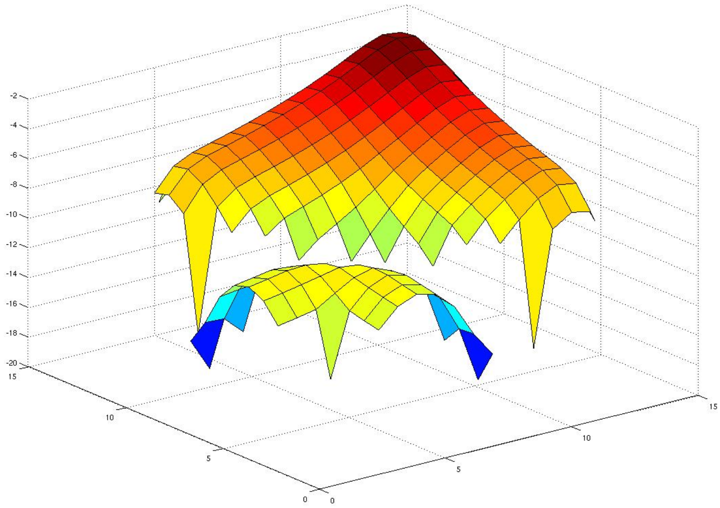

Figure 9.4: Distribution function for $\lambda = 2$, probability for exceeding the threshold is 0.0048, log scale plot

Note that problem (P) is equivalent to

$$
\begin{aligned}
(P^3) \qquad & \sup_{\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3 \in \mathscr{M}_{\mathfrak{F}}} \int_H h(x)\,\mathrm{d}\,\mathcal{F}_3(x) \\
& \text{s.t.} \int_{\Phi_s} \phi_s(x)\,\mathrm{d}\,\mathcal{F}_1(x) = a_s, \quad (s = 1,\dots,M), \\
& \int_{\Psi_t} \psi_t(x)\,\mathrm{d}\,\mathcal{F}_2(x) = b_t, \quad (t = 1,\dots,N), \\
& \int_{\Phi} 1\,\mathrm{d}\,\mathcal{F}_3(x) = 1, \\
& \mathcal{F}_1 = \mathcal{F}_2 = \mathcal{F}_3 \geq 0.
\end{aligned}
$$

As before, let $\{\Pi_i\}, i = 1,\dots K$ be the set of all the original partitions. Denote $T_1 = \Pi_1$ and construct $T_k = T_{k-1} \cup \Pi_k$.

Consider the following problem:

$$
\begin{aligned}
(P^E) \qquad \sup_{\mathcal{F}_1,\mathcal{F}_2,\mathcal{F}_3 \in \mathscr{M}_{\mathfrak{F}}} \quad & \int_H h(x)\,\mathrm{d}\,\mathcal{F}_3(x) \\
\text{s.t.} \quad & \int_\Phi 1\,\mathrm{d}\,\mathcal{F}_3(x) = 1, \\
& \int_{\Phi_s} \phi_s(x)\,\mathrm{d}\,\mathcal{F}_1(x) = a_s, \quad (s = 1,\dots,M), \\
& \int_{\Psi_t} \psi_t(x)\,\mathrm{d}\,\mathcal{F}_2(x) = b_t, \quad (t = 1,\dots,N), \\
& \int_{T_i} [\mathrm{d}\,\mathcal{F}_1(x) - \mathrm{d}\,\mathcal{F}_2(x)] = 0, \quad i = 1,\dots K, \\
& \int_{T_i} [\mathrm{d}\,\mathcal{F}_1(x) - \mathrm{d}\,\mathcal{F}_3(x)] = 0, \quad i = 1,\dots K, \\
& \mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3 \geq 0.
\end{aligned}
$$

Note that the last two equality constraints above do not guarantee that $\mathcal{F}_1 = \mathcal{F}_2 = \mathcal{F}_3$, so (P^E) is a relaxation of (P^3). As we will see after constructing the dual form of (P^E), however, problems (P) and (P^E) are equivalent.
To do this we again use the general duality framework from Chapter 4.
Let $\bar{\Phi} := \{1,2,3\} \times \Phi$ and

$$
\Sigma = \{0, 1, \dots, M, \dots, M+N, \dots, M+N+2K\}.
$$

Construct the objective and the constraint functions as

$$
c(i,x) = \begin{cases} h(x), & \text{if } i = 3, x \in H \\ 0, & \text{otherwise.} \end{cases}
$$

$$
B(z,1,x) = \begin{cases} 1, & \text{if } z = 0, x \in \Phi \\ \phi_z(x), & \text{if } z = 1,\dots,M, x \in \Phi_z \\ 1, & \text{if } M+N < z \leq M+N+K, x \in T_{z-M-N} \\ 1, & \text{if } M+N+K < z \leq M+N+2K, x \in T_{z-M-N-K} \\ 0, & \text{otherwise.} \end{cases}
$$

$$
B(z,2,x) = \begin{cases} \psi_{z-M}(x), & \text{if } z = M+1,\dots,M+N, x \in \Psi_{z-M} \\ -1, & \text{if } M+N < z \leq M+N+K, x \in T_{z-M-N} \\ 0, & \text{otherwise.} \end{cases}
$$

$$
B(z,3,x) = \begin{cases} -1, & \text{if } M+N+K < z \leq M+N+2K, x \in T_{z-M-N-K} \\ 0, & \text{otherwise.} \end{cases}
$$

$$b(z) = \begin{cases} 1, & \text{if } z = 0 \\ a_z, & \text{if } z = 1, \dots, M, \\ b_{z-M}, & \text{if } z = M+1, \dots, M+N, \\ 0, & \text{otherwise.} \end{cases}$$

With this construction, it is easy to show that in our framework the dual problem becomes

$$\begin{aligned} \inf_{s \in \mathbb{R}^{M+1}, t \in \mathbb{R}^N, \tau, \delta \in \mathbb{R}^K} & \sum_{i=1}^{M} s_i a_i + \sum_{i=1}^{N} t_i b_i + s_0 \\ \text{s.t. } & \sum_{i=1}^{M} s_i \phi_i(x) + \sum_{i=1}^{K} \tau_i + \sum_{i=1}^{K} \delta_i \geq 0, \quad \forall x \in \Phi, \\ & \sum_{i=1}^{N} t_i \psi_i(x) - \sum_{i=1}^{K} \tau_i \geq 0, \quad \forall x \in \Phi, \\ & s_0 - \sum_{i=1}^{K} \delta_i \geq h(x) \quad \forall x \in \Phi. \end{aligned}$$

Denoting $\tau = \sum_{i=1}^{K} \tau_i$ and $\delta = \sum_{i=1}^{K} \delta_i$ we get a much simpler linear programming problem:

$$\begin{aligned} (D^E) \qquad \inf_{s \in \mathbb{R}^{M+1}, t \in \mathbb{R}^N, \tau, \delta \in \mathbb{R}} & \sum_{i=1}^{M} s_i a_i + \sum_{i=1}^{N} t_i b_i + s_0 \\ \text{s.t. } & \sum_{i=1}^{M} s_i \phi_i(x) + \tau + \delta \geq 0, \quad \forall x \in \Phi, \\ & \sum_{i=1}^{N} t_i \psi_i(x) - \tau \geq 0, \quad \forall x \in \Phi, \\ & s_0 - \delta \geq h(x) \quad \forall x \in \Phi. \end{aligned}$$

Note that the dual problem is not dependent on the choice of T_i, so we would get the same linear programming problem for any finite set of T_i. From the latter fact we conclude that, by solving (D^E), we will obtain the optimal value of (P^E), since we can choose any large number of partitions T_i.
When ϕ_i, ψ_j and h are chosen as linear functions, and Φ is a box, this scheme produces a problem with $M + N + 3$ variables and $2^n(M + N + 1)$ constraints. Note that in the standard form we would have many more constraints, for example, for two the dimensional marginal problem we would have $M + N + 1$ variables and $2^n MN$ constraints.

9.3 The Polynomial Optimisation Approach

Consider again the problem from Section 4.4 with piecewise linear functions ϕ_s, ψ_t, h, $s = 1, \ldots, M$, $t = 1, \ldots, N$.

$$
\begin{aligned}
\text{(P)} \quad \sup_{\mathcal{F} \in \mathscr{M}_{\mathfrak{F}}} & \int_{\Phi} h(x) \, \mathrm{d}\, \mathcal{F}(x) \\
\text{s.t.} & \int_{\Phi} \phi_s(x) \, \mathrm{d}\, \mathcal{F}(x) \leq a_s, \quad (s = 1, \ldots, M), \\
& \int_{\Phi} \psi_t(x) \, \mathrm{d}\, \mathcal{F}(x) = b_t, \quad (t = 1, \ldots, N), \\
& \int_{\Phi} 1 \, \mathrm{d}\, \mathcal{F}(x) = 1, \\
& \mathcal{F} \geq 0,
\end{aligned}
$$

where we discuss the problem for $\Phi = \cup \Pi_\kappa \subset \mathbb{R}^n$, where for $\kappa = (\kappa_1, \ldots, \kappa_n)$ we denote $\Pi_\kappa := [l^1_{\kappa_1 - 1}, l^1_{\kappa_1}] \times [l^2_{\kappa_2 - 1}, l^2_{\kappa_2}] \times \cdots \times [l^n_{\kappa_n - 1}, l^n_{\kappa_n}]$.

In this section we assume that we know in advance that some groups of risks are mutually independent from each other. This information can be obtained either from an expert opinion or, in the case of stock prices, from the independence of different industries.

Without loss of generality, assume that these groups are $Y_1 := (X_1, \ldots, X_{n_1})$, $Y_2 := (X_{n_1+1}, \ldots, X_{n_2})$, $\ldots$, $Y_m := (X_{n - n_{m-1} + 1}, \ldots, X_n)$.

Denote the restriction of function ϕ_s on a box Π_κ by ϕ_s^κ, which is linear by the definition of functions ϕ_s. We can then write

$$
\phi_s^\kappa(x_1, \ldots, x_n) = \phi_{s,1}^\kappa(y_1) + \phi_{s,2}^\kappa(y_2) + \cdots + \phi_{s,m}^\kappa(y_m), \tag{9.22}
$$

since linear functions decouple.

Further, the independence of $Y_1, \ldots, Y_m$ implies

$$
\mathrm{d}\, \mathcal{F}(x_1, \ldots, x_n) = \mathrm{d}\, \mathcal{F}_1(y_1) \cdot \cdots \cdot \mathrm{d}\, \mathcal{F}_m(y_m).
$$

For each integral in the constraints we therefore have

$$
\begin{aligned}
& \int_{\Phi} \phi_s(x) \, \mathrm{d}\, \mathcal{F}(x) = \sum_{\kappa} \int_{\Pi_\kappa} \phi_s^\kappa(x) \, \mathrm{d}\, \mathcal{F}(x) = \\
& \sum_{\kappa} \int_{\Pi_\kappa} \left(\phi_{s,1}^\kappa(y_1) + \cdots + \phi_{s,m}^\kappa(y_m) \right) \mathrm{d}\, \mathcal{F}_1(y_1) \cdot \ldots \mathrm{d}\, \mathcal{F}_m(y_m) = \\
& \sum_{\kappa} \Bigg(\int_{\Pi_\kappa} \phi_{s,1}^\kappa(y_1) \, \mathrm{d}\, \mathcal{F}_1(y_1) \cdot \cdots \cdot \mathrm{d}\, \mathcal{F}_m(y_m) + \cdots + \\
& \int_{\Pi_\kappa} \phi_{s,m}^\kappa(y_m) \, \mathrm{d}\, \mathcal{F}_1(y_1) \cdot \ldots \mathrm{d}\, \mathcal{F}_m(y_m) \Bigg)
\end{aligned}
$$

Denote $z^\kappa_{s,j} := \int_{\Pi_\kappa|_{y_j}} \phi^\kappa_{s,j}(y_j)\,\mathrm{d}\,\mathcal{F}_j(y_j)$ and $\tau^\kappa_j := \int_{\Pi_\kappa|_{y_j}} \mathrm{d}\,\mathcal{F}_j(y_j)$ for $j = 1,\dots,m$. We obtain

$$\int_\Phi \phi_s(x)\,\mathrm{d}\,\mathcal{F}(x) = \sum_\kappa \left(z^\kappa_{s,1}\tau^\kappa_2 \dots \tau^\kappa_m + \dots + z^\kappa_{s,m}\tau^\kappa_1 \dots \tau^\kappa_{m-1}\right) \tag{9.23}$$

Extending this notation, the original problem (P) becomes

$$\begin{aligned}
\sup_{z,w,h,\tau,\mathcal{F}} \quad & \sum_\kappa \left(h^\kappa_1\tau^\kappa_2 \dots \tau^\kappa_m + \dots + h^\kappa_m\tau^\kappa_1 \dots \tau^\kappa_{m-1}\right) \\
& \sum_\kappa \left(z^\kappa_{s,1}\tau^\kappa_2 \dots \tau^\kappa_m + \dots + z^\kappa_{s,m}\tau^\kappa_1 \dots \tau^\kappa_{m-1}\right), \quad (s = 1,\dots M) \\
& \sum_\kappa \left(w^\kappa_{t,1}\tau^\kappa_2 \dots \tau^\kappa_m + \dots + w^\kappa_{t,m}\tau^\kappa_1 \dots \tau^\kappa_{m-1}\right), \quad (t = 1,\dots n) \\
& \sum_\kappa \left(\tau^\kappa_1\tau^\kappa_2 \dots \tau^\kappa_m\right) = 1,
\end{aligned}$$

where

$$\begin{aligned}
h^\kappa_j &:= \int_{\Pi_\kappa|_{y_j}} h^\kappa_j(y_j)\,\mathrm{d}\,\mathcal{F}_j(y_j), \quad \forall j,\kappa \\
w^\kappa_{t,j} &:= \int_{\Pi_\kappa|_{y_j}} \psi^\kappa_{t,j}(y_j)\,\mathrm{d}\,\mathcal{F}_j(y_j), \quad \forall t,j,\kappa.
\end{aligned}$$

With this procedure we have constructed a polynomial optimisation problem which, in the worst case, has more decision variables than the dual formulation of the original problem. As we observe in many practical applications and, in particular, in our model for risk management, however, this leads to a smaller sized problem.
Note that this method also works when, instead of having linear functions in the constraints, we have functions that can be introduced in the following form:

$$\phi^\kappa_s(y_1,\dots,y_m) = \sum_{i=1}^{l} \phi^\kappa_{s,i,1}(y_1)\cdot\dots\cdot\phi^\kappa_{s,i,m}(y_m),$$

which is particularly practical for piecewise quadratic functions.

9.4 Dimension Reduction Technique

We have already seen that, for higher dimensions, direct implementation of the standard numerical techniques is not reasonable in practice because of the size of the linear programming problem. In this section we introduce an algorithm which yields comparatively simpler problem.

As in Chapter 7, we assume Φ to be a box in $\mathbb{R}^n$ and the partition Π_i, $i = 1, \ldots, K$ to be such that $\cup \Pi_i = \Phi$, $\cap \Pi_i = \emptyset$.

Recall that we require a numerical solution to problem (P) from Section 4.4.

$$
\begin{aligned}
(P) \quad & \sup_{\mathcal{F} \in \mathscr{M}_{\mathfrak{F}}} \int_{\Phi} h(x) \, \mathrm{d}\,\mathcal{F}(x) \\
& \text{s.t.} \int_{\Phi} \phi_s(x) \, \mathrm{d}\,\mathcal{F}(x) \leq a_s, \quad (s = 1, \ldots, M), \\
& \text{s.t.} \int_{\Phi} \psi_t(x) \, \mathrm{d}\,\mathcal{F}(x) = b_t, \quad (t = 1, \ldots, N), \\
& \int_{\Phi} 1 \, \mathrm{d}\,\mathcal{F}(x) = 1, \\
& \mathcal{F} \geq 0,
\end{aligned}
$$

Here we consider a case in which the functions h, ϕ_s and ψ_t are piecewise linear on the partition Π_i, $i = 1, \ldots, K$.

Define the linear functions $h^i, \phi_s^i, \psi_t : \Pi_i \to \mathbb{R}$ as

$$h^i(x) = h(x), \quad \text{if } x \in \Pi_i \tag{9.24}$$

$$\phi_s^i(x) = \phi_s(x), \quad \text{if } x \in \Pi_i \tag{9.25}$$

$$\psi_t^i(x) = \psi_t(x), \quad \text{if } x \in \Pi_i, \tag{9.26}$$

$i = 1, \ldots, K$, $s = 1, \ldots, M$, $t = 1, \ldots, N$.

Taking into consideration that the polytops Π_i do not intersect, problem (P) is equivalent to

$$
\begin{aligned}
(P') \quad & \sup_{\mathcal{F} \in \mathscr{M}_{\mathfrak{F}}} \sum_{i=1}^{K} \int_{\Pi_i} h_i(x) \, \mathrm{d}\,\mathcal{F}(x) \\
& \text{s.t.} \sum_{i=1}^{K} \int_{\Pi_i} \phi_s^i(x) \, \mathrm{d}\,\mathcal{F}(x) \leq a_s, \quad (s = 1, \ldots, M), \\
& \text{s.t.} \sum_{i=1}^{K} \int_{\Pi_i} \psi_t(x) \, \mathrm{d}\,\mathcal{F}(x) = b_t, \quad (t = 1, \ldots, N), \\
& \sum_{i=1}^{K} \int_{\Pi_i} 1 \, \mathrm{d}\,\mathcal{F}(x) = 1, \\
& \mathcal{F} \geq 0.
\end{aligned}
$$

Furthermore, for any linear function $g : \mathbb{R}^n \to \mathbb{R}$ we have

$$
\begin{aligned}
g(x) = & \langle \alpha, x \rangle + \beta = \\
& \alpha_1 x_1 + \alpha_2 x_2 + \cdots + \alpha_n x_n + \beta = \\
& x_1(\alpha_1 - \alpha_2) + \alpha_2(x_1 + x_2) + \cdots + \alpha_n x_n + \beta.
\end{aligned}
$$

If we use the notation $\phi_s^i = \langle \alpha_s^i, x\rangle + \beta_s^i$ we obtain

$$\sum_{i=1}^{K} \int_{\Pi_i} \phi_s^i(x)\,\mathrm{d}\,\mathcal{F}(x) = \sum_{i=1}^{K} (\alpha_s^{i,1} - \alpha_s^{i,2}) m_i + \sum_{i=1}^{K} \int_{\Pi_i} \bar{\phi}_s^i(x)\,\mathrm{d}\,\mathcal{F}(x),$$

where $m_i = \int_{\Pi_i} \mathrm{d}\,\mathcal{F}$ and

$$\bar{\phi}_s^i(x) = \phi_s^i(x) - (\alpha_s^{i,1} - \alpha_s^{i,2}) x_1.$$

Moreover, since $\bar{\phi}_s^i(x)$ is a function of $\bar{x} = (x_1 + x_2, x_3, \ldots, x_n)$, we relax problem (P') to the following:

$$\begin{aligned}
(P'') \qquad \sup_{\mathcal{G}_1,\ldots,\mathcal{G}_K \in \mathscr{M}_{\mathfrak{G}}} \;& \sum_{i=1}^{K} \int_{\bar{\Pi}_i} h_i(\bar{x})\,\mathrm{d}\,\mathcal{G}_i(\bar{x}) + \sum_{i=1}^{K} (h_i^1 - h_i^2) m_i \\
\text{s.t.}\;& \sum_{i=1}^{K} \int_{\bar{\Pi}_i} \bar{\phi}_s^i(\bar{x})\,\mathrm{d}\,\mathcal{G}_i(y) + \sum_{i=1}^{K} (\alpha_s^{i,1} - \alpha_s^{i,2}) m_i \le a_s, \quad (s = 1, \ldots, M), \\
\text{s.t.}\;& \sum_{i=1}^{K} \int_{\bar{\Pi}_i} \bar{\psi}_t(\bar{x})\,\mathrm{d}\,\mathcal{G}_i(\bar{x}) + \sum_{i=1}^{K} (\gamma_t^{i,1} - \gamma_t^{i,2}) m_i = b_t, \quad (t = 1, \ldots, N), \\
& \sum_{i=1}^{K} \int_{\bar{\Pi}_i} 1\,\mathrm{d}\,\mathcal{G}_i(\bar{x}) = 1, \\
& \mathcal{G} \ge 0 \quad m_i \in \left[\min\{\Pi_i|_{x_1}\} \max\{\Pi_i|_{x_1}\}\right].
\end{aligned}$$

where γ_t^i are the same for ψ as α_s^i for ϕ, and

$$\bar{\Pi}_i := \{(x_1 + x_2, x_3, \ldots, x_n) : (x_1, x_2, x_3, \ldots, x_n) \in \Pi_i\}$$

Note that y is now in $n-1$ dimensional space. Moreover, most of the coefficients $(\alpha_s^{i,1} - \alpha_s^{i,2})$ and $(\gamma_t^{i,1} - \gamma_t^{i,2})$ are zero in most of the practical applications, or the quantities m_i are known for many boxes (we have many observed points in these boxes in order to approximate the average and the probability of being there), so that the new optimisation problem becomes easier to solve.

The same approach can now be applied iteratively to other pairs of variables that haven't been contracted yet, e.g., $(x_3, x_4), (x_5, x_6)$ etc..

Let us now find the dual of problem (P'') again using our framework for general duality theory from Chapter 4.

We use the following notation:

$$\Phi = \{1, \dots, 2K\} \times \bar{\Phi}$$
$$\Gamma = \{1, \dots M\}$$
$$\Sigma = \{0, \dots N\}$$
$$c(i, \bar{x}) = \begin{cases} h_i(\bar{x}), & \text{if } i = 1, \dots K, \bar{x} \in \bar{\Pi}_i \\ h^1_{i-k} - h^2_{i-K}, & \text{if } i = K+1, \dots 2K \\ 0, & \text{otherwise.} \end{cases}$$

$$A(s, i, \bar{x}) = \begin{cases} \phi^i_s(\bar{x}), & \text{if } i = 1, \dots K, \quad \bar{x} \in \bar{\Pi}_i \\ \alpha^{i-K,1}_s - \alpha^{i-K,2}_s, & \text{if } i = K+1, \dots 2K \\ 0, & \text{otherwise.} \end{cases}, \quad s = 1, \dots M$$

$$B(t, i, \bar{x}) = \begin{cases} \psi^i_t(\bar{x}), & \text{if } i = 1, \dots K, \quad \bar{x} \in \bar{\Pi}_i \\ \gamma^{i-K,1}_t - \gamma^{i-K,2}_t, & \text{if } i = K+1, \dots 2K. \\ 0, & \text{otherwise.} \end{cases}, \quad t = 1, \dots N$$

$$B(0, i, \bar{x}) = 1.$$

This gives us the following dual problem:

$$(D'') \inf_{g \in \mathbb{R}^M_+, z \in \mathbb{R}^{N+1}} \sum_{s=1}^{M} a_s g_s + \sum_{t=1}^{N} b_t z_t + z_0$$

$$\text{s.t.} \sum_{s=1}^{M} \phi^i_s(\bar{x}) g_s + \sum_{t=1}^{N} \psi^i_t(\bar{x}) z_t + z_0 \geq h_i(\bar{x}), \quad (\forall \bar{x} \in \bar{\Pi}_i, i = 1, \dots K)$$

$$\sum_{s=1}^{M} (\alpha^{i-K,1}_s - \alpha^{i-K,2}_s) g_s + \sum_{t=1}^{N} (\gamma^{i-K,1}_t - \gamma^{i-K,2}_t) z_t + z_0 \geq h^1_i - h^2_i,$$
$$i = 1, \dots K,$$

$g_s \geq 0, \quad (s = 1, \dots, M)$.

As mentioned above, $\bar{x}$ is now in $n-1$ dimensional space, which provides us with a linear programming problem with the same number of variables but with a smaller number of inequality constraints. To illustrate this point, take for example a case in which of all the Π_i are boxes in $\mathbb{R}^n$. In the original dual problem we would have $2^n K + M$ inequality constraints, whereas in the reduced problem we have $2^{n-1}K + K + M$ inequalities.

Applying the maximum number of contractions of pairs of variables, we obtain a relaxation with $2^{\frac{n}{2}} K + \frac{n}{2} K + M$ inequality constraints.

Appendix A

Finding the Closest Point on a Simplex

We can see that on each iteration of Algorithm 6.5.2 we solve an optimisation problem of the following type:

$$\min_{x \in \Delta_n} \frac{1}{2} \|x - z\|^2, \tag{A.1}$$

where $z \in \mathbb{R}^n$.

In this Appendix we will describe how to solve this problem.

Writing Karush-Kuhn-Tucker conditions for problem (A.1), we get the following system of equations:

$$\begin{array}{rcll} x_i - z_i + \lambda - \mu_i & = & 0, & i = 1, \ldots, n \\ x_i & \geq & 0, & i = 1, \ldots, n \\ \mu_i x_i & = & 0, & i = 1, \ldots, n \\ \mu_i & \geq & 0, & i = 1, \ldots, n \\ \displaystyle\sum_{i=1}^{n} x_i & = & 1, & \end{array}$$

which is equivalent to

$$\begin{array}{rcll} \text{(KKT)}\ x_i(x_i - z_i + \lambda) & = & 0, & i = 1, \ldots, n \\ x_i - z_i + \lambda & \geq & 0, & i = 1, \ldots, n \\ x_i & \geq & 0, & i = 1, \ldots, n \\ \displaystyle\sum_{i=1}^{n} x_i & = & 1. & \end{array}$$

Let (x^*, λ^*) be the optimal solution of (KKT). Denote $\mathbb{N} := \{1, \ldots, n\}$ and

$$\begin{aligned} A &= \{i \in \mathbb{N} : x_i^* > 0\}, \\ B &= \{i \in \mathbb{N} : x_i^* = 0\}. \end{aligned}$$

Then, for each $k \in A$,

$$x_k^* - z_k + \lambda^* = 0. \tag{A.2}$$

Which implies

$$\begin{aligned} \lambda^* &= \frac{\sum_A z_i - 1}{m}, \\ x_k^* &= z_k - \lambda^* > 0, \end{aligned}$$

where $m = \|A\|$, for each $k \in B$, $z_k \leq \lambda^*$. We therefore have the following algorithm for finding the optimal solution x^*.

Algorithm A.0.1

- *Sort z in ascending order (order x with respect to sorted z).*
- *Set $S = 0$.*
- *Loop $k = 1, \ldots, n$*
    1. $S \leftarrow S + z_{n-k+1}$.
    2. *If* $(z_{n-k+1} > \frac{S-1}{k} \quad AND \quad z_{n-k} \leq \frac{S-1}{k})$, *terminate.*
    3. $k \leftarrow k + 1$.
- *Output*

$$\begin{aligned} \lambda^* &= \frac{S-1}{k}, \\ x_{g(i)}^* &= \begin{cases} 0, & \text{if } i \leq n-k \\ z_i - \lambda^*, & \text{if } i > n-k, \end{cases} \end{aligned}$$

where $g(i)$ is the original index of z_i before sorting.

Appendix B

Simulated 'Onion'-Shaped Loss Data for Risk Management

Table B.1: Simulated 'Onion'-Shaped Loss Data for Dimension 2

85.4	63.4	92.4	59.4	124.2	184.1	81.4	75.1	73.1	87.1	60.7	86.4	89.8	66.4
70.0	65.7	89.8	86.0	75.0	67.5	82.7	81.2	68.9	61.1	120.8	172.1	131.4	71.3
88.4	66.8	76.5	77.5	66.2	79.0	70.0	62.6	74.7	98.6	76.6	70.5	98.2	656.9
160.9	67.0	101.8	72.3	78.7	68.7	65.8	65.4	75.7	68.3	74.4	70.7	73.1	93.0
92.3	66.5	73.9	67.6	75.3	76.4	66.7	448.9	70.1	72.6	81.7	66.0	66.1	65.8
64.5	67.2	63.1	132.1	68.6	68.8	86.9	77.1	67.8	69.8	64.6	99.4	74.4	104.3
112.6	115.5	79.5	99.1	76.4	123.9	67.3	64.7	101.1	82.4	64.7	123.8	77.1	62.5
63.9	70.7	84.8	63.3	68.2	55.1	73.6	82.7	63.5	82.5	79.7	79.8	66.3	69.5
90.1	89.5	128.1	134.9	66.5	127.3	63.5	69.2	81.9	72.3	172.5	69.4	101.6	78.0
73.5	107.8	71.1	72.4	132.9	136.6	84.3	458.4	64.9	63.5	71.1	95.7	68.1	81.6
60.7	69.6	68.9	107.1	66.7	79.4	244.0	65.9	93.5	105.2	114.8	82.0	65.3	88.1
103.0	74.1	421.0	80.1	64.2	69.1	66.3	62.6	67.2	66.1	126.3	91.5	67.1	67.5
83.0	74.9	67.5	69.9	84.8	86.1	65.3	74.1	85.6	87.5	70.5	65.8	81.3	67.1
74.3	66.0	62.8	67.5	64.6	73.7	68.2	90.6	77.4	72.4	64.3	61.7	89.3	159.8
73.5	66.4	72.2	76.7	67.2	92.6	90.5	75.3	68.8	70.0	65.7	64.3	83.7	89.7
71.8	68.4	81.5	69.6	76.3	69.4	67.4	115.1	63.8	74.6	70.3	71.0	74.8	80.2
102.9	65.5	64.5	66.5	66.2	77.2	69.9	72.4	87.9	81.2	96.3	64.9	71.0	74.3
83.9	101.7	67.5	117.7	117.8	77.2	65.2	73.1	64.8	65.7	69.6	59.3	85.5	61.9
102.4	93.3	95.6	99.2	80.6	66.6	68.2	82.6	76.4	59.3	65.7	67.5	74.0	76.7
69.9	62.4	77.1	82.8	130.4	65.7	73.6	67.9	111.7	70.8	65.9	97.2	492.4	81.6
77.3	78.6	62.9	67.0	58.7	86.5	123.1	76.2	70.4	83.2	67.8	69.4	63.1	60.7
58.4	78.3	71.5	123.3	279.3	74.7	64.5	67.4	78.7	71.0	161.7	68.5	69.2	65.2
62.4	108.4	121.8	71.4	76.2	66.7	64.9	72.7	111.9	115.1	168.5	151.4	70.4	68.6
72.9	75.2	90.5	69.6	82.6	80.8	107.5	105.5	71.5	70.1	245.4	77.5	75.1	72.7
77.8	77.1	89.3	103.8	102.6	69.2	92.3	109.1	91.4	106.1	74.2	72.8	75.6	87.3
66.2	59.5	74.4	75.3	67.0	64.9	79.0	60.0	81.9	102.5	65.7	74.9	60.9	68.8
106.0	80.5	63.7	68.6	76.9	77.8	58.2	64.3	79.5	81.1	79.2	73.2	73.3	70.6
58.1	64.2	111.9	120.2	107.1	63.5	66.2	78.6	65.3	66.1	79.0	58.6	92.1	59.4
64.4	65.5	71.3	73.8	90.5	114.0	104.1	89.6	88.7	86.1	65.8	286.3	75.7	59.2
86.1	70.2	71.7	66.9	84.6	89.1	69.4	69.5	128.5	61.0	71.7	85.4	57.9	63.1
65.6	128.1	70.1	79.4	75.8	148.2	95.1	67.3	90.7	71.1	69.6	130.7	91.8	66.5
65.0	76.0	68.3	61.3	82.5	69.0	81.3	69.1	65.5	60.9	82.5	155.3	71.0	91.1
67.8	78.7	210.8	72.5	73.2	69.2	78.8	132.4	1196.2	1202.4	63.7	82.5	81.4	68.4
84.4	61.3	68.1	76.9	112.4	99.1	74.6	79.0	73.2	101.1	64.1	68.5	65.5	131.9
81.6	76.3	79.3	72.3	73.4	63.2	66.5	75.5	66.8	65.5	62.9	67.8	66.3	72.2
133.0	73.5	76.0	67.6	73.6	77.2	66.2	73.8	70.5	71.4	71.9	69.6	123.7	55.9
63.5	57.4	88.9	68.2	63.7	168.5	67.9	62.2	62.6	59.7	82.6	62.1	78.7	106.0
72.2	73.9	52.5	105.8	93.0	101.7	64.7	53.3	63.7	69.1	69.3	76.5	91.3	80.7
95.0	70.6	171.9	73.6	95.2	100.9	93.8	90.7	66.8	64.6	73.0	49.3	187.3	268.7
107.3	79.4	81.1	88.5	64.6	62.6	79.9	72.7	197.7	82.3	95.1	100.4	78.7	77.7
65.4	62.9	75.1	81.2	66.9	80.8	64.9	74.8	110.5	68.8	54.0	75.3	68.4	71.8
88.8	87.3	64.8	64.0	121.2	81.2	87.7	69.1	85.3	468.9	78.8	63.0	61.1	75.7
63.7	71.8	84.3	107.3	115.3	161.9	75.5	76.6	87.0	116.1	69.4	63.3	69.9	67.4
77.1	168.0	83.8	69.8	70.7	135.7	60.9	80.3	78.2	111.6	96.5	74.6	276.4	66.0
88.8	94.6	64.5	72.3	64.5	63.7	117.3	112.5	61.0	86.2	63.6	63.5	64.3	80.4
65.4	89.7	82.9	88.8	63.5	70.2	63.6	64.8	95.2	61.8	90.0	74.3	63.3	62.0
61.7	69.3	67.0	66.4	61.3	268.6	65.4	76.2	67.1	66.8	82.9	139.5	75.9	70.9
108.6	66.4	70.0	80.9	111.2	84.1	66.9	65.9	422.0	460.9	84.4	63.9	119.3	1190.6
76.5	68.8	93.1	77.3	105.8	66.3	73.3	75.6	74.4	75.8	127.5	69.5	81.8	67.9
69.9	67.4	60.5	116.3	84.0	74.2	68.7	69.2	78.6	74.5	84.8	67.3	71.8	62.7
66.4	64.8	70.6	104.5	139.1	111.7	71.6	65.8	187.7	74.5	67.6	83.5	104.1	73.8
96.9	269.5	155.1	147.1	115.5	73.1	63.4	139.2	60.7	70.2	152.1	148.1	74.9	69.6
93.2	70.8	1338.2	91.4	149.1	65.0	61.1	64.1	67.7	66.0	75.5	60.3	65.4	61.8
808.2	67.7	78.4	96.2	72.7	99.1	81.7	156.9	67.2	81.9	84.8	72.1	64.0	75.8
73.8	83.5	64.6	67.0	236.5	86.6	123.0	87.7	65.4	66.9	64.3	98.1	73.4	70.1
82.1	87.1	68.0	70.9	69.5	69.0	108.0	70.2	75.0	68.3	71.6	75.0	63.3	106.2

