# An empirical analysis of scenario generation methods for stochastic optimization

Nils Löhndorf

WU Vienna University of Economics and Business
Department of Information Systems and Operations
Welthandelsplatz 1, 1020 Vienna, Austria
nils.loehndorf@wu.ac.at, +43 31336 5629

May 8, 2016

**Abstract**

This work presents an empirical analysis of popular scenario generation methods for stochastic optimization, including quasi-Monte Carlo, moment matching, and methods based on probability metrics, as well as a new method referred to as *Voronoi cell sampling*. Solution quality is assessed by measuring the error that arises from using scenarios to solve a multi-dimensional newsvendor problem, for which analytical solutions are available. In addition to the expected value, the work also studies scenario quality when minimizing the expected shortfall using the conditional value-at-risk. To quickly solve problems with millions of random parameters, a reformulation of the risk-averse newsvendor problem is proposed which can be solved via Benders decomposition. The empirical analysis identifies *Voronoi cell sampling* as the method that provides the lowest errors, with particularly good results for heavy-tailed distributions. A controversial finding concerns evidence for the ineffectiveness of widely used methods based on minimizing probability metrics under high-dimensional randomness.

*Keywords:* stochastic optimization, sample average approximation, scenario generation, vector quantization, probability metrics, moment matching, Monte Carlo methods, conditional value-at-risk

## 1 Introduction

A wide range of real-world problems that occur in finance, industrial engineering, operations, or marketing involve decision-making under uncertainty. If a statistical model can be used to describe

this uncertainty, the decision problem can be modeled as a stochastic optimization problem. The most widely used technique to solve real-world stochastic optimization problems is sample average approximation, where the probability distribution is approximated by a set of discrete scenarios (Birge and Louveaux, 1997; Shapiro, 2003). Since the complexity of an optimization problem scales with the number of scenarios, a lot of work on stochastic optimization is dedicated to techniques that reduce the number of scenarios while retaining the quality of the stochastic solution. The goal of techniques for scenarios generation is hereby to select a set of scenarios which minimizes the approximation error.

Methods for generating a reduced set of scenarios from univariate distributions are well established and typically referred to as variance reduction techniques (Higle, 1998; Shapiro, 2003). Adapting these techniques for the more general multivariate case, however, is not straightforward. Let us briefly summarize what can be considered as state-of-the art to generate scenarios of multivariate random variables for sample average approximation: quasi-Monte Carlo methods, methods based on probability metrics, and moment matching.

A popular class of methods for Monte Carlo sampling in higher dimensions are quasi-Monte Carlo methods which have their roots in number theory (Niederreiter, 1992). Quasi-Monte Carlo methods rely on so called low-discrepancy sequences which produce a sequence of vectors that cover the unit hypercube as uniformly as possible, e.g., Sobol sequences. If combined with an adequate transformation, these sequences can be treated just like pseudo-random numbers and as such can be used to sample random number sequences from a large number of multivariate distributions. See Glasserman (2004) for an overview.

The idea behind scenario reduction using probability metrics is to compute the closest approximation of the probability distribution by a discrete distribution with smaller support. A probability metric serves as the objective criterion and can be related to the error from implementing the optimal solution of a stochastic optimization problem using sample average approximation (Pflug, 2001; Heitsch and Römisch, 2003; Pennanen and Koivu, 2005; Heitsch and Römisch, 2007). A greedy algorithm that reduces a time series sample to a scenario tree is proposed in Heitsch and Römisch (2003).

If the scenario tree has only two stages, the problem of finding the closest discrete approximation, can be modeled as optimal quantization problem. Quantization has its roots in signal processing, where a finite number of quantizers is needed to transmit a stationary signal (Gersho and Gray, 1992). Quantizers are real-valued vectors comparable to what a decision-maker would refer to as a scenario if the signal was a sample of prices or demands. An optimal quantization of the signal sets the location of the quantizers such that the average distortion is minimized, which is equivalent to finding the closest discrete approximation. Optimal quantization has reemerged as a method for

numerical integration, mostly driven by the work of Pagès (1998) and Graf and Luschgy (2000). See Pagès and Printems (2008) for a review of optimal quantization applied in finance.