Continued on next page

Table B.1 – *Continued from previous page*

64.1	103.7	95.0	169.9	66.1	62.3	65.5	95.6	62.5	300.9	77.8	70.5	72.1	129.5
72.2	83.0	67.5	75.1	62.5	69.3	81.4	97.8	63.3	85.1	64.3	69.9	67.5	268.0
64.1	257.3	94.9	72.8	97.7	76.7	68.1	69.5	94.9	69.7	61.0	68.4	65.0	73.3
73.9	78.0	69.3	87.4	63.5	91.5	69.2	70.5	67.0	67.5	68.8	66.6	84.3	95.3
72.1	65.5	83.5	103.1	61.4	62.0	74.2	71.4	111.3	301.0	83.3	62.6	64.3	74.4
70.9	68.9	66.2	65.3	69.7	67.3	70.1	82.2	87.8	340.1	112.6	102.8	64.9	78.8
116.0	69.6	67.5	57.8	75.6	66.3	100.9	69.4	67.5	64.2	62.9	72.6	65.1	139.2
62.9	71.7	65.3	79.7	110.2	70.8	77.3	73.4	61.7	62.7	171.1	64.2	73.8	66.2
67.3	74.6	104.2	67.3	296.4	65.1	102.4	60.9	68.4	64.8	100.0	74.7	96.9	74.0
67.6	65.2	74.4	86.4	604.0	98.4	64.8	81.8	99.1	84.6	91.9	101.8	79.8	77.9
82.8	69.4	71.4	75.0	134.8	73.9	70.9	69.5	63.5	71.8	64.7	167.7	68.0	72.4
74.7	69.9	86.0	77.0	69.0	123.2	71.4	109.5	111.4	100.6	72.4	67.4	62.2	66.4
82.7	63.1	94.6	76.0	120.5	101.6	67.5	67.7	129.8	89.3	54.8	88.9	72.1	61.0
63.5	109.9	73.9	77.6	61.7	70.5	71.3	72.1	69.3	76.3	76.7	70.5	100.2	106.6
72.7	123.1	136.4	137.5	66.2	65.1	99.2	132.4	151.6	76.1	72.7	81.4	70.2	73.5
70.6	71.4	114.4	98.9	70.9	68.3	172.7	175.3	66.2	63.8	93.5	104.5	80.8	168.7
68.0	72.9	59.2	53.1	97.8	66.2	60.6	68.7	68.5	77.7	58.9	70.9	67.8	63.5
93.4	75.6	103.0	103.7	71.3	66.4	64.2	68.3	102.9	94.8	66.7	75.9	92.1	83.7
82.0	64.7	80.4	71.2	61.8	70.9	102.7	99.3	67.1	63.2	104.3	65.4	70.8	159.6
72.3	93.4	74.5	95.2	75.9	76.3	68.3	79.3	68.7	64.7	68.8	72.5	66.1	68.4
92.1	67.4	79.1	80.3	78.4	87.5	137.0	75.2	61.0	251.4	88.7	89.9	70.9	60.8
65.9	65.2	62.8	69.7	124.5	124.9	103.1	99.4	97.2	82.6	105.4	113.6	75.0	67.9
63.9	92.4	83.1	70.2	63.0	73.9	86.1	58.6	78.8	98.2	71.8	96.8	58.6	64.2
65.8	59.3	67.1	74.8	80.2	74.6	83.1	71.4	80.4	82.2	69.5	66.4	79.1	62.0
85.6	77.0	65.0	62.2	163.8	63.6	70.8	67.8	101.4	77.4	70.0	112.2	67.5	77.1
65.6	72.4	73.0	72.6	67.4	83.1	82.7	89.8	73.2	79.6	69.6	61.4	197.5	102.1
71.4	69.3	85.4	69.9	65.0	63.0	73.6	141.6	97.9	66.4	124.9	61.9	65.9	76.2
67.5	173.6	69.5	104.1	76.2	66.1	128.8	86.1	74.3	65.9	67.9	62.6	79.1	86.1
115.4	74.8	440.7	67.9	64.9	66.3	73.6	71.2	82.4	65.8	73.7	92.6	78.4	75.1
106.8	69.3	91.7	62.6	61.0	71.0	60.9	64.0	86.9	81.8	79.6	91.6	71.2	80.3
55.9	78.4	61.4	64.2	65.4	69.8	77.6	61.0	67.8	132.6	63.7	71.9	76.1	73.2
68.7	159.9	65.5	79.3	65.0	67.7	67.2	66.0	64.6	140.7	84.2	57.4	64.8	61.3
74.4	52.3	68.5	62.5	63.2	79.7	68.8	73.7	68.6	112.3	61.9	82.4	62.9	68.3
73.8	72.8	98.6	68.6	105.6	73.0	116.1	78.2	102.8	98.9	158.5	73.2	81.2	166.7
68.5	59.2	59.8	118.6	63.3	71.8	65.0	147.8	126.2	68.4	74.5	66.8	62.2	66.4
154.5	124.5	74.9	93.6	92.7	93.4	71.0	69.1	86.0	117.3	96.7	67.4	64.8	79.2
160.1	119.9	96.5	85.8	118.4	83.5	70.6	67.9	86.3	64.7	82.8	118.0	63.1	70.4
61.6	58.0	83.0	67.1	75.9	86.5	72.4	67.5	71.6	73.3	63.3	66.6	80.4	93.8
80.6	68.9	75.7	89.4	71.5	69.7	57.6	157.1	67.9	68.2	80.9	63.5	79.2	412.0
67.1	67.9	65.8	66.1	74.3	80.3	82.9	67.2	86.0	67.3	62.9	66.2	91.2	63.6
79.2	188.9	94.5	83.9	72.7	66.5	78.0	66.5	84.2	90.3	77.2	65.9	72.2	67.3
78.9	78.6	161.4	160.3	66.7	80.0	70.3	90.4	66.5	108.0	78.4	98.9	72.4	70.3
70.6	65.8	67.3	73.8	73.1	70.1	70.7	68.8	78.5	75.2	68.6	131.0	83.4	81.7
131.9	103.8	67.5	73.4	104.4	100.4	67.6	62.0	70.7	75.9	87.6	94.6	90.1	80.1
210.8	203.7	64.8	73.4	90.7	96.7	91.4	66.1	68.3	74.1	63.4	83.5	95.6	76.7
117.0	117.6	76.7	65.2	95.3	82.5	65.0	71.4	72.5	74.4	68.1	74.3	65.9	84.1
67.3	90.8	69.6	92.9	60.2	142.3	260.1	68.6	72.1	64.4	61.9	74.0	69.0	66.6
71.5	169.8	128.4	74.8	127.0	116.9	75.3	102.7	68.8	201.4	75.8	71.1	63.9	64.6
70.6	84.6	96.1	71.4	80.5	65.1	114.8	69.7	69.4	61.0	102.2	385.6	78.9	64.1
64.8	61.0	71.4	193.8	99.2	87.4	65.2	574.6	104.0	78.4	80.3	91.5	85.7	105.4
64.8	73.2	75.2	76.3	65.6	82.2	164.5	63.1	74.9	85.1	76.6	76.3	100.3	72.2
86.0	85.8	68.2	69.4	63.2	73.7	64.2	74.8	240.7	76.0	68.5	64.4	67.2	78.0
66.8	94.3	57.1	66.7	53.0	63.2	70.1	60.9	107.0	322.1	90.1	104.4	108.6	67.6
78.9	94.8	74.3	72.9	107.1	116.0	107.1	97.3	66.6	92.1	104.5	122.2	67.8	90.6
90.2	171.8	51.4	67.1	66.4	70.8	71.5	70.1	96.4	70.4	69.1	60.5	77.0	72.8
70.1	140.9	157.6	62.7	99.3	126.0	69.5	353.9	61.8	61.9	68.0	157.1	67.1	80.6
74.9	146.0	68.2	72.6	66.0	68.8	90.3	73.1	79.9	77.4	71.6	70.0	65.5	70.1
87.1	82.1	65.0	62.0	68.7	63.2	86.9	113.7	74.3	72.5	74.2	271.6	93.9	64.7
70.7	90.9	70.2	65.3	85.1	91.3	354.3	75.6	89.9	71.2	73.7	70.8	75.8	88.0
62.5	92.7	88.8	61.0	69.9	55.7	134.1	135.3	66.0	65.7	80.2	236.3	97.0	65.2
70.8	115.5	82.0	61.6	72.0	64.9	62.4	116.7	66.0	69.1	87.4	71.1	68.6	65.4
69.6	77.5	94.0	74.2	126.8	99.1	70.4	75.6	93.0	92.8	70.4	69.9	63.0	91.6
61.6	111.6	85.0	88.9	70.2	55.5	63.9	77.3	69.7	65.4	98.1	94.0	78.4	76.1
67.0	69.7	132.3	80.7	63.1	65.0	64.2	82.0	74.7	71.1	69.3	67.6	62.8	67.5
75.1	68.0	81.6	78.8	73.2	91.5	97.0	73.1	73.4	75.0	69.9	71.2	69.2	93.3
72.1	67.4	64.3	74.1	66.4	67.6	73.6	73.6	70.9	72.6	98.3	121.1	78.7	80.3
59.1	68.9	60.8	107.6	84.3	76.8	66.3	71.4	91.4	83.7	68.4	74.0	81.8	70.9
77.9	70.8	67.7	164.1	61.2	64.6	141.6	81.4	58.7	63.9	71.2	72.4	75.9	129.5
66.8	65.7	65.8	58.8	63.2	69.0	117.7	73.2	69.2	83.4	65.4	72.9	65.5	81.3
67.0	75.7	136.5	209.4	73.1	72.9	57.9	96.7	63.3	63.2	75.1	87.5	119.2	135.7
63.5	82.3	81.5	76.2	70.8	63.3	78.1	72.8	93.0	114.2	84.0	80.7	70.6	64.0
145.8	67.8	72.1	271.8	72.4	80.5	63.9	74.8	75.6	62.8	73.6	88.9	79.2	69.1
71.0	128.7	65.7	65.6	99.1	180.4	94.4	96.7	73.6	63.8	81.2	56.4	68.9	75.5
121.4	72.6	78.0	74.7	59.3	65.0	112.3	72.7	108.3	78.9	66.1	77.8	73.2	79.2
84.7	123.7	71.5	65.9	65.6	70.7	115.1	73.8	125.6	62.2	63.9	71.6	88.3	79.7
67.0	65.2	65.6	102.5	120.4	66.8	81.2	73.4	69.6	96.8	87.9	62.3	69.6	80.5
81.7	74.6	67.3	64.2	58.8	57.3	79.7	61.4	88.9	115.7	77.9	73.0	66.8	70.0
74.3	67.4	81.6	65.8	70.7	60.1	66.8	82.2	89.5	71.0	72.9	63.6	73.9	91.0
72.9	73.5	91.8	71.9	73.9	74.5	65.8	219.5	69.9	147.8	97.8	73.0	74.8	80.9
83.2	73.1	72.9	69.5	57.4	116.3	76.3	60.9	69.3	63.9	59.6	71.7	74.5	98.1
64.7	101.3	85.9	116.6	62.9	82.9	65.1	77.9	63.3	66.1	147.8	74.8		
72.4	67.3	59.4	92.9	68.1	66.9	67.0	93.2	70.0	59.7	126.5	67.3		
69.9	76.2	91.1	93.4	75.9	70.5	92.5	70.4	105.0	61.8	71.6	75.7		
70.4	59.0	70.6	69.4	74.6	65.4	78.5	70.2	64.6	115.5	69.6	65.3		
81.1	79.4	75.4	143.2	68.6	68.6	68.1	63.4	64.6	71.8	78.4	65.0		
137.8	142.8	72.0	95.2	78.3	81.8	75.4	58.7	129.4	140.5	105.1	105.3		
76.2	183.6	71.9	72.0	150.7	81.5	64.6	78.1	102.0	59.5	64.9	58.1		
68.4	71.8	111.6	68.0	72.2	67.9	61.6	69.7	97.2	94.1	78.8	61.9		

Table B.2: Simulated 'Onion'-Shaped Loss Data for Dimension 3

102.1	66.9	67.3	75.7	79.9	77.2	80.6	80.6	70.1	88.3	82.9	90.7	81.7	110.9	71.2
91.8	79.6	77.6	119.9	118.8	110.9	87.9	173.9	81.0	81.0	83.9	67.8	83.2	67.3	70.7
60.4	69.8	69.4	67.6	76.4	68.5	65.5	71.3	121.7	78.3	78.0	70.8	123.1	131.1	111.8
67.6	71.5	67.0	82.9	106.1	91.1	70.3	74.7	95.6	86.8	70.8	69.6	81.0	75.7	82.7
71.9	89.7	73.9	76.0	73.4	64.6	97.5	725.1	94.0	167.2	67.3	134.5	77.1	65.5	96.2
72.6	69.6	68.9	103.9	70.1	65.0	73.8	84.3	69.3	117.6	155.4	69.7	69.5	68.7	103.6
108.4	123.1	108.6	82.1	74.8	73.3	72.1	75.2	78.1	92.8	84.0	67.3	125.8	71.4	83.9
134.7	99.2	112.6	82.5	77.2	70.8	69.7	72.4	132.5	71.9	99.5	69.7	231.1	247.8	230.6
112.9	101.6	74.8	77.2	74.0	88.7	69.2	80.3	64.7	127.9	64.3	92.8	70.1	70.0	74.7
91.1	69.9	71.7	84.6	81.6	70.0	85.6	81.8	87.0	76.6	72.3	119.6	80.6	82.8	120.0
76.5	99.0	86.5	103.3	107.2	104.8	100.6	93.9	85.7	77.5	79.2	90.6	76.6	77.8	97.0
70.2	75.8	239.3	67.4	69.9	73.9	86.0	96.8	72.6	183.3	147.8	111.9	68.3	61.9	70.1
91.9	71.0	67.5	92.5	100.6	93.0	89.1	85.9	83.7	204.7	116.0	119.2	74.4	98.8	68.3
78.4	92.7	78.7	70.0	100.4	76.3	84.2	76.0	87.5	114.9	103.4	77.9	73.2	82.6	80.2
161.4	157.0	154.2	97.9	105.1	78.5	92.1	83.0	83.3	76.3	80.3	75.3	87.1	87.8	88.7
108.2	72.1	76.5	111.3	74.9	78.8	71.5	122.2	65.5	73.3	71.0	82.3	73.1	85.2	73.3
64.2	67.3	69.9	69.7	65.6	72.7	67.6	73.6	83.3	72.2	63.8	67.4	75.6	101.9	89.1
77.6	82.9	86.5	74.8	79.2	79.7	83.5	69.2	77.2	74.0	75.7	95.7	88.1	83.3	81.5
71.7	75.5	236.9	89.0	87.3	89.9	75.0	70.1	67.2	64.8	70.3	101.1	132.0	72.4	70.9
83.0	82.0	98.0	116.2	67.8	75.2	107.0	104.6	109.5	68.0	62.9	77.5	86.9	81.4	81.3
77.0	64.6	70.2	73.1	79.8	75.0	91.9	72.7	75.1	66.1	67.4	74.8	83.0	66.1	71.4
101.1	86.8	90.9	78.3	86.3	77.6	84.1	73.9	72.9	70.5	75.4	71.7	75.2	68.3	74.4
81.7	85.9	86.6	68.3	68.1	61.1	94.5	67.1	75.1	116.1	66.4	71.5	83.3	126.3	90.8
71.5	123.8	91.4	96.1	126.1	134.1	91.7	90.4	83.8	64.8	63.6	81.5	102.5	98.7	98.6
92.7	104.0	98.6	145.0	137.1	146.1	74.7	79.7	82.2	78.0	75.0	83.0	75.3	65.6	86.0
65.6	69.1	83.1	115.8	75.2	88.7	71.9	78.6	69.2	67.1	113.5	69.7	71.3	121.5	66.4
83.4	107.7	95.8	64.2	69.3	69.7	97.5	124.9	77.1	71.3	60.4	64.4	128.8	112.5	104.3
63.0	66.9	63.3	82.8	79.4	69.5	108.0	75.8	78.6	91.9	82.2	77.9	67.3	85.3	242.6
123.3	121.4	129.0	62.1	71.1	81.3	69.3	68.7	75.9	70.7	70.5	66.3	68.4	76.1	96.8
70.6	86.8	69.1	75.2	89.8	108.0	67.7	66.7	102.8	76.4	77.7	79.5	75.4	75.8	74.1
96.3	71.9	69.2	65.5	66.5	68.9	65.8	72.4	68.8	67.5	199.1	71.0	105.0	86.4	77.9
89.9	82.4	86.1	82.1	66.2	79.5	165.4	135.6	166.5	86.1	83.2	121.1	92.0	118.5	148.9
100.4	217.2	73.5	76.6	68.6	70.5	71.1	72.8	70.0	91.9	79.7	87.3	72.4	69.8	79.3
108.8	104.2	99.7	71.2	71.3	78.5	70.6	63.6	71.0	92.3	85.4	62.2	82.2	66.2	67.7
69.4	59.7	66.5	88.2	89.7	92.2	66.9	77.8	73.1	74.9	73.8	86.5	125.2	68.6	118.2
78.5	81.1	86.2	70.5	83.9	74.8	151.4	83.9	103.5	78.7	173.9	93.9	81.3	86.0	482.2
91.4	90.7	110.7	72.0	143.3	69.1	90.8	65.8	92.1	73.8	67.3	113.0	67.6	68.0	84.1
71.9	73.5	330.6	97.0	112.0	90.0	64.7	75.8	75.9	69.3	113.5	92.0	87.2	84.9	83.8
72.0	424.1	72.1	81.0	85.9	91.1	72.6	89.1	100.0	88.9	67.5	96.3	69.4	77.1	105.3
74.4	65.0	76.4	69.0	103.9	83.8	69.2	73.8	64.5	71.7	76.7	70.1	74.9	78.4	77.3
70.3	70.0	74.0	114.9	82.0	86.6	101.4	64.5	69.8	69.5	62.9	76.6	81.0	69.2	75.3
85.8	92.9	92.5	71.9	68.3	90.8	64.0	76.4	65.6	127.0	80.2	68.6	81.3	71.6	70.1
64.0	91.5	104.4	66.2	67.6	69.7	88.2	89.2	65.0	104.9	117.5	124.1	99.5	78.8	69.9
183.3	166.2	69.8	141.1	76.6	75.8	90.5	75.4	241.4	72.3	265.9	71.6	67.7	72.7	73.9
82.4	70.1	73.4	64.7	75.3	64.2	77.6	67.2	72.0	111.6	127.7	117.8	115.4	103.9	103.9
83.8	103.5	67.2	79.6	76.6	77.6	80.8	179.4	178.5	93.1	69.9	124.2	71.6	87.2	71.7
74.0	68.6	68.8	73.6	69.3	64.7	77.2	71.8	76.0	79.3	83.0	70.3	69.1	65.8	83.3
84.3	68.0	70.1	64.1	59.6	55.5	165.6	86.5	86.2	81.5	126.7	72.9	76.4	69.9	76.5
73.2	77.6	94.5	75.4	73.4	74.2	71.6	76.1	177.9	70.7	103.0	69.8	92.0	72.1	77.8
114.8	68.9	65.5	80.1	72.6	64.0	79.3	71.0	75.6	123.8	90.2	116.2	201.7	106.9	113.2
219.9	1279.3	102.5	68.6	78.6	63.9	92.2	99.0	91.7	79.9	86.9	115.9	67.3	75.4	109.3
76.6	76.1	77.5	79.0	74.0	102.0	68.3	66.0	157.6	66.1	66.6	45.6	76.7	72.2	70.0
69.5	347.9	76.3	94.5	81.3	77.0	86.4	116.2	92.4	73.9	80.0	70.9	232.2	234.7	226.6
74.9	75.7	69.4	65.6	72.0	72.4	75.1	72.7	98.6	107.2	74.6	299.8	80.5	110.9	79.7
60.7	71.0	74.5	87.4	130.3	92.0	118.6	114.3	98.5	73.6	94.6	451.5	68.2	83.0	72.2
123.6	95.1	94.1	82.6	67.8	95.4	69.5	77.1	100.4	68.4	75.7	76.4	71.0	126.4	79.2
71.3	72.4	82.3	91.5	97.3	117.1	74.5	64.5	67.7	66.7	64.1	73.9	93.7	105.9	92.5
89.7	95.9	78.8	75.8	68.4	98.2	92.9	118.8	91.2	73.7	70.7	77.9	68.0	80.7	76.8
151.4	66.0	117.9	74.2	69.5	69.0	78.8	96.1	65.3	96.2	83.2	77.8	82.3	76.7	88.8
75.1	85.1	70.5	72.2	92.9	103.3	72.4	94.9	96.2	75.8	69.2	79.1	212.6	230.3	200.8
68.9	119.1	66.8	65.1	72.3	69.2	85.4	109.0	86.3	73.8	90.8	75.3	67.3	74.8	68.4
72.2	76.4	75.5	77.3	70.2	82.5	80.3	77.8	76.7	77.2	103.1	67.9	71.8	252.6	66.8
89.8	127.9	103.5	77.0	67.3	92.0	79.1	95.1	69.0	74.7	62.3	68.4	64.9	67.4	61.7
68.1	77.0	69.0	93.4	76.3	73.3	104.2	90.4	63.9	67.6	83.7	81.9	65.0	71.1	79.7
90.7	96.8	113.8	69.1	81.5	90.1	64.6	102.8	95.4	71.8	86.2	76.8	72.8	74.7	79.9
92.8	64.0	64.3	75.4	76.8	69.8	74.8	73.1	78.7	99.7	101.9	90.7	104.3	78.8	71.9
74.0	72.4	68.3	80.1	65.9	72.2	103.7	82.3	77.4	69.2	94.8	73.5	69.1	75.0	65.6
73.7	73.3	78.1	69.6	64.8	63.2	89.1	71.9	72.9	417.2	424.7	415.5	73.7	74.8	71.6
244.0	95.7	89.4	66.9	70.8	65.8	86.8	76.6	60.9	73.4	66.6	75.7	94.5	77.5	95.4
73.6	189.2	77.8	78.8	81.9	65.6	98.8	63.9	73.3	64.9	99.5	68.9	67.1	83.3	74.8
73.3	68.5	76.8	74.3	74.2	79.9	242.6	164.5	188.2	95.2	77.8	70.8	86.7	79.9	94.6
76.2	139.8	70.5	300.7	293.5	294.3	90.9	80.5	81.6	72.9	94.3	80.5	67.7	66.6	69.0
84.5	99.7	75.6	68.0	61.6	92.3	70.7	69.4	70.2	68.1	67.7	82.9	52.9	73.2	351.0
81.7	79.0	69.7	114.8	98.2	83.0	117.1	83.3	76.3	88.9	72.2	209.1	118.1	184.5	113.4
241.1	72.2	74.9	374.5	382.3	376.8	94.9	60.9	70.2	72.2	121.8	81.4	72.0	93.6	77.7
68.0	106.9	79.9	76.1	79.0	72.9	72.8	75.0	68.2	85.9	67.7	83.1	96.8	81.3	75.0
74.5	135.7	78.5	81.9	73.4	62.8	74.4	75.8	78.6	246.7	235.1	230.2	102.6	71.5	65.7
89.7	89.1	95.9	118.9	89.5	89.3	65.7	66.4	67.8	81.5	69.1	69.5	83.1	79.6	69.3
158.8	250.8	155.2	85.7	64.8	85.4	82.2	72.4	84.6	71.9	73.6	70.1	88.1	157.2	73.7
124.7	123.6	121.5	68.2	87.9	67.7	72.3	70.4	75.2	94.2	63.2	89.5	74.5	84.4	68.9
76.7	106.8	88.5	71.9	79.8	72.7	75.1	93.8	113.2	68.2	88.5	72.5	72.2	74.6	88.8
69.3	107.6	68.3	74.8	87.2	74.8	354.0	85.2	94.0	64.7	68.7	117.2	238.5	225.4	248.7
88.2	88.6	75.5	102.7	127.9	81.3	76.1	74.8	78.9	73.4	93.4	79.7	74.5	117.0	68.2
89.4	76.5	71.1	71.3	67.5	78.5	70.5	67.6	105.4	79.8	71.9	79.1	70.5	82.1	68.2
93.7	65.0	95.0	70.2	66.2	130.2	64.4	91.1	66.9	70.8	66.9	276.8	74.7	87.5	70.2
63.8	68.5	69.3	105.8	70.6	68.7	74.1	79.5	75.1	140.9	203.9	140.3	71.1	75.7	99.2
96.8	80.1	76.1	71.5	73.4	73.1	77.4	79.1	70.3	64.9	72.1	78.4	62.1	73.4	67.5
68.6	69.9	83.6	119.6	71.5	71.5	88.1	84.1	94.5	82.4	108.8	80.3	132.7	122.0	126.9
99.2	75.3	74.9	66.0	83.3	87.2	101.2	99.9	357.2	76.6	66.8	85.4	93.0	91.3	86.4

Continued on next page

Table B.2 – *Continued from previous page*

67.6	101.6	68.4	118.8	75.0	77.7	69.8	72.8	84.2	68.1	70.3	92.8	68.6	67.9	66.3
78.0	70.0	74.6	61.6	68.2	67.3	74.2	95.1	69.3	79.1	146.4	75.5	113.8	156.5	115.5
106.9	86.2	142.2	84.0	158.3	114.4	118.4	126.4	137.5	79.8	77.9	98.5	147.9	77.8	78.3
74.3	62.0	85.9	65.0	73.5	70.9	84.4	70.6	79.2	120.3	100.3	194.7	73.1	77.3	81.9
78.9	87.4	66.1	80.0	79.7	90.2	106.8	113.7	71.3	67.3	64.9	69.2	99.8	100.0	163.1
76.2	76.2	78.7	77.8	86.0	76.2	85.3	86.4	79.4	91.8	79.0	72.5	84.4	79.8	92.4
71.9	76.1	103.5	78.5	510.4	78.7	82.8	64.3	65.9	90.3	88.3	88.4	73.4	66.5	72.3
72.3	144.8	77.6	73.9	85.8	88.2	82.9	92.8	66.5	78.4	86.2	80.1	69.3	72.7	120.2
94.2	80.2	95.9	69.8	78.1	64.4	184.1	77.9	74.6	64.2	57.0	177.6	66.7	72.7	73.5
86.0	84.2	66.8	76.2	75.9	88.3	67.2	94.7	67.0	169.0	158.6	161.1	82.5	65.4	81.2
69.7	83.7	128.7	64.8	135.8	72.9	116.5	75.5	76.6	118.7	181.0	107.3	68.3	83.7	76.6
72.5	69.8	94.2	71.7	64.3	113.5	80.1	80.9	73.3	89.9	101.6	73.1	74.3	185.9	107.3
85.0	67.0	87.1	85.9	105.2	78.4	62.2	144.6	92.0	83.2	62.3	57.2	93.1	87.3	117.5
85.5	74.3	74.0	84.7	84.1	197.3	77.2	69.8	71.6	83.0	69.7	158.6	174.6	97.6	104.2
69.9	67.4	158.1	98.0	88.3	79.9	76.3	115.7	77.2	215.4	70.1	84.4	71.1	88.6	84.5
71.2	69.6	73.4	78.0	72.1	117.3	66.2	68.8	68.3	70.8	73.5	69.4	68.6	68.0	69.2
93.9	85.6	91.9	76.4	126.5	105.2	97.2	69.3	77.7	74.3	113.6	82.3	75.5	71.9	71.5
81.8	95.4	75.8	87.1	98.1	80.2	69.3	72.4	80.8	72.2	65.0	68.6	192.5	64.9	159.8
79.6	216.2	103.2	122.1	126.4	119.6	87.3	82.8	82.6	69.7	71.2	73.5	80.6	65.2	89.5
98.8	89.3	95.9	96.6	214.4	72.1	86.3	82.1	86.6	90.2	105.4	95.2	68.2	75.8	106.1
79.5	77.1	68.6	67.0	68.5	75.2	74.9	71.5	73.8	97.8	103.9	88.3	65.2	86.0	75.4
77.3	71.3	79.6	73.4	68.7	108.8	359.2	80.1	76.2	68.0	106.8	70.1	70.7	81.7	79.5
70.8	77.5	65.7	126.0	92.0	69.7	649.4	80.6	73.5	74.1	73.5	99.5	70.0	71.2	78.1
69.0	65.4	65.0	188.8	185.8	181.2	121.3	108.8	116.1	68.7	73.4	64.8	69.8	103.3	84.8
66.4	466.7	86.0	68.9	105.6	71.4	79.5	89.1	81.1	96.3	69.1	74.1	73.3	75.6	75.1
94.6	89.5	81.6	94.1	68.2	73.5	71.7	65.9	92.7	75.6	143.1	81.3	113.7	120.9	104.8
72.9	73.4	67.0	72.7	80.7	76.4	68.4	81.4	82.2	70.5	76.0	79.2	74.7	67.9	89.2
71.0	61.3	65.9	114.9	91.8	96.2	89.8	82.9	80.5	81.5	78.5	72.3	85.3	71.8	112.9
67.1	74.3	69.9	84.8	87.6	69.1	116.4	113.9	81.2	87.8	73.3	67.0	158.8	98.9	113.9
81.0	81.7	83.3	82.1	72.1	77.4	82.1	65.2	76.3	123.0	67.8	76.2	125.8	79.8	68.8
79.9	73.0	66.3	73.3	80.5	82.0	107.8	91.1	75.2	154.1	154.2	162.5	75.0	88.9	77.3
83.6	97.2	68.5	87.7	76.7	90.5	70.1	77.6	73.2	67.5	107.4	72.4	79.1	92.1	66.1
87.4	90.2	93.2	80.3	73.0	85.5	87.1	89.2	90.7	68.0	69.0	70.3	63.3	78.4	75.3
87.7	112.3	73.2	77.2	83.9	69.7	72.8	82.7	69.5	82.2	76.6	71.9	144.9	70.2	89.0
122.7	89.0	80.3	73.4	77.3	82.4	78.1	88.2	123.3	74.8	84.1	77.5	76.1	93.1	76.5
79.5	72.4	122.7	73.0	71.2	72.2	70.0	82.3	77.9	73.4	70.6	222.1	95.6	110.2	119.7
72.0	69.5	85.5	71.5	71.2	96.6	69.3	76.0	68.4	78.2	78.8	87.2	75.7	69.4	86.2
68.3	72.0	65.0	79.1	228.4	68.4	87.0	77.4	68.2	86.7	110.7	75.8	73.3	151.8	76.9
128.7	98.2	154.6	119.5	79.6	83.0	143.6	76.8	68.5	217.2	193.1	194.9	97.2	80.7	79.7
65.6	77.1	71.3	76.2	78.4	217.4	76.8	75.6	83.2	111.6	101.6	165.0	77.7	69.0	72.8
82.7	87.9	86.3	71.1	77.4	85.9	99.4	95.3	91.4	95.4	95.8	90.7	96.1	108.8	139.3
81.1	64.5	67.0	71.3	71.0	71.4	93.8	71.5	109.7	66.9	57.4	99.8	73.6	75.5	70.3
73.0	81.9	69.8	69.7	67.6	76.8	73.0	80.4	133.5	67.6	124.3	68.7	62.2	64.7	81.9
72.0	68.9	57.9	77.2	68.4	71.9	72.1	71.7	188.6	162.2	96.7	79.1	69.5	82.2	126.9
83.9	83.4	70.9	121.4	98.2	236.3	65.5	69.1	68.4	80.4	113.2	79.1	77.4	173.6	72.4
84.0	87.7	106.0	75.1	79.8	75.6	111.7	82.7	118.4	163.1	86.1	91.9	88.6	76.0	70.0
67.3	69.8	66.5	66.4	78.5	84.3	76.2	192.1	89.8	91.6	76.0	100.8	69.6	57.7	77.9
112.1	71.5	109.8	88.9	83.9	89.7	87.5	68.8	135.2	219.0	73.3	81.0	69.5	69.6	70.2
71.4	70.6	78.6	84.8	70.7	85.0	72.7	111.8	68.5	187.2	106.5	103.5	79.2	68.3	86.0
68.6	73.0	72.2	256.0	191.1	230.8	114.5	216.9	67.6	70.0	82.0	102.9	70.8	67.5	69.4
88.9	65.2	88.0	112.2	75.8	95.3	80.6	134.2	72.4	64.4	457.2	73.1	58.6	66.6	66.6
86.9	76.2	126.4	76.8	78.4	75.2	98.8	74.4	74.2	96.6	102.2	87.8	82.2	82.6	105.1
77.4	75.1	72.7	90.9	202.4	74.0	80.1	142.2	64.3	76.6	116.8	79.5	68.1	220.1	70.2
79.6	73.6	166.6	68.7	79.6	74.8	69.9	74.0	69.9	72.6	77.0	83.6	69.5	63.1	96.6
86.0	77.4	78.3	71.7	139.3	104.5	97.3	97.5	120.7	73.2	89.9	68.9	124.4	77.4	77.3
93.2	64.4	178.1	75.9	116.9	87.4	94.1	200.0	105.4	78.6	70.2	68.4	80.4	70.6	89.3
100.4	59.1	75.2	68.4	65.1	70.4	73.3	93.1	164.3	82.2	74.1	107.9	71.0	180.2	65.6
86.4	83.9	81.0	92.5	101.0	97.8	103.0	81.5	66.8	75.1	107.4	75.4	69.1	72.1	155.3
84.5	72.4	54.9	141.9	133.3	156.7	75.4	92.6	67.5	98.7	59.2	81.6	125.6	130.9	122.8
81.0	90.7	73.2	77.1	113.3	59.3	69.2	69.5	71.2	96.2	77.6	83.7	147.5	91.2	85.3
73.3	69.4	82.4	71.6	81.8	67.0	148.5	82.4	98.7	73.0	75.1	81.5	70.6	73.6	76.7
64.0	68.4	75.0	78.9	75.1	63.4	85.1	81.7	81.6	83.5	130.2	75.8	98.3	82.7	88.7
78.0	65.2	73.4	97.9	118.4	72.7	77.5	115.1	76.7	70.6	69.7	66.0	68.8	66.0	78.0
134.0	144.1	194.4	215.9	214.5	222.0	71.5	81.3	67.9	67.6	65.1	69.2	78.8	78.0	105.2
114.0	71.2	78.5	138.8	81.8	84.2	77.1	75.4	74.6	101.0	70.7	79.9	72.6	71.5	80.6
71.9	129.3	70.3	168.5	117.8	74.9	67.7	75.7	68.0	98.2	107.8	107.5	92.9	76.3	183.2
76.4	106.9	72.6	75.7	67.8	394.3	65.7	68.8	98.0	66.9	73.2	111.4	78.4	73.0	73.4
75.3	66.4	85.8	72.1	82.3	69.2	82.7	66.0	71.8	75.9	93.2	78.4	125.2	96.0	80.7
172.6	73.2	84.4	107.2	73.6	70.5	66.9	77.1	88.3	77.2	125.0	101.8	71.9	67.2	81.5
73.9	75.5	65.6	74.5	91.6	66.5	78.4	76.5	70.3	100.1	85.1	89.9	65.9	69.6	78.8
61.7	75.3	65.6	168.4	174.4	175.8	135.0	110.2	98.7	53.4	76.5	70.6	75.5	72.7	74.8
71.2	157.5	76.1	68.3	102.0	75.4	75.5	66.3	76.8	73.2	61.3	107.8	80.1	100.1	132.7
66.4	71.5	68.7	67.7	77.7	80.2	157.7	85.3	93.8	72.3	166.7	82.1	74.8	76.3	66.9
75.0	99.7	69.1	168.6	93.9	135.7	87.5	74.7	66.3	91.3	76.8	118.1	78.0	68.0	65.8
225.9	270.3	231.2	65.7	70.7	70.2	73.3	65.9	81.8	65.8	68.7	70.3	69.2	78.0	65.9
261.6	69.3	73.6	75.1	117.7	101.1	79.5	97.2	75.0	327.1	130.6	76.5	70.2	102.4	90.7
73.8	100.2	89.5	70.2	70.0	82.2	114.4	80.1	83.4	63.2	71.2	87.5	77.7	66.2	91.1
74.2	79.5	87.5	70.5	72.7	66.8	72.7	68.3	66.6	79.2	85.5	138.1	113.7	118.3	121.9
70.6	70.5	87.6	86.3	81.0	66.7	71.1	78.1	65.0	67.2	88.3	70.3	97.7	68.6	100.2
78.6	72.4	68.1	79.6	93.2	74.0	70.7	93.6	101.3	92.0	87.3	96.3	105.3	80.3	68.1
67.7	64.6	73.8	78.3	59.6	66.7	70.2	62.1	79.3	103.8	75.1	71.0	70.6	71.2	75.7
72.6	68.7	69.5	80.7	78.7	84.0	88.9	83.7	128.5	69.6	75.4	102.8	83.8	73.5	73.4
77.1	105.2	80.0	153.4	165.0	168.5	71.8	65.6	67.0	77.5	75.1	72.5	82.6	71.8	128.7
74.5	77.7	85.9	79.2	80.5	127.9	86.7	77.7	96.6	92.5	92.1	84.8	84.6	75.1	81.1
584.9	76.3	82.1	69.0	72.6	76.9	66.7	71.3	52.6	69.8	81.0	74.7	139.5	90.1	99.4
100.2	98.6	105.4	68.4	78.2	62.7	84.1	93.0	103.0	112.7	91.1	95.7	93.1	67.6	74.8
207.2	69.1	66.2	96.8	99.4	110.0	64.7	72.6	65.1	69.6	67.2	66.9	89.3	112.5	79.3
103.2	83.3	70.7	76.3	79.6	72.2	75.8	89.8	70.4	71.3	68.0	144.7	69.9	68.5	94.1
89.4	78.6	64.3	74.6	102.0	77.9	78.0	69.7	66.5	93.0	90.2	97.1	75.8	82.5	80.2
82.1	87.7	84.1	120.9	69.1	64.1	144.0	98.5	74.4	68.6	75.2	81.9	74.1	74.0	106.6

Continued on next page

Table B.2 – *Continued from previous page*

77.9	78.2	75.7	82.6	80.3	73.1	73.2	63.1	92.5	61.4	77.8	88.4	66.7	70.3	80.4
2275.1	79.0	79.9	77.0	86.7	68.8	69.3	81.0	83.9	99.8	69.7	67.9	129.7	454.9	106.5
241.1	150.6	149.8	77.3	94.5	76.2	69.2	77.5	70.8	67.8	67.9	78.1	70.3	73.9	67.4
73.7	66.1	168.3	69.5	79.8	75.2	76.0	307.2	92.7	72.5	78.8	72.1	91.9	74.0	108.1
67.7	74.7	113.1	71.7	103.6	213.9	77.1	75.6	85.6	68.2	82.0	138.4	68.1	62.2	68.3
77.0	72.3	74.2	284.3	66.2	73.3	68.2	63.0	72.4	143.8	85.2	93.0	70.7	115.1	121.8
74.9	84.8	68.3	87.4	178.7	84.5	78.3	71.3	182.5	78.4	93.3	79.5	82.6	92.0	83.8
67.6	71.9	67.3	75.8	73.3	131.2	83.5	71.1	113.3	73.2	74.5	92.4	72.7	66.1	80.4
82.0	72.4	75.2	97.5	82.0	67.8	83.0	73.3	82.2	107.3	69.3	69.9	280.8	83.9	65.6
84.0	68.0	71.1	87.5	101.9	88.6	150.0	140.0	131.5	69.4	77.7	91.1	68.9	69.3	69.7
69.2	153.2	172.6	98.1	80.0	72.4	76.5	67.6	83.4	71.3	61.1	81.0	71.0	68.8	64.3
97.5	68.4	69.6	80.5	77.1	78.3	108.7	71.4	77.9	91.6	102.1	92.3	70.3	149.8	63.6
77.9	68.3	69.3	72.5	64.9	79.8	67.5	66.7	152.8	71.5	74.1	82.7	76.2	77.8	73.9
72.2	85.5	83.3	91.1	93.3	92.1	74.6	72.9	173.9	66.8	81.2	124.9	136.2	71.5	74.2
71.3	64.1	77.0	91.7	75.8	76.7	73.6	70.1	76.6	123.4	77.3	85.6	572.7	579.0	573.1
70.8	70.2	72.6	90.6	76.1	107.3	200.9	72.4	82.2	84.3	85.7	83.5	71.6	70.9	68.5
102.7	94.1	75.0	95.4	108.4	223.0	99.8	78.5	80.7	386.7	76.3	86.6	76.7	106.8	82.4
63.4	82.6	79.5	91.1	68.5	85.2	69.2	80.9	78.0	74.7	169.3	63.8	72.7	93.0	71.8
62.3	69.1	121.2	80.3	80.8	105.0	69.4	71.9	69.5	91.0	106.3	69.1	74.1	82.4	225.9
65.3	86.1	74.8	88.9	84.3	79.7	90.6	89.0	69.8	130.7	133.4	159.2	70.1	128.2	70.9
78.8	71.7	64.3	105.7	87.1	121.2	66.7	64.8	80.7	72.5	94.7	65.5	99.2	115.0	73.9

Table B.3: Simulated 'Onion'-Shaped Loss Data for Dimension 4

70.7	71.9	75.8	70.0	82.9	68.1	65.8	73.4	78.0	78.4	69.7	72.4
81.9	67.3	74.0	89.8	116.1	100.2	104.3	154.1	68.8	75.7	72.9	109.6
75.8	76.2	76.2	73.2	73.7	71.8	97.8	71.1	141.2	65.2	79.0	64.7
65.1	71.5	70.7	107.6	63.5	72.2	67.5	71.6	76.5	69.9	79.2	72.0
66.7	71.7	69.2	89.4	120.2	63.7	78.0	83.8	96.0	76.3	89.9	83.6
83.2	90.7	84.6	86.9	183.0	100.6	97.5	72.7	78.0	122.0	73.6	61.2
379.4	63.7	68.0	84.2	89.4	98.3	68.6	78.6	63.8	78.8	79.1	70.8
92.3	95.0	78.8	75.2	64.4	72.8	134.7	89.3	84.3	129.2	90.5	87.0
82.1	91.0	89.5	85.3	65.8	78.9	71.1	76.9	102.1	105.3	125.5	120.6
76.2	63.6	62.8	63.1	68.2	101.8	71.9	168.7	77.9	87.7	94.8	85.6
149.0	109.1	103.1	107.3	76.5	81.2	68.1	81.2	77.2	94.3	71.7	81.4
79.5	67.5	68.0	79.5	75.5	67.1	74.6	71.2	108.1	104.6	89.8	86.9
79.6	70.9	80.9	71.3	173.4	172.8	185.2	176.1	83.8	86.3	81.9	81.8
73.0	97.6	74.1	158.7	70.1	69.9	98.8	74.0	69.3	80.5	65.6	116.2
100.5	102.5	225.5	86.5	105.5	120.7	114.8	105.3	78.7	77.1	81.5	77.7
67.0	115.7	61.0	83.7	144.3	68.2	53.8	58.5	82.8	95.7	97.1	86.1
89.2	85.1	126.8	84.1	92.9	241.6	68.0	79.4	64.7	66.3	74.0	68.7
71.9	90.2	80.5	80.9	70.4	79.1	74.3	299.7	128.7	126.9	207.8	115.7
89.1	85.3	72.4	82.5	86.9	141.3	82.4	83.2	62.7	95.6	90.6	109.6
99.2	100.4	102.5	107.7	74.1	81.8	76.7	139.8	65.9	83.4	73.8	172.0
104.9	69.4	66.0	66.4	74.8	102.0	76.9	76.2	74.1	78.2	76.1	78.9
141.6	115.0	126.2	121.8	70.8	84.4	83.4	79.3	69.9	71.2	73.2	68.0
76.3	70.0	141.7	71.5	64.9	64.0	74.2	100.6	145.9	102.5	124.1	96.0
86.3	94.7	68.1	79.2	71.6	75.0	74.0	68.8	70.8	73.7	63.6	83.4
177.5	173.9	185.1	167.6	98.8	141.0	119.8	319.7	79.1	95.7	75.1	70.7
110.9	67.5	66.1	64.5	82.1	80.7	78.7	79.6	71.9	72.6	100.3	88.1
77.0	138.4	65.9	63.4	74.5	286.9	67.2	79.9	71.7	372.3	71.5	71.2
108.1	206.2	98.1	105.0	67.8	75.5	70.1	69.2	99.8	72.4	76.1	122.9
70.1	75.4	71.3	66.9	62.3	67.8	134.9	82.8	330.4	168.7	80.3	78.4
86.4	83.8	83.4	80.6	67.4	74.4	71.6	67.9	64.2	93.1	68.9	78.9
83.6	71.8	94.4	71.7	75.6	65.1	101.6	68.1	181.5	84.7	79.7	78.7
779.0	140.5	121.6	67.7	69.7	79.2	69.6	73.7	76.6	72.1	70.0	94.8
79.7	96.6	111.1	71.0	76.8	79.8	86.6	67.1	92.0	87.1	87.0	84.1
58.5	73.4	70.4	69.5	114.2	118.2	118.0	119.2	88.4	89.3	87.9	186.4
89.2	88.1	108.8	104.7	96.0	74.9	74.7	65.7	93.9	71.5	68.4	82.1
81.3	87.2	72.8	70.0	252.9	69.1	67.1	261.4	141.7	124.8	116.6	131.0
74.0	72.9	86.5	75.5	122.2	114.6	170.1	112.2	84.9	83.7	79.4	107.2
94.8	90.0	79.3	75.1	75.2	83.2	78.3	141.4	71.8	100.7	70.2	79.4
220.4	96.3	74.1	177.3	71.9	110.9	73.6	73.1	113.6	115.9	115.1	115.8
95.9	69.9	66.2	73.7	71.4	75.3	64.8	62.7	80.8	75.9	76.4	83.8
71.1	69.5	68.7	235.4	74.0	80.8	82.7	79.6	68.7	77.1	80.4	80.5
72.5	85.0	86.3	94.2	89.2	124.7	61.4	67.4	350.8	83.5	78.1	74.6
140.7	72.6	71.8	76.1	73.1	97.7	97.0	74.1	75.6	78.5	67.9	71.0
93.4	68.7	74.2	69.3	75.0	79.8	121.7	72.7	71.8	69.6	74.7	77.2
82.8	72.1	84.0	160.1	69.4	62.8	67.3	59.6	81.9	72.9	90.0	70.3
76.7	76.4	77.5	75.7	82.2	94.1	77.0	82.7	73.3	80.7	79.9	104.8
62.9	66.1	87.6	65.4	76.8	75.6	87.1	96.5	82.7	67.6	117.6	70.0
86.0	84.3	68.4	78.8	82.3	105.8	68.9	77.1	79.9	72.2	74.9	91.1
79.5	81.1	92.2	95.1	76.3	85.7	72.4	67.9	79.5	113.2	82.7	90.9
87.0	73.5	66.6	100.4	74.8	85.9	97.1	96.0	59.7	56.3	56.9	57.0
87.2	88.1	91.4	66.2	70.9	73.3	65.7	66.5	72.3	81.6	70.3	69.8
78.4	65.0	71.5	108.1	69.7	78.2	91.1	69.8	90.3	68.7	65.7	70.2
72.8	98.7	103.2	89.7	72.3	71.8	73.4	65.6	87.5	92.4	97.2	94.7
80.2	90.8	83.0	89.9	73.9	73.1	72.6	82.4	99.1	84.3	71.8	95.7
101.5	72.8	67.8	84.6	100.6	86.1	77.7	73.0	85.0	96.8	71.9	81.8
70.6	75.1	74.1	69.3	91.5	79.0	78.4	82.5	77.3	71.5	110.1	151.7
76.3	105.9	71.3	133.2	70.3	73.5	77.6	72.3	75.6	65.3	68.2	257.9
71.0	72.3	99.7	75.6	96.4	79.5	75.5	67.6	75.8	133.4	69.5	118.4
58.5	77.4	80.2	87.3	60.2	75.4	103.9	81.9	74.7	73.8	98.4	81.0
102.4	105.2	96.1	154.4	78.6	76.4	68.1	65.9	83.2	81.2	87.5	106.1

Continued on next page

Table B.3 – *Continued from previous page*

77.0	68.5	76.6	104.3	92.9	80.3	95.8	85.3	89.3	70.9	71.9	151.9
87.4	75.4	71.2	80.6	364.1	77.7	70.9	86.7	82.5	71.5	107.7	139.2
67.6	126.9	211.0	69.5	70.6	81.5	65.3	82.9	78.3	166.5	83.0	88.4
74.3	87.9	81.5	65.2	90.3	171.7	102.9	81.1	70.9	92.8	71.7	80.7
87.9	99.2	95.5	99.6	99.6	82.4	75.3	66.8	81.3	106.1	86.1	79.0
82.1	108.3	71.6	197.2	83.7	69.2	65.6	146.6	112.8	119.2	111.8	113.5
66.9	187.1	71.3	68.0	66.3	86.7	420.0	73.7	78.7	71.0	67.0	70.9
79.5	83.6	72.5	84.5	81.3	76.9	95.9	75.0	66.8	124.7	74.9	69.7
77.7	82.1	75.5	85.9	155.6	72.6	114.7	71.4	125.3	74.5	71.8	68.0
75.5	67.9	82.8	70.0	67.0	70.9	71.2	80.2	69.3	99.7	86.4	85.9
92.1	71.5	68.4	70.3	86.0	81.0	78.7	79.0	75.1	74.4	67.9	66.1
157.5	144.5	81.4	83.4	89.8	87.5	101.0	98.3	73.0	75.8	68.4	87.7
76.2	73.2	95.8	84.5	257.7	231.0	235.5	234.9	83.2	78.2	440.0	385.5
108.3	90.9	110.0	88.8	150.9	73.5	72.0	70.5	98.3	77.4	74.5	73.6
85.3	81.2	323.2	95.7	100.6	141.6	82.1	205.8	84.6	88.9	91.3	76.6
165.8	68.6	137.5	65.7	106.3	69.6	66.2	67.3	76.5	84.0	80.2	78.7
76.0	598.6	73.1	71.0	68.1	83.6	67.5	65.6	133.1	160.0	146.6	149.9
66.3	68.8	72.6	72.8	82.9	66.5	72.7	99.7	67.4	80.6	79.0	68.4
88.9	78.6	139.2	86.3	66.3	73.5	84.6	75.3	76.0	80.0	101.0	66.0
80.6	66.6	95.9	91.0	65.5	63.2	65.3	67.3	76.7	72.3	74.6	73.3
67.2	79.5	71.6	102.9	81.9	70.7	68.0	68.5	75.9	78.0	94.6	78.9
70.0	74.6	75.6	74.3	93.3	68.0	73.0	81.3	80.4	73.4	95.4	71.3
61.9	70.0	101.3	65.9	77.4	80.0	74.5	77.8	77.0	78.1	105.2	77.5
111.2	81.1	92.1	94.3	72.6	72.5	90.5	74.9	92.0	102.4	88.4	100.8
62.7	74.4	79.4	106.5	71.0	73.6	72.9	84.7	90.4	95.6	74.9	67.6
162.6	147.4	141.3	148.1	75.5	96.4	76.7	66.9	68.8	74.0	69.7	89.5
76.1	73.9	67.4	71.6	75.1	73.3	71.6	72.3	141.6	77.2	76.9	75.5
66.6	75.9	93.9	69.6	75.9	97.7	74.9	74.1	81.0	82.0	101.2	79.2
68.8	77.8	93.0	70.1	69.4	70.6	82.7	63.9	86.8	79.8	138.6	87.4
74.8	72.0	75.3	76.9	78.9	74.4	73.4	73.1	86.8	79.9	80.3	77.3
71.7	144.8	423.4	77.1	65.6	72.9	68.3	65.0	70.6	75.2	83.5	83.4
66.8	67.8	67.6	100.4	79.6	74.1	67.8	73.8	2427.3	101.1	96.8	111.6
71.0	99.2	65.1	81.8	242.4	72.0	59.2	76.5	106.5	130.6	124.4	117.0
76.1	95.2	83.2	80.9	76.3	66.4	99.6	70.7	72.0	74.6	67.1	87.9
77.1	75.6	87.2	76.8	77.9	79.2	81.7	79.0	81.9	92.5	68.9	59.3
68.7	168.9	63.0	66.7	78.3	65.6	70.2	67.2	116.2	74.2	61.2	74.9
74.3	81.0	72.2	66.1	75.0	94.7	78.4	71.2	70.3	71.9	72.3	102.9
71.3	74.3	74.9	153.7	72.6	127.7	82.1	205.7	77.6	73.4	67.2	95.1
76.3	72.9	73.2	100.1	80.0	117.9	77.7	97.1	66.7	64.8	104.4	69.7
75.8	67.2	96.4	73.6	73.6	67.8	69.5	182.2	76.9	73.5	75.5	73.2
77.3	75.2	126.5	70.3	59.8	78.8	67.2	80.0	81.6	69.0	73.9	75.9
72.6	63.0	65.0	62.5	65.9	91.1	75.7	93.5	93.2	66.6	78.2	69.9
69.8	85.2	78.1	71.9	94.6	61.5	72.8	74.8	67.6	71.5	70.5	116.3
103.3	102.4	208.6	106.8	81.4	66.8	89.0	68.4	61.7	59.3	112.3	102.5
101.4	88.5	91.4	243.4	69.2	66.0	69.9	79.9	74.8	90.0	104.1	68.6
135.5	97.6	110.1	95.7	83.8	73.7	74.7	88.4	69.7	68.3	76.9	75.8
80.0	71.8	63.1	67.9	96.0	86.5	133.0	78.7	63.7	72.5	77.0	72.6
80.3	97.8	151.9	69.2	95.2	155.1	84.4	199.4	78.6	85.5	146.6	78.0
72.3	74.8	75.9	74.8	79.7	97.3	62.0	74.1	92.1	74.1	72.6	102.8
66.6	71.7	74.8	80.6	91.6	72.8	75.8	76.9	72.5	69.8	84.4	73.4
105.7	81.5	82.1	86.6	122.2	95.9	110.6	109.0	97.6	77.9	72.3	74.8
97.0	90.9	128.3	140.8	73.9	72.5	75.7	69.7	76.7	86.6	125.5	66.4
241.1	220.3	245.8	222.7	141.3	65.6	69.4	58.8	92.8	69.8	71.9	93.6
60.3	74.6	72.8	67.8	77.8	102.2	75.1	92.4	71.4	99.9	69.1	68.5
65.9	60.2	70.6	65.9	86.3	81.6	76.8	87.5	67.3	84.6	67.5	72.0
63.0	139.9	73.0	70.8	111.5	112.2	116.0	123.9	69.9	105.9	78.1	71.3
91.8	74.2	72.2	83.6	84.5	81.5	126.2	82.4	251.8	133.3	67.6	68.5
454.2	165.6	168.8	166.5	83.5	391.9	92.0	69.7	90.1	101.9	77.5	76.6
117.6	84.3	103.6	97.3	73.1	75.8	114.0	145.6	76.0	72.3	90.9	68.4
101.4	90.6	110.3	90.3	66.6	69.5	72.2	61.9	70.1	68.1	132.4	69.6
69.7	51.4	73.7	77.7	95.2	67.9	90.0	95.4	116.4	76.0	91.0	99.4
87.8	93.6	112.6	83.8	108.7	75.8	77.8	73.0	88.7	82.1	116.7	67.5
63.7	69.5	79.6	77.9	69.3	69.1	73.4	76.4	74.8	83.2	71.7	82.7
63.7	82.2	92.8	77.5	65.3	160.8	69.5	98.6	63.7	157.0	75.8	88.3
67.4	74.0	286.4	73.3	79.5	78.1	85.2	81.1	76.2	103.6	70.1	124.2
69.0	84.8	56.8	68.9	110.7	89.2	113.6	85.2	75.0	333.0	76.2	93.4
113.8	110.8	74.2	74.3	79.9	74.5	69.5	77.3	111.6	84.7	73.5	137.6
69.4	97.6	121.7	101.9	68.3	74.7	76.0	77.5	141.8	75.1	86.6	91.4
77.6	154.3	75.6	100.8	70.8	110.7	67.6	112.5	238.6	243.4	290.9	240.4
79.6	76.2	80.7	71.7	191.1	203.5	194.3	186.4	74.4	68.7	234.5	69.5
104.6	74.1	115.3	135.3	71.5	66.0	92.7	69.1	80.3	131.5	88.2	82.6
72.7	82.8	71.6	61.5	70.2	70.4	67.3	73.2	85.7	81.9	78.4	75.2
70.3	100.5	90.6	73.8	104.2	119.8	78.2	69.9	69.5	95.8	70.6	75.0
101.9	147.6	78.5	96.5	80.0	85.4	91.1	95.9	74.3	94.9	87.9	66.0
70.4	71.9	76.5	78.8	90.3	66.3	71.2	146.8	68.4	75.2	80.6	86.3
128.8	127.4	117.3	135.0	65.4	66.4	66.3	75.8	65.6	73.7	109.2	62.5
74.6	70.9	66.8	64.0	71.0	78.5	67.8	77.5	58.2	198.0	92.8	70.0
103.9	116.9	69.8	77.8	102.3	333.4	76.4	72.8	86.9	75.5	97.4	75.7
68.6	66.6	78.6	117.0	70.4	236.3	61.2	61.4	91.2	77.5	77.1	101.0
63.2	71.7	130.4	66.4	81.8	83.7	93.1	85.7	68.9	68.4	80.9	79.7
67.4	128.5	85.8	77.7	69.9	92.9	98.3	176.4	75.3	75.0	80.6	88.5
85.2	142.6	74.9	72.0	158.9	92.5	106.2	98.1	97.0	93.2	97.2	139.8
106.0	100.9	105.4	104.5	86.0	119.7	81.2	69.3	70.7	69.4	76.9	64.7
69.6	73.7	102.0	67.5	268.5	151.4	126.8	129.0	76.5	81.8	77.7	75.6
81.3	76.1	72.6	144.4	77.7	73.1	76.7	73.0	93.0	76.8	80.1	161.8
271.6	108.0	92.1	90.0	141.4	81.6	90.5	83.6	69.6	73.6	70.8	88.3
66.6	70.2	70.7	62.6	73.4	88.8	96.4	87.8	103.5	105.3	89.5	98.1
78.6	70.2	80.7	70.6	90.0	215.7	74.9	70.1	85.6	80.3	94.4	82.3
69.9	65.9	68.3	118.9	84.6	136.9	80.1	153.0	77.7	82.4	86.0	82.0
70.9	76.9	153.6	95.9	341.5	365.6	352.5	341.8	126.4	112.9	124.1	136.9

Continued on next page

Table B.3 – *Continued from previous page*

61.1	177.6	82.5	63.8	87.3	71.3	73.0	95.6	76.7	84.4	90.8	82.4
106.7	78.2	77.3	65.9	68.2	71.0	82.3	75.0	75.1	72.8	58.2	69.5
64.3	89.0	74.7	65.6	74.1	87.0	78.9	75.8	79.7	71.4	180.1	81.6
81.7	109.9	125.7	81.8	73.9	77.9	72.5	87.0	107.6	129.1	112.6	126.0
77.0	70.4	98.3	112.2	79.6	75.7	85.6	66.6	70.3	106.0	66.7	80.3
93.3	77.3	66.3	69.2	74.4	100.3	82.8	74.7	167.2	74.7	104.5	90.2
90.2	88.2	96.9	103.3	80.5	172.1	68.6	62.5	67.2	72.7	126.0	73.1
94.5	72.7	77.3	90.0	85.6	98.7	203.2	78.7	177.4	131.8	141.5	144.4
108.1	101.3	108.0	136.3	110.3	128.6	112.9	117.8	94.1	66.9	75.4	60.8
97.9	76.3	75.4	86.0	72.0	68.4	81.1	78.0	71.9	65.9	76.2	129.4
109.9	74.3	79.8	77.1	127.6	86.7	77.0	85.5	116.4	106.4	100.1	100.5
100.8	101.6	180.6	104.7	70.1	87.3	74.1	74.8	102.9	102.0	95.2	100.7
103.1	68.0	80.3	131.5	86.8	88.6	68.7	105.8	71.6	73.4	76.9	69.0
68.1	118.4	75.4	92.5	76.7	88.6	74.3	80.2	70.9	90.3	255.4	75.6
150.3	120.4	197.1	122.7	80.1	68.8	93.9	64.2	70.8	94.5	66.5	70.6
109.9	64.0	81.0	91.1	66.2	65.1	76.3	71.8	78.6	87.5	73.4	74.0
67.3	72.6	68.1	65.4	72.7	83.2	70.5	71.5	74.2	66.0	84.1	70.2
115.9	103.9	129.1	110.6	335.6	335.2	342.9	334.9	99.9	71.0	85.0	76.5
105.3	100.9	111.0	107.7	81.4	110.9	72.7	82.0	71.5	65.4	85.5	91.0
71.5	82.7	166.4	67.7	66.3	66.5	75.8	64.4	71.2	73.9	74.0	64.9
89.7	97.8	90.4	102.6	67.1	92.5	92.7	203.1	71.1	165.9	71.5	88.2
75.9	88.6	65.9	73.4	81.7	95.8	85.6	76.2	68.8	60.4	63.7	102.6
128.9	69.9	76.7	175.4	72.7	71.0	70.1	72.4	72.4	71.6	67.0	70.7
77.0	77.9	83.8	90.2	91.7	75.0	80.8	83.8	80.2	137.4	75.5	109.1
74.2	82.6	73.1	68.1	87.2	86.5	98.6	80.0	198.7	197.6	197.8	203.1
84.0	89.1	79.7	76.6	85.7	83.7	68.4	94.2	67.3	68.3	83.7	61.7
83.1	82.0	67.2	137.5	76.4	115.6	68.1	61.5	99.9	70.6	69.0	73.3
55.9	79.0	128.2	117.8	71.5	78.2	114.5	86.7	82.6	148.9	75.3	78.3
212.8	69.0	76.7	94.2	68.9	72.3	71.9	79.6	108.3	86.7	86.0	84.3
67.2	107.0	69.8	66.6	72.5	74.0	129.2	141.1	75.2	74.7	71.7	76.0
67.7	74.5	71.1	72.6	74.5	87.6	68.6	74.6	96.4	77.8	74.1	79.8
75.7	85.9	106.5	89.8	69.4	73.9	72.8	75.3	89.6	116.8	76.5	70.9
69.2	67.7	65.6	64.9	111.9	77.4	111.6	97.7	118.3	72.3	70.6	76.2
84.5	78.9	113.4	91.9	85.1	87.9	74.8	72.2	82.3	73.4	81.2	71.9
97.1	87.3	208.1	80.0	82.0	290.5	90.8	130.5	56.2	124.9	80.2	69.4
73.4	81.0	117.4	76.5	83.9	80.7	110.7	81.1	170.2	122.2	123.4	73.9
129.6	72.2	106.0	207.1	76.6	164.7	113.4	82.9	73.5	89.2	83.6	68.6
135.1	68.8	74.8	67.5	67.9	92.1	68.0	115.8	64.9	204.2	69.2	69.1
100.7	107.9	98.9	87.6	80.4	74.5	70.5	70.1	80.2	162.1	103.5	78.6
91.0	89.0	88.5	100.8	89.5	82.6	88.1	83.1	63.8	86.0	69.6	71.0
65.7	68.1	122.9	73.0	115.5	66.0	148.6	73.0	70.2	68.7	68.9	65.4
88.2	74.5	79.8	132.7	66.5	68.0	95.9	92.4	75.1	70.2	78.0	64.8
98.7	123.9	93.8	89.7	111.5	99.7	61.7	73.4	78.7	83.7	86.3	69.2
84.8	83.2	103.6	136.5	68.7	90.9	64.6	144.2	75.8	62.9	75.8	156.1
72.2	75.5	70.5	69.5	116.7	87.6	71.5	79.9	77.1	82.9	62.9	69.0
106.1	83.6	298.1	75.9	77.8	75.9	79.6	84.2	94.1	96.9	91.7	99.2
90.5	102.3	74.5	75.8	86.7	77.5	66.1	71.6	69.6	79.1	60.3	81.6
154.2	156.7	161.0	158.6	92.6	71.7	97.8	73.1	73.7	138.8	80.6	77.8
94.3	133.3	100.5	97.1	103.7	104.2	127.4	134.8	112.8	89.9	66.0	78.0
100.0	81.3	3013.6	70.7	111.5	86.2	79.3	85.6	85.3	78.5	88.1	88.1
80.8	70.1	75.2	71.6	87.8	80.3	79.0	75.7	70.5	64.7	69.1	75.0
77.8	111.1	82.3	137.4	69.7	82.5	69.7	63.6	67.8	102.2	63.6	320.6
71.4	78.1	565.6	67.8	176.0	98.6	78.4	73.2	80.0	73.6	75.5	92.8
68.5	75.0	70.8	69.6	79.5	72.4	70.5	67.8	101.2	76.2	72.8	74.2
79.3	75.9	70.7	77.7	128.9	93.3	84.5	79.0	103.0	132.5	136.0	120.1
115.6	86.3	75.1	69.9	77.7	67.8	81.7	79.7	94.8	78.9	71.0	121.0
86.4	69.0	79.0	64.6	104.9	101.3	78.0	83.1	80.0	64.1	116.9	72.7
70.9	74.9	66.6	112.0	72.4	75.9	76.8	77.6	76.8	65.4	67.3	75.4
80.0	72.6	66.2	95.1	112.5	82.3	98.9	71.1	83.8	66.0	72.6	74.2
65.7	79.2	78.2	71.8	74.0	79.6	147.1	72.0	80.1	86.7	72.1	80.8
104.3	77.9	85.8	70.9	81.5	126.3	84.4	86.7	88.2	68.3	67.6	69.8
91.7	207.1	80.7	146.0	65.6	85.3	688.2	72.5	210.2	223.7	224.9	214.3
72.9	76.9	70.9	67.4	80.7	77.9	83.5	90.0	79.0	73.2	72.7	70.5
75.6	338.4	99.2	72.8	267.8	266.6	280.0	276.7	95.1	75.7	68.5	73.8
76.2	79.8	74.5	73.5	77.3	82.3	90.4	75.8	248.1	79.3	71.1	87.6
79.3	82.1	85.1	68.8	80.2	71.9	89.0	80.8	110.7	97.5	90.3	94.1
69.6	70.5	103.6	66.8	77.7	72.9	82.4	78.6	77.9	71.2	71.4	75.9
95.1	74.6	77.8	68.7	68.8	1361.9	102.4	71.9	75.7	85.2	80.0	74.7
101.6	75.1	126.1	56.8	80.3	98.7	87.6	91.0	100.7	70.9	87.8	129.4
73.4	78.6	84.3	98.4	68.3	59.6	67.5	330.4	65.4	65.5	74.5	75.3
66.9	162.8	80.6	80.3	70.7	80.0	74.7	135.0	71.2	70.9	65.9	76.6
94.8	100.9	97.4	97.1	70.6	72.8	69.5	86.5	73.4	72.1	72.3	92.2
86.9	138.9	70.4	74.4	73.6	73.8	64.8	72.5	74.4	70.0	71.0	69.4
72.8	121.3	80.8	110.8	65.9	71.2	70.2	107.3	88.2	68.5	67.9	85.9
72.0	123.4	77.2	67.7	95.0	70.7	70.2	112.2	68.4	130.3	80.7	77.0
86.5	413.8	77.6	75.5	72.0	127.3	75.1	102.9	224.5	74.7	83.1	77.7
90.5	81.8	79.8	81.3	78.4	179.6	100.8	73.5	79.3	85.1	80.9	69.8
64.6	69.7	67.5	67.3	214.3	80.3	75.2	78.4	76.2	75.3	81.6	1337.0
70.6	69.3	66.5	66.2	70.6	63.0	67.7	75.0	71.7	74.9	96.8	84.3
71.4	96.8	70.9	68.4	114.1	106.0	76.7	74.2	137.4	87.5	69.3	68.0
167.5	159.6	160.9	160.7	90.1	64.4	73.8	94.8	88.3	79.8	68.2	73.3
69.4	67.6	311.5	96.5	76.4	68.9	102.7	71.9	178.7	179.6	176.9	204.9
85.8	67.5	95.1	81.9	66.8	74.9	73.6	75.5	92.5	73.9	85.8	71.7
83.7	67.5	88.1	72.7	67.9	82.6	73.6	92.0	59.1	80.4	135.3	70.8
69.8	118.3	67.6	74.2	210.5	213.9	197.8	193.4	106.6	99.0	237.6	103.1
75.7	95.3	86.6	89.3	88.5	91.8	88.4	96.8	73.4	71.2	75.0	79.8
75.1	68.4	82.5	95.8	71.4	89.4	69.9	162.0	60.1	66.6	67.0	71.6
69.0	70.8	68.5	72.1	76.4	114.3	77.0	75.9	65.8	68.9	66.8	75.5
71.2	81.9	80.1	73.2	71.7	82.4	83.6	78.1	188.9	153.7	165.8	177.3
121.3	83.3	75.8	87.6	76.2	77.2	79.2	87.5	77.5	75.4	197.7	70.1

Continued on next page

Table B.3 – *Continued from previous page*

70.1	69.0	78.1	72.6	76.5	114.6	69.2	69.1	71.7	125.5	75.7	83.0
73.2	77.2	100.6	162.5	70.5	71.7	134.2	66.7	66.9	86.6	71.3	70.6
71.8	68.6	82.9	71.4	85.8	85.4	80.4	85.5	103.1	110.0	101.5	104.8
69.7	146.0	75.2	69.2	67.1	68.0	73.6	91.3	83.5	418.6	70.5	67.8
77.9	72.7	91.7	80.3	102.1	83.8	141.1	80.6	77.4	78.1	68.2	77.2
171.2	180.1	176.5	181.3	89.4	111.6	100.6	117.3	79.6	69.2	77.1	73.2
84.0	94.8	89.8	178.3	77.3	128.3	66.3	76.0	77.9	77.6	83.1	88.7
73.7	68.6	71.5	85.0	74.1	75.5	71.9	71.2	362.6	70.7	92.7	80.9
83.6	73.0	75.9	67.9	67.3	76.5	70.1	79.8	100.0	72.2	76.4	74.7
85.3	78.4	78.4	78.4	119.6	75.5	92.1	98.9	96.6	83.1	83.0	89.8
74.9	71.7	67.5	74.9	77.9	90.5	83.8	190.3	87.4	72.2	80.7	72.7
61.6	68.9	67.8	74.6	72.7	80.1	75.8	120.4	94.6	77.4	92.4	86.4
94.9	81.8	85.7	85.4	108.1	66.8	75.3	117.9	65.7	128.9	67.4	112.6
125.7	154.7	125.2	129.1	81.0	79.8	80.4	95.0	89.9	74.7	81.3	79.7
85.4	67.7	265.2	72.5	81.5	159.7	96.1	227.7	179.1	70.5	85.9	72.1
65.4	72.8	66.3	68.0	80.3	83.7	93.7	176.6	64.5	76.4	72.6	125.9
114.6	56.6	61.2	62.3	73.4	73.0	90.3	76.6	473.2	70.4	71.0	66.5
74.6	67.5	70.8	94.4	73.7	69.3	71.1	82.5	81.6	117.5	72.4	88.4
81.8	119.6	97.5	83.1	81.6	87.1	109.6	103.8	71.8	73.7	85.9	72.9
76.1	74.3	210.0	65.5	73.9	83.8	73.7	78.4	74.1	75.9	122.6	62.8
147.0	71.6	68.3	73.3	73.0	73.0	101.2	70.6	77.6	78.4	82.3	91.5
65.9	87.6	68.6	73.2	149.4	77.7	191.2	80.9	82.3	90.6	371.0	122.7
67.5	128.7	71.7	79.0	78.0	71.7	70.1	83.0	70.9	68.4	70.0	70.9
74.2	71.3	72.4	76.3	62.9	75.0	72.6	63.5	71.8	70.0	85.2	88.4
90.1	87.4	74.1	75.1	187.4	174.1	185.6	171.5	68.5	113.8	112.5	72.2
68.4	69.7	74.2	177.9	77.1	67.9	62.3	67.1	103.5	74.1	99.0	77.5
65.1	60.3	63.7	78.1	78.6	361.7	77.7	86.4	194.1	173.4	170.5	181.7
70.2	72.9	114.6	68.5	88.8	69.2	91.3	69.7	68.0	77.8	71.4	65.9
112.4	69.3	88.6	71.9	75.6	70.2	75.7	72.4	156.9	68.0	72.3	69.2
71.2	69.4	71.6	75.7	70.1	74.2	65.0	74.2	96.1	80.5	67.7	123.7
92.4	96.2	95.0	106.7	81.0	128.3	85.1	138.8	108.0	93.2	105.2	104.0
82.2	86.7	89.3	86.3	64.0	72.8	71.0	87.9	73.8	235.0	82.6	205.8
69.6	75.9	94.1	80.1	69.5	75.5	70.6	68.2	64.8	72.2	111.6	85.9
109.4	92.3	138.1	165.7	106.6	75.5	68.6	67.9	76.6	88.3	74.4	191.0
68.5	53.0	85.1	71.4	70.3	66.1	71.3	75.7	72.3	90.6	74.2	72.1
75.1	113.0	67.4	94.9	92.8	131.8	98.1	101.8	71.3	78.4	136.4	77.4
99.8	82.3	110.0	75.0	80.7	80.7	83.4	78.9	70.0	86.7	74.5	100.1
128.1	77.8	69.6	96.5	69.9	86.7	70.1	66.0	80.8	69.3	67.3	77.4
129.7	158.7	76.6	80.5	68.8	76.0	78.8	70.3	65.9	72.5	201.1	67.5
134.7	91.9	84.9	100.3	73.3	77.1	77.4	65.4	73.7	76.0	101.6	62.9
406.9	78.7	85.9	62.8	130.7	83.3	78.8	84.1	91.0	82.8	78.1	85.5
207.4	202.4	206.8	206.4	74.2	88.9	76.5	83.7	78.8	76.2	77.2	98.9
77.8	90.2	87.6	72.5	406.5	66.2	85.4	91.4	150.5	69.6	79.5	93.8
72.2	78.4	104.1	77.6	69.7	73.1	73.7	75.4	74.6	127.4	135.1	67.4
108.0	105.1	102.4	98.4	72.8	73.5	64.0	72.0	82.6	97.3	73.1	66.5
82.0	81.6	65.4	66.8	66.6	69.2	68.4	71.5	80.8	85.5	99.1	282.3
63.5	55.2	221.0	71.3	93.0	72.5	91.1	73.0	179.8	238.8	188.3	185.4
91.4	101.3	171.6	98.1	67.7	74.0	72.7	74.2	79.2	68.0	67.0	76.2
93.1	74.7	171.6	73.9	106.7	78.7	76.0	81.2	73.9	69.2	74.0	72.9
88.9	70.7	94.4	67.5	82.4	75.2	72.9	61.9	67.9	92.0	76.7	75.7
100.9	67.3	79.0	67.3	76.4	105.9	70.6	71.2	131.2	75.7	78.5	82.5
133.4	59.6	163.4	64.4	269.6	140.3	172.4	121.5	98.8	100.7	98.2	93.4
66.6	73.8	69.4	74.8	94.1	87.8	85.6	90.8	73.0	63.7	76.4	71.6
90.0	93.1	112.3	92.4	76.1	69.5	67.9	75.3	71.8	68.8	74.7	68.6
67.1	72.3	63.9	99.5	226.4	73.4	84.9	74.3	161.6	79.6	72.7	74.8
67.7	65.1	86.4	70.7	119.9	111.1	71.7	66.1	73.9	71.5	122.3	72.8
78.6	74.5	73.1	71.6	73.3	65.7	66.3	78.7	286.1	199.2	208.7	196.0
115.9	92.2	84.7	105.8	82.8	95.6	78.0	90.4	74.9	75.2	76.9	77.5
81.2	64.5	100.6	92.4	69.1	69.3	99.9	73.6	77.0	94.5	69.6	178.7
70.3	68.6	71.2	79.4	70.9	69.2	80.9	152.9	68.1	104.6	63.7	120.7
145.8	77.0	78.4	85.3	60.4	71.8	78.5	84.6	77.5	68.5	70.2	76.0
75.4	67.2	73.2	75.6	78.8	68.7	69.1	74.0	123.0	72.3	75.8	144.9
540.3	555.3	550.7	537.5	71.2	90.1	71.3	72.5	90.7	74.8	116.8	87.0
75.3	83.0	114.1	71.2	85.5	78.0	70.0	98.2	70.6	74.0	212.6	95.2
117.5	71.0	76.5	97.9	95.0	79.1	74.6	78.2	108.8	102.9	73.3	96.1
112.2	86.9	92.3	76.5	71.4	120.5	68.2	67.6	81.0	83.9	72.0	83.1
78.6	111.5	75.1	66.4	72.5	72.0	74.1	96.5	81.3	72.3	156.4	70.8
143.8	162.7	80.2	66.4	65.4	91.0	70.7	74.2	73.6	125.6	75.5	76.7
69.0	86.1	64.9	75.2	77.7	93.3	66.9	73.9	86.9	84.8	104.1	87.9
79.3	72.4	82.9	68.5	73.5	85.1	74.8	76.0	100.1	77.1	305.9	71.3
426.4	104.9	197.6	66.7	75.1	66.5	69.3	73.9	83.6	69.5	75.3	106.4
76.2	104.6	75.5	72.7	64.6	67.7	65.8	106.3	78.2	69.9	101.5	63.0
100.3	68.9	65.8	78.3	68.7	90.4	106.3	69.0	75.5	71.2	966.4	83.3
133.7	111.2	127.1	95.7	76.6	68.1	77.9	71.6	148.5	79.0	83.2	223.8
71.0	84.3	125.1	83.7	77.1	75.3	71.3	263.6	101.4	64.5	68.9	65.6
75.2	85.5	74.0	78.2	118.9	90.7	90.0	83.0	207.8	77.2	79.3	84.5
74.2	111.2	79.8	77.0	79.4	78.5	74.2	79.8	118.0	96.8	98.9	107.6
95.2	70.1	69.5	75.5	96.5	116.5	91.4	96.2	66.0	73.8	64.6	105.9
81.2	66.2	83.3	79.4	72.3	73.4	88.5	72.3	70.4	63.0	77.3	106.3
71.2	71.3	77.1	75.8	85.3	83.4	76.2	354.1	143.1	92.9	94.1	93.4
77.7	76.8	68.1	166.0	90.4	84.0	102.5	78.1	154.4	135.0	145.4	125.7
72.2	80.6	68.9	74.2	119.5	71.9	77.5	76.2	91.7	77.8	286.4	128.0
76.8	78.5	88.2	189.3	91.1	75.9	70.1	77.4	194.3	75.7	69.7	84.0
90.4	69.2	118.4	64.3	70.9	80.6	84.5	67.6	82.4	140.3	79.1	82.6
65.2	71.9	76.7	77.5	62.5	151.0	107.2	68.1	70.9	111.0	157.6	82.0
98.8	88.8	88.5	116.2	92.8	181.2	100.7	69.7	149.4	116.1	116.6	131.5
79.3	107.2	77.7	70.6	106.3	111.7	119.0	137.7	67.5	63.5	84.5	101.5
68.2	65.2	68.9	68.2	83.1	72.6	77.1	206.2	78.6	73.4	74.9	202.7
116.0	144.0	94.4	104.8	81.4	97.3	70.2	81.5	77.9	72.7	93.6	72.3
89.4	84.7	76.0	74.0	70.7	72.7	70.1	70.0	134.3	192.9	102.5	106.2