Another successful method to generate scenarios for stochastic optimization is moment matching. The original idea is due to Fleishman (1978) for the univariate which has been adapted by Høyland et al. (2003) who propose a heuristic algorithm for the multivariate case. Mehrotra and Papp (2013) use optimization to find a set of moment matching scenarios. In contrast to optimal quantization, where a random sample from the probability distribution is drawn to generate scenarios, moment matching tries to define another discrete distribution that exhibits the same first four moments as the original distribution.

In the extant literature, only few studies compare the performance of different scenario generation techniques across multiple different optimization problems, with respect to solution quality of sample average approximation. Studies with a focus on Monte Carlo methods are Koivu (2005), Linderoth et al. (2006), Freimer et al. (2012), and Homem-de Mello et al. (2011). A notable exception are Dempster et al. (2011) who compare the performance of Monte Carlo methods, moment matching, and methods based on probability metrics on a specific asset liability management problem.

The goal of this research is study the error that arises from using a small set of scenarios to approximate a continuous multivariate distribution with the objective to numerically solve a multidimensional newsvendor problem, for which analytical solutions are still available. The computational study compares the current state-of-the-art, including quasi-Monte Carlo, moment matching, as well as optimal quantization, and considers a range of distributions including normal, uniform, log-normal, as well as a heavy-tailed t-distribution.

Discovered as a by-product of initial experimental work, the work furthermore introduces *Voronoi cell sampling* as an alternative method for scenario generation. Voronoi cell sampling integrates stratified sampling with probability metrics in a simple yet effective way and performs favorably compared to the other methods.

Since the objective function of the profit maximizing newsvendor is separable, minimization of the expected shortfall is also considered, which is modeled using the conditional value-at-risk formulation (CVaR) of Rockafellar and Uryasev (2002). To measure the error of approximating the continuous distribution by a small set of scenarios, a tailored solution approach based on Benders decomposition is developed. The method is capable of approximating a solution to the CVaR newsvendor problem with several million random parameters which then serves as proxy of the true solution.

The paper is organized as follows. Section 2 gives a brief introduction to stochastic optimization in general, describes existing approaches to generate scenarios for stochastic optimization, and

introduces Voronoi cell sampling as a new scenario reduction method. Section 3 presents the newsvendor models that are used to evaluate the sample average approximation error. Section 4 outlines the experimental design and discusses the results. Section 5 concludes with a summary and gives implications for future work.

# 2 Methodological Background

## 2.1 Sample Average Approximation

Denote $x$ as a decision variable defined over the feasible set $X \subseteq \mathbb{R}$, $z$ as a $d$-dimensional random realization of random variable $Z$ that is defined by the distribution function $F : \mathbb{R}^d \to [0, 1]$, and $c(x, z)$ as the cost function. The stochastic optimization problem is to minimize

$$C(x) = \int c(x, z) dF(z) \tag{1}$$

by choosing an optimal decision

$$x^* \in \text{argmin}_{x \in X} C(x). \tag{2}$$

With the exception of very simple well-behaved problems, solutions are typically calculated numerically by either drawing a sample from $F$ or by approximating $F$ by a discrete distribution $\hat{F}$. Using $\hat{F}$ in place of $F$ yields the problem of minimizing

$$\hat{C}(x) = \sum_{\hat{z} \in \hat{F}} \hat{p}(z) c(x, \hat{z}), \tag{3}$$

where $\hat{p}(z)$ is the probability of the mass points of $\hat{F}$.

This approach is referred to by different names in different communities, e.g., numerical integration (Judd, 1998), Monte Carlo methods (Glasserman, 2004), or sample average approximation (Shapiro, 2013). Since the optimization literature mostly refers to this method by the latter name, the term sample average approximation (SAA) will be used from hereon.

## 2.2 Monte Carlo and Quasi-Monte Carlo

A common approach to solve a stochastic optimization problem using SAA is based on Monte Carlo sampling. To use Monte Carlo sampling for SAA, we generate a set of $M$ uniformly distributed, pseudo-random realizations $u_1 \ldots, u_M$ with $u_i \in [0, 1]^d$, and then construct a sample from $F$ by an appropriate transformation $U \to Z$ (Glasserman, 2004).