Continued on next page

Table B.3 – *Continued from previous page*

66.2	68.5	69.8	81.3	74.8	74.8	75.0	71.5	89.3	82.0	92.7	233.1
73.3	77.5	74.9	65.4	75.5	77.4	97.7	116.8	88.0	69.4	69.0	75.6
84.1	80.9	89.8	83.1	83.9	84.3	85.3	86.9				
76.7	85.0	83.1	138.1	71.0	79.9	71.2	76.9				

Table B.4: Simulated 'Onion'-Shaped Loss Data for Dimension 5

78.9	84.2	170.1	76.9	72.0	78.6	79.6	83.9	78.2	103.4	190.4	68.4	74.5	71.2	73.0
73.1	74.5	109.4	148.5	91.5	86.2	90.1	105.2	92.7	86.8	97.2	70.1	162.4	77.2	70.0
67.9	73.4	77.8	75.0	78.2	63.2	66.2	74.3	98.3	68.4	74.9	88.8	81.7	187.9	70.8
79.2	75.0	73.9	70.7	73.3	89.9	79.1	86.9	86.0	77.1	69.3	73.8	179.2	69.8	222.9
80.9	78.3	84.0	74.8	87.8	115.2	92.0	93.9	92.4	90.8	88.3	185.9	70.5	67.0	74.0
136.3	196.3	120.4	130.0	114.6	71.0	88.3	70.4	158.8	109.3	72.2	69.5	71.5	76.0	73.1
72.9	73.2	75.0	78.8	97.8	136.9	146.7	87.6	88.6	88.3	111.4	159.3	64.3	74.0	69.9
83.9	81.2	72.4	69.6	76.4	65.4	72.0	81.0	77.1	75.9	68.5	90.7	78.7	66.7	70.3
78.8	61.6	74.9	77.2	74.3	94.8	99.4	88.0	98.0	107.3	104.4	69.9	68.3	69.8	105.7
114.8	106.9	78.4	105.3	70.4	73.5	68.0	127.2	86.1	75.2	89.5	81.3	192.5	101.6	62.0
82.0	77.2	91.7	81.6	89.8	81.8	71.8	81.0	78.2	78.7	81.6	71.9	74.4	80.0	71.1
74.8	72.4	70.3	78.0	82.7	74.6	70.8	75.0	182.0	221.9	75.5	73.9	104.4	69.6	72.5
87.9	81.1	92.9	79.7	79.6	71.8	91.9	69.3	71.9	74.6	90.7	72.0	71.1	87.8	118.9
72.3	88.3	69.7	73.5	69.1	82.4	70.8	78.5	82.4	76.1	80.3	99.1	73.5	74.3	72.6
132.8	121.0	120.0	128.9	124.6	86.7	133.4	60.1	68.1	71.2	84.4	82.0	80.7	73.1	79.9
70.3	76.1	138.1	73.2	64.7	87.2	66.3	76.0	201.9	78.4	67.9	77.7	65.7	69.9	74.2
69.3	75.8	65.2	67.3	77.7	84.2	66.5	94.6	75.3	71.9	72.1	78.6	81.2	75.2	69.0
295.6	67.9	87.8	79.3	68.7	79.5	89.7	74.6	66.7	73.8	222.5	78.7	77.4	87.9	79.2
84.4	93.2	88.6	200.0	73.1	78.1	84.5	61.1	71.9	68.2	87.0	74.2	67.8	78.6	122.9
68.4	63.6	72.7	78.6	119.4	77.1	71.8	102.0	64.7	67.0	73.2	117.9	74.9	122.1	74.9
212.4	103.0	89.7	71.1	98.8	70.8	64.8	102.9	89.3	70.8	77.6	83.9	88.1	69.0	117.2
83.5	68.3	104.7	94.4	66.6	164.2	62.7	64.4	67.5	59.5	67.7	85.7	71.9	78.6	68.9
88.8	118.3	119.2	94.7	90.6	115.3	73.6	72.9	82.2	75.1	73.9	68.4	67.3	72.0	70.5
73.5	64.0	54.7	72.6	80.8	79.7	245.9	66.8	69.2	72.8	81.6	66.6	76.0	73.0	90.1
179.2	79.1	77.4	82.2	76.9	78.1	72.6	69.1	84.0	113.6	83.1	78.6	81.9	68.5	71.1
156.2	151.4	178.2	169.7	160.3	84.0	98.5	87.4	79.5	77.8	78.2	76.6	77.1	82.2	357.2
65.2	113.2	76.8	85.8	74.0	74.8	106.9	73.0	66.1	67.5	151.2	66.2	108.3	69.2	71.8
73.4	84.2	63.0	74.8	59.8	82.1	98.9	347.5	80.0	102.9	82.5	75.7	78.7	80.3	68.0
68.2	83.4	77.3	73.3	91.2	119.7	137.9	191.2	141.6	127.8	86.7	92.3	165.1	82.8	75.4
76.7	78.5	68.8	92.4	69.3	72.3	79.5	68.2	68.7	70.6	91.6	90.0	69.8	72.2	63.7
89.3	90.3	82.4	94.1	77.7	73.4	75.3	70.3	72.9	70.5	76.7	79.6	68.1	98.7	72.4
77.5	98.8	85.5	106.8	85.1	72.6	71.9	85.2	103.1	65.7	192.0	76.3	69.1	76.2	70.2
170.5	85.3	89.7	101.6	89.4	83.8	122.4	72.9	67.5	104.4	73.4	70.1	78.3	83.1	78.2
107.1	76.4	65.0	86.0	88.2	70.9	68.8	70.8	78.8	72.7	72.7	71.8	82.8	115.3	80.0
90.7	79.3	84.0	85.4	87.0	104.4	83.8	75.2	85.5	79.8	167.2	59.8	70.2	69.4	302.8
85.4	79.5	94.2	108.1	71.5	77.6	88.2	97.6	74.4	75.0	70.0	69.2	67.2	71.1	86.5
68.1	126.4	65.9	66.9	71.9	71.9	90.2	100.3	81.7	71.7	76.1	70.6	70.8	155.3	69.5
91.5	92.5	85.3	79.8	72.2	84.6	171.1	75.7	73.1	93.8	75.1	96.8	71.1	72.9	69.3
95.4	68.5	138.9	60.7	68.8	73.6	96.5	72.5	86.3	74.0	160.4	139.1	112.0	70.1	70.2
75.5	286.1	74.9	181.8	75.1	75.0	93.7	91.9	71.0	94.0	118.8	97.3	86.1	91.0	96.2
67.8	68.3	70.9	71.5	85.6	80.2	84.2	102.5	77.5	78.2	70.6	67.4	67.4	79.1	71.6
79.2	79.6	164.2	172.9	67.4	84.7	88.5	80.4	65.1	131.6	131.1	72.9	73.0	118.0	80.9
102.7	69.5	75.8	71.4	78.5	86.1	70.9	79.8	71.6	62.1	136.8	98.6	98.8	99.5	187.8
80.8	78.3	74.0	65.8	98.5	73.1	87.4	78.5	68.8	71.6	110.2	87.9	310.6	100.3	102.8
76.6	134.0	70.4	81.3	69.6	120.2	74.2	97.6	91.3	70.8	89.3	72.2	72.0	64.9	86.4
72.1	75.0	75.0	133.8	81.5	99.5	88.5	86.7	116.6	118.9	61.3	100.3	77.4	94.5	68.5
84.1	96.6	76.6	158.0	78.7	270.9	290.4	264.7	259.9	280.5	76.5	86.6	72.8	92.9	73.8
73.1	134.1	69.9	84.5	87.1	71.7	76.2	409.2	68.9	76.4	65.9	117.2	68.5	72.1	68.9
84.4	71.4	76.8	90.1	74.5	87.3	85.7	87.3	92.0	177.9	106.5	97.9	101.5	100.2	98.5
70.1	74.8	70.9	144.8	77.9	79.8	79.6	86.4	80.6	71.5	135.5	136.2	135.7	474.3	158.4
73.3	86.0	71.5	147.7	77.8	72.9	66.7	78.6	88.8	74.0	72.5	82.1	78.5	79.9	79.2
166.8	194.8	236.9	157.9	164.0	66.3	76.1	66.4	66.7	97.4	73.9	73.2	75.7	80.8	81.5
80.8	83.3	105.9	91.1	127.8	63.2	81.0	84.3	111.7	126.5	67.8	81.5	63.4	61.6	75.4
80.5	91.9	71.2	85.3	129.8	66.3	101.8	71.1	68.0	68.6	75.9	82.3	76.9	83.6	86.7
84.8	117.5	98.3	84.8	88.8	69.2	125.3	78.4	69.6	65.2	79.5	77.6	89.7	103.8	70.9
71.2	74.4	67.2	72.2	81.6	72.5	110.6	74.8	83.4	71.1	113.4	70.0	62.2	142.2	62.0
148.8	158.1	156.2	153.1	154.6	105.1	164.9	104.1	106.7	113.4	115.5	79.7	78.4	74.7	94.0
82.5	70.8	101.7	68.7	63.0	85.0	65.9	79.6	144.9	88.5	70.2	87.4	81.1	91.1	71.0
86.8	73.9	70.8	77.0	82.7	73.1	90.9	83.1	72.7	66.1	66.1	73.9	70.7	84.2	83.9
112.1	91.3	62.4	73.2	68.7	91.7	74.0	77.5	126.1	87.6	71.9	70.8	75.0	72.0	89.1
65.1	93.3	70.8	64.7	307.1	75.7	163.3	77.3	79.1	88.4	80.1	81.8	108.0	76.9	79.3
77.9	67.0	68.9	130.8	67.3	83.9	78.6	73.4	64.9	79.7	89.1	88.8	113.3	77.2	75.8
189.8	188.3	204.8	194.3	196.6	70.5	94.4	86.8	70.1	72.5	80.5	89.9	110.0	85.8	81.0
78.1	84.4	87.6	112.2	83.4	67.3	63.1	78.2	223.6	89.0	64.7	66.3	71.8	62.3	65.1
79.5	70.6	70.8	78.4	71.9	68.1	69.1	69.5	71.9	81.0	67.9	69.9	66.5	92.8	78.1
85.8	67.2	66.0	97.9	78.2	129.6	130.4	142.9	142.0	143.7	79.8	66.7	117.2	87.5	109.5
78.2	93.3	65.4	73.3	105.6	85.0	66.9	94.4	81.7	93.6	86.7	77.8	84.0	91.8	83.2
76.7	86.0	72.7	67.3	72.6	90.0	73.0	70.7	86.2	93.8	72.6	67.4	100.1	73.2	84.2
86.6	84.3	83.8	86.7	79.3	152.8	149.7	148.2	164.7	154.5	88.9	80.4	72.5	86.5	74.0
76.2	106.6	73.1	109.7	93.9	71.5	152.1	70.1	68.0	232.4	90.5	76.4	96.8	72.6	74.4
100.8	88.6	66.9	61.4	70.4	184.0	72.8	100.9	82.6	71.1	155.6	99.8	98.3	92.9	91.8
79.2	69.7	77.6	162.1	79.8	80.5	89.3	63.9	72.1	65.3	66.4	68.2	56.4	72.1	70.9
72.2	70.0	68.4	87.1	71.6	68.7	94.9	112.0	64.5	71.7	71.3	68.0	63.7	70.3	87.7
67.2	88.8	80.8	68.9	68.0	94.8	71.6	377.0	68.7	72.9	119.2	68.2	74.7	226.3	75.1
81.5	77.7	74.6	90.2	77.1	90.0	83.8	101.2	88.4	83.7	75.6	255.1	106.3	111.8	70.7
66.8	79.6	71.6	70.5	103.1	67.8	66.7	100.5	84.5	73.0	78.0	94.7	81.9	81.8	1116.1
79.3	75.3	77.6	86.2	76.5	72.6	72.4	68.4	93.2	78.9	87.2	87.7	93.0	85.0	103.0

Continued on next page

Table B.4 – *Continued from previous page*

76.6	68.1	71.6	89.9	77.9	645.2	122.7	123.6	133.2	120.2	73.4	213.1	70.4	84.3	76.1
75.0	74.8	70.9	82.0	73.9	128.0	122.3	126.3	161.7	142.7	149.6	113.5	85.8	85.1	74.4
71.9	127.3	82.3	64.0	77.1	65.9	225.5	72.0	77.4	76.4	426.8	72.7	72.7	59.3	78.0
67.5	74.9	132.6	177.5	66.7	66.3	71.2	86.4	77.3	85.4	88.7	67.9	95.3	92.8	78.3
82.6	79.3	83.2	86.3	81.2	78.8	85.0	192.5	77.5	70.5	80.4	91.1	70.9	68.5	68.1
78.4	72.7	97.1	105.2	91.1	131.7	80.8	88.3	92.8	75.6	86.2	97.0	79.0	73.5	77.1
69.7	82.7	88.5	71.6	87.0	83.3	68.0	77.2	77.3	72.8	72.4	91.3	68.1	93.1	73.0
79.3	81.2	75.4	83.9	75.0	144.3	128.8	84.8	71.8	73.5	91.6	316.1	72.5	87.4	63.7
121.0	82.8	91.0	82.1	69.6	72.0	86.4	64.2	73.8	59.7	75.6	83.1	89.6	111.5	68.0
70.5	71.5	71.2	72.0	69.0	82.5	111.5	85.9	69.5	147.9	112.5	84.7	171.3	69.9	74.6
99.7	322.0	80.2	84.4	81.6	66.2	90.3	74.0	219.2	68.1	82.5	104.0	96.2	79.3	185.2
936.3	97.2	83.0	86.1	73.9	68.3	70.9	72.2	67.9	65.8	111.0	90.6	66.5	189.0	124.5
79.7	110.0	79.2	70.7	103.9	185.0	122.7	77.6	81.1	75.4	67.2	114.1	68.6	92.2	69.0
121.7	70.4	65.5	66.4	68.1	130.8	84.6	108.4	76.6	70.6	115.5	95.8	96.2	111.2	98.3
73.6	137.4	79.4	109.1	66.4	78.0	77.0	67.0	72.0	71.8	75.4	69.7	68.8	302.2	86.4
103.8	104.0	114.3	97.0	216.5	69.8	71.4	73.4	113.7	78.9	130.7	114.7	110.1	143.6	115.5
76.1	69.7	74.8	77.5	65.9	107.6	122.3	111.2	194.6	104.3	91.3	72.0	82.2	77.7	75.2
85.9	83.9	125.6	70.5	67.3	92.0	98.9	98.0	202.8	106.7	93.2	74.2	137.2	68.2	133.6
83.8	74.2	91.4	69.8	106.5	80.6	82.8	70.7	69.4	87.8	71.1	266.9	73.9	76.3	88.7
67.8	78.5	69.9	78.0	68.8	96.6	95.4	94.8	92.4	413.9	66.1	73.7	83.4	75.2	30.8
90.6	204.9	84.6	96.5	77.4	115.2	122.0	114.4	111.3	161.4	80.2	69.7	77.8	72.0	81.8
99.2	94.2	99.3	99.1	95.3	447.2	70.3	84.8	70.8	72.7	73.2	62.2	133.5	66.0	64.8
65.0	118.3	71.5	84.1	65.4	102.2	59.7	99.6	80.7	209.2	71.9	67.5	58.7	61.8	79.2
118.5	78.4	80.9	88.2	81.0	107.8	221.7	103.2	102.0	107.2	75.5	78.2	77.0	93.6	74.2
72.7	74.8	115.0	81.9	75.0	78.4	70.0	72.5	71.1	71.5	78.1	139.9	79.9	80.7	81.2
81.0	63.8	85.6	112.7	65.6	74.0	71.5	66.8	78.7	71.0	73.6	72.7	60.7	72.5	190.5
68.9	74.8	61.4	91.1	74.4	78.7	107.0	72.6	79.6	71.5	80.1	82.0	86.3	84.8	81.1
96.2	82.1	86.3	80.9	100.1	92.4	120.4	68.9	68.3	70.2	83.8	81.2	103.5	88.1	86.1
115.8	70.6	107.2	82.9	72.5	70.1	72.8	69.5	70.9	69.8	69.7	174.0	72.3	73.4	169.9
155.0	100.1	75.0	834.0	79.2	127.8	71.2	75.1	67.4	69.6	79.9	69.5	66.6	96.1	69.9
98.1	127.8	48.8	87.9	70.0	90.5	95.1	125.8	100.6	105.5	89.6	82.5	87.6	94.9	84.4
93.0	69.7	69.5	71.9	80.3	303.5	150.7	105.4	88.8	83.8	67.3	77.5	77.9	62.1	88.9
103.4	97.4	90.8	123.0	88.2	62.7	93.2	133.0	79.5	72.7	153.3	79.1	95.0	71.0	70.5
109.2	88.3	142.8	77.1	136.1	78.6	65.0	81.1	176.6	75.2	77.2	70.8	79.5	63.6	75.6
71.4	65.3	92.7	83.2	67.1	66.8	67.4	67.0	213.3	68.5	82.2	92.8	56.3	127.5	73.2
111.9	76.9	126.2	100.7	68.4	81.1	71.6	85.1	88.1	77.2	74.3	108.9	74.1	83.1	68.9
87.3	81.4	94.2	78.3	84.9	48.9	68.1	100.1	102.0	74.1	96.4	119.1	73.9	91.4	82.4
75.1	77.9	83.6	79.1	82.9	73.3	73.0	88.2	98.1	103.0	67.8	68.4	80.9	65.7	76.5
110.1	85.7	89.6	87.1	88.2	88.3	83.8	84.8	84.1	80.3	145.9	74.6	68.5	82.8	73.5
72.7	90.8	297.3	108.8	84.3	81.5	79.4	85.7	76.3	131.8	520.5	91.1	89.7	101.6	130.8
86.1	66.6	72.2	74.0	76.0	91.4	76.4	78.7	79.2	85.1	93.3	74.5	67.9	100.2	90.8
104.8	170.1	67.1	66.8	67.2	85.8	303.1	84.3	88.6	85.7	73.1	95.6	86.0	81.1	98.8
73.8	75.4	73.2	80.1	84.9	76.6	65.6	67.7	70.7	85.5	71.3	76.1	81.7	72.1	72.5
78.9	72.4	73.5	72.9	78.7	226.3	96.6	108.7	113.8	112.9	76.3	65.8	103.6	80.3	113.5
88.8	71.0	66.1	100.3	63.6	80.0	91.3	78.2	87.5	84.8	106.8	107.0	99.3	86.3	83.7
67.7	74.0	78.9	94.6	64.2	68.7	84.0	102.9	117.1	80.5	68.3	70.6	68.9	69.9	72.0
108.9	97.9	199.0	95.6	102.1	153.6	129.8	136.0	121.5	130.2	88.5	82.0	70.9	83.1	77.8
104.2	86.7	70.7	71.7	79.3	69.3	77.6	79.1	67.4	70.5	77.8	74.8	86.0	81.9	85.7
70.2	81.1	79.6	73.1	77.2	105.4	90.3	88.7	161.8	102.0	62.7	160.8	68.6	71.4	120.9
324.1	83.6	81.6	97.7	73.8	83.1	69.1	74.8	77.7	72.9	90.1	65.6	76.4	170.2	87.8
78.5	118.2	97.9	96.1	110.2	196.4	70.6	86.4	77.9	102.3	77.1	240.6	65.1	69.6	69.6
81.8	99.0	114.5	117.1	87.3	90.7	65.1	68.3	79.2	70.6	76.5	85.0	85.6	109.1	73.3
71.5	77.0	79.2	76.0	73.9	88.0	78.6	74.1	138.9	83.2	86.6	65.4	85.6	81.2	66.1
73.6	267.0	66.9	71.0	82.1	80.2	82.9	71.0	65.6	67.8	131.6	68.5	69.7	87.5	71.1
163.9	115.3	76.1	75.7	79.7	81.1	79.8	95.5	493.0	73.4	131.4	80.3	85.7	93.1	78.9
73.6	122.2	102.7	66.5	103.0	77.7	72.0	75.3	206.0	90.4	78.1	213.5	71.1	88.9	76.7
82.8	76.4	80.0	134.0	69.0	448.5	446.0	452.5	446.4	448.9	76.8	71.0	80.1	79.8	105.7
68.3	75.7	73.3	79.0	73.2	79.0	94.2	114.7	65.7	73.4	78.6	86.2	101.4	104.5	75.3
84.0	76.5	85.9	81.1	90.9	75.0	79.7	76.7	106.7	78.9	131.0	69.2	94.3	76.7	89.6
69.6	139.9	137.7	67.5	72.0	70.7	82.6	68.0	69.0	169.7	93.4	124.6	73.0	78.2	76.5
70.8	85.8	62.7	71.6	54.0	77.3	98.4	76.0	72.7	72.7	82.1	80.6	150.6	74.3	91.2
82.9	110.3	87.0	69.8	70.5	69.0	68.9	65.2	68.5	75.2	75.9	71.1	80.7	342.4	68.7
91.2	85.3	76.0	77.1	83.5	79.9	69.3	74.6	70.9	84.4	116.7	86.5	98.7	72.5	72.2
90.3	70.2	73.1	67.8	77.6	106.9	75.2	76.6	78.3	78.1	88.3	112.9	86.8	106.7	75.5
70.6	71.6	72.5	72.0	95.3	65.3	74.8	64.9	72.4	73.2	76.8	78.9	83.0	72.0	72.1
83.5	132.5	67.4	95.3	73.7	159.9	81.8	77.7	72.1	80.0	73.2	71.3	70.8	133.4	109.9
65.2	72.9	71.8	80.8	84.7	68.2	81.4	80.1	87.2	91.9	125.1	83.7	97.0	73.4	75.0
96.3	93.8	104.4	95.4	164.8	86.6	74.8	75.2	90.4	73.2	96.9	74.3	96.2	71.6	71.2
88.9	84.5	68.1	70.6	71.8	75.3	69.2	74.3	63.9	64.3	73.7	68.7	83.8	118.1	69.7
119.9	71.8	130.1	66.0	80.9	98.6	74.0	62.0	68.1	63.5	93.0	117.4	98.6	228.1	99.4
72.0	72.9	84.4	89.8	69.9	79.6	75.0	73.0	68.7	87.9	74.6	80.5	75.2	70.3	78.1
78.0	68.9	70.7	68.7	73.3	249.1	86.7	78.7	77.1	111.4	80.9	77.4	67.2	107.9	86.6
62.9	77.2	72.9	69.3	80.8	199.7	68.8	87.9	71.4	69.3	77.2	74.0	64.8	76.5	82.6
83.6	76.7	72.9	67.9	70.4	91.8	74.4	70.3	74.8	75.7	82.9	92.0	77.2	76.6	79.7
186.9	71.5	89.2	79.8	77.4	83.3	80.5	110.6	76.8	78.0	86.7	72.9	65.4	75.7	72.8
84.3	94.2	77.9	69.3	73.6	185.1	118.9	121.5	120.1	122.0	82.5	69.2	62.0	95.8	69.5
74.4	79.3	83.4	136.8	76.7	80.2	71.2	75.2	67.7	66.0	62.4	115.8	76.2	65.0	63.2
142.5	157.2	132.2	131.3	130.7	72.0	62.9	64.0	70.4	64.2	77.9	80.9	73.1	72.8	146.1
104.4	138.8	99.9	108.6	104.2	78.2	89.5	70.1	76.1	279.8	106.4	104.3	92.1	85.8	82.3
74.9	68.9	72.3	100.3	73.0	67.6	79.4	67.2	79.1	72.4	81.4	95.4	71.8	152.8	71.0
82.0	80.1	75.5	80.1	78.5	77.3	77.6	72.5	72.1	75.2	75.6	70.6	247.7	111.7	66.0
155.5	127.0	150.9	166.7	138.2	216.4	219.4	220.5	223.6	219.4	85.7	141.1	88.6	89.4	100.3
81.4	72.9	80.9	79.6	106.1	107.3	98.0	260.3	107.6	100.3	77.8	67.6	70.4	1927.4	66.7
75.0	207.7	64.8	102.0	91.0	72.2	87.9	64.1	67.2	66.9	81.5	84.1	77.2	85.5	69.9
74.2	67.8	78.9	67.1	77.9	124.8	67.3	77.7	74.1	68.1	107.3	131.5	103.0	119.1	101.5
84.0	85.9	70.4	66.3	83.1	76.5	176.9	76.1	89.0	98.7	74.0	79.8	97.5	73.2	75.9
80.5	85.0	76.6	70.5	77.5	70.1	85.2	64.2	168.0	66.1	77.1	89.6	79.1	85.4	83.2
114.6	64.6	72.2	83.6	71.0	71.3	71.5	92.7	69.8	66.2	90.0	120.9	84.7	77.4	97.3
108.3	69.1	69.2	637.4	86.3	115.8	68.8	74.5	66.9	85.3	73.6	75.3	129.5	67.5	84.3
72.6	75.0	79.4	103.4	69.4	69.6	88.8	73.8	65.7	80.1	66.3	85.6	69.0	107.5	110.7

Continued on next page

Table B.4 – *Continued from previous page*

118.4	113.0	138.0	113.7	121.8	69.6	76.4	66.8	87.1	63.5	68.8	84.8	125.0	87.8	80.2
78.5	76.7	112.6	141.6	68.8	254.2	72.5	100.2	81.8	83.8	71.3	67.0	86.3	67.2	67.0
72.7	68.4	81.0	76.3	67.2	71.0	76.8	74.8	82.3	80.8	72.5	178.2	82.2	75.4	75.0
71.7	67.1	81.1	122.6	72.4	106.3	104.3	117.1	110.8	109.4	78.1	84.8	99.1	90.5	82.7
74.1	95.2	68.9	72.6	74.9	80.7	79.5	95.1	81.4	95.3	91.8	81.7	442.3	81.4	95.7
96.5	101.1	95.7	96.8	117.2	70.5	84.5	72.2	97.6	109.9	77.6	73.4	92.2	71.5	149.6
86.0	108.4	99.3	90.0	93.4	98.7	105.6	115.3	127.3	108.8	79.3	98.9	82.3	61.1	87.9
81.7	88.7	93.6	83.1	91.4	106.1	113.8	97.1	102.8	110.2	74.3	75.5	65.7	102.3	63.8
98.9	77.3	142.1	103.2	76.4	73.7	74.9	71.3	75.9	76.8	103.2	69.4	75.5	63.8	66.9
70.9	73.6	71.0	78.9	70.1	71.1	69.6	88.6	69.9	128.5	117.3	119.6	135.7	119.2	114.7
80.1	72.0	65.9	76.1	67.7	73.4	113.1	63.5	67.5	65.4	73.1	86.5	86.1	114.6	76.9
74.5	65.2	69.9	74.1	77.7	74.1	78.1	77.6	75.6	72.7	75.8	73.9	92.8	71.3	63.7
75.8	102.3	83.6	147.4	113.5	152.4	198.0	146.9	146.2	151.1	71.6	64.7	73.0	71.7	108.6
103.1	179.9	91.4	72.8	71.6	73.1	72.9	97.0	81.9	70.7	79.6	203.3	76.1	77.3	76.2
81.5	71.8	97.3	69.0	75.4	111.1	110.0	103.0	107.3	105.8	83.0	78.2	78.6	131.0	77.4
80.6	57.1	90.3	109.2	75.3	341.1	343.6	332.9	336.0	347.6	94.3	114.6	102.4	80.8	117.9
125.4	66.2	75.7	101.7	70.4	81.8	74.4	81.2	64.1	83.8	76.5	112.3	89.2	66.7	145.9
89.7	80.7	75.5	79.5	69.6	69.1	68.1	68.7	91.7	75.2	67.6	74.9	111.1	70.9	73.9
75.1	81.8	91.8	77.2	78.4	236.6	107.7	105.6	116.6	102.1	72.9	81.5	104.9	135.8	70.2
81.4	69.0	74.3	74.6	57.8	145.9	71.6	69.1	116.7	125.4	69.5	67.7	64.0	75.5	193.5
68.8	110.4	122.5	69.9	64.5	97.7	138.4	108.8	100.2	107.2	178.1	92.5	101.4	95.1	86.7
70.1	75.8	70.4	76.2	72.8	66.0	199.4	68.4	71.8	69.2	73.7	73.0	211.0	104.6	86.6
77.1	102.1	96.8	73.8	74.9	69.5	75.5	69.7	83.2	155.1	102.1	121.9	137.5	117.2	118.2
124.3	125.8	150.7	149.0	120.6	66.7	62.1	67.9	79.6	74.1	83.9	85.6	91.3	144.6	86.9
91.6	69.1	73.8	59.3	77.0	65.1	71.2	71.4	75.2	73.2	108.0	76.2	68.3	82.9	70.4
98.6	93.5	99.6	93.2	111.9	72.3	76.1	101.3	94.9	69.6	78.9	69.1	269.5	75.1	80.2
65.6	66.5	82.9	60.7	73.3	69.1	80.5	84.1	71.0	94.4	68.0	73.9	71.2	72.6	79.7
81.9	392.5	79.5	82.7	88.5	66.4	86.2	64.3	82.0	117.2	84.0	79.6	71.6	135.6	68.7
83.6	85.4	76.3	78.2	169.0	92.7	75.1	75.9	76.0	70.8	98.4	80.1	69.4	120.2	80.5
251.2	127.8	110.0	138.4	107.3	97.4	84.6	81.0	90.1	79.1	74.8	69.9	69.8	76.0	66.3
81.1	67.0	64.0	82.4	77.8	64.4	68.4	96.5	145.5	66.9	93.9	88.4	79.3	95.0	216.7
70.6	71.1	69.6	309.0	72.1	69.5	78.6	66.0	90.7	70.4	68.2	72.1	72.8	72.1	180.8
72.8	78.7	107.7	65.5	78.8	71.2	84.3	72.1	64.0	70.8	96.2	111.3	76.4	93.8	74.5
65.9	101.2	86.7	65.3	78.5	67.2	59.7	74.4	82.7	64.8	746.3	88.0	136.8	70.9	74.0
68.0	74.1	73.2	66.6	104.2	96.8	70.9	68.9	72.8	99.6	71.2	74.8	92.5	109.8	77.6
62.8	183.9	83.3	85.0	76.5	95.4	94.6	88.5	105.4	101.2	81.2	63.0	63.8	79.6	66.5
81.7	98.3	76.6	87.4	96.6	176.7	87.0	82.4	70.5	100.2	69.7	59.2	74.5	72.2	69.0
79.4	73.3	73.8	131.5	76.5	69.6	88.1	71.1	73.4	80.9	118.8	96.4	87.7	98.5	88.8
67.4	79.5	78.5	72.2	73.3	80.9	95.1	76.4	507.4	100.0	67.4	200.6	95.1	69.2	73.5
112.8	107.5	96.8	96.7	94.0	97.6	77.0	87.1	121.8	71.0	72.9	78.6	135.3	277.5	80.6
61.4	71.8	68.8	71.9	75.5	83.8	86.3	81.7	88.4	86.7	80.3	87.8	75.3	72.9	127.9
67.5	114.9	78.5	80.5	462.7	71.7	75.3	95.4	66.7	97.2	91.2	64.1	70.6	73.4	86.1
75.6	63.6	72.0	150.8	77.6	73.2	85.4	93.5	72.2	83.3	68.2	78.4	73.2	71.0	70.8
179.5	98.3	85.0	90.7	105.6	70.9	90.9	68.0	97.6	68.1	72.6	82.0	75.8	94.4	86.3
92.5	272.7	87.5	74.0	74.8	93.1	185.6	75.1	130.0	81.8	111.6	82.3	73.2	77.0	128.8
88.3	75.8	72.9	73.5	75.6	79.9	71.4	73.0	100.5	172.0	69.5	89.4	70.7	72.2	69.2
122.9	107.7	69.1	64.5	131.5	73.1	66.0	67.0	90.4	87.6	80.9	75.5	108.3	79.5	196.3
97.0	146.6	81.1	83.2	91.6	78.5	79.2	85.3	77.2	69.5	68.5	66.1	71.8	70.7	74.5
85.7	100.8	89.6	76.6	86.1	82.6	115.9	62.2	401.7	68.8	84.7	86.4	69.8	83.2	105.5
68.6	83.0	80.0	83.9	134.9	70.1	82.4	71.8	73.8	73.0	71.8	113.5	69.6	67.9	70.1
68.7	131.4	72.2	75.8	91.6	104.6	70.7	68.8	76.7	202.9	90.0	76.6	73.6	82.8	71.5
67.8	157.8	77.1	82.4	71.9	139.7	79.9	90.5	146.2	96.3	79.9	129.7	74.7	90.5	71.6
110.9	73.0	76.1	81.0	76.2	75.3	84.2	77.3	79.7	73.8	95.1	87.2	115.4	83.7	95.0
89.6	129.6	79.2	86.0	77.5	121.3	185.3	129.1	103.1	99.6	113.6	142.3	128.3	77.4	90.7
77.1	76.7	104.8	161.7	71.5	82.7	91.4	71.8	79.6	76.1	87.2	61.3	76.1	72.2	86.0
69.5	142.2	83.7	87.7	131.4	67.9	126.9	69.3	159.1	77.6	81.4	78.3	127.3	75.2	81.3
107.1	70.9	64.2	75.2	93.5	80.6	65.7	75.8	117.5	71.0	67.8	66.0	81.2	87.4	69.0
73.0	98.0	80.2	75.6	73.2	65.7	76.5	73.4	112.6	85.7	133.3	139.8	67.8	72.1	71.0
85.4	80.2	88.8	81.3	88.1	65.3	70.7	139.3	415.1	74.7	68.6	62.0	69.0	78.4	86.9
60.9	68.8	71.4	77.0	65.3	88.4	176.6	66.0	67.7	114.7	86.0	83.9	84.2	75.3	74.8
109.8	76.9	109.6	70.0	74.4	110.5	121.8	74.1	79.1	72.4	82.6	75.7	75.6	77.8	86.9
121.9	97.5	94.3	86.3	88.4	137.1	77.4	63.2	67.6	70.4	150.6	134.2	70.4	72.2	91.2
122.4	109.4	113.6	113.4	172.3	84.1	97.2	74.5	83.6	75.9	81.4	71.5	75.6	69.3	75.8
67.8	76.0	77.2	77.1	72.6	72.6	73.2	73.6	65.5	133.9	240.6	69.2	66.2	81.6	71.0
165.7	137.4	64.3	75.5	70.6	87.1	88.6	87.8	88.7	86.2	86.6	82.0	114.5	78.5	89.3
84.1	68.0	70.5	84.6	61.9	80.5	187.8	76.1	126.9	75.5	118.9	101.7	130.0	99.5	104.4
82.8	81.9	92.6	112.2	84.2	73.6	69.0	67.9	72.0	70.6	209.8	70.1	70.4	68.8	79.8
69.3	88.4	76.8	66.3	59.7	76.2	90.5	70.7	74.1	74.5	73.6	67.4	74.4	64.9	68.9
114.2	75.2	81.9	81.5	77.2	95.3	89.8	79.3	79.1	79.2	75.0	74.0	79.5	72.3	70.3
60.4	69.6	80.9	91.5	77.2	74.1	78.8	79.9	69.2	79.2	80.7	76.7	75.0	92.3	76.6
78.0	76.7	85.7	73.5	118.8	67.8	99.0	176.7	84.1	64.4	78.1	123.2	77.1	86.3	92.7
87.5	79.6	81.9	77.1	84.3	70.4	68.4	68.2	72.1	75.2	77.8	223.8	102.2	85.2	84.9
92.3	79.3	150.0	78.1	85.0	66.1	65.9	82.7	122.1	71.2	76.4	63.9	68.4	136.1	72.7
80.5	70.7	69.9	68.2	86.0	80.4	89.2	79.1	99.9	81.7	80.8	84.9	110.8	85.5	77.5
68.6	70.5	79.0	102.6	76.1	70.6	78.4	74.8	69.7	71.7	98.5	88.0	90.6	99.2	85.6
75.5	82.6	74.1	87.9	70.4	70.2	117.1	71.0	93.5	72.1	75.2	73.2	83.0	69.5	71.7
87.5	83.7	99.7	93.1	92.7	80.4	89.2	86.0	71.9	223.4	95.3	89.5	82.0	84.8	87.4
62.8	72.5	79.3	77.1	69.8	101.2	129.4	149.3	102.2	106.7	208.8	246.8	209.1	237.0	208.2
72.5	84.9	86.4	67.7	72.8	100.4	74.1	89.2	72.5	70.6	70.6	72.2	77.7	76.1	94.7
89.9	79.0	88.3	73.3	82.3	82.3	83.4	87.1	93.3	78.6	70.2	153.1	70.0	78.6	106.7
97.8	80.1	71.1	82.1	84.2	80.2	78.0	81.0	131.7	122.6	104.1	70.3	68.1	66.6	68.2
86.9	99.5	92.8	110.9	96.4	93.9	93.0	125.4	157.4	165.7	73.2	92.1	162.1	84.8	101.9
77.6	187.6	72.4	77.1	73.5	75.2	78.3	73.0	78.4	78.7	72.9	116.6	88.0	71.3	84.0
100.9	81.5	81.4	79.8	85.5	90.6	101.2	161.2	101.5	103.2	69.5	71.6	66.4	77.0	68.8
782.8	71.1	70.9	83.0	67.6	79.1	75.5	77.1	73.8	76.5	98.4	84.9	74.6	73.7	76.2
84.4	157.0	72.3	78.1	89.5	150.0	74.7	358.6	82.4	79.6	77.3	74.5	135.5	332.0	76.2
72.9	87.0	69.0	74.8	72.1	151.7	74.8	74.9	83.9	70.2	108.8	153.4	65.5	72.7	81.3
62.7	68.7	96.2	63.7	67.5	65.9	83.0	120.7	82.5	119.0	67.7	74.5	115.5	81.1	200.9
79.0	82.9	79.5	75.6	76.6	71.3	67.3	85.7	82.6	68.9	81.4	79.2	76.5	75.2	147.2
77.0	75.7	77.8	80.0	89.9	281.4	285.5	281.1	275.0	283.3	67.5	82.1	79.7	68.2	73.3

Continued on next page

Table B.4 – *Continued from previous page*

201.1	118.3	146.3	132.3	200.2	72.0	88.8	67.2	86.8	231.7	120.1	96.5	97.8	96.2	116.7
72.0	71.9	84.3	94.5	72.1	94.4	104.2	96.5	465.7	95.4	92.6	70.1	69.6	76.3	70.4
79.2	87.9	73.5	107.6	75.9	112.4	126.5	141.7	111.6	176.8	81.6	83.0	72.3	73.0	66.8
79.1	122.1	176.3	82.2	130.8	77.3	74.9	80.1	70.8	78.8	77.7	78.5	79.5	91.3	77.8
82.6	77.8	99.0	73.0	83.2	105.5	88.9	100.5	180.5	85.4	109.3	100.7	73.2	72.7	95.9
78.5	70.6	89.9	71.5	77.0	66.4	81.9	75.0	71.7	70.3	96.0	68.6	71.0	74.3	73.9
75.9	71.2	119.7	98.6	92.4	94.1	143.9	86.6	87.4	402.6	81.1	79.4	73.7	67.9	170.8
193.6	141.4	72.9	100.8	82.1	81.1	89.2	78.4	82.1	83.0	73.7	78.6	129.0	73.3	83.1
101.4	97.6	101.1	99.3	101.0	80.8	84.5	73.4	73.9	76.5	68.6	72.3	64.8	68.3	68.6
59.9	78.2	73.0	140.5	74.1	66.6	69.7	78.0	72.8	66.3	67.7	65.1	89.5	66.3	68.3
107.3	112.3	104.6	106.2	106.3	89.0	101.9	77.6	91.5	84.5	77.5	88.9	96.1	73.9	70.0
78.2	83.5	80.5	69.1	74.5	71.1	91.9	72.3	66.2	123.9	74.2	68.5	80.3	92.0	72.5
75.5	187.4	72.7	80.6	84.2	76.9	68.7	70.6	106.2	137.0	75.4	80.3	94.5	76.2	74.8
88.4	89.9	78.6	86.7	77.1	67.1	73.6	67.1	69.8	67.4	75.0	152.3	65.4	170.4	108.0
96.1	103.4	104.9	99.2	91.1	84.7	81.2	84.3	85.3	79.9	75.3	99.2	83.5	77.8	68.4
94.3	91.8	80.0	72.3	153.2	82.2	69.3	147.3	101.0	72.7	84.6	79.1	74.7	105.1	79.2
58.9	74.8	80.6	188.0	66.4	68.4	89.0	71.8	76.8	211.0	75.8	71.9	75.3	276.2	61.5
75.5	77.2	77.5	82.1	71.4	89.2	78.2	90.3	78.9	121.0	77.7	69.3	76.1	71.5	73.8
74.9	74.1	76.6	84.0	104.4	81.0	74.3	92.6	71.3	106.7	68.2	67.9	67.9	78.6	67.6
108.3	72.6	70.8	78.5	74.4	70.3	166.9	67.1	62.5	115.6	107.9	107.2	113.3	107.6	147.7
68.9	65.4	69.0	67.5	104.3	81.3	80.1	174.1	122.3	159.6	69.7	107.5	67.9	87.8	67.4
64.6	72.9	64.1	80.4	67.8	79.6	79.5	81.3	82.8	83.2	78.4	138.5	96.5	103.6	78.5
148.2	94.1	91.1	273.4	87.5	76.4	70.4	391.6	81.9	82.9	71.1	106.1	75.9	76.2	79.4
85.6	77.5	86.5	137.4	78.9	68.1	87.1	414.1	68.2	71.3	72.1	92.9	72.1	74.2	71.1
296.2	75.6	75.6	75.6	86.7	162.0	152.6	153.0	158.8	192.0	91.5	285.9	72.2	78.0	73.0
62.8	83.5	75.8	75.4	74.8	90.0	74.0	78.6	89.1	140.8	70.1	75.6	77.2	76.7	71.2
83.6	69.7	72.3	84.2	71.7	65.9	71.7	74.3	73.4	67.6	84.3	89.5	75.5	99.0	82.3
142.2	89.1	87.0	112.7	167.9	68.6	67.5	72.4	103.3	64.7	117.8	87.0	72.4	73.1	86.1
96.9	83.7	118.7	128.8	84.5	83.9	120.5	82.1	90.0	71.7	88.2	84.5	98.1	117.5	82.1
68.2	62.4	164.4	109.1	69.0	68.8	90.8	74.6	78.2	97.9	83.8	73.8	71.8	72.4	67.3
85.6	82.0	76.9	66.4	169.4	99.4	84.0	64.0	164.1	68.7	75.7	86.6	89.8	79.6	135.5
78.0	80.4	84.0	71.8	74.7	76.1	62.6	73.7	76.8	98.6	214.0	93.8	69.2	63.8	67.9
73.5	66.5	73.5	78.3	85.7	65.5	70.1	82.3	97.9	84.1	94.9	95.9	94.6	110.9	96.0
65.4	72.2	66.1	69.6	65.5	74.9	73.4	72.2	189.8	72.9	76.2	124.2	77.9	80.1	77.9
67.8	115.5	73.6	92.0	68.7	69.8	75.4	79.3	129.8	74.7	66.0	82.1	78.6	72.0	111.3
92.6	86.7	83.2	82.4	89.0	94.2	81.0	87.8	81.8	98.6	69.4	94.7	82.9	70.5	125.9
75.5	75.0	69.8	79.5	105.6	70.7	172.8	65.1	78.8	68.1	204.3	107.0	114.3	113.3	118.2
81.9	73.6	87.2	71.9	71.2	67.1	71.8	115.3	70.7	69.7	62.3	109.5	69.1	98.6	64.9
77.7	115.5	88.7	124.4	84.9	74.2	71.1	314.4	68.5	56.8	71.2	70.5	67.6	73.6	98.4
68.4	77.4	72.2	128.7	74.4	77.4	81.4	193.5	84.1	74.0	78.2	80.4	78.5	70.8	78.3
102.0	77.0	89.8	69.1	69.8	87.5	65.6	80.2	82.7	72.8	71.7	75.4	74.3	89.0	85.7
80.4	299.6	72.6	80.9	67.5	85.9	87.0	89.1	86.1	87.1	77.6	70.9	68.2	89.6	80.1
80.3	78.9	95.4	77.0	73.7	206.9	120.1	131.5	128.6	129.4	73.7	67.0	66.9	86.4	784.3
94.4	89.2	90.9	84.7	85.3	100.5	92.0	91.5	98.9	97.4	288.5	77.6	69.3	80.8	67.2
74.8	72.2	86.3	84.1	84.1	71.6	70.3	68.3	73.8	70.0	185.6	240.6	215.3	228.2	185.0
71.0	70.2	80.8	94.8	77.3	76.3	110.7	83.2	92.1	69.6	84.9	83.2	96.7	92.2	80.2
174.3	185.5	179.2	175.8	171.1	68.5	69.9	69.9	73.0	83.3	68.6	76.0	73.7	85.4	225.8
77.7	69.8	81.8	71.1	92.2	101.2	110.7	101.9	110.8	95.1	207.8	77.9	71.5	247.9	75.3
519.2	516.7	515.5	516.2	515.3	73.9	84.1	69.3	69.1	90.6	74.1	65.5	69.2	75.2	71.5
100.7	95.3	160.4	98.2	92.8	69.2	71.6	100.5	71.1	75.7	109.3	87.7	80.8	82.5	89.0
245.1	73.9	67.8	81.6	108.6	74.8	77.5	78.5	76.2	107.7	109.4	80.0	78.5	77.1	79.1
69.8	85.9	129.3	60.6	76.6	67.5	68.1	66.3	155.0	73.2	506.5	70.9	110.4	73.3	71.8
70.6	80.8	105.6	82.9	142.6	69.7	107.6	93.9	65.3	89.3	66.2	68.9	88.5	135.2	111.8
127.2	106.1	103.9	105.0	108.9	228.0	61.3	72.2	66.2	64.9	89.0	74.1	77.8	108.8	79.1
76.2	92.8	83.7	101.5	89.2	98.2	107.1	90.2	109.0	88.6	75.3	143.6	102.4	82.4	88.2
72.5	67.1	68.5	72.2	65.9	69.7	74.1	73.5	81.4	70.3	75.3	74.4	218.3	77.8	76.2
68.8	78.8	87.2	78.8	78.3	69.8	73.1	73.2	106.3	61.9	85.3	74.6	189.7	80.1	69.6
111.4	109.7	68.3	65.3	79.3	74.0	80.5	71.6	103.9	67.1	83.8	116.4	86.9	90.9	70.8
168.6	65.1	69.0	77.1	72.8	91.6	112.2	77.5	87.4	84.3	80.3	77.9	86.5	90.0	89.6
72.1	77.4	74.0	71.3	75.3	91.3	70.6	81.0	91.2	69.9	493.7	71.5	66.7	73.2	72.1
103.9	168.3	97.8	96.7	131.1	84.5	81.4	76.5	68.6	75.7	73.1	120.1	68.6	76.8	145.3
83.4	72.4	68.0	98.6	73.1	93.6	110.0	94.0	387.9	91.3	80.1	69.0	72.0	123.0	77.7
76.7	82.0	79.8	89.3	89.5	69.7	74.3	65.6	70.1	72.7	134.3	85.7	90.7	91.5	94.6
68.6	75.9	66.9	153.0	91.3	72.8	70.9	88.4	70.4	94.5	121.2	171.6	113.9	112.2	111.7
69.4	78.1	74.8	1112.9	79.7	90.2	73.2	68.5	70.6	84.4	84.0	69.4	78.4	86.2	74.7
372.9	400.4	374.3	372.4	367.5	79.2	64.8	66.8	125.2	93.2	154.2	68.6	79.6	74.0	70.9
79.7	83.3	78.4	81.8	94.9	73.3	79.2	80.1	96.4	1537.3	181.4	208.9	134.6	149.4	138.5
78.6	80.1	68.6	75.3	94.3	73.9	106.5	105.8	95.4	82.2	66.3	71.5	77.6	68.7	202.7
66.2	125.6	214.9	86.9	124.1	74.7	93.4	89.7	79.8	73.0	85.2	105.7	86.4	90.2	82.0
76.5	199.8	86.8	96.4	82.4	91.4	82.1	78.5	77.7	139.8	73.8	81.3	82.0	119.9	86.4
69.1	73.2	77.5	107.8	76.6	150.6	228.4	202.7	143.4	146.8	109.4	124.5	107.2	115.3	108.1
73.9	75.0	80.5	77.4	81.0	69.3	85.9	120.5	83.0	66.7	79.5	64.3	82.6	68.4	113.9
64.3	84.3	77.3	88.2	66.5	82.0	81.3	77.1	249.7	66.2	70.2	73.2	81.4	82.8	70.8
68.5	66.3	72.1	71.9	77.7	137.4	141.3	136.5	129.6	141.6	64.1	78.2	86.8	75.1	80.3
86.3	88.1	102.6	92.2	78.1	76.2	81.6	101.2	75.0	79.9	84.5	67.7	67.4	65.3	69.6
74.9	102.9	82.8	78.8	75.8	102.4	65.4	65.2	83.1	67.2					
80.7	77.0	81.8	81.6	76.3	80.3	221.5	71.4	102.1	92.7					

Table B.5: Simulated 'Onion'-Shaped Loss Data for Dimension 6

72.5	59.3	75.6	264.0	77.0	100.3	369.3	74.0	79.6	84.3	73.6	85.7
65.6	65.6	145.3	117.9	107.1	70.9	61.8	63.3	105.9	70.2	114.5	71.6
72.7	69.8	59.9	64.8	67.2	63.2	65.8	88.2	71.9	83.3	69.4	78.3
87.1	60.6	69.8	65.6	85.0	98.5	94.7	67.6	71.6	65.8	69.3	68.6

Continued on next page

Table B.5 – *Continued from previous page*

78.4	58.2	94.2	75.1	102.5	59.1	71.8	59.7	60.9	75.5	63.0	119.7
74.0	70.2	73.8	182.9	63.0	88.3	85.2	60.8	75.9	67.0	95.8	77.4
75.2	66.7	68.8	80.4	64.5	61.6	65.3	120.0	67.2	67.5	62.7	70.2
72.9	73.2	62.0	57.2	113.3	67.2	145.9	101.6	71.4	69.3	72.3	144.4
85.0	66.0	77.3	88.4	66.0	62.4	82.9	84.6	68.4	178.7	68.0	72.5
68.6	73.3	74.6	62.4	80.3	64.4	107.5	133.2	105.7	110.1	108.7	131.1
85.5	61.6	134.7	67.9	70.3	76.2	65.1	70.1	63.6	139.6	65.7	65.5
77.5	68.5	69.0	75.4	78.0	70.2	73.6	70.0	74.2	76.2	82.2	131.2
62.1	66.6	62.5	65.2	83.8	68.3	70.9	61.4	63.0	65.9	69.5	52.0
102.3	169.7	67.8	68.4	62.6	80.2	57.8	64.6	83.2	64.9	50.4	59.9
63.2	67.5	69.1	63.6	77.7	65.9	168.7	81.4	122.8	75.0	65.9	71.0
76.6	79.6	98.3	128.4	81.5	76.9	80.0	86.1	111.5	84.3	88.9	78.2
77.7	87.0	74.4	136.5	72.2	67.4	152.8	68.7	104.5	104.6	69.6	63.7
65.9	62.3	68.6	51.4	63.2	74.8	85.0	104.0	80.1	82.5	86.4	108.0
64.6	57.2	74.2	82.6	76.5	74.9	70.5	68.4	139.8	104.6	84.1	72.8
60.6	69.0	75.8	68.5	94.8	112.6	74.1	60.1	216.3	63.4	64.4	80.7
112.1	106.9	141.0	110.2	106.3	106.6	64.2	104.1	61.9	66.9	68.6	75.9
210.8	63.6	65.2	62.3	66.1	85.7	63.7	62.0	66.7	65.0	58.2	63.9
74.7	59.0	56.5	65.9	73.0	61.7	73.1	70.4	65.7	71.8	65.5	75.7
63.1	155.2	56.1	106.6	65.2	98.3	66.0	59.4	62.3	86.9	56.6	60.8
160.7	64.2	76.6	103.4	64.3	107.1	63.0	58.8	66.3	69.2	64.6	78.3
74.7	70.2	64.9	89.6	139.3	68.0	115.9	61.5	76.4	81.4	59.0	69.2
79.9	79.3	77.7	75.8	83.2	68.9	66.5	87.2	68.1	65.6	131.2	94.3
75.4	75.1	64.9	429.4	64.4	71.1	65.7	54.7	68.8	62.4	70.7	93.2
59.9	58.6	67.4	134.8	64.3	65.6	97.1	62.5	66.0	62.5	67.7	63.4
80.4	67.0	100.5	74.1	111.3	67.1	76.0	65.7	69.0	85.1	62.9	79.1
60.6	55.2	69.0	77.6	199.9	68.6	149.3	78.0	89.5	61.1	162.8	62.9
72.6	64.1	66.4	79.8	257.4	77.7	68.8	89.7	71.3	75.5	68.2	99.4
83.0	381.3	72.6	74.2	59.0	72.4	98.0	98.1	77.2	80.9	164.0	81.5
88.0	123.0	60.9	63.1	115.1	61.0	73.4	64.3	70.8	67.0	65.3	63.0
59.3	65.4	76.5	66.4	61.8	125.8	63.3	65.5	66.7	63.5	60.9	69.2
84.9	216.6	107.4	74.3	75.1	131.7	99.0	86.1	102.0	68.0	78.7	67.6
83.5	71.0	70.7	80.7	80.0	78.0	93.1	72.8	69.6	66.6	71.7	65.7
81.4	67.8	60.3	60.0	62.8	63.9	94.6	80.7	102.9	88.7	75.1	73.8
137.4	70.6	96.0	62.2	71.3	83.0	68.3	76.4	77.8	252.9	76.8	64.8
136.9	73.4	61.8	63.6	60.6	65.7	64.3	64.5	58.9	74.2	97.6	104.1
68.8	71.7	77.5	68.5	66.8	69.2	87.4	67.0	57.3	110.3	64.0	83.9
168.4	95.5	72.0	60.8	67.7	62.9	78.5	66.9	66.2	75.3	69.5	82.1
61.6	58.8	67.4	71.5	140.1	53.1	64.7	66.8	79.0	69.1	72.4	63.3
64.2	72.0	84.4	66.0	112.1	71.3	197.5	66.6	82.9	66.5	76.0	58.6
272.7	93.3	63.2	81.2	77.0	130.2	63.0	73.4	61.6	68.5	160.2	70.1
70.7	65.4	66.8	77.5	64.6	74.8	69.7	64.6	70.7	62.6	93.2	69.7
64.4	65.0	116.7	75.9	86.2	65.3	71.6	76.6	100.6	63.1	109.2	61.3
80.6	61.9	62.2	87.6	68.2	69.6	158.3	130.7	72.8	59.6	111.3	58.6
77.5	81.9	67.1	74.0	67.8	64.3	130.2	66.1	72.2	55.2	67.3	70.1
82.6	72.5	74.8	76.2	69.2	79.2	69.2	125.9	180.8	68.7	64.7	65.3
64.5	95.1	59.4	137.3	113.3	63.9	73.1	75.6	82.7	66.1	73.4	76.0
56.0	65.8	78.0	62.3	63.4	74.5	80.1	88.6	82.5	75.0	98.7	88.8
69.9	70.7	112.6	71.1	68.3	64.1	61.0	67.9	65.8	60.9	61.7	65.7
204.7	77.0	81.4	71.2	216.0	79.2	63.0	71.8	65.6	56.4	66.8	66.0
69.7	71.3	65.9	67.6	69.5	82.5	301.5	73.8	65.5	63.5	134.2	95.6
59.3	68.3	79.0	110.9	66.9	63.2	70.7	92.3	68.9	64.4	68.8	75.0
71.1	68.2	69.5	303.2	61.7	227.6	66.1	77.1	66.4	75.1	72.4	68.8
65.6	58.6	67.9	65.6	79.8	64.7	69.7	59.2	99.0	62.6	70.8	115.1
70.7	62.0	74.1	127.4	59.3	87.0	98.9	101.2	92.6	94.7	95.4	96.7
59.3	68.7	58.7	56.7	66.7	60.2	91.4	70.8	178.9	68.0	62.7	71.2
71.8	81.9	64.9	55.8	72.6	76.8	81.3	150.3	65.4	88.1	68.9	68.0
76.6	73.5	84.8	77.4	73.7	80.5	154.0	120.7	166.9	76.5	83.7	88.8
65.6	67.9	82.3	62.6	58.0	59.2	65.9	67.7	63.8	77.9	66.3	68.3
76.7	64.8	61.4	67.0	124.4	105.4	152.0	68.6	103.5	65.1	68.5	61.8
60.5	65.6	64.8	54.1	69.9	93.7	66.1	63.8	72.7	63.1	127.3	66.5
61.4	117.2	82.7	67.9	61.4	111.4	196.4	58.4	74.0	70.2	80.2	78.7
64.5	62.2	75.7	100.1	59.7	110.1	54.3	87.7	60.0	61.4	65.8	69.2
65.4	64.8	291.9	65.3	74.8	62.4	74.0	66.7	65.2	75.5	59.2	79.2
85.3	63.1	69.5	76.4	66.6	83.1	61.5	112.9	120.1	62.0	61.8	66.3
73.7	79.6	61.1	553.1	96.8	62.3	74.6	87.3	76.1	74.6	106.1	72.1
67.3	65.2	80.1	62.2	105.0	89.0	73.4	63.4	65.5	65.9	76.4	75.9
77.0	78.2	94.6	95.9	83.9	78.1	89.3	62.9	63.7	72.0	82.3	63.2
64.0	73.2	63.3	83.2	216.2	59.2	68.5	67.4	79.4	70.0	78.1	90.3
101.2	125.3	69.8	67.4	73.7	73.4	68.8	79.3	85.3	74.9	58.0	66.8
62.9	100.9	68.8	159.6	62.8	62.7	106.1	92.6	109.6	96.2	97.8	91.7
71.7	73.5	82.6	63.8	79.0	83.5	70.0	81.0	70.3	72.2	71.8	66.9
65.5	62.7	62.8	71.5	69.5	59.9	69.0	62.6	70.4	67.8	64.2	62.3
58.5	61.7	61.3	186.0	65.6	75.8	64.5	69.9	68.7	63.5	57.8	62.0
121.2	96.7	74.0	450.0	113.2	71.3	98.7	87.0	69.1	70.9	69.3	81.4
189.2	67.2	82.8	82.8	64.9	68.6	66.1	75.7	106.7	70.2	92.1	57.6
73.8	123.5	69.3	82.5	114.6	49.1	38.5	77.3	66.5	72.4	100.2	66.5
66.6	86.5	66.8	296.8	96.6	64.1	72.3	85.0	67.8	74.3	72.9	64.5
66.0	130.2	62.8	67.4	60.7	72.5	129.0	138.9	131.4	140.7	127.0	135.8
73.2	68.0	72.9	67.8	158.1	159.6	66.8	79.6	76.5	94.9	90.9	59.3
92.8	262.4	66.5	57.8	90.8	66.8	64.4	68.6	66.3	63.9	196.5	64.5
65.6	63.2	68.9	73.4	93.6	64.5	68.7	68.7	101.2	182.1	59.9	69.8
68.3	60.9	118.4	72.3	72.6	85.9	63.4	73.1	72.8	53.8	69.8	97.9
66.7	75.9	72.7	60.1	62.9	70.1	104.5	92.5	105.2	77.5	85.6	77.6
196.6	71.9	57.3	84.3	73.2	121.3	70.2	111.3	111.1	86.2	63.4	160.9
92.0	88.4	68.3	301.3	74.3	64.8	56.9	89.9	72.6	64.5	56.4	66.0
73.9	72.6	67.7	62.1	61.6	103.5	64.6	60.2	62.8	61.6	61.0	85.6
99.4	89.2	95.1	85.3	77.1	99.2	64.9	90.1	62.2	57.1	63.6	61.8
63.1	67.2	64.6	67.8	60.6	92.1	67.3	98.6	286.2	62.8	67.1	64.5
70.0	63.8	75.2	60.6	65.2	72.0	68.5	149.8	63.0	56.9	64.3	62.8

Continued on next page

Table B.5 – *Continued from previous page*

74.1	73.4	77.5	70.7	78.1	93.2	68.1	231.6	115.8	73.5	95.6	70.2
64.6	64.1	67.5	87.9	118.8	66.5	65.5	77.4	62.1	70.8	61.4	64.3
59.2	96.6	63.2	65.6	129.7	64.9	74.3	59.6	77.5	60.5	61.1	60.3
220.0	76.8	63.7	60.8	61.8	68.3	62.6	72.0	65.6	84.7	73.0	93.8
101.4	98.8	237.7	93.2	103.8	185.3	97.8	66.0	59.6	59.4	62.2	69.5
80.8	61.7	62.8	4003.8	121.9	59.9	57.8	65.8	63.9	70.5	75.8	64.6
82.0	81.3	62.3	76.9	60.0	64.8	66.1	65.4	48.1	65.5	64.0	63.4
61.6	60.7	70.6	60.3	61.8	71.8	66.1	513.9	80.8	85.2	77.7	53.5
87.6	77.7	76.7	68.2	71.9	66.5	179.0	198.2	126.2	134.3	131.4	132.4
58.1	71.0	73.9	97.6	104.9	69.5	71.7	94.8	62.6	72.3	149.3	73.9
126.7	64.4	72.8	61.9	67.6	84.4	69.9	68.2	65.1	82.2	79.1	69.8
81.8	59.1	62.1	66.4	58.1	59.5	61.3	75.8	66.8	69.8	71.8	68.5
66.7	77.7	64.1	67.4	68.2	111.7	59.6	66.6	67.4	90.2	81.8	81.1
63.4	1429.5	95.3	67.7	82.9	65.6	51.3	63.4	83.2	60.5	65.5	60.1
98.7	82.7	118.2	71.0	70.6	76.4	50.9	65.4	57.6	57.8	89.5	62.3
74.1	77.6	103.2	65.6	63.8	86.8	66.0	56.3	59.4	59.7	124.3	268.2
67.1	147.3	62.2	75.8	66.7	67.9	63.1	61.4	63.3	137.8	69.4	65.7
67.8	89.4	88.7	163.3	85.8	74.5	93.3	63.4	66.4	90.3	64.8	74.2
149.1	543.9	64.6	63.5	61.6	71.1	61.3	105.0	69.6	69.5	76.0	64.5
70.8	67.7	79.6	57.4	115.2	55.3	66.7	80.4	58.8	72.1	67.7	67.7
74.3	83.7	80.5	66.6	113.4	72.8	65.8	130.1	70.1	83.0	140.2	66.6
101.9	66.8	78.0	67.5	64.8	89.8	106.6	115.7	98.9	101.9	97.4	177.9
84.1	86.3	87.7	80.5	81.1	79.9	112.6	102.5	106.7	444.1	100.8	110.9
98.3	164.2	64.7	167.1	61.3	65.6	76.3	112.6	227.7	62.0	78.4	2539.1
60.2	67.9	218.7	66.7	65.7	77.1	61.9	65.2	84.6	70.6	59.1	92.0
63.7	76.5	76.2	66.6	65.7	74.2	190.1	108.6	101.4	113.5	134.1	105.4
76.5	82.0	93.1	76.3	82.4	79.2	72.5	228.1	70.4	63.2	74.3	73.1
107.2	77.1	68.7	72.9	65.4	81.6	77.8	77.1	136.5	134.4	64.6	81.3
66.1	127.9	76.8	80.3	66.9	70.4	100.4	95.3	66.8	137.6	66.9	62.6
55.5	69.3	68.8	68.1	77.0	60.4	76.7	74.6	75.5	82.4	82.5	84.6
87.8	90.2	63.3	79.8	62.7	129.7	113.3	61.3	67.7	70.9	64.6	81.6
110.6	67.1	70.1	96.4	64.8	59.3	159.8	84.8	74.8	93.1	65.5	71.0
74.5	63.4	64.6	83.3	318.1	69.2	275.2	68.4	67.5	67.9	154.6	75.9
82.4	68.2	112.5	63.7	70.0	75.6	80.6	67.5	94.8	156.7	61.1	72.5
63.4	77.5	70.0	68.8	66.4	67.6	313.9	77.8	80.9	62.6	63.3	72.0
121.1	104.8	133.8	124.5	105.1	106.6	67.3	64.8	105.8	61.7	69.8	196.4
57.6	59.8	545.5	68.0	65.2	64.7	150.7	64.3	75.2	70.5	59.8	75.2
69.9	70.4	61.3	74.7	67.5	81.0	67.7	66.5	66.7	72.5	68.2	125.6
75.6	75.2	133.2	68.2	64.3	113.5	66.4	65.4	86.9	71.3	80.9	71.2
66.6	90.0	76.7	78.8	72.5	75.1	87.6	63.0	65.3	196.7	56.8	60.6
64.1	67.3	62.0	67.7	242.4	74.9	81.4	70.8	66.3	86.8	63.9	67.1
64.2	65.3	63.3	64.9	76.2	62.9	88.7	89.0	81.9	76.6	130.6	240.8
100.8	69.7	118.1	62.3	66.1	78.1	89.1	65.0	64.6	74.1	64.8	61.5
74.8	60.8	76.0	66.6	69.6	80.6	76.3	63.0	253.8	63.4	89.1	62.3
99.0	87.0	65.3	60.5	65.0	67.6	115.0	116.2	523.2	122.4	113.4	122.7
146.8	60.9	64.5	70.8	73.4	70.7	59.7	116.7	68.3	62.3	58.7	59.5
56.0	64.5	77.7	62.3	58.7	200.5	77.3	63.1	77.2	75.8	82.0	121.4
92.7	101.1	91.9	96.5	172.1	95.3	82.7	56.8	63.8	68.7	65.7	182.5
64.4	96.7	77.5	62.3	74.5	76.1	68.2	72.1	73.0	67.8	89.0	75.1
64.8	75.9	61.5	69.1	62.9	67.2	91.7	68.7	93.3	62.1	104.8	65.4
66.8	117.3	62.2	67.9	87.6	61.8	58.9	71.3	69.6	73.0	70.4	74.6
64.4	79.8	57.9	164.6	86.6	61.4	75.8	76.1	103.3	65.0	59.9	65.5
72.1	78.1	68.9	72.9	83.8	68.0	86.4	70.3	75.7	68.8	75.2	71.2
87.9	84.5	92.2	67.4	88.0	67.5	61.3	99.2	93.6	65.9	59.1	70.4
88.6	138.2	69.0	76.3	56.7	59.0	62.7	81.3	59.4	120.3	58.3	67.6
70.0	104.0	65.5	69.8	62.9	67.4	68.0	70.1	70.4	66.9	79.6	347.2
69.3	115.4	75.0	131.3	93.0	71.3	99.5	106.6	108.1	123.1	156.4	115.1
68.8	71.9	83.5	92.0	96.4	92.6	58.3	71.1	80.6	59.1	60.6	56.9
76.2	64.4	67.9	73.6	63.6	62.9	71.1	71.4	67.4	67.0	67.7	136.4
72.1	73.2	69.0	77.6	65.4	62.3	140.9	71.8	87.9	83.7	63.7	58.1
95.7	89.7	86.2	105.0	81.7	96.0	68.8	63.6	63.0	65.0	129.6	69.2
258.5	77.9	58.3	74.6	75.5	61.5	173.4	64.8	184.1	222.8	84.6	69.0
96.8	68.8	68.7	75.7	87.8	67.7	67.8	145.7	58.8	64.7	61.1	72.6
64.3	65.7	66.5	72.6	66.8	54.9	70.6	65.0	68.9	75.9	196.4	64.4
66.5	69.2	64.8	83.4	65.6	123.9	76.0	150.8	71.5	76.8	74.8	111.3
72.0	103.8	84.3	375.4	83.9	67.5	86.4	69.8	70.3	72.7	62.0	78.5
63.3	75.4	77.6	65.4	144.9	88.2	68.8	60.3	75.3	65.5	68.0	61.6
94.1	59.1	68.4	60.9	177.3	70.3	85.2	62.6	112.4	60.5	80.1	64.3
62.9	81.3	61.4	71.4	73.6	87.6	70.5	63.5	62.7	79.3	64.6	103.0
124.9	98.4	62.8	58.5	78.1	69.2	57.0	385.8	141.6	67.1	73.6	71.3
65.9	63.3	73.1	165.1	73.3	75.3	65.2	61.6	73.1	72.2	64.8	177.9
69.0	67.5	65.0	68.6	68.6	64.7	76.5	230.7	65.9	85.7	90.0	64.8
103.5	117.0	65.3	59.5	67.6	171.0	84.1	103.4	73.2	150.9	73.6	63.0
62.0	168.6	91.7	78.5	66.4	62.4	64.8	75.3	65.8	89.9	104.1	72.9
133.4	64.2	60.0	136.2	47.1	81.6	66.4	80.3	75.4	72.4	69.8	64.2
61.3	62.4	66.3	82.7	75.5	64.2	121.7	74.3	79.2	65.4	71.4	67.0
70.6	77.1	61.7	81.7	56.7	134.9	74.3	201.2	92.4	88.3	107.1	82.5
66.2	72.4	65.3	67.3	56.7	60.3	79.7	68.5	67.0	63.8	76.6	68.9
75.0	65.2	75.6	65.9	60.0	69.9	64.6	69.4	72.7	64.7	85.7	63.1
76.8	69.2	59.2	60.2	68.2	70.8	68.4	683.9	71.8	81.3	63.6	99.8
108.3	65.1	82.6	75.4	90.9	68.1	82.7	71.4	79.5	78.2	77.9	75.1
60.6	90.8	134.6	87.3	62.7	63.8	64.6	77.3	54.6	66.9	68.0	57.4
71.9	69.6	66.2	95.9	68.0	88.3	75.4	70.0	77.4	82.7	391.3	66.4
65.3	66.8	66.8	64.6	65.0	64.1	70.4	76.1	67.7	66.5	71.7	78.9
55.3	69.7	66.3	58.5	60.7	64.7	56.2	63.7	76.5	70.7	75.6	69.2
64.1	96.2	77.4	61.7	54.5	63.2	76.0	87.4	105.0	55.2	66.5	59.5
86.5	72.8	72.5	117.7	82.5	67.8	180.6	185.1	218.9	192.7	189.6	178.5
63.7	62.6	66.8	80.4	66.5	74.9	72.1	72.8	78.9	73.7	73.4	75.1
54.2	64.2	77.3	62.8	73.6	68.6	79.9	71.8	64.9	112.5	63.6	65.6
66.6	78.0	68.3	70.6	99.2	64.0	72.1	57.3	87.9	83.7	103.5	112.3