If we view $z_1 \ldots, z_M$ as a sample of random realizations from the same distribution as $F$, the

SAA is given by

$$\hat{C}(x) = \frac{1}{M}\sum_{i=1}^{M} c(x, z_i), \tag{4}$$

with the variance of the estimate given by

$$\hat{\sigma}^2_{MC} = \frac{1}{M}\mathbb{V}ar[c(x, Z)]. \tag{5}$$

Although the Monte Carlo estimate $\hat{C}_{MC}(x)$ converges to $C(x)$ with probability one as $M \to \infty$, it is desirable to reduce the variance of the estimate more quickly to speed up convergence of the error bounds (Shapiro, 2003). In this respect, a number of variance reduction techniques have been proposed for Monte Carlo sampling. See Glasserman (2004) or Shapiro (2003) for an overview.

A popular approach to reduce the variance for small sample sizes is to use so-called low-discrepancy sequences sequences that cover the unit hypercube as uniformly as possible (Niederreiter, 1992). It can be shown that using such low-discrepancy sequences instead of pseudo-random numbers speeds up the rate of convergence of the Monte Carlo estimate. Since these sequences are also referred to as quasi-random numbers, the sampling method is typically referred to as quasi-Monte Carlo. See Koivu (2005) for an analysis of using quasi-Monte Carlo sampling for SAA.

## 2.3 Probability Metrics

Another approach to reduce the number of scenarios is to address the approximation error directly. Let us therefore define the approximation error of SAA as the absolute deviation in objective values

$$e(C, \hat{C}) = \left| \max_x C(x) - C(\text{argmin}_x \hat{C}(x))) \right|. \tag{6}$$

Since the approximation error is difficult if not impossible to compute in practical applications, theoretical results on the stability of stochastic programs provide an upper bound of the error based on the theory of probability metrics.

Let us introduce the following metric. Denote $L_r(c)$ as the cost function's Lipschitz constant of order $r$,

$$L_r(c(x)) = \inf\{L : |c(x, z_1) - c(x, z_2)| \leq L|z_1 - z_2| \max(1, |z_1|^{r-1}, |z_2|^{r-1}) \ \forall \ z_1, z_2\} \leq \bar{L}_r, \tag{7}$$

with $\bar{L}_r$ as upper bound. This gives us a measure of how fast the cost $c(\cdot, z)$ changes in $z$. Then,

denote $d_r$ as the Wasserstein distance between $F$ and $\hat{F}$, such that

$$d_r(F, \hat{F}) = \left( \inf_{g \in M(F,\hat{F})} \left\{ \int ||z - \hat{z}||^r \, dg(z, \hat{z}) \right\} \right)^{\frac{1}{r}}, \tag{8}$$

where the infimum is taken over the space of probability distributions, $M(F, \hat{F})$, on the product space $\mathbb{R}^d \times \mathbb{R}^d$ with marginal distributions $F$ and $\hat{F}$ that satisfy (7).

In Pflug (2001), it is shown that the approximation error is bounded by the product of $\bar{L}_r$ and $d_r$, i.e.,

$$e(C, \hat{C}) \leq 2\bar{L}_r d_r(F, \hat{F}). \tag{9}$$

Since $\bar{L}_r$ is constant for a given $r$ and $c$, the results suggests that we can minimize the approximation error by minimizing the distance between the two distributions. Note that the bound is not unique for a specific problem, but holds for the general class of problems for which the objective function fulfills the Lipschitz condition.

Since the integral in (8) makes the distance computationally inaccessible, typically the distance between two discrete distributions is measured. Denote $M$ as the number of mass points of the reduced distribution and $N$ as the number of mass points of the true distribution, then the distance between the two distributions is given by

$$d_r(F, \hat{F}) = \left( \min_{y_{i,j} \in [0,1]} \left\{ \sum_{i=1}^{N} \sum_{j=1}^{M} y_{i,j} ||z_i - \hat{z_j}||^r \,\middle|\, \sum_{j=1}^{M} y_{i,j} = p_i, \sum_{i=1}^{N} y_{i,j} = q_j \right\} \right)^{\frac{1}{r}}, \tag{10}$$

with $p_i$, $q_i$ being the probabilities of the respective mass points. The optimal value of (10) is the optimal value of a minimum cost flow problem (Heitsch and Römisch, 2003). Assuming a (discrete) uniform distribution for $Z$, the problem can be be solved by assigning each $z_i$ the nearest neighbor in the outcome space of $\hat{Z}$.