Continued on next page

Table B.5 – *Continued from previous page*

64.8	72.3	74.5	196.8	66.3	90.9	88.5	72.6	73.9	77.8	78.6	70.1
68.7	120.7	63.5	63.5	117.9	62.6	71.0	431.5	66.8	76.3	75.0	72.1
58.3	100.4	62.8	65.6	62.0	71.9	94.3	73.7	71.8	67.7	84.1	80.5
71.6	74.7	89.7	71.4	82.2	78.7	93.2	122.7	88.4	63.6	66.5	66.2
90.7	56.3	99.1	90.4	59.4	84.2	240.1	71.3	74.4	66.7	69.6	69.0
62.0	61.1	59.2	68.6	81.2	82.8	66.9	68.3	73.7	69.6	62.2	62.4
58.1	62.2	65.7	64.1	64.8	61.3	67.2	69.5	84.0	72.9	67.1	95.6
73.4	77.7	95.0	86.0	173.6	79.3	90.6	69.0	71.9	138.9	99.8	72.7
60.4	72.2	61.5	87.2	63.0	69.3	60.4	64.0	137.4	68.5	58.7	64.3
109.0	174.3	123.0	117.1	175.0	137.0	115.5	65.7	65.9	92.7	102.4	62.6
81.6	67.4	63.5	76.4	59.7	64.8	63.5	148.1	151.2	73.0	64.1	156.8
66.2	64.2	69.2	70.4	132.9	441.5	118.0	58.6	62.8	64.2	96.5	85.6
70.0	69.5	70.4	90.4	130.6	64.4	62.4	64.8	63.3	73.5	65.2	67.6
58.6	69.5	60.0	69.1	67.1	71.8	62.5	143.6	69.8	79.1	83.6	72.6
68.0	56.7	76.5	64.0	61.0	71.4	110.2	101.6	102.8	65.9	65.7	131.8
54.0	61.0	66.4	82.0	69.6	59.2	70.3	88.2	84.8	61.0	60.6	60.8
63.1	82.6	65.9	56.2	103.9	76.9	61.3	299.8	102.0	59.2	67.4	79.8
78.5	59.9	121.6	63.8	80.4	98.5	77.5	72.1	71.9	75.2	75.4	476.4
115.7	59.8	62.0	64.7	52.5	60.6	83.0	68.3	68.3	82.4	73.0	64.3
70.6	76.8	74.0	73.7	92.0	63.6	68.6	86.8	79.5	85.1	58.5	101.7
72.3	74.9	78.5	65.2	61.9	69.9	71.8	69.3	65.2	62.0	65.2	122.2
102.4	166.7	66.2	81.8	71.1	70.4	75.4	154.5	69.8	1107.9	70.6	80.3
75.6	99.3	70.2	143.7	76.4	78.5	63.6	82.1	69.4	112.3	66.4	117.7
240.4	249.7	267.6	254.6	243.7	258.2	60.8	65.2	137.6	89.7	68.4	59.1
829.2	76.4	76.2	108.9	79.1	74.0	83.3	55.7	72.3	61.6	62.3	66.8
73.2	122.2	66.8	63.6	83.9	82.5	72.2	64.0	60.6	100.9	59.0	69.9
63.0	60.6	62.9	75.1	64.5	63.2	52.3	60.2	64.9	66.7	66.0	75.2
95.1	110.9	91.7	95.6	113.4	106.0	113.5	68.2	89.5	63.1	67.8	70.2
98.0	63.5	54.9	92.1	61.8	175.5	73.4	84.0	67.3	82.1	79.1	66.2
80.4	66.5	62.6	98.3	99.2	80.7	99.9	62.1	86.2	75.3	63.5	71.8
57.8	59.7	75.0	109.1	70.0	68.4	59.7	64.1	62.2	66.3	85.7	237.9
74.2	131.4	71.9	67.8	64.6	74.8	64.1	353.4	67.9	63.8	72.8	194.9
70.9	66.2	67.0	63.6	49.7	67.6	68.6	72.5	88.1	66.8	66.8	92.4
74.6	68.9	79.8	71.2	72.5	84.9	65.9	71.2	63.0	64.7	68.5	74.1
67.5	199.0	62.8	75.8	68.9	165.4	64.1	82.0	65.5	62.6	93.1	81.8
73.0	66.6	72.1	73.0	65.7	105.2	61.8	68.5	70.6	75.6	81.2	60.4
61.0	73.1	65.5	57.8	61.3	76.7	60.9	213.5	64.1	63.2	67.4	85.2
93.1	67.6	65.1	61.3	66.9	61.4	62.0	64.1	84.2	68.7	63.8	64.2
87.0	71.4	63.3	70.1	73.4	66.7	60.7	63.2	62.4	69.7	269.5	66.1
60.3	92.6	63.3	58.5	60.1	62.3	85.7	91.4	92.4	105.9	80.5	80.8
66.8	81.3	94.2	70.2	100.9	83.8	582.5	61.8	63.9	199.8	60.4	62.3
65.4	63.3	62.4	61.6	78.2	81.4	61.1	80.6	66.3	74.5	98.8	61.3
85.6	71.3	111.1	68.1	96.4	78.8	64.1	62.8	60.9	65.1	62.8	73.4
57.1	76.5	73.9	83.7	59.7	67.0	68.1	89.9	122.9	79.2	72.0	78.8
85.7	72.6	89.2	124.3	86.3	76.9	81.7	92.4	76.1	65.9	61.4	66.9
63.5	62.3	65.9	73.5	66.1	81.3	94.9	96.7	78.7	72.1	79.6	79.2
74.5	81.4	70.9	77.6	86.0	75.8	67.6	64.2	82.3	60.5	57.7	60.0
82.2	70.2	58.5	54.4	63.9	94.0	64.3	66.7	71.3	65.4	61.1	81.1
76.5	60.9	60.5	63.3	59.9	65.8	67.6	61.7	63.1	58.0	66.0	78.3
74.6	67.5	165.7	76.9	63.1	127.8	64.8	60.2	80.5	68.8	283.5	70.4
54.3	80.5	67.1	78.1	61.2	103.0	66.2	63.4	69.5	66.6	79.5	61.1
82.8	90.3	92.5	82.1	88.1	103.5	73.1	75.3	97.2	75.6	71.2	70.8
66.1	92.7	84.8	71.9	223.0	73.1	65.3	62.5	86.7	110.7	64.0	62.8
69.1	77.0	63.1	298.2	71.4	66.4	109.7	66.7	68.0	75.6	57.0	71.1
61.5	61.8	70.3	103.5	93.8	72.9	78.3	98.0	85.4	81.9	97.8	79.3
66.8	63.6	72.5	182.1	74.0	65.5	88.4	145.4	68.6	134.6	66.8	73.2
76.0	59.8	58.5	61.9	72.9	109.6	62.9	67.5	172.9	64.4	77.2	73.2
100.1	59.6	80.8	66.7	79.2	99.5	62.7	72.2	70.7	79.8	65.1	92.0
65.0	82.8	68.6	85.6	64.3	63.9	63.5	63.6	106.8	58.4	67.7	63.0
71.2	71.1	76.6	114.9	69.4	80.1	68.5	65.8	67.8	75.8	371.9	69.3
72.8	74.6	88.6	75.9	71.2	201.7	170.5	88.1	86.4	103.3	82.6	102.7
120.2	70.5	56.2	66.2	60.0	65.1	81.9	85.1	93.2	72.8	103.2	114.0
63.1	84.9	65.9	57.0	68.2	74.5	66.2	67.0	75.2	64.0	63.2	73.1
62.6	57.8	67.5	112.3	96.3	61.5	70.4	63.9	71.3	65.5	68.3	157.2
61.3	63.7	63.0	82.5	63.7	212.7	86.5	76.1	72.7	75.3	68.2	87.1
107.6	74.8	58.1	76.0	70.7	69.5	83.3	83.9	76.4	61.7	72.0	61.2
69.5	73.6	66.9	61.9	74.7	88.8	71.4	76.1	74.2	73.1	61.3	73.1
68.2	115.3	74.7	67.4	63.6	75.0	68.6	71.7	67.1	61.3	65.9	196.1
106.7	70.4	83.2	66.8	87.2	66.5	100.0	66.7	328.1	86.4	76.0	97.4
70.1	68.3	65.1	93.4	64.4	69.2	77.7	72.1	64.0	62.8	80.8	73.2
146.1	55.9	67.5	69.6	62.9	84.0	76.6	71.0	67.5	61.8	64.3	61.9
74.2	69.6	145.6	81.9	68.1	69.3	64.8	259.7	78.5	261.9	107.2	64.0
55.8	63.4	86.4	75.3	65.0	80.0	77.3	72.5	82.6	75.8	64.6	69.7
134.6	151.5	91.5	63.6	65.3	313.9	65.1	65.5	52.5	72.4	59.6	76.3
63.1	74.6	66.5	115.5	83.2	68.0	72.5	79.7	79.6	70.9	65.8	79.0
62.8	64.2	69.0	57.0	64.3	63.2	50.9	57.5	58.4	71.4	70.7	91.2
62.6	181.8	184.8	61.9	74.0	67.0	74.5	63.7	68.5	65.1	64.1	61.8
105.9	83.7	121.9	70.6	80.7	87.2	64.6	65.0	62.9	68.6	63.5	65.2
75.5	72.4	69.1	120.4	77.8	224.7	79.1	96.0	75.4	72.3	65.8	91.3
61.3	67.2	99.7	71.1	60.0	64.1	84.6	70.4	62.6	218.2	93.6	69.9
64.0	62.2	62.1	76.2	152.7	65.5	73.6	81.2	75.6	56.5	66.1	110.7
64.4	62.6	60.7	62.8	64.4	64.3	84.4	91.7	78.4	77.7	96.1	86.2
65.9	57.8	63.5	80.4	61.0	91.6	80.9	64.4	67.8	71.8	69.1	63.7
66.4	132.4	79.2	88.3	78.7	55.0	69.4	77.3	70.0	67.4	63.1	63.7
1954.6	1945.9	1948.1	1950.2	1945.5	1947.7	66.3	63.8	99.1	76.9	64.2	75.9
60.7	69.6	65.1	69.9	88.9	85.9	72.7	66.6	67.6	68.5	252.0	149.5
70.0	61.8	65.1	61.1	62.5	85.7	75.6	76.5	99.1	70.6	54.4	126.5
70.1	64.0	66.4	62.5	97.1	91.6	62.3	83.0	67.7	85.2	116.6	64.5
78.9	67.8	59.8	66.9	118.6	84.1	80.5	65.6	69.8	66.9	60.2	110.3
61.1	68.1	56.4	73.0	64.3	63.2	78.4	71.9	87.8	78.6	69.5	75.2

Continued on next page

Table B.5 – *Continued from previous page*

78.2	65.0	74.1	67.4	69.2	63.4	64.0	109.0	76.8	68.5	67.3	68.4
125.0	103.5	80.2	71.4	66.1	250.4	87.5	97.1	96.9	101.1	90.5	106.0
62.8	71.2	57.5	105.6	68.6	81.3	119.8	64.6	64.5	64.2	100.6	111.7
67.5	71.4	65.2	80.5	62.6	60.4	75.4	81.1	60.9	85.5	69.2	66.2
63.9	125.3	185.0	63.9	63.9	182.8	68.3	126.7	71.6	92.4	84.7	70.6
80.9	117.3	73.4	80.1	69.5	73.4	75.4	61.9	65.6	67.1	79.9	67.2
81.2	88.5	570.2	74.8	75.9	123.3	382.3	402.3	179.3	130.7	130.8	164.6
74.1	64.3	118.4	78.6	66.1	105.8	72.2	69.5	66.6	76.5	77.4	78.8
65.4	63.5	62.5	61.8	66.9	60.8	72.4	67.9	73.7	104.4	85.1	72.8
73.7	69.8	135.1	342.4	75.0	80.0	52.0	58.1	59.0	61.6	63.4	65.2
68.5	52.9	68.9	65.2	61.0	70.8	68.8	67.2	61.9	64.7	82.0	67.9
63.0	61.1	63.7	61.5	58.9	72.8	83.2	70.1	133.3	66.5	60.5	63.6
66.8	62.4	64.6	88.2	327.0	75.4	103.5	89.6	186.0	86.2	87.2	107.1
60.9	69.5	69.9	84.4	74.2	63.1	87.7	64.0	77.0	70.7	217.2	66.8
61.9	73.5	102.4	87.0	71.6	63.8	62.8	63.4	92.5	71.2	62.0	68.7
67.9	94.4	67.2	199.7	59.6	72.6	72.7	64.7	67.0	79.7	70.8	68.9
63.9	56.4	63.1	51.0	67.6	98.0	65.9	69.2	90.6	61.0	67.8	81.0
80.9	102.8	67.0	140.2	75.8	85.5	60.8	62.7	63.8	65.4	72.6	72.3
109.9	79.0	79.1	71.0	89.4	74.0	86.0	101.6	107.2	72.2	86.4	78.6
70.2	78.3	83.0	59.8	63.0	75.7	138.7	71.6	79.8	66.9	72.0	67.5
80.3	63.6	106.2	145.6	66.6	86.6	121.8	247.8	117.2	116.1	116.6	319.9
71.9	62.2	58.8	65.4	62.3	61.4	74.1	61.3	84.1	64.3	121.3	64.9
62.5	64.9	78.0	65.2	72.5	67.9	106.4	213.1	98.9	66.3	65.7	66.3
73.9	61.9	72.2	72.1	61.6	322.1	67.8	96.5	72.6	116.9	61.5	63.1
66.3	89.4	72.9	100.9	74.8	68.6	73.6	75.1	68.2	63.9	63.9	68.6
67.8	77.7	114.2	65.9	83.9	65.3	76.3	60.1	67.2	64.3	74.9	87.1
77.4	70.5	72.4	74.3	73.3	77.1	74.1	95.0	61.9	74.4	101.3	119.2
83.0	77.0	54.4	61.7	65.6	63.8	72.9	82.1	68.0	71.4	67.8	70.1
68.5	70.4	65.1	64.6	66.3	73.0	69.1	82.4	162.8	70.6	71.8	69.6
62.9	60.6	78.2	59.8	67.4	61.5	68.6	70.4	101.5	62.0	70.4	101.9
333.2	342.3	356.6	333.3	337.1	322.7	124.2	153.6	125.6	128.2	126.6	130.6
66.7	72.0	69.6	158.9	79.3	64.8	62.9	72.4	60.1	66.1	69.0	62.7
71.2	61.3	87.7	75.2	61.8	64.7	391.5	391.3	429.7	391.0	394.8	391.0
182.7	59.6	180.5	63.9	64.4	73.9	85.3	75.7	105.6	71.0	75.3	94.9
72.8	66.2	116.8	163.9	65.5	69.0	86.6	71.4	77.4	105.5	78.1	79.7
80.7	67.1	62.2	70.9	60.7	66.3	72.2	56.2	152.6	93.7	62.3	59.3
71.5	99.6	193.9	73.5	186.0	66.3	246.3	78.2	114.7	70.4	86.0	70.4
63.7	74.0	60.8	64.0	65.6	76.9	79.3	120.8	84.7	79.6	79.9	78.5
74.4	92.1	73.8	70.3	66.1	66.3	68.8	84.1	67.6	62.7	85.9	75.3
81.1	64.4	93.0	64.7	71.5	67.9	70.9	80.2	68.8	94.0	78.0	173.4
75.9	59.2	71.2	97.9	63.7	63.9	95.8	88.5	117.8	83.0	89.6	80.0
67.7	74.0	74.0	59.4	64.9	82.1	75.8	65.0	62.7	66.1	62.4	112.6
72.4	75.2	81.2	71.8	77.0	75.4	78.8	62.3	61.8	86.6	63.3	65.5
65.4	125.1	62.8	64.6	63.9	94.3	62.5	66.5	61.6	76.0	151.6	150.5
63.3	58.9	75.4	143.0	65.6	102.9	69.0	113.1	64.7	64.8	97.3	60.6
60.4	66.7	63.5	62.3	58.6	70.4	65.5	77.3	65.7	69.2	87.1	99.0
303.4	57.5	62.1	66.0	78.5	77.6	58.4	68.1	68.3	67.2	75.3	63.3
78.1	61.8	71.6	70.5	76.4	63.9	91.8	72.9	90.6	66.7	64.8	75.5
65.3	60.6	63.2	61.7	93.8	79.6	72.7	161.6	114.5	77.1	81.5	87.9
75.1	98.9	66.1	64.0	95.2	60.5	69.8	94.8	75.1	212.1	84.0	72.0
73.8	78.8	79.0	82.4	104.9	75.0	129.5	131.8	131.1	133.9	139.9	127.1
68.9	60.2	68.4	89.2	69.6	66.1	66.4	64.3	73.5	79.3	74.6	77.1
73.9	140.6	87.7	75.6	49.3	86.8	71.1	98.3	70.4	82.3	135.0	273.7
71.0	55.9	61.3	70.9	106.5	107.9	68.4	80.9	82.1	86.6	87.6	78.5
87.2	151.8	63.0	70.5	65.8	62.8	63.1	67.8	59.4	74.9	66.2	68.8
59.2	59.7	78.3	65.2	60.4	77.8	59.3	64.4	96.0	61.6	65.2	181.1
78.9	92.4	72.0	736.4	64.4	66.9	60.3	75.6	76.8	58.8	60.0	95.3
64.0	67.7	58.9	65.1	83.9	58.8	81.2	77.7	86.8	64.8	65.3	70.6
68.4	74.2	81.1	131.6	78.7	79.1	79.5	90.3	102.6	78.2	60.3	73.9
66.6	68.2	66.3	73.9	72.1	74.3	69.9	74.6	100.1	101.3	77.3	126.0
65.8	91.3	66.8	69.0	67.1	94.9	83.0	99.7	70.2	75.9	82.8	71.0
206.5	74.8	90.5	88.2	94.8	75.4	60.0	96.5	62.3	60.1	552.1	78.5
65.4	64.6	77.1	77.2	71.8	63.1	81.4	78.8	80.0	80.7	99.7	79.0
67.6	70.6	63.9	84.4	86.6	68.7	74.1	102.8	80.9	89.4	98.7	79.3
65.3	74.1	67.8	71.0	76.2	92.9	59.5	63.5	64.9	77.3	63.3	71.9
169.2	70.7	88.1	77.1	74.8	68.4	70.4	72.0	79.5	88.9	66.6	67.8
65.8	62.8	99.9	63.6	137.4	62.8	68.7	60.5	254.5	81.8	60.8	89.0
65.2	64.5	91.3	61.3	58.2	64.6	94.9	100.1	69.8	81.0	63.6	62.3
78.2	68.4	81.4	67.9	122.3	67.6	63.1	96.8	78.6	70.4	63.5	102.6
58.6	106.2	75.9	68.6	75.2	75.8	78.1	77.8	68.8	79.2	145.8	115.0
60.8	66.9	86.9	60.2	65.6	69.5	62.8	238.3	75.6	72.3	69.8	63.1
152.5	66.3	83.2	68.0	70.6	76.6	88.4	56.9	73.0	63.6	64.0	123.8
72.0	67.6	72.1	66.6	65.9	89.8	60.4	68.7	60.3	54.2	66.1	81.4
66.4	72.1	73.8	65.7	148.8	71.8	83.2	62.0	70.1	97.9	54.9	106.2
111.9	86.6	132.1	91.4	99.7	98.6	104.3	75.5	83.1	91.0	89.2	77.3
67.0	69.1	65.3	247.8	75.7	66.6	74.2	69.8	62.1	63.1	62.8	72.7
67.8	61.1	74.6	95.0	71.0	79.3	106.1	109.6	104.3	129.0	108.0	114.2
61.0	62.4	61.8	62.2	75.4	60.2	68.7	64.1	78.6	58.8	70.2	71.7
65.8	64.5	61.0	94.3	190.8	65.7	63.3	94.3	58.9	70.8	65.7	65.1
61.2	66.1	71.8	76.5	67.2	74.2	63.2	60.8	64.0	60.9	65.7	69.9
61.4	64.9	70.2	71.9	62.9	67.9	70.9	76.8	75.2	59.9	65.0	63.3
124.4	83.2	75.7	72.6	81.2	139.0	67.0	65.5	93.5	64.4	61.8	72.1
158.0	71.8	72.1	73.1	63.0	80.4	69.5	69.5	70.5	71.2	70.5	67.0
131.2	60.9	67.1	72.8	73.3	57.7	65.8	90.7	114.0	59.2	59.0	67.2
65.4	67.2	102.5	78.2	61.5	62.3	109.6	70.6	70.6	120.8	67.1	62.5
154.4	63.9	67.2	67.5	72.6	66.3	64.0	65.7	118.8	177.6	70.1	59.9
74.0	68.9	76.0	77.6	94.5	78.2	65.6	71.4	77.8	64.5	68.0	76.9
65.9	93.6	78.0	78.5	62.3	90.4	101.6	84.5	87.8	98.4	117.0	88.6
78.1	65.6	74.2	63.6	75.9	62.1	99.5	87.6	93.9	128.7	92.2	99.7
97.1	101.0	68.5	73.1	73.3	85.4	95.2	133.4	80.5	307.6	164.2	65.8

Continued on next page

Table B.5 – *Continued from previous page*

76.1	89.8	62.1	150.2	75.4	68.4	59.6	134.3	70.8	56.6	70.3	60.2
66.5	64.8	69.3	62.4	80.6	60.9	110.3	61.2	62.9	73.7	61.2	125.0
59.4	62.1	60.3	67.7	61.1	76.4	96.5	72.7	67.8	58.7	64.2	89.2
60.9	63.5	64.9	71.5	77.0	94.5	94.3	101.4	92.1	170.3	96.0	89.9
74.9	62.9	72.9	64.8	59.8	55.6	108.2	60.3	73.1	66.0	70.1	65.3
75.6	67.9	73.6	156.6	102.7	70.8	64.2	64.3	63.5	73.8	56.9	77.2
69.5	83.5	68.2	106.4	68.0	64.2	74.4	66.7	62.6	86.7	65.3	67.8
74.5	65.6	103.4	74.7	71.8	71.1	78.7	95.1	60.8	541.9	63.1	67.4
64.7	73.9	76.1	101.8	85.6	101.9	103.8	59.6	71.4	63.4	121.6	62.9
74.5	100.0	64.1	70.0	224.8	74.2	337.5	82.3	60.9	74.3	69.1	93.9
77.4	75.1	74.7	78.0	70.2	75.6	86.6	112.5	86.2	83.7	96.4	88.4
90.2	963.1	91.1	67.3	69.9	73.6	64.6	65.0	136.0	54.6	62.9	71.7
60.1	64.5	59.6	67.7	71.1	72.2	61.6	75.3	65.0	59.0	54.3	57.0
86.1	72.1	68.9	109.7	73.6	63.2	77.7	86.9	167.8	78.4	75.0	82.8
68.2	74.3	100.0	63.6	76.9	67.4	81.1	73.6	65.4	64.3	66.9	77.5
77.9	58.7	71.9	67.5	85.6	66.7	71.2	79.2	85.1	75.7	70.3	97.6
57.4	64.2	65.3	64.3	76.6	93.5	63.5	83.1	62.4	142.5	66.8	60.5
73.3	108.8	63.5	63.4	279.7	97.5	59.0	112.2	68.7	65.7	65.5	76.3
207.0	152.2	176.8	176.5	153.2	167.8	56.0	58.8	84.7	76.8	62.2	59.4
69.8	70.2	61.0	59.8	63.7	66.0	58.7	171.1	63.5	59.4	79.1	63.2
202.0	61.4	66.8	66.1	356.6	58.6	67.6	65.2	91.6	74.9	54.7	62.0
75.2	75.5	76.9	117.5	74.3	74.7	60.0	59.0	72.9	73.6	77.7	61.7
63.6	58.4	61.4	57.2	62.7	76.4	61.9	69.8	72.0	68.5	72.0	69.0
68.6	64.8	70.9	62.9	84.7	87.0	72.8	65.1	59.1	76.9	62.6	76.9
66.8	126.8	88.9	93.1	63.5	64.5	66.5	62.6	78.9	67.6	63.9	113.7
56.1	141.1	81.8	64.1	67.3	61.0	72.3	58.1	99.8	64.2	62.8	120.7
64.7	65.9	61.8	69.2	73.4	158.2	73.5	201.3	79.9	67.7	69.7	73.0
58.5	64.6	72.1	64.0	68.0	76.6	83.7	73.9	75.5	70.2	67.1	75.6
77.1	67.2	61.5	134.3	186.3	71.6	67.1	68.7	67.8	65.2	68.6	69.0
63.4	64.4	100.0	122.5	62.8	80.2	78.2	70.2	76.5	85.9	117.9	111.5
68.9	62.6	199.1	61.1	67.4	61.0	74.6	69.7	66.6	97.0	238.8	67.1
105.0	72.7	70.0	69.6	64.1	74.9	60.9	56.4	57.2	72.4	97.6	71.2
84.2	75.9	59.1	65.1	70.0	997.1	70.2	67.8	61.7	90.3	56.5	59.4
70.3	105.8	66.7	75.5	75.0	158.9	59.4	72.6	73.6	79.5	67.1	60.0
63.4	66.1	78.2	75.0	73.3	82.1	62.2	65.6	79.4	94.7	59.8	89.5
66.1	64.2	63.2	74.0	89.4	70.3	227.6	70.0	86.2	122.4	67.9	87.7
69.6	85.1	73.0	65.4	74.0	72.7	64.5	62.5	70.0	148.0	61.2	73.5
68.4	78.8	68.3	90.0	104.6	60.8	81.6	61.2	69.0	114.3	61.1	63.6
131.8	69.6	71.4	75.0	102.2	68.7	71.5	73.8	66.2	60.7	71.5	61.8
59.4	152.4	69.3	66.4	77.9	120.0	80.9	64.8	56.9	64.6	85.3	99.6
219.5	66.8	63.3	65.3	76.5	68.9	67.7	74.7	100.9	71.5	61.7	59.9
64.1	64.5	59.5	65.3	65.6	70.6	65.2	136.6	66.9	65.9	186.9	119.2
100.1	73.7	116.3	114.6	68.9	69.1	62.3	69.9	63.8	69.9	71.2	104.3
78.3	92.1	80.3	107.3	78.0	65.6	70.7	62.8	63.6	85.8	62.9	58.5
63.4	65.3	82.3	73.5	70.8	63.9	67.0	63.7	72.2	82.9	65.6	201.4
90.9	101.6	63.8	70.9	63.9	82.3	70.0	74.3	111.2	71.7	77.9	106.6
93.0	99.0	92.4	86.4	109.7	92.2	62.9	71.1	68.9	66.4	90.1	97.6
67.2	122.3	67.6	77.4	67.9	73.4	73.9	61.4	106.2	73.4	60.8	69.7
68.3	73.2	69.7	60.8	82.7	76.7	73.1	67.4	82.0	79.5	89.5	72.3
62.9	66.2	78.2	84.0	67.3	71.0	73.3	66.1	192.1	65.6	87.4	67.9
64.0	71.5	72.2	66.5	69.9	69.3	65.9	75.0	68.2	77.9	66.5	80.7
67.7	73.8	89.1	58.4	67.9	70.0	68.6	97.4	66.5	109.4	186.9	81.6
65.3	69.8	71.3	65.6	71.3	113.4	78.8	465.1	62.1	72.8	62.2	64.4
70.3	64.2	94.1	70.2	68.3	67.5	67.8	69.1	73.5	63.3	86.3	62.4
60.0	64.2	61.3	80.5	75.5	64.4	69.7	63.7	83.1	76.7	97.7	72.7
84.0	63.8	64.7	95.7	79.7	59.0	70.7	55.3	53.9	64.0	64.6	63.5
60.3	76.1	95.1	65.8	65.0	61.0	85.0	72.6	72.6	62.4	84.8	88.6
64.1	73.3	61.4	91.4	70.1	76.8	56.0	65.8	80.7	88.6	63.7	63.1
60.6	65.3	63.9	70.9	60.8	66.2	80.4	69.8	72.7	77.7	74.1	581.7
60.2	68.4	60.1	79.9	65.4	62.2	76.1	62.4	71.6	62.8	65.4	76.4
83.0	61.5	59.6	77.6	78.8	66.9	66.4	63.5	55.2	61.0	82.6	87.5
66.7	89.2	62.8	65.6	60.1	63.3	63.5	63.9	73.8	64.9	128.6	66.2
68.7	60.2	70.6	77.7	61.9	80.0	79.5	103.4	76.2	71.7	65.0	67.4
92.9	110.1	98.3	130.3	93.0	99.2	76.7	75.7	66.2	149.0	57.4	51.8
59.4	61.5	145.0	71.8	64.1	60.7	77.3	61.8	147.0	71.6	57.0	62.5
67.5	92.5	72.8	73.6	163.6	59.7	70.5	81.7	76.0	64.3	63.2	111.1
68.1	68.9	133.9	59.5	68.6	88.0	122.8	81.4	73.5	71.3	63.4	61.8
49.6	63.6	68.9	87.2	68.9	71.6	77.3	79.6	94.5	73.2	83.8	58.5
359.6	73.8	68.0	68.5	59.5	88.8	80.2	84.2	86.6	77.5	4789.9	67.9
94.3	64.4	65.5	69.7	89.5	64.3	60.3	77.0	88.9	65.5	89.7	62.4
73.9	61.8	77.2	100.5	71.9	63.6	64.6	64.6	66.6	61.6	60.3	71.0
64.5	62.2	81.0	59.2	66.3	62.6	65.7	69.5	103.6	198.0	63.8	85.7
132.1	67.6	76.6	72.7	85.2	68.9	61.1	64.5	72.8	61.1	83.3	68.6
87.4	192.3	89.1	74.4	117.8	117.5	69.8	62.8	67.3	98.0	65.5	64.9
81.5	63.4	85.1	64.8	57.3	64.4	64.7	63.8	67.2	76.5	61.8	67.3
76.3	71.5	117.6	61.0	65.2	96.6	136.7	125.3	122.7	129.8	143.9	119.1
73.6	64.9	65.3	65.8	67.0	60.0	68.3	73.0	61.4	69.1	56.3	98.0
60.6	78.3	87.4	67.8	62.7	95.7	63.6	64.2	69.9	70.1	54.9	76.5
347.2	68.0	96.8	60.8	71.0	74.6	57.9	78.8	72.6	73.0	61.7	53.3
54.1	72.7	116.9	59.5	60.6	85.7	59.4	56.7	64.1	69.3	63.9	67.8
64.8	78.2	72.0	66.8	74.0	76.4	57.7	77.0	149.2	62.6	64.7	73.3
61.1	83.3	100.0	66.8	62.7	59.3	77.8	117.4	66.5	66.0	62.1	64.7
75.4	83.9	86.6	106.2	76.6	83.9	83.1	72.7	86.7	154.6	72.5	113.8
59.7	92.0	64.1	60.4	67.1	87.1	79.1	82.6	76.3	89.6	86.7	76.4
74.5	68.5	72.3	78.0	62.7	63.9	70.1	83.5	88.7	63.2	77.3	74.6
66.2	72.4	66.6	71.0	88.1	63.9	73.9	96.2	69.4	125.6	64.2	59.1
70.6	74.4	65.6	65.5	68.9	67.1	69.1	62.5	65.0	114.9	70.6	78.5
76.1	78.4	76.3	71.7	79.8	77.0	66.5	63.2	71.8	82.2	81.5	103.3
108.3	105.3	88.3	75.8	83.0	118.8	69.0	87.0	70.4	76.6	64.7	75.1
88.7	64.3	65.5	74.2	107.0	81.0	621.7	612.0	617.3	620.6	611.0	612.3

Continued on next page

Table B.5 – *Continued from previous page*

65.0	61.3	77.0	58.5	64.9	64.7	113.1	258.0	63.9	63.8	85.6	59.5
72.1	81.4	67.7	67.7	88.3	68.2	87.8	64.9	64.5	66.5	67.0	68.1
119.9	103.7	82.7	379.8	79.1	79.6	65.8	68.0	70.6	73.6	73.3	77.1
64.9	83.4	77.9	158.3	101.4	67.1	65.3	93.8	61.4	60.1	63.6	70.7
74.8	79.8	85.5	77.1	72.2	74.8	67.4	218.9	64.9	60.3	60.8	60.5
66.2	63.5	87.0	68.8	86.7	63.5	68.6	68.3	90.3	67.2	65.7	72.1
70.2	64.1	94.9	61.6	69.2	66.0	73.2	96.2	77.4	84.1	120.9	251.6
62.1	62.3	72.0	99.1	59.3	59.3	86.6	65.1	66.2	98.1	66.7	82.0
88.5	64.7	73.8	82.1	94.0	62.8	108.4	65.3	59.2	103.1	79.6	59.8
68.6	90.4	93.4	76.4	77.2	87.5	62.7	67.4	64.5	87.6	66.9	67.3
60.2	85.9	90.0	69.1	62.9	57.9	71.7	66.3	60.7	75.4	71.5	68.5
66.1	73.8	62.8	70.5	60.1	66.7	68.1	68.7	101.2	74.1	177.1	80.0
60.2	129.3	72.7	68.4	64.3	75.0	67.0	70.3	65.2	62.7	62.7	58.6
69.8	61.4	78.2	99.7	76.0	64.7	62.2	74.5	65.4	63.4	66.5	74.9
67.8	131.0	118.5	66.8	67.7	90.5	63.1	71.8	78.8	64.8	64.8	69.5
68.8	72.5	60.5	75.8	66.8	68.2	80.1	73.7	71.6	70.7	75.8	100.6
92.3	70.1	67.2	74.8	60.1	62.2	63.2	61.7	60.9	68.5	82.4	81.6
84.6	71.1	61.2	58.0	66.2	92.5	86.1	99.9	81.4	84.9	124.5	81.6
180.9	63.4	64.9	71.4	109.2	67.5	67.1	69.6	65.5	70.2	70.3	69.6
64.6	86.6	65.2	79.8	73.9	223.8	71.6	66.9	88.6	70.0	64.5	110.0
104.4	147.8	113.8	120.0	117.3	108.2	86.0	86.6	72.3	114.8	72.6	72.1
66.3	68.5	77.3	58.8	92.9	71.4	67.2	63.2	63.4	101.6	91.4	63.8
93.1	81.8	77.4	78.6	74.9	75.1	72.5	74.8	69.4	76.6	62.0	66.7
60.7	59.2	60.7	59.0	67.9	62.5	65.2	61.4	65.0	63.8	70.4	74.6
73.5	91.0	78.5	84.6	78.8	78.9	65.9	84.0	63.2	60.1	53.7	53.6
69.9	92.9	62.0	91.1	59.0	61.7	69.4	175.8	65.4	72.8	66.6	66.6
77.0	75.9	64.3	66.2	68.1	73.2	61.0	63.4	65.5	68.5	78.6	58.9
154.1	123.1	90.0	64.1	64.1	74.7	80.2	80.1	67.0	99.5	83.3	70.0
71.7	74.9	70.9	73.7	72.2	81.6	72.4	64.4	55.6	62.9	65.5	62.8
65.6	64.9	75.5	86.2	95.4	67.1	70.8	61.9	84.9	68.7	69.3	67.1
70.4	62.0	98.5	64.3	298.0	86.5	271.8	59.8	50.2	61.2	70.0	59.8
64.2	65.9	71.9	65.6	64.6	69.9	63.2	86.8	64.4	81.8	63.0	61.4
70.7	75.7	74.7	85.2	69.9	79.9	92.6	67.9	69.9	70.5	125.4	61.6
68.1	61.3	71.2	66.0	83.0	82.0	75.2	64.6	126.2	121.3	64.6	71.2
70.7	66.9	74.3	62.7	59.7	65.8	57.2	90.5	64.1	78.8	103.5	81.4
146.5	69.4	70.3	69.1	62.8	63.1	83.0	74.3	62.3	63.4	64.4	60.6
70.7	124.5	96.5	74.0	74.6	85.9	74.5	72.1	115.3	92.6	63.5	69.3
77.6	65.5	61.1	100.4	66.7	70.1	64.8	66.0	67.7	92.4	76.7	80.5
94.1	69.0	72.4	71.7	65.5	71.9	67.6	63.9	60.9	65.5	72.8	65.2
70.0	126.0	61.8	64.4	83.8	89.0	92.8	62.7	87.6	68.7	91.1	63.0
72.7	64.2	79.5	61.0	64.1	67.9	85.0	77.4	108.4	72.4	76.1	66.9
68.8	223.6	76.8	69.1	61.4	70.2	71.7	131.6	59.6	68.2	74.6	70.2
64.7	46.8	91.0	70.4	154.3	67.0	76.1	66.9	61.7	61.2	62.0	70.2
64.0	94.2	68.2	69.4	89.8	66.9	113.8	123.3	157.1	115.7	127.9	166.0
63.9	72.5	76.6	77.5	68.6	73.8	71.6	91.9	69.1	74.6	75.8	133.8
86.8	63.9	89.2	67.6	749.1	83.7	80.8	85.1	83.4	82.0	81.4	99.1

Table B.6: Simulated 'Onion'-Shaped Loss Data for Dimension 7

73.4	102.5	89.6	99.7	64.2	72.5	68.3	83.2	101.0	82.5	71.1	99.3	65.3	75.3
121.1	158.4	70.3	95.0	69.3	68.6	68.6	126.9	72.1	66.4	74.5	62.9	68.3	72.5
70.4	77.5	66.9	72.0	69.0	72.1	65.5	93.2	65.6	86.8	75.4	83.4	71.6	91.9
62.9	66.5	73.1	78.0	80.0	79.2	63.7	66.1	120.0	64.9	67.3	62.9	82.4	72.9
99.3	113.6	77.3	92.8	83.8	98.5	104.1	69.0	123.4	71.8	71.8	89.3	74.4	72.2
166.5	82.0	57.7	69.6	73.6	67.8	95.6	70.3	95.1	90.7	66.5	69.2	71.4	70.8
72.2	70.3	71.0	88.2	73.1	70.5	75.2	81.6	62.3	89.1	64.9	91.6	71.7	69.4
139.0	145.9	167.1	137.9	141.7	141.0	154.9	79.6	78.9	75.8	85.0	81.0	127.1	83.9
223.3	215.2	208.7	218.0	215.8	216.6	191.6	82.6	108.8	69.7	74.8	88.9	65.3	103.1
71.8	69.3	65.6	66.7	66.4	199.9	62.8	85.5	134.0	71.5	68.1	76.6	64.8	68.3
74.0	66.2	68.9	64.4	61.0	67.3	129.8	72.0	76.0	67.7	132.3	72.5	64.4	69.6
92.6	63.0	77.6	80.9	127.1	65.8	63.5	87.1	97.9	83.8	78.7	66.4	73.9	131.5
116.2	69.4	91.9	83.5	70.6	72.7	75.0	73.0	72.3	72.9	164.2	71.1	65.9	67.6
70.4	75.4	63.4	56.3	63.9	65.7	68.3	67.8	67.8	80.2	67.0	85.7	67.5	79.7
84.5	94.8	144.6	70.6	64.1	65.1	66.2	60.7	61.0	65.3	67.5	65.9	73.3	66.5
72.5	70.7	70.0	82.0	83.7	557.6	97.1	141.0	128.6	546.4	177.4	126.6	131.0	134.4
95.4	67.4	68.9	72.0	66.0	70.6	68.7	90.1	79.7	107.8	81.5	82.8	80.5	86.0
67.9	121.5	248.9	58.3	82.7	63.5	86.2	82.8	66.2	86.1	168.8	65.5	77.3	81.0
78.1	74.6	75.5	75.2	66.9	74.0	87.9	68.1	99.1	71.1	77.2	88.5	69.2	103.1
104.5	110.3	111.5	95.5	100.8	92.0	100.4	119.5	53.1	74.7	68.4	63.0	158.6	65.8
67.0	66.8	85.9	58.9	93.0	73.2	73.3	162.0	153.5	116.2	110.5	113.4	95.8	115.2
70.9	87.8	80.4	78.0	70.7	167.1	69.8	67.0	72.5	65.0	75.7	65.0	73.8	92.3
94.8	65.1	76.1	91.3	121.2	70.6	74.6	94.2	126.7	74.4	97.0	69.9	83.2	70.0
76.0	81.1	71.0	64.5	185.3	68.8	60.2	65.5	72.4	68.8	51.5	79.1	84.8	71.1
66.6	69.5	300.2	68.2	60.5	63.9	77.8	103.5	65.0	61.5	110.3	66.3	59.6	78.4
106.6	68.5	68.8	69.8	67.0	70.8	70.4	66.5	80.2	76.2	81.9	71.2	84.2	71.0
68.9	156.1	77.6	64.2	101.4	86.6	68.6	97.4	80.1	70.3	74.3	69.2	70.5	69.1
89.4	65.5	69.5	69.2	114.6	65.6	98.7	184.4	74.3	63.5	69.9	87.5	63.6	70.6
65.5	70.9	72.9	65.4	63.5	69.2	68.7	68.6	72.4	64.5	74.9	76.6	67.5	75.7
64.4	62.6	67.9	94.5	64.2	108.2	129.8	103.3	68.4	67.4	70.7	85.7	78.9	95.9
93.4	70.1	134.1	72.0	77.2	79.2	77.0	106.2	102.9	112.3	119.3	108.9	228.4	125.8
73.5	78.5	89.8	65.2	79.0	98.3	83.2	94.4	59.1	90.5	55.3	75.5	80.2	72.1
63.6	86.1	79.2	75.2	72.7	68.7	74.7	89.5	85.5	151.4	82.2	128.3	80.8	86.6
76.7	68.4	87.7	105.5	72.7	57.8	76.2	87.1	77.6	68.0	64.2	70.0	113.3	87.8
72.9	76.8	65.8	71.0	66.4	66.7	68.6	76.8	79.6	116.1	79.6	80.2	96.8	111.2

Continued on next page

Table B.6 – *Continued from previous page*

78.4	68.7	76.5	155.8	63.1	79.8	67.2	95.5	65.3	63.2	60.6	62.7	60.9	63.1
68.3	98.9	62.9	76.1	104.0	167.3	79.9	69.2	68.2	70.1	69.3	70.7	84.6	71.6
75.6	65.9	73.8	70.0	78.3	80.1	78.9	78.2	80.1	72.4	85.3	77.1	80.7	152.7
76.1	72.5	70.7	119.9	84.3	76.3	70.3	65.6	66.8	73.3	133.5	211.9	69.2	67.5
69.7	67.0	64.4	91.8	64.7	65.5	62.9	149.3	68.2	71.7	84.0	109.9	114.0	69.3
73.4	81.6	80.9	75.8	71.0	71.4	75.2	63.8	69.4	77.7	66.1	100.3	64.0	82.3
61.5	70.4	79.0	154.7	72.8	74.2	95.3	66.4	89.1	69.4	85.9	68.7	99.1	71.9
64.9	65.4	90.8	61.8	69.7	66.4	58.2	61.0	524.6	74.4	63.7	91.7	61.3	68.9
72.9	68.9	72.1	122.2	359.4	75.2	82.8	78.8	77.3	71.9	153.5	81.2	95.1	76.0
68.4	77.9	71.1	69.1	70.5	70.9	112.0	66.6	64.7	93.1	68.8	65.3	82.5	79.3
76.1	71.5	75.2	106.3	60.9	114.1	86.3	79.8	136.0	89.6	67.1	67.1	78.9	95.6
112.0	70.3	271.3	69.8	77.3	71.9	66.8	116.0	77.5	150.4	63.0	82.2	81.3	234.7
100.0	70.8	59.2	113.6	68.9	61.5	62.3	115.4	65.8	68.4	101.7	63.9	64.5	88.1
79.8	62.5	65.7	87.2	80.0	93.1	91.3	78.5	69.8	71.1	72.1	70.0	76.0	71.3
80.9	120.2	266.0	120.0	67.4	60.8	72.1	224.4	74.2	75.8	67.9	59.8	66.1	99.4
75.2	89.8	83.7	67.1	62.4	64.0	70.7	82.1	83.8	91.9	129.9	85.9	143.9	85.4
84.7	104.7	74.9	70.3	151.1	68.8	68.0	70.5	72.1	67.0	68.5	71.0	70.9	71.5
106.6	314.2	67.7	135.7	74.6	64.2	69.7	70.5	64.1	74.8	77.1	65.6	62.4	118.8
71.9	163.6	79.1	78.8	66.6	60.2	100.1	73.5	165.9	72.2	62.6	72.5	82.4	62.8
82.0	371.1	75.2	622.9	84.4	79.9	82.5	79.4	72.1	77.0	79.0	76.0	79.0	72.5
161.7	69.7	81.3	83.1	147.3	66.6	63.0	278.1	188.7	179.1	181.6	184.4	216.3	179.1
71.0	82.2	129.9	72.7	85.1	124.0	71.5	143.3	85.1	90.5	87.0	86.8	88.3	83.5
87.0	158.6	110.0	84.7	350.2	78.1	95.5	102.1	100.7	83.2	107.4	76.5	66.7	192.6
810.3	302.2	74.0	65.4	74.2	64.4	57.9	66.7	66.5	121.7	71.4	65.1	76.0	77.5
64.7	69.0	113.2	74.0	73.9	76.7	79.7	74.5	72.3	100.3	65.9	88.4	71.8	66.5
143.4	61.5	68.0	67.0	63.6	67.8	63.9	108.6	74.7	94.6	72.4	86.3	80.4	71.3
61.2	70.6	84.6	132.8	66.4	73.0	56.5	73.3	66.0	57.4	75.7	76.1	113.2	61.0
154.5	73.6	85.8	80.1	67.6	65.8	74.2	67.6	64.1	74.1	67.8	65.5	64.8	75.9
73.3	70.4	68.1	294.3	114.3	62.5	64.7	145.5	80.6	81.8	122.8	75.2	71.1	169.5
67.5	76.3	65.5	97.0	68.6	71.8	61.2	75.7	75.9	82.7	76.0	164.7	89.3	94.8
70.0	79.0	78.9	75.6	57.3	78.0	101.8	83.6	182.3	88.1	109.0	87.2	68.6	70.9
73.6	79.5	70.2	72.2	103.2	68.6	76.8	62.0	77.0	76.1	98.1	69.8	72.9	64.9
69.2	65.9	62.7	57.8	72.3	64.2	74.9	59.2	116.7	70.5	80.6	74.9	68.1	67.1
75.3	92.3	75.9	73.3	70.7	70.1	87.5	83.0	78.1	78.9	120.2	80.1	105.2	80.3
68.5	238.4	115.4	130.8	77.5	83.1	75.1	76.1	78.5	82.3	79.8	79.3	75.9	64.3
127.5	103.5	199.4	105.3	108.8	100.5	97.3	69.7	107.4	66.5	72.0	75.6	138.1	68.6
65.8	67.2	80.6	68.0	70.7	74.9	88.2	68.3	71.3	74.4	66.2	63.4	67.7	88.2
73.0	71.5	81.2	84.1	69.5	80.8	55.4	77.0	73.5	78.2	127.6	126.4	88.1	73.2
75.2	68.6	66.2	77.7	101.6	76.2	70.1	66.0	68.2	66.9	63.6	62.4	70.5	63.7
82.0	78.9	70.6	80.9	71.0	61.0	74.0	67.4	68.4	66.9	111.5	67.8	77.9	66.1
65.4	74.5	65.5	63.8	64.2	67.5	71.0	102.4	65.9	64.5	60.4	105.8	62.6	68.0
107.4	78.8	74.1	100.3	79.7	75.3	80.3	144.9	80.5	82.6	63.5	91.4	65.1	69.7
82.3	86.6	72.8	73.9	88.3	75.1	130.3	67.5	84.5	69.8	67.5	63.1	75.1	69.8
81.3	63.1	59.6	79.2	70.2	66.0	83.4	67.4	72.8	63.5	65.7	67.7	67.0	73.4
76.8	69.3	81.7	74.5	69.3	96.1	67.8	1075.1	1081.3	1083.1	1075.6	1075.0	1074.1	1074.3
129.2	69.1	68.5	63.1	64.5	123.6	74.1	72.1	113.6	76.7	68.7	78.2	64.3	68.0
101.0	291.5	111.1	96.9	107.5	96.6	100.6	65.4	58.5	65.8	69.1	67.4	66.0	101.3
67.4	67.0	65.6	86.6	68.7	69.1	66.9	60.2	71.3	59.8	84.4	90.8	75.4	65.5
79.2	84.3	77.6	81.3	85.8	68.1	78.0	75.5	72.9	63.3	76.7	84.7	66.4	66.9
64.8	70.5	133.8	74.7	73.8	75.6	67.5	70.8	71.0	225.5	155.5	74.2	70.7	80.4
76.0	74.4	78.5	64.9	94.4	64.1	86.3	193.0	187.8	249.8	200.7	190.3	188.2	206.7
73.9	76.0	78.8	84.6	73.2	83.1	67.7	85.6	71.0	74.8	68.3	83.5	78.9	113.5
82.6	66.8	72.9	64.4	95.7	95.8	65.2	75.6	74.0	80.6	303.7	107.7	77.8	72.5
92.2	110.0	105.3	97.8	280.5	88.8	84.3	69.1	64.6	79.3	64.9	71.6	74.5	68.3
101.5	75.0	82.7	72.7	73.1	103.5	112.3	72.1	75.0	69.3	86.4	73.5	70.7	78.7
88.8	96.1	74.3	79.6	71.8	71.9	75.5	77.2	76.2	64.8	95.9	62.3	64.0	71.2
71.4	63.2	70.1	86.2	71.1	74.2	67.0	67.4	62.2	108.0	76.0	70.0	981.6	66.3
61.1	74.0	295.0	60.7	64.0	70.2	68.2	72.7	218.5	84.3	374.7	67.2	60.0	59.4
74.6	68.8	67.6	66.7	66.5	67.0	78.6	64.3	57.2	78.0	62.2	63.7	65.0	83.8
98.2	86.9	84.6	136.7	83.6	109.7	83.6	81.5	78.3	138.6	75.1	75.0	74.8	76.9
101.3	73.1	74.6	85.7	85.7	73.8	99.5	82.1	70.2	78.3	74.7	76.0	66.4	81.0
74.5	73.0	89.3	71.8	74.8	86.8	80.1	72.6	67.0	68.4	125.5	65.6	73.0	72.1
273.5	267.6	264.0	261.3	264.2	266.3	261.4	71.8	69.5	67.1	93.2	69.4	64.4	72.3
67.8	69.0	110.1	79.0	68.4	72.7	81.7	62.1	63.1	76.1	67.1	74.7	53.1	88.7
70.9	90.0	77.8	66.3	89.6	70.0	72.7	90.5	315.5	318.3	88.1	88.6	102.7	199.6
89.1	90.6	78.1	76.1	76.5	88.7	85.6	93.8	96.3	59.6	64.3	140.1	68.1	106.3
93.7	65.5	70.6	63.2	65.5	67.3	142.6	121.7	86.7	69.9	65.7	71.6	80.0	77.2
67.0	67.5	71.3	155.4	81.0	68.1	101.7	75.0	72.7	83.4	65.2	71.5	74.2	70.2
89.0	91.4	67.2	80.7	76.9	76.9	71.3	73.6	80.5	92.1	64.8	73.6	77.3	122.2
82.8	79.1	75.8	80.3	87.7	87.2	75.2	73.0	66.7	73.7	71.0	69.2	66.4	94.4
98.5	71.1	72.2	93.9	62.7	72.0	74.8	84.9	96.7	79.0	73.5	73.5	74.0	67.8
93.4	79.9	85.1	85.8	67.6	65.0	134.3	68.5	74.6	79.3	129.6	69.6	72.8	68.4
147.7	114.1	100.9	94.3	87.7	90.7	233.1	84.8	64.0	124.4	72.8	65.3	253.2	74.3
67.1	65.2	62.9	80.9	71.4	68.3	68.9	69.6	85.1	70.2	69.8	72.3	73.6	80.1
118.8	104.8	145.7	100.9	115.7	105.8	104.0	81.2	82.1	119.9	88.9	66.3	88.1	78.0
65.2	85.0	80.5	69.4	67.0	62.1	91.2	69.0	71.0	72.6	67.4	95.1	99.4	72.2
67.9	75.6	74.6	72.2	81.8	69.2	82.8	67.0	65.3	77.4	64.5	105.0	93.2	57.0
80.2	87.3	67.2	67.9	73.8	73.7	70.2	80.4	73.5	67.5	64.2	193.0	104.2	71.9
63.4	65.9	50.7	84.8	71.4	70.4	77.4	76.8	75.4	69.9	181.8	65.4	76.6	69.3
62.6	62.0	114.4	63.1	107.9	85.2	62.9	129.0	84.3	82.1	61.4	78.3	178.4	65.5
70.1	85.5	97.2	78.7	68.1	109.9	62.1	76.3	66.5	63.4	60.8	65.5	74.9	68.4
70.6	97.7	66.6	73.3	64.2	228.5	91.3	157.9	138.2	179.1	130.9	220.3	135.0	133.7
82.8	142.9	80.7	77.9	128.9	177.8	76.1	86.3	65.6	63.5	84.3	140.7	66.8	387.3
85.4	69.8	78.0	115.4	73.9	72.4	72.8	73.4	73.4	70.7	71.5	73.9	62.9	86.0
198.0	77.3	80.0	83.8	139.7	69.3	84.3	69.9	85.3	70.9	114.2	72.4	73.9	107.7
65.6	60.4	88.1	63.5	69.8	66.5	60.3	84.6	68.8	67.6	86.4	84.0	66.2	68.6
89.5	96.2	86.9	133.3	103.6	95.9	88.9	86.5	81.0	105.9	88.9	79.8	103.4	418.7
91.0	91.1	89.9	100.2	88.6	95.9	93.9	70.1	67.0	72.4	69.9	73.0	65.9	69.7
343.8	102.4	96.9	158.6	98.4	96.7	101.4	67.3	69.5	62.9	62.3	65.9	113.6	100.3
62.2	95.4	64.9	68.3	62.4	61.7	67.0	73.7	67.8	73.0	70.7	158.1	74.1	59.8

Continued on next page

Table B.6 – *Continued from previous page*

75.6	174.4	59.0	75.1	72.0	125.7	406.4	76.8	103.8	66.7	61.6	109.8	105.6	64.2
59.7	97.0	60.6	55.5	78.2	63.4	64.5	70.7	74.7	70.0	67.8	68.0	77.0	66.1
73.9	74.2	86.0	88.5	75.6	71.0	76.5	286.1	68.1	160.3	319.1	74.8	67.0	63.4
63.3	113.7	82.0	72.0	66.4	66.6	131.7	69.3	98.7	93.4	71.2	68.5	71.9	73.3
71.0	64.2	66.7	75.9	66.1	70.1	75.3	89.1	105.9	65.0	65.0	72.4	67.5	71.6
74.1	72.4	400.8	67.1	73.5	70.2	73.6	81.2	73.7	79.2	110.1	101.8	88.2	66.9
88.8	81.1	81.1	85.5	83.1	89.3	90.0	67.5	98.7	76.6	65.9	66.9	66.2	98.1
82.6	66.5	63.5	75.6	66.5	115.0	81.6	63.9	57.2	83.4	90.4	87.1	66.9	75.8
78.7	66.9	73.5	86.3	69.0	94.9	81.1	73.4	79.2	71.2	83.6	76.9	77.4	98.7
72.4	73.4	79.1	59.2	67.5	100.5	58.8	78.1	75.9	135.3	77.0	78.5	84.7	79.1
85.4	105.7	67.9	69.7	73.2	65.1	132.3	74.4	78.5	77.1	74.5	85.7	109.3	89.8
74.3	66.1	71.2	76.9	79.2	65.2	64.1	74.8	69.0	64.3	64.8	64.5	82.3	94.0
66.6	75.8	78.8	79.6	73.0	146.2	76.7	66.1	67.8	71.3	81.8	59.7	65.3	66.6
61.7	68.8	77.7	60.4	70.2	59.9	61.2	97.4	95.0	326.0	104.9	85.9	92.2	86.1
67.4	69.6	64.1	933.2	67.7	69.0	81.0	71.6	67.6	75.6	85.1	67.7	73.5	69.3
73.7	67.9	79.7	76.7	64.1	81.1	64.9	68.1	70.5	69.6	86.2	82.9	85.6	63.9
69.7	67.1	65.0	97.8	102.0	79.7	71.5	67.7	75.5	92.7	90.3	74.6	59.7	58.7
78.7	124.4	75.8	76.8	82.2	1169.2	70.6	102.6	102.2	201.8	120.8	103.0	113.5	97.1
100.6	72.3	75.6	270.0	68.6	74.3	76.3	85.4	71.5	72.5	78.1	92.8	88.5	89.3
108.0	175.7	136.6	107.8	69.6	69.1	78.3	68.4	65.3	88.9	62.7	66.2	75.5	60.7
65.0	78.8	90.0	70.9	68.0	64.6	70.4	112.0	66.1	91.3	119.5	67.3	70.2	133.0
73.5	69.5	65.0	67.1	74.0	62.4	61.4	91.8	64.2	61.2	68.6	71.1	64.8	81.3
64.2	72.7	70.0	71.2	72.9	70.9	65.1	75.8	79.8	75.1	75.1	79.0	97.6	75.5
75.6	81.5	62.4	64.3	65.1	65.2	71.4	73.9	69.6	67.8	68.1	70.7	75.7	77.4
188.7	64.8	72.4	69.1	64.0	68.2	85.9	81.2	65.3	84.0	88.5	75.6	67.2	74.6
74.6	82.5	71.3	75.0	66.5	70.6	83.8	61.8	78.3	84.7	64.9	89.3	67.8	70.1
128.1	95.6	62.4	66.9	81.8	77.0	90.3	68.7	79.9	62.2	64.7	66.0	63.3	74.0
67.4	79.3	66.5	65.7	75.7	68.4	67.3	76.3	84.7	86.1	107.6	71.4	76.8	66.9
92.2	97.5	104.7	93.1	136.1	117.7	89.4	75.0	68.2	78.3	66.0	68.5	64.4	60.2
107.1	104.3	88.9	98.6	112.4	104.1	90.3	87.3	64.8	122.6	87.8	80.6	63.6	59.8
130.4	140.2	128.9	187.2	163.9	147.1	136.1	108.0	72.7	68.5	94.8	74.6	69.7	71.1
65.2	67.8	73.0	63.3	65.9	65.6	61.8	89.5	93.6	94.9	100.0	117.7	99.7	87.9
92.7	119.0	132.0	142.0	84.0	79.7	90.2	66.9	79.9	74.0	63.9	75.6	65.1	60.3
78.6	84.9	67.5	72.5	65.5	82.6	110.1	72.5	74.8	81.3	70.2	76.8	73.8	77.7
104.7	99.9	71.8	74.6	65.9	73.6	83.6	82.9	66.2	65.9	83.5	245.9	66.7	145.6
78.0	75.1	74.6	73.1	78.6	142.8	73.0	82.1	71.1	74.7	98.7	64.6	64.7	70.1
293.5	301.1	317.4	292.9	298.8	295.8	292.3	87.6	86.0	72.6	62.3	71.9	68.5	80.5
71.2	66.7	79.7	1135.4	74.9	93.0	108.2	102.8	130.6	70.1	64.4	65.0	68.1	80.8
63.1	64.8	69.1	97.7	80.7	67.6	90.7	81.4	95.7	92.3	83.7	91.9	71.1	165.5
76.7	86.5	116.3	96.6	89.5	72.2	75.9	82.4	70.4	67.4	77.5	72.0	74.1	69.3
55.5	84.6	72.6	62.2	60.3	81.9	66.5	72.2	64.0	64.0	99.6	64.8	139.7	66.1
74.7	92.3	68.7	86.4	67.0	68.8	70.0	68.0	73.5	66.8	72.8	57.4	64.4	66.0
69.2	67.7	92.6	86.2	118.6	70.0	453.2	68.5	83.2	73.6	95.9	63.9	68.6	91.4
82.8	69.7	90.0	78.8	98.2	85.2	74.3	117.2	64.9	81.2	81.2	73.3	110.4	59.8
83.7	70.6	69.0	65.2	80.6	81.7	69.3	96.0	66.5	70.1	80.3	70.4	80.5	70.8
200.4	59.8	70.5	81.0	66.4	65.4	70.0	65.2	85.7	63.0	115.1	110.0	65.7	70.8
72.2	72.6	116.5	86.6	94.6	82.7	76.1	69.4	75.4	70.1	75.1	69.4	77.5	70.7
66.0	70.4	91.1	67.9	59.1	68.9	62.3	72.3	64.7	69.2	65.2	74.3	73.7	79.5
117.4	70.5	79.6	85.2	90.3	69.7	67.5	68.0	84.8	71.5	86.7	65.0	81.6	72.9
63.3	72.4	81.2	77.9	74.5	80.2	112.7	94.0	81.3	83.4	96.2	98.2	113.5	87.5
188.1	137.1	61.2	65.3	63.6	108.4	67.7	83.5	77.4	79.8	109.4	77.9	81.6	92.7
74.5	66.9	68.5	73.3	62.1	112.8	88.3	216.8	181.8	181.1	184.2	184.7	196.3	179.3
81.5	89.2	100.4	129.3	106.5	86.0	158.6	76.9	93.0	70.9	75.8	93.7	107.8	103.6
96.1	148.3	71.9	79.8	68.8	64.4	72.5	88.3	67.4	77.4	69.2	62.4	69.4	70.8
72.6	64.8	61.5	60.9	62.5	65.8	77.6	135.8	73.9	74.4	65.2	65.2	382.9	69.2
66.8	141.7	65.7	124.9	69.2	81.7	82.4	75.7	784.3	101.6	84.9	93.7	70.2	71.1
72.6	64.6	98.1	86.9	61.9	141.8	71.9	62.3	68.6	81.2	64.3	77.2	69.9	79.3
107.7	69.8	68.8	86.0	65.9	65.7	65.6	66.8	70.5	84.5	69.0	73.1	73.2	69.5
104.0	104.5	100.8	108.0	489.6	91.8	104.5	62.4	66.9	144.7	63.9	94.3	76.1	230.2
68.2	114.9	76.2	75.2	68.2	69.2	74.8	61.7	66.4	67.1	78.4	65.8	70.7	76.2
97.6	74.6	63.6	79.7	70.4	69.2	167.5	90.9	66.2	235.6	67.6	306.3	120.5	64.2
74.4	60.4	69.9	69.2	63.9	64.6	75.9	77.5	64.9	67.2	81.6	104.6	70.8	64.8
80.8	64.6	64.5	78.1	71.8	67.5	62.8	77.5	70.3	70.4	64.3	65.7	65.8	108.2
64.0	75.0	70.8	68.1	68.1	70.1	64.5	66.9	73.3	64.8	144.8	54.1	88.6	87.6
62.4	68.4	61.3	119.8	94.4	115.1	71.1	60.5	112.3	70.6	78.4	145.7	81.6	64.5
67.1	75.3	91.9	101.2	63.7	72.4	85.5	67.3	78.4	79.4	85.3	74.8	108.5	90.4
842.5	75.9	76.8	81.1	70.9	81.7	78.6	80.7	67.3	219.2	167.6	67.1	61.4	63.0
62.1	77.4	76.8	67.2	77.3	61.9	72.6	61.4	69.4	81.3	89.6	69.0	65.7	63.9
64.0	94.6	64.0	102.8	71.1	73.8	75.1	64.6	98.5	71.4	78.3	66.9	63.6	70.3
130.0	64.8	62.1	66.2	87.9	63.6	68.6	82.5	68.9	67.4	74.5	67.1	100.6	67.4
87.9	66.2	65.1	78.7	70.3	62.9	90.5	86.2	67.1	99.4	74.0	69.2	62.5	116.8
68.2	75.0	67.9	78.2	72.7	72.4	88.3	89.2	118.0	89.3	148.4	145.5	141.5	119.7
67.9	63.0	62.5	65.9	68.1	63.7	71.3	96.7	93.4	97.6	163.0	92.3	138.5	94.7
70.6	101.5	61.0	67.4	65.3	87.5	60.8	125.6	69.8	78.8	71.6	72.2	70.7	73.7
77.7	93.4	77.5	74.2	87.6	79.3	73.5	77.7	72.3	73.3	81.9	65.3	88.8	63.6
86.7	74.7	69.1	75.1	74.1	87.7	81.7	68.8	94.7	102.2	64.8	96.2	70.2	78.1
172.9	126.3	124.0	162.7	127.2	164.6	210.0	74.6	74.7	93.6	133.0	71.2	81.0	151.3
218.7	81.7	89.0	74.1	64.3	68.0	246.4	64.1	70.4	73.3	83.4	70.7	67.3	63.1
67.4	91.7	69.1	70.1	65.3	75.5	68.4	65.0	197.8	70.5	194.6	63.7	307.0	59.9
72.7	84.9	112.6	77.7	71.0	72.8	63.4	68.3	66.0	67.6	68.2	78.3	65.7	70.7
209.7	87.9	110.7	94.9	124.8	91.0	89.0	65.0	66.2	62.2	85.0	60.5	63.9	66.3
77.4	66.7	72.6	85.4	68.0	82.4	84.1	80.7	70.5	71.9	66.6	77.4	69.8	79.0
63.0	72.0	68.1	98.4	95.5	77.0	61.3	81.2	74.0	249.0	64.8	69.3	85.8	63.1
74.6	73.7	69.4	65.9	106.8	156.6	73.7	65.2	61.7	90.7	113.6	84.0	66.2	72.3
63.2	65.7	61.7	90.5	68.2	69.0	59.1	398.3	85.3	74.0	91.0	82.8	83.1	95.9
75.5	104.6	89.4	62.2	192.7	98.3	75.4	162.3	68.5	92.7	69.6	73.1	64.3	59.8
78.7	80.4	78.5	78.2	70.4	72.3	83.5	72.7	88.1	73.1	77.5	136.5	68.9	68.4
77.9	79.2	82.1	122.3	97.4	153.8	66.7	71.3	101.6	73.5	69.3	70.2	80.3	72.8
80.6	67.0	72.1	67.2	185.6	70.6	66.8	65.8	77.2	99.2	73.1	68.2	66.8	69.0
76.4	64.2	69.2	70.4	74.6	72.1	107.3	74.8	81.1	74.9	66.6	84.4	102.6	68.2

Continued on next page

Table B.6 – *Continued from previous page*

78.7	71.3	67.4	84.9	75.3	83.6	87.2	71.2	66.8	70.0	70.1	95.2	68.0	61.6
88.2	83.3	66.4	64.3	69.6	76.1	68.5	109.6	67.6	108.8	178.7	65.7	87.9	73.4
76.1	80.9	89.8	77.4	84.0	77.8	93.6	620.7	69.0	100.5	72.4	75.6	71.1	60.9
203.2	171.1	66.6	61.7	63.9	73.4	67.1	76.7	66.4	215.7	84.5	107.8	107.1	61.9
78.7	67.6	70.1	78.5	77.7	69.9	83.5	86.6	80.1	81.8	89.9	87.8	160.5	70.4
66.6	75.1	78.0	133.3	74.3	70.9	65.8	66.1	76.7	75.8	70.8	76.0	89.3	110.9
176.0	166.9	164.6	194.9	196.5	183.9	186.5	62.3	105.4	73.3	128.7	66.6	64.3	75.4
73.5	93.4	67.1	69.9	71.8	108.0	87.9	66.4	68.3	103.0	65.9	72.5	62.2	59.0
145.0	152.5	139.4	191.4	555.7	156.2	156.5	90.3	157.6	71.4	216.2	91.1	74.1	50.8
78.0	72.3	69.5	72.6	75.5	72.6	98.7	76.9	75.2	83.6	70.3	69.5	74.6	78.9
150.2	157.9	96.1	97.2	95.5	93.6	91.1	77.1	64.5	86.8	117.5	118.1	75.6	73.5
79.1	90.8	73.8	183.0	98.1	72.5	81.4	70.0	68.2	69.6	93.1	876.0	78.8	66.2
78.9	68.0	70.4	89.3	111.2	74.3	67.7	70.1	71.2	69.7	78.7	64.2	79.2	75.5
72.4	112.1	90.0	75.0	68.1	96.8	90.5	75.2	84.3	71.5	70.8	72.8	108.2	77.2
107.5	79.4	102.7	75.0	70.9	67.2	99.8	131.9	174.5	142.5	164.3	132.2	133.9	135.5
70.1	95.7	72.8	73.1	69.3	73.5	100.6	70.9	75.0	75.6	76.3	94.0	76.3	102.7
72.1	73.6	112.8	87.3	67.3	202.6	70.2	58.6	74.9	68.7	71.7	113.1	74.3	72.5
76.8	76.9	74.2	81.8	140.0	81.4	102.4	107.3	83.6	90.0	113.5	85.0	82.4	92.5
68.4	62.1	180.1	78.7	65.8	75.6	89.4	68.2	126.9	62.8	77.7	76.1	67.2	66.2
67.4	70.3	76.2	63.9	163.4	135.0	171.6	67.9	146.3	67.9	78.7	64.7	63.2	73.1
84.3	78.3	68.9	70.0	69.6	72.5	120.3	180.3	98.4	95.5	85.1	84.9	88.4	87.7
138.4	68.1	67.9	63.2	68.5	63.7	79.2	85.0	63.1	61.5	88.3	66.7	84.9	91.3
94.0	81.1	90.6	81.9	79.2	74.1	91.3	80.2	72.0	64.9	64.3	66.5	72.3	62.0
66.0	59.9	63.4	69.4	76.1	73.7	72.5	97.2	100.9	112.0	381.9	95.0	113.7	103.0
83.5	86.6	66.5	71.9	295.7	175.6	90.6	73.9	82.0	91.1	72.6	87.7	224.1	91.8
93.6	90.4	58.9	63.9	80.1	89.3	61.5	73.3	60.7	94.5	83.6	60.7	79.7	66.6
70.2	65.1	93.5	81.7	68.5	117.0	90.5	71.5	65.8	112.9	88.8	65.4	75.0	67.0
74.2	69.1	64.8	65.1	64.2	68.8	60.5	64.8	65.5	68.9	75.7	69.9	65.8	66.6
63.9	70.1	70.4	63.0	64.0	66.7	68.5	61.9	75.9	75.8	92.4	82.5	72.0	76.2
124.7	150.4	112.0	100.4	212.2	106.3	109.4	72.8	520.2	69.8	101.8	106.5	84.1	63.2
141.3	137.6	137.0	215.4	138.7	144.8	160.1	114.9	453.5	124.1	137.7	131.1	207.9	126.6
94.7	105.7	92.8	89.6	87.7	96.8	92.0	67.1	110.7	71.0	75.2	136.9	62.0	92.1
67.6	72.5	60.2	72.7	67.7	71.1	108.0	83.7	70.3	59.1	66.9	67.6	97.3	66.3
56.5	68.6	76.1	182.5	70.9	64.9	89.4	70.8	65.5	100.4	291.3	64.3	67.7	121.9
65.9	72.4	95.5	70.0	71.0	72.1	77.5	74.4	62.5	72.5	90.8	69.9	71.2	71.5
62.7	64.1	79.2	66.6	86.7	66.0	83.9	63.2	58.8	66.7	275.2	74.8	55.8	262.9
72.0	62.0	66.7	78.1	512.1	64.9	65.0	72.3	72.2	64.8	73.1	71.2	60.0	72.4
75.7	136.8	69.0	92.9	123.7	69.2	71.7	66.8	160.5	90.1	60.3	69.6	71.3	93.8
84.3	82.6	81.2	85.7	94.3	84.5	84.7	76.6	68.5	72.1	80.3	106.3	67.7	66.6
68.8	69.4	155.6	88.1	170.8	66.5	72.1	82.9	62.8	68.1	84.2	66.3	226.1	62.4
76.1	87.4	75.8	159.3	69.7	70.9	73.2	387.2	373.6	370.9	373.4	374.0	453.7	371.5
216.7	74.8	76.4	72.7	187.0	124.8	75.5	68.6	72.0	80.8	148.1	79.7	65.5	137.8
80.8	64.4	86.2	87.6	70.2	86.5	75.6	75.4	115.1	84.1	74.6	68.9	68.4	65.2
70.9	57.2	64.5	77.4	69.1	120.5	78.2	348.4	365.9	357.3	346.9	347.5	356.0	354.8
82.1	80.7	114.4	65.3	76.8	133.3	79.7	67.2	76.9	76.9	75.7	66.5	67.2	95.6
148.0	163.3	148.3	154.4	145.5	144.9	147.0	73.8	147.6	78.3	121.5	73.7	71.8	77.0
68.0	65.5	81.6	72.5	80.0	68.4	79.5	67.2	67.3	65.6	64.9	73.1	97.9	80.0
307.6	69.3	66.0	83.1	75.0	64.2	90.7	65.4	64.4	69.6	70.4	76.3	64.8	84.0
69.8	84.8	73.5	76.1	75.3	68.5	68.4	94.5	80.6	80.5	88.5	86.1	83.3	87.5
389.2	77.3	68.3	145.8	69.9	68.8	67.9	66.3	73.0	61.3	68.2	80.0	74.8	62.5
63.6	70.2	86.3	64.9	74.7	76.9	60.5	73.7	82.0	82.8	67.5	109.4	70.1	73.7
100.3	110.3	99.1	100.8	98.5	107.7	104.3	68.2	70.3	87.2	80.1	88.8	72.3	73.8
71.4	68.3	68.2	88.4	85.9	64.7	78.3	76.2	64.0	67.2	73.7	64.2	64.1	75.1
71.5	70.0	78.4	76.1	104.5	168.8	89.9	235.7	65.2	65.3	70.9	63.0	70.4	93.7
88.8	69.4	70.5	113.8	82.4	92.3	66.7	79.0	96.0	97.8	72.2	67.3	68.9	74.4
138.6	68.5	73.8	82.6	74.4	74.3	70.2	84.5	85.2	86.3	85.1	79.9	101.1	164.0
108.2	161.1	66.9	78.0	76.0	71.6	67.5	78.8	225.3	62.1	59.9	61.9	52.6	61.2
63.8	112.2	68.8	64.3	74.3	68.5	154.0	77.0	68.7	124.3	93.9	61.6	68.7	80.1
64.0	66.7	69.2	145.6	65.7	58.1	59.4	80.1	86.7	70.4	73.2	68.4	69.0	66.8
73.7	76.6	110.4	106.3	73.9	69.9	70.8	83.8	69.5	73.5	80.9	72.6	107.8	68.5
63.5	84.2	94.4	66.2	64.1	139.1	92.2	110.8	68.0	73.0	64.8	85.9	64.3	64.0
274.2	101.0	92.9	87.5	118.0	120.4	86.7	122.1	65.1	63.8	70.9	78.2	109.7	101.8
63.7	96.3	65.5	84.8	216.2	118.8	62.0	67.8	155.9	77.0	69.8	69.3	63.7	71.9
92.0	76.5	73.8	74.8	73.9	85.3	67.4	77.8	80.6	100.6	169.6	93.0	80.4	77.5
108.0	80.1	67.0	108.0	61.0	69.8	124.7	72.9	86.0	65.1	74.4	70.9	85.4	77.2
78.3	76.7	75.6	76.3	64.7	73.0	94.0	75.3	66.9	87.0	63.3	64.3	75.2	78.9
97.7	95.7	79.9	79.1	80.8	91.1	78.4	73.9	104.6	69.4	85.0	68.3	81.6	72.5
71.5	74.7	82.0	110.2	94.1	73.1	88.1	63.8	63.0	63.3	57.8	72.0	74.2	63.3
67.1	70.2	72.0	116.0	91.7	67.5	57.6	191.9	155.4	154.8	169.0	157.1	154.1	171.6
68.4	66.0	112.7	69.1	152.0	69.1	68.0	233.6	241.5	229.5	218.7	213.5	215.8	212.5
68.7	75.3	269.7	65.2	116.7	347.3	71.2	102.9	67.7	68.1	69.9	64.1	67.7	84.9
65.6	218.4	67.2	59.4	64.8	59.6	66.1	92.5	82.0	83.7	83.6	80.6	112.9	78.7
90.2	62.4	61.0	63.4	80.7	94.4	62.3	150.5	66.3	66.8	66.8	119.7	78.4	212.9
65.3	73.5	180.8	104.5	66.9	80.3	162.7	68.2	71.8	66.6	69.5	76.6	117.0	86.6
420.7	101.9	78.8	89.1	80.8	79.8	86.9	80.7	69.2	102.5	69.8	69.8	89.4	87.4
70.6	82.4	80.7	81.1	93.3	175.3	61.5	74.0	67.8	68.3	71.2	100.6	149.4	109.4
70.5	114.2	64.6	65.2	72.5	69.9	119.9	68.0	76.7	95.5	65.7	69.9	67.4	71.1
145.1	145.4	144.5	135.5	133.4	141.6	137.2	71.5	87.4	185.5	68.9	97.8	66.7	75.8
93.1	65.7	63.1	234.1	78.6	62.8	75.5	61.4	73.6	67.0	71.5	235.3	80.0	100.0
69.4	77.1	151.5	92.4	66.3	83.2	75.2	71.3	67.7	62.0	84.5	81.7	72.2	87.0
71.0	67.2	79.3	71.5	70.8	70.3	71.8	182.3	82.1	152.6	94.7	90.9	91.3	87.6
80.6	70.8	90.5	86.1	90.7	69.2	74.8	109.3	63.7	63.9	71.0	59.4	71.9	65.4
69.2	79.6	72.8	69.6	119.4	71.9	121.0	81.1	96.9	147.2	74.5	164.0	90.2	129.6
64.2	74.3	73.9	65.0	67.6	89.3	134.8	107.8	63.1	64.6	68.1	128.0	68.4	74.8
83.0	65.5	93.1	74.1	69.9	66.1	102.4	81.1	76.9	73.7	86.7	72.4	72.8	72.9
63.0	192.4	70.6	61.8	70.8	62.7	68.5	79.2	136.0	82.8	94.5	76.9	109.5	77.7
70.1	68.2	85.9	64.8	73.3	104.5	70.4	70.9	85.5	72.5	97.2	69.3	87.4	116.0
67.3	73.6	65.1	81.1	71.0	62.7	59.2	64.8	65.1	104.1	84.2	73.2	66.5	68.5
69.7	105.8	80.7	94.9	72.5	159.4	73.7	98.9	96.8	91.5	95.5	95.5	131.5	87.9
69.7	97.6	70.5	63.3	76.4	80.7	67.3	123.0	76.1	69.7	70.1	71.3	57.8	68.3