Hence, by finding the discrete distribution $\hat{F}$ that is closest to $F$, we can minimize the upper bound, which minimizes the approximation error of the SAA if $\bar{L}_r$ is sufficiently small. For this reason, a probability metric can be used as objective criterion to search for an optimal discrete distribution, by making the tacit assumption that the minimizer of the upper bound also minimizes the approximation error.

### 2.3.1 Optimal Quantization

A computational approach to address the problem of minimizing the upper bound is optimal quantization. The underlying optimization problem is to find a set of mass point such that the average distance between the sample distribution and the discrete distribution described through

the mass points is minimized. Assuming a set of equiprobable sample vectors $z_1, \ldots, z_N$ coming from a continuous distribution, the corresponding optimization problem becomes

$$\min_{\hat{z}_1,\ldots,\hat{z}_M} \left\{ \left( \min_{y_{i,j}\in\{0,1\}, q_j \geq 0} \left\{ \sum_{i=1}^{N}\sum_{j=1}^{M} y_{i,j}||z_i - \hat{z}_j||^r \,\middle|\, \sum_{j=1}^{M} y_{i,j} = 1, \sum_{i=1}^{N} y_{i,j} = q_j N \right\} \right)^{\frac{1}{r}} \right\}, \tag{11}$$

which yields optimal mass points $(\hat{z}_1^*, q_1^*), \ldots, (\hat{z}_M^*, q_M^*)$ (Heitsch and Römisch, 2003).

The optimization problem in (11) is a non-convex optimization problem that belongs to the class of $\mathcal{NP}$-hard problems, so that typically local optimizers are used to tackle the problem. A well-known strategy to find a local optimum is to start with an initial guess of $\hat{z}$, e.g. by randomly drawing $M$ elements from $z_1, \ldots, z_N$, and iterate between finding optimal $q_1^*, \ldots, q_M^*$ by solving

$$\min_{y_{i,j}\in\{0,1\}, q_j \geq 0} \left\{ \sum_{i=1}^{N}\sum_{j=1}^{M} y_{i,j}||z_i - \hat{z}_j||^r \,\middle|\, \sum_{j=1}^{M} y_{i,j} = 1, \sum_{i=1}^{N} y_{i,j} = q_j N \right\}, \tag{12}$$

for $\hat{z}_j$ fixed, and then finding optimal $\hat{z}_1^*, \ldots, \hat{z}_M^*$ by solving

$$\min_{\hat{z}_1,\ldots,\hat{z}_M} \left\{ \sum_{i=1}^{N}\sum_{j=1}^{M} y_{i,j}||z_i - \hat{z}_j||^r \right\}, \tag{13}$$

for $y_{ij}$ fixed.

If the distance is computed using the squared Euclidean distance ($r = 2$), this strategy corresponds to Lloyd's method I, which is better known as the k-means clustering algorithm (MacQueen, 1967). In case of k-means, minimization of (12) simplifies to setting $y_{ij} = 1$ if $j = \text{argmin}_j\{||z_i - \hat{z}_j||_2\}$ and minimization of (13) simplifies to computing $z_j$ as mean (centroid) of all $z_i$ where $y_{ij} = 1$. (Since the quadratic case is the one that is computationally most appealing, any reference to quantization from hereon assumes that $r = 2$.)

An interesting property of using quantizers as scenarios is that the corresponding optimal objective value is a lower bound, which is supported by the following proposition.

**Proposition 2.1** *If* $\hat{C}(x) = \sum_{j=1}^{M} q_j c(x, \hat{z}_j)$ *and* $c(x, z)$ *convex in* $z$, *then* $\min_x \hat{C}(x) \leq \min_x C(x)$.

**Proof** Denote $\Omega_j$ as the $j$-th partition of the outcome space that contains all points closest to quantizer $\hat{z}_j$, so that

$$\begin{aligned} &\Omega_j \equiv \Omega_j(\hat{z}_j) = \left\{ z_i : j = \text{argmin}_k\{||z_i - \hat{z}_k||^2\}, i = 1, \ldots, N \right\}, \; j = 1, \ldots, M, \\ &q_j \equiv q_j(\hat{z}_j) = \int_{\Omega_j} f(z)dz, \; j = 1, \ldots, M. \end{aligned} \tag{14}$$

For $r = 2$ the optimal quantizers