Continued on next page

Table B.6 – *Continued from previous page*

68.8	65.1	102.0	79.9	59.0	67.8	71.3	132.9	131.1	131.4	167.2	141.9	140.3	279.8
68.9	63.0	70.6	63.2	130.3	219.1	74.4	113.8	105.6	112.2	162.9	104.8	96.5	100.8
62.1	73.4	72.0	100.6	60.7	70.7	77.9	77.2	70.7	76.8	744.5	120.5	73.2	75.8
67.3	70.7	94.5	72.1	73.1	72.6	75.2	69.2	68.9	68.6	82.4	81.7	64.2	68.5
68.7	77.4	76.9	66.3	81.6	84.4	100.5	91.3	110.4	80.2	85.6	100.8	79.4	80.9
73.8	83.8	78.6	143.4	69.7	74.4	55.0	349.7	71.3	75.8	75.3	72.1	77.7	64.4
73.2	68.9	92.6	65.9	66.0	60.6	72.9	70.4	109.2	80.9	77.4	68.2	119.7	72.7
92.6	100.6	68.1	70.6	81.5	70.1	106.7	71.3	67.1	71.2	65.3	60.8	68.0	61.9
84.0	66.1	82.7	70.6	75.8	75.8	72.0	76.1	79.8	70.4	97.1	73.3	154.2	76.3
70.1	75.3	66.0	66.8	70.8	67.1	67.4	63.9	134.1	68.1	67.7	64.1	75.8	83.0
83.3	117.2	81.9	76.6	89.0	80.1	80.2	124.8	103.1	116.6	97.4	107.6	106.8	106.5
57.1	67.0	62.6	58.5	95.0	56.3	68.9	109.8	131.2	113.0	85.8	88.1	79.1	73.7
78.9	77.2	71.2	69.9	81.5	129.4	74.6	70.1	99.5	72.5	68.9	85.3	92.7	62.4
60.7	61.3	71.8	68.7	71.3	64.8	66.3	97.7	70.1	73.7	74.2	92.6	78.0	74.8
76.1	76.7	104.5	70.4	73.0	70.9	77.1	68.4	71.6	63.7	75.6	185.7	64.2	71.8
74.4	95.6	65.9	77.2	128.4	68.3	73.2	68.4	86.1	75.2	81.3	97.5	67.3	69.5
74.7	64.8	68.7	64.7	102.2	66.6	74.4	79.1	80.4	108.6	84.4	82.4	106.9	73.2
114.6	95.9	91.1	102.9	90.4	104.8	118.3	76.0	74.1	62.4	66.2	83.3	65.5	62.1
69.2	147.7	70.0	68.9	78.8	75.3	79.2	65.5	113.7	68.4	70.7	68.2	70.8	95.8
69.7	86.4	65.0	78.3	73.5	65.2	139.5	145.1	95.5	87.3	1629.9	73.6	68.8	65.8
68.9	71.2	129.5	89.0	65.6	73.9	74.6	175.0	97.0	72.5	68.7	73.4	69.1	69.8
63.2	79.4	69.7	115.6	66.1	77.5	75.5	93.7	63.4	78.2	63.7	70.0	68.7	65.8
66.5	67.5	60.5	66.0	66.1	64.9	63.9	70.8	59.8	77.7	68.2	209.2	70.7	80.3
102.6	80.7	71.9	73.1	70.1	81.3	79.8	74.2	74.2	66.1	71.4	58.6	95.4	158.1
73.8	127.0	69.8	79.5	60.8	64.5	53.6	72.4	113.8	81.5	73.8	78.8	83.8	68.1
67.9	61.4	73.5	68.7	101.4	79.4	68.6	89.1	61.1	80.1	66.8	124.8	74.4	100.1
115.6	71.9	75.2	95.8	77.4	90.6	103.8	200.0	73.0	61.5	161.6	108.7	91.0	75.3
67.1	91.1	71.4	63.4	66.4	112.4	64.9	105.9	128.4	111.3	111.2	113.4	107.2	102.9
74.7	69.2	79.0	65.1	69.4	66.7	72.1	67.8	69.0	72.3	68.8	76.0	93.7	68.4
71.9	85.6	92.7	71.7	74.0	77.8	77.8	61.5	67.1	100.2	68.1	101.2	60.8	69.1
94.7	79.8	83.5	74.8	69.8	121.8	70.5	77.2	63.2	68.8	66.9	113.5	70.6	65.0
72.8	66.7	82.2	233.4	85.8	65.8	72.7	65.9	79.0	69.4	122.2	69.9	68.0	68.4
74.3	80.1	69.8	70.0	76.5	128.9	67.8	112.4	91.2	88.6	71.8	80.1	85.5	78.1
83.4	69.3	65.7	94.7	65.7	92.2	64.2	78.5	74.8	99.4	58.2	87.2	83.7	126.4
75.6	59.2	68.9	78.1	60.6	68.1	119.3	109.8	71.0	80.3	102.3	66.9	153.2	85.1
72.2	76.6	78.1	65.6	68.8	70.3	102.6	82.4	69.1	73.1	76.6	88.5	77.6	75.8
66.5	77.8	64.1	92.9	65.9	75.0	72.0	66.3	151.4	74.3	71.5	96.0	72.3	70.3
90.0	68.9	56.7	76.4	68.3	73.7	70.7	133.8	74.1	77.0	85.2	71.4	77.4	78.7
71.5	94.3	309.4	90.5	67.3	65.3	75.6	79.9	109.1	83.6	98.7	71.7	83.3	78.5
66.9	78.4	72.1	69.2	78.3	73.5	76.8	95.8	128.7	95.9	63.8	84.1	89.0	108.8
72.0	75.9	67.7	170.2	68.5	81.8	66.7	68.2	64.5	83.3	64.2	81.7	79.7	68.6
79.0	65.1	98.7	123.6	84.2	69.8	71.4	67.6	84.4	68.1	99.5	91.8	79.5	69.0
71.9	143.2	67.6	70.5	65.9	95.9	66.6	67.0	67.8	68.7	66.3	87.9	92.5	81.5
112.7	102.7	102.8	335.9	98.8	99.4	97.0	488.0	61.7	71.4	64.6	64.4	64.1	87.2
73.1	135.8	71.3	64.9	98.1	70.5	131.1	61.3	66.6	64.8	69.3	67.8	71.2	64.6
155.1	70.0	86.8	80.0	68.2	137.0	64.4	64.2	68.3	98.7	84.3	71.5	63.8	73.2
71.1	77.9	60.6	74.9	73.2	78.2	67.2	84.0	99.0	86.9	108.3	128.8	90.8	79.7
74.9	80.6	90.3	71.0	85.4	73.7	164.8	77.9	88.1	279.9	85.2	74.6	101.1	102.3
61.3	72.2	101.5	69.6	61.6	72.4	670.4	113.2	94.8	102.7	108.4	92.4	96.0	98.3
74.2	128.1	76.5	76.8	99.8	73.8	68.5	88.5	73.5	78.8	80.9	83.5	76.5	74.0
78.8	75.5	76.2	72.7	75.7	70.4	84.0	68.7	73.7	75.8	65.7	76.2	67.9	64.8
63.7	64.1	73.8	60.1	138.7	84.7	109.1	64.3	73.7	69.1	64.2	67.0	63.9	77.2
67.3	57.1	68.9	70.2	65.8	70.2	94.7	65.8	63.1	70.3	58.2	59.8	58.8	115.7
83.3	437.0	98.0	141.7	83.4	82.0	101.0	64.8	72.8	63.0	66.3	66.5	85.4	67.3
64.5	66.5	75.3	65.3	62.6	69.3	68.0	94.6	83.4	74.2	71.7	75.4	65.4	135.1
73.7	67.4	67.5	80.1	76.2	114.6	73.0	63.7	63.0	65.1	64.0	86.9	65.7	62.6
78.5	104.3	78.1	61.9	61.7	73.4	69.2	64.5	89.0	82.5	101.0	60.8	176.1	70.8
80.5	79.7	79.1	122.0	117.4	93.2	94.9	83.1	77.7	82.8	77.4	124.4	110.9	83.1
63.5	71.7	68.9	706.9	70.5	211.9	69.6	69.8	60.3	85.3	67.9	65.4	76.1	63.7
64.6	123.1	68.2	72.3	76.6	83.9	72.7	68.1	84.8	64.5	71.7	74.3	171.2	63.5
70.5	83.5	67.9	97.0	71.3	111.4	70.5	98.3	73.2	72.0	79.7	69.4	73.0	96.4
71.1	64.7	67.2	58.6	115.4	62.6	58.6	85.4	91.0	97.0	72.7	82.8	66.8	83.5
62.9	115.1	67.9	62.5	88.6	66.3	59.5	130.7	127.3	123.4	134.2	128.0	147.3	124.7
64.7	78.1	71.1	92.1	60.1	78.0	121.4	93.9	84.5	67.0	115.3	77.9	78.3	86.5
76.0	135.3	97.3	75.9	98.9	248.9	74.1	72.4	97.8	90.7	78.3	85.2	75.0	78.0
84.6	74.4	136.2	68.8	69.5	69.3	73.0	68.1	68.3	71.9	77.3	69.9	68.3	66.0
95.7	65.5	68.3	64.2	78.4	74.1	68.5	67.6	69.6	111.9	65.8	88.9	75.0	67.6
70.5	84.3	64.3	72.9	67.4	90.6	68.4	73.9	94.4	93.5	68.0	63.9	73.4	72.4
187.5	125.5	88.9	87.5	102.1	292.6	94.4	79.5	85.9	80.8	106.6	84.5	115.0	84.8
66.6	77.6	76.6	431.6	59.2	145.2	69.3	76.2	76.5	84.4	85.0	70.5	78.8	168.0
231.6	95.4	68.1	93.7	72.0	68.9	53.6	72.4	70.5	85.5	79.6	59.6	85.3	78.8
70.4	87.5	94.1	74.4	73.3	84.8	95.1	79.5	82.7	72.9	68.0	64.6	65.2	61.7
189.6	74.6	70.5	91.0	84.4	69.2	69.6	101.8	66.4	88.0	68.4	61.3	76.8	72.5
69.1	77.1	73.2	83.1	71.7	84.2	57.5	81.1	69.1	73.0	82.8	63.9	104.1	82.9
74.0	72.2	99.9	195.6	79.7	73.7	76.6	457.2	72.0	65.0	79.6	66.4	113.9	65.4
69.8	64.6	67.9	59.3	83.8	137.1	206.3	67.6	64.2	62.0	61.7	63.8	85.2	94.4
114.1	67.0	72.0	82.7	79.4	67.8	59.6	65.1	65.7	78.3	70.8	61.3	78.9	63.0
76.1	84.3	71.4	78.6	76.2	84.1	76.3	86.3	78.6	72.8	73.1	81.5	65.4	68.2
76.7	166.0	72.6	96.4	66.5	69.0	69.6	78.0	73.8	76.1	80.9	64.9	74.2	67.4
67.7	70.8	68.7	88.6	72.3	67.6	66.3	71.1	70.6	74.4	128.4	72.8	79.4	132.8
98.2	80.5	103.3	75.6	119.1	103.5	72.0	81.8	80.6	68.7	65.7	66.9	253.8	76.1
111.6	89.5	107.0	91.3	105.3	93.9	95.1	134.7	117.5	114.3	111.4	112.0	111.8	127.3
97.8	82.9	102.5	81.5	96.6	71.8	84.7	67.8	69.1	85.2	111.1	79.8	62.9	68.7
76.9	83.3	81.5	85.3	88.0	88.7	84.5	71.6	70.5	65.5	86.2	69.7	63.7	67.8
68.9	56.0	71.9	67.6	64.6	75.0	74.5	89.7	107.0	74.0	95.4	73.2	70.6	72.3
67.7	65.6	84.0	63.6	69.9	67.9	64.7	66.1	68.9	65.6	65.0	63.5	67.8	71.5
67.3	68.5	67.5	63.5	181.2	77.5	94.2	183.1	199.2	216.8	201.0	191.8	189.5	184.4
88.6	84.8	78.6	66.3	65.4	67.5	77.6	66.9	78.5	84.3	74.3	107.9	70.2	65.5
66.2	75.5	69.5	63.1	67.9	78.6	61.7	71.2	91.3	47.0	80.1	66.5	91.2	73.2
63.0	72.7	229.5	86.2	89.4	75.3	68.3	64.5	80.1	124.4	121.9	74.2	61.9	68.6

Continued on next page

Table B.6 – *Continued from previous page*

153.9	64.9	70.8	63.3	66.7	66.7	94.7	67.6	67.3	120.6	72.4	74.9	68.9	63.2
62.3	68.0	65.2	67.9	65.0	89.7	61.8	71.3	120.7	76.5	69.4	73.7	70.5	69.4
69.2	78.5	94.9	72.8	95.5	71.1	65.5	66.3	127.2	72.0	82.1	99.2	65.3	64.7
71.9	70.2	67.8	72.3	73.3	71.6	200.0	118.5	119.6	66.8	84.8	59.7	64.5	59.2
70.1	70.5	85.9	77.2	79.6	80.7	65.7	73.6	69.0	70.4	102.3	66.5	72.3	114.2
78.8	88.8	69.0	73.4	73.5	67.3	80.4	66.0	65.7	105.2	92.4	89.5	77.3	150.8
61.4	64.9	67.9	73.9	64.5	72.6	65.7	67.9	82.3	78.4	56.5	64.7	67.8	244.6
97.1	80.9	66.7	64.7	66.7	69.7	70.4	75.4	90.2	67.4	71.2	62.8	65.5	118.9
70.1	136.0	97.9	72.3	216.1	72.1	103.3	73.6	72.7	218.9	64.6	65.7	66.1	65.3
79.5	67.4	150.8	73.7	67.7	68.7	72.7	74.7	63.6	64.2	63.9	59.1	67.6	69.1
67.1	64.8	85.2	76.7	71.1	67.5	64.2	117.6	67.5	65.7	65.7	62.1	64.3	61.3
74.7	74.7	73.4	75.8	72.0	83.8	68.4	69.8	67.1	71.9	278.7	145.6	63.5	69.7
65.2	76.1	91.5	71.2	79.0	97.3	65.4	71.5	74.5	66.4	64.5	84.5	154.4	64.8
81.7	69.6	69.2	80.3	69.2	110.5	65.7	70.3	91.1	80.1	74.7	72.5	98.3	72.2
73.4	79.6	71.6	70.2	74.0	61.5	65.5	123.4	75.0	79.9	984.6	72.0	77.4	63.1
65.2	297.0	70.8	80.8	63.1	46.3	80.8	291.3	73.7	77.4	68.6	75.7	77.7	97.5
66.1	69.1	64.6	171.4	64.1	81.1	65.4	96.2	74.8	69.6	66.9	73.1	66.7	66.6
63.8	70.1	74.7	63.6	112.5	65.5	73.0	64.0	58.2	67.4	61.4	74.4	61.0	63.6
74.0	97.1	76.0	77.9	218.5	71.5	68.9	53.5	62.0	66.1	70.6	52.8	67.9	63.1
72.0	65.2	80.8	73.3	71.2	66.2	60.5	64.2	68.1	92.2	63.3	71.2	65.2	74.7
66.0	64.2	72.7	71.3	71.7	104.4	69.8	114.8	60.4	65.1	68.8	131.2	68.3	63.7
81.0	86.7	78.0	75.2	72.0	530.4	79.0	96.2	88.9	78.9	80.1	63.2	64.2	89.0
59.5	67.7	60.8	73.9	72.3	60.0	63.5	61.7	70.4	65.4	63.0	70.5	67.3	77.7
423.6	428.7	426.0	436.6	425.6	477.1	428.1	83.7	60.5	64.7	71.7	88.6	121.8	66.4
69.2	78.2	67.8	62.5	62.1	72.6	63.1	71.3	79.2	87.1	82.6	167.6	101.4	68.1
112.3	63.1	67.8	64.6	67.5	159.1	77.0	84.0	91.5	79.2	80.9	81.5	111.4	81.0
63.9	75.7	91.2	66.1	73.4	72.7	65.6	97.0	101.8	88.7	82.2	77.9	82.2	88.3
74.8	67.1	90.7	81.1	72.1	67.3	72.4	73.9	61.9	66.1	122.8	65.3	89.0	66.6
77.2	63.2	95.9	78.5	68.9	66.0	80.3	65.9	67.8	66.3	66.3	64.6	99.8	67.4
86.7	69.1	66.0	153.3	158.3	185.1	76.0	105.8	71.5	71.4	182.6	67.6	65.2	421.3
74.1	61.6	66.3	67.7	63.1	86.0	68.0	114.0	75.1	94.9	62.4	67.5	82.6	68.3
131.9	69.6	79.0	96.7	71.8	138.5	68.4	63.7	117.4	173.0	61.9	64.2	68.9	63.6
59.0	66.6	61.6	56.2	74.6	70.6	66.5	81.0	62.8	82.1	73.8	59.1	193.1	90.7
71.3	65.5	83.5	78.2	66.9	64.1	63.3	56.1	60.4	116.7	84.0	63.0	125.3	69.7
66.3	65.2	65.6	69.6	67.1	66.4	63.6	67.0	65.2	70.1	66.5	75.6	69.3	65.9
66.6	344.7	65.4	73.1	79.1	75.9	75.1	61.8	63.7	59.1	81.5	68.6	111.3	212.5
78.0	77.7	67.6	68.2	70.6	78.5	70.3	166.4	68.0	78.5	78.8	73.2	80.6	73.0
104.3	285.7	63.4	145.0	64.7	60.5	85.7	254.8	73.9	67.7	83.5	65.3	68.4	83.2
95.2	76.9	76.9	61.5	71.4	267.2	70.1	72.3	75.5	63.3	78.4	107.6	67.6	73.4
63.8	60.6	100.0	92.3	74.0	65.2	63.7	80.6	78.4	72.2	79.3	72.0	70.5	2427.5
78.8	70.9	75.0	86.4	65.7	65.2	69.7	60.0	74.1	79.6	158.4	66.8	65.0	71.0
73.6	100.5	71.1	70.5	71.8	68.7	94.6	66.8	68.3	69.6	54.2	66.1	93.8	67.6
65.4	74.1	68.2	67.5	78.0	69.7	95.3	64.4	69.1	71.1	61.5	85.8	62.9	68.0
71.8	195.7	71.8	221.6	78.2	64.7	94.1	68.9	81.6	68.4	69.5	67.6	62.3	68.3
72.1	79.7	84.0	73.0	70.8	65.7	98.8	98.5	103.0	99.2	132.3	102.6	111.0	219.1
73.2	81.6	60.6	87.1	67.9	84.8	77.6	73.6	105.6	72.9	71.4	141.0	75.2	72.9
66.0	101.1	153.5	66.1	89.4	81.8	63.4	68.8	60.4	69.8	111.1	64.7	196.0	68.4
72.0	69.6	68.0	72.8	74.3	72.7	74.1	128.6	99.9	170.0	106.4	95.7	106.8	102.8
62.5	55.8	224.0	67.1	66.1	186.4	77.7	84.4	75.3	73.7	95.3	76.7	107.1	84.9
65.8	68.2	65.2	68.6	65.3	62.5	56.7	73.8	97.4	65.9	70.3	68.2	70.4	68.4
78.0	243.7	64.3	71.6	75.1	67.7	58.1	140.3	75.9	71.4	75.8	116.0	116.1	85.6
68.1	74.0	70.3	66.0	101.6	73.1	83.7	68.1	73.6	62.7	79.6	65.0	102.2	79.7
104.2	101.7	99.2	116.1	104.0	129.6	232.4	62.7	75.0	154.2	63.4	96.9	86.0	81.0
69.3	69.5	76.0	71.0	82.3	67.4	72.4	65.7	68.0	70.5	69.9	73.5	179.1	70.8
76.8	70.3	76.6	77.1	68.9	66.3	65.3	66.9	75.8	70.3	85.3	58.1	85.6	72.7
103.9	73.9	88.9	71.5	170.9	63.8	68.6	76.3	75.2	69.9	71.7	105.5	108.5	70.7
125.9	108.5	119.3	106.9	114.6	108.4	123.0	68.9	69.3	96.0	88.3	83.5	72.4	68.4
64.7	91.2	69.5	80.1	65.7	62.7	66.5	63.4	65.8	162.4	83.0	64.6	74.8	62.7
69.3	82.6	89.6	65.7	89.0	60.6	59.7	64.0	71.8	65.5	72.5	295.1	67.4	62.6
84.1	81.2	93.5	84.4	83.1	84.9	87.4	80.7	77.6	111.7	85.6	251.4	86.1	134.3
70.1	99.6	72.9	82.9	71.7	71.7	71.2	58.6	62.2	65.4	64.2	64.3	73.4	72.5
67.4	99.6	83.0	83.4	228.6	71.5	73.7	80.3	75.7	64.9	59.4	61.8	73.6	90.1
132.3	202.2	128.2	99.2	108.6	107.6	92.9	148.0	97.3	69.1	71.2	68.4	73.2	70.1
283.0	58.2	60.9	63.3	63.5	54.4	192.3	67.5	98.9	63.2	79.4	68.4	77.2	71.1
66.1	83.9	64.1	70.6	59.4	61.7	85.6	82.8	70.0	72.3	77.1	357.3	66.7	70.1
76.4	68.3	69.0	119.8	74.3	70.0	70.9	67.1	65.4	72.3	74.7	71.3	107.4	76.3
82.4	95.9	65.6	79.6	65.2	67.9	77.5	85.2	74.4	103.6	78.2	71.6	76.9	120.1
65.4	123.2	78.9	81.7	68.6	64.6	64.7	83.7	68.2	78.7	67.2	64.6	113.3	73.4
78.8	107.8	66.3	105.6	100.0	67.6	70.0	67.0	118.1	72.7	88.2	54.7	70.1	72.3
64.8	74.7	68.4	104.6	73.5	68.5	70.9	99.5	75.4	61.7	61.6	142.1	106.3	78.4
61.3	76.0	63.3	248.6	68.9	84.4	61.8	137.8	88.2	79.0	101.5	77.0	81.8	84.0
78.8	70.8	98.0	71.0	74.6	65.9	74.8	60.8	67.2	67.1	71.0	68.0	78.8	63.5
72.5	93.4	82.1	79.8	64.0	71.1	93.6	124.1	120.5	85.8	117.7	100.3	110.5	93.7
73.2	77.2	59.7	80.8	63.5	103.4	64.5	67.7	107.8	84.6	68.8	71.3	71.8	75.1
148.8	66.3	73.5	147.3	71.5	98.0	62.5	84.8	66.5	66.4	78.0	82.2	66.0	72.4
63.1	70.6	97.8	69.5	65.3	67.3	102.2	90.4	779.4	75.3	119.4	78.9	74.0	123.1
99.5	77.4	79.0	73.3	81.8	94.7	80.4	81.3	81.6	79.9	88.3	76.5	67.4	74.6
71.4	135.4	67.0	65.9	70.1	67.2	69.4	71.6	78.0	64.5	75.6	69.0	170.5	71.0
71.5	69.1	99.0	65.6	67.9	99.7	71.2	85.0	73.4	64.3	116.0	69.7	63.2	71.3
184.8	72.4	67.3	65.7	73.1	98.0	83.5	178.4	59.7	63.2	66.4	67.6	66.8	65.8
176.2	59.8	78.9	125.3	65.3	63.5	71.7	71.1	64.4	61.2	76.6	68.9	68.3	82.5
65.1	79.4	71.0	79.0	80.3	61.7	64.5	67.6	115.7	64.4	56.1	96.2	78.3	67.8
61.7	71.0	69.8	83.5	175.0	122.5	79.0	72.3	68.0	75.3	68.1	60.8	115.3	146.0
98.8	122.4	97.4	106.5	186.3	114.5	104.8	94.4	133.5	84.5	95.6	68.2	70.3	74.3
73.4	65.5	75.3	72.7	64.6	66.2	73.0	77.0	65.0	59.8	69.4	69.7	65.1	61.5
75.0	70.1	70.8	88.6	98.1	76.7	70.9	59.1	76.0	64.2	74.2	93.9	56.9	77.8
133.3	126.4	127.4	118.8	115.0	148.9	151.9	318.4	64.9	65.2	80.1	82.2	64.6	70.1
68.2	66.7	78.2	71.7	187.1	82.6	78.3	66.3	66.0	85.4	78.8	72.4	169.9	73.5
66.8	68.8	124.1	70.6	70.5	66.7	65.9	75.2	95.2	82.9	88.5	112.8	73.6	88.4
111.5	68.9	73.8	65.8	156.4	65.8	66.6	71.0	65.9	64.5	70.9	63.9	59.8	66.1

Continued on next page

Table B.6 – *Continued from previous page*

78.1	66.4	72.8	66.8	70.9	66.6	76.7	75.5	69.8	67.0	67.3	70.7	76.8	74.3
206.0	75.5	77.3	88.2	88.5	78.9	89.4	73.9	66.7	85.2	63.7	64.2	64.9	62.6
70.7	87.7	177.1	73.8	68.6	71.0	77.8	77.8	91.8	76.4	105.2	73.1	75.5	77.1
63.5	71.7	67.3	84.5	82.0	75.1	65.0	89.9	107.3	83.7	80.0	190.7	85.9	74.9
82.0	118.3	74.6	83.5	69.3	75.0	75.9	61.5	61.5	72.4	76.0	117.7	74.0	80.5
65.1	67.2	83.3	68.3	67.1	63.6	67.6	71.9	67.0	119.7	69.1	72.2	73.6	68.4
83.5	72.3	81.0	65.5	76.9	162.1	85.0	70.6	74.4	75.6	76.9	116.3	188.8	66.2
94.6	81.1	70.4	71.4	67.6	58.7	83.2	88.2	68.5	77.6	86.2	68.3	86.3	70.3
73.1	73.5	82.6	73.2	86.7	58.4	75.7	75.1	71.3	65.3	76.0	69.2	65.5	81.0
72.9	69.4	66.0	327.2	68.3	67.4	59.9	134.6	73.1	75.1	74.2	79.0	203.5	83.5
65.6	61.8	79.3	63.8	66.2	69.8	63.9	79.0	83.6	89.8	68.9	70.9	68.7	86.4
79.0	81.5	110.0	110.6	86.1	318.4	84.0	65.0	68.2	67.2	68.2	68.7	116.7	64.9
2529.0	79.5	65.2	70.4	79.1	52.3	231.0	78.5	77.5	78.2	85.8	75.5	75.7	80.8
75.1	66.0	65.2	61.4	73.4	74.8	88.4	106.3	70.8	78.7	82.1	74.7	65.3	65.7
95.9	69.9	90.2	71.4	70.5	64.7	70.5	71.8	71.9	68.8	67.7	100.1	63.8	68.6

Bibliography

[1] F.R. Bach. Consistency of trace norm minimization. *J. Mach. Learn. Res.*, 9:1019–1048, June 2008.

[2] M. Beiglböck, P. Henry-Labordére, and F. Penkner. Model-independent bounds for option prices: a mass transport approach. *Finance and Stochastics*, 17(3):477–501, 2013.

[3] F. Bellini and V. Bignozzi. Elicitable risk measures. *Quant. Finance*, 15(5):725–733, 2015.

[4] D. Bertsimas and I. Popescu. On the relation between option and stock prices: A convex optimization approach, 2000.

[5] D. Bertsimas and I. Popescu. Optimal inequalities in probability theory: A convex optimization approach. *SIAM Journal on Optimization*, 15:780–804, 2000.

[6] J. F. Bonnans and A. Shapiro. *Perturbation Analysis of Optimization Problems.* Springer, 2000.

[7] J.F. Bonnans and X. Tan. A model-free no-arbitrage price bound for variance options. *Applied Mathematics and Optimization*, 68(1):43–73, 2013.

[8] L.M. Bregman. The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming. {*USSR*} *Computational Mathematics and Mathematical Physics*, 7(3):200 – 217, 1967.

[9] H. Brown, D. Hobson, and L.C.G. Rogers. The maximum maximum of a martingale constrained by an intermediate law. *Probability Theory and Related Fields*, 119(4):558–578, 2001.

[10] H. Brown, D. Hobson, and L.C.G. Rogers. Robust hedging of barrier options. *Mathematical Finance*, 11(3):285–314, 2001.

[11] P. Carr and R. Lee. Hedging variance options on continuous semimartingales. *Finance and Stochastics*, 14(2):179–207, 2010.

[12] V. Chvátal. *Linear Programming*. Series of books in the mathematical sciences. W.H. Freeman, 1983.

[13] A.M.G. Cox and J. Obłój. Robust hedging of double touch barrier options. *SIAM Journal on Financial Mathematics*, 2(1):141–182, 2011.

[14] A.M.G. Cox and J. Obłój. Robust pricing and hedging of double no-touch options. *Finance and Stochastics*, 15(3):573–605, 2011.

[15] A.M.G. Cox and J. Wang. Root's barrier: Construction, optimality and applications to variance options. *Annals of Applied Probability*, 23(3):859–894, 2013.

[16] J.C. Cox and S.A. Ross. The valuation of options for alternative stochastic processes. *Journal of Financial Economics*, 3(1):145 – 166, 1976.

[17] A. d'Aspremont and L. El Ghaoui. Static arbitrage bounds on basket option prices. *Mathematical Programming*, 106(3):467–489, 2006.

[18] S. Daum and R. Werner. A novel feasible discretization method for linear semi-infinite programming applied to basket option pricing. *Optimization*, 60(10-11):1379–1398, 2011.

[19] M. Davis, J. Obłój, and V. Raval. Arbitrage bounds for prices of weighted variance swaps. *Mathematical Finance*, 24(4):821–854, 2014.

[20] M. Degen, P. Embrechts, and D. D. Lambrigger. The quantitative modeling of operational risk: between g-and-h and evt. *ASTIN Bulletin*, 38, 2007.

[21] K. Dutta and J. Perry. A tale of tails: an empirical analysis of loss distribution models for estimating operational risk capital. Working Papers 06-13, Federal Reserve Bank of Boston, 2006.

[22] P. Embrechts, A. Hőing, and G. Puccetti. Worst var scenarios. *Insurance: Mathematics and Economics*, 37(1):115 – 134, 2005.

[23] P. Embrechts and G. Puccetti. Bounds for functions of dependent risks. *Finance and Stochastics*, 10(3):341–352, 2006.

[24] P. Embrechts and G. Puccetti. Bounds for functions of multivariate risks. *Journal of Multivariate Analysis*, 97(2):526 – 547, 2006.

[25] P. Embrechts, G. Puccetti, and L. Rüschendorf. Model uncertainty and var aggregation. 37(8):2750 – 2764, 2013.

[26] M. Fazel. Matrix rank minimisation with applications. *Ph.D. thesis, Stanford University*, 2002.

[27] J.R. Gabriel and O. Hernández-Lerma. Strong duality of the general capacity problem in metric spaces. *Mathematical Methods of Operations Research*, 53(1):25–34, 2001.

[28] B. D. Grundy. Option prices and the underlying asset's return distribution. *The Journal of Finance*, 46(3):1045–1069, 1991.

[29] J. Gustafsson, J.P. Nielsen, P. Pritchard, and D. Roberts. Quantifying operational risk guided by kernel smoothing and continuous credibility. *SSRN: http://ssrn.com/abstract=627342, http://dx.doi.org/10.2139/ssrn.627342*, 2005.

[30] J.M. Harrison and D.M. Kreps. Martingales and arbitrage in multiperiod securities markets. *Journal of Economic Theory*, 20(3):381 – 408, 1979.

[31] R. Hauser, S. Shahverdyan, and P. Embrechts. Duality in risk aggregation. 99:375–392, 2015.

[32] P. Henry-Labordére. Automated option pricing: numerical methods. *International Journal of Theoretical and Applied Finance*, 16(08):1350042, 2013.

[33] P. Henry-Labordére, P., J. Obłój, P. Spoida, and N. Touzi. Maximum maximum of martingales given marginals. *SSRN: http://ssrn.com/abstract=2031461, http://dx.doi.org/10.2139/ssrn.2031461*, 2013.

[34] D. Hobson and M. Klimmek. Model-independent hedging strategies for variance swaps. *Finance and Stochastics*, 16(4):611–649, 2012.

[35] D. Hobson, P. Laurence, and T. Wang. Static-arbitrage optimal subreplicating strategies for basket options. *Insurance: Mathematics and Economics*, 37(3):553 – 572, 2005.

[36] D. Hobson, P. Laurence, and T. Wang. Static-arbitrage upper bounds for the prices of basket options. *Quantitative Finance*, 5(4):329–342, 2005.

[37] D. Hobson and A. Neuberger. Robust bounds for forward start options. *Mathematical Finance*, 22(1):31–56, 2012.

[38] R. B. Holmes. *Geometric Functional Analysis and its Applications*. Springer, New York, 1975.

[39] K. Isii. The extrema of probability determined by generalized moments (i) bounded random variables. *Annals of the Institute of Statistical Mathematics*, 12(2):119–134, 1960.

[40] S. Ito, S.Y Wu, T.J Shiu, and K.L Teo. A numerical approach to infinite-dimensional linear programming in l_1 spaces. *Journal of Industrial and Management Optimization*, 6(1):15–28, 2010.

[41] J. H. B. Kemperman. The general moment problem, a geometric approach. *The Annals of Mathematical Statistics*, 39(1):93–122, 1968.

[42] H.C. Lai and S.Y. Wu. Extremal points and optimal solutions for general capacity problems. *Mathematical Programming*, 54(1-3):87–113, 1992.

[43] J. B. Lasserre. Global optimization with polynomials and the problem of moments. *SIAM Journal on Optimization*, 11:796–817, 2001.

[44] Andrew W. Lo. Semi-parametric upper bounds for option prices and expected payoffs. *Journal of Financial Economics*, 19(2):373 – 387, 1987.

[45] S. Ma, D. Goldfarb, and L. Chen. Fixed point and bregman iterative methods for matrix rank minimization. *Mathematical Programming*, 128(1-2):321–353, 2011.

[46] M. Makarov. Extreme value theory and high quantile convergence. *Journal of Operational Risk*, 1(2):51–57, 2006.

[47] A. J. McNeil, R. Frey, and P. Embrechts. *Quantitative Risk Management*. Princeton University Press, Princeton, NJ, 2006.

[48] G. Mignola and R. Ugoccioni. Tests of extreme value theory. *Operational Risk*, 6(10):32–35, 2005.

[49] A.M. Monteiro, R.H. Tütüncü, and L.N. Vicente. Recovering risk-neutral probability density functions from options prices using cubic splines and ensuring nonnegativity. *European Journal of Operational Research*, 187(2):525 – 542, 2008.

[50] R. N. Nelsen. *An Introduction to Copulas (Springer Series in Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[51] Y. Nesterov. A method of solving a convex programming problem with convergence rate $o(1/k^2)$. *Soviet Math. Dokl.*, 27(2), 1983.

[52] Y. Nesterov. Smooth minimization of non-smooth functions. *Mathematical Programming*, 103(1):127–152, 2005.

[53] J. Nešlehová, P. Embrechts, and V. Chavez-Demoulin. Infinite mean models and LDA for operational risk. *Journal of Operational Risk*, 1(1):3–25, 2006.

[54] P. A. Parrilo. Semidefinite programming relaxations for semialgebraic problems. *Mathematical Programming*, 96(2):293–320, 2003.

[55] J. Peña, J. Vera, and L.F. Zuluaga. Static-arbitrage lower bounds on the prices of basket options via linear programming. *Quantitative Finance*, 10(8):819–827, 2010.

[56] J. Peña, J. Vera, and L.F. Zuluaga. Computing arbitrage upper bounds on basket options in the presence of bidask spreads. *European Journal of Operational Research*, 222(2):369 – 376, 2012.

[57] I. Pólik and T. Terlaky. A survey of the s-lemma. *SIAM Review*, 49:371–418.

[58] G. Puccetti and L. RüSchendorf. Computation of sharp bounds on the distribution of a function of dependent risks. *J. Comput. Appl. Math.*, 236(7):1833–1840, 2012.

[59] G. Puccetti and L. RüSchendorf. Sharp bounds for sums of dependent risks. *J. Comput. Appl. Math.*, 50(1):42–53, 2013.

[60] D. Ramachandran and L. Rüschendorf. A general duality theorem for marginal problems. *Probability Theory and Related Fields*, 101(3):311–319, 1995.

[61] S. M. Robinson. Regularity and stability for convex multivalued functions. *Mathematics of Operations Research*, 1(2):130–143, 1976.

[62] R. T. Rockafellar. *Convex Analysis.* Princeton University Press, Princeton, NJ, USA, 1970.

[63] R. T. Rockafellar. Conjugate duality and optimization. *Regional Conference Series in Applied Mathematics, SIAM*, 1974.

[64] M. Rubinstein. Implied binomial trees. *The Journal of Finance*, 49(3):771–818, 1994.

[65] P. Shah. No-arbitrage bounds on american put options with a single maturity. *SSRN:http://hdl.handle.net/1721.1/36232*, 2006.

[66] A. Shapiro. On duality theory of conic linear problems. In M.. Goberna and M. López, editors, *Semi-Infinite Programming*, volume 57 of *Nonconvex Optimization and Its Applications*, pages 135–165. Springer US, 2001.

[67] F. Stebegg. Model-independent pricing of asian options via optimal martingale transport. 2014.

[68] R. Wang, L. Peng, and J. Yang. Bounds for the sum of dependent risks and worst value-at-risk with monotone marginal densities. *Finance Stoch.*, 17(2):395–417, 2013.

[69] R. C. Williamson and T. Downs. Probabilistic arithmetic. I. numerical methods for calculating convolutions and dependency bounds. *Int. J. Approx. Reasoning*, 4(2):89–158, 1990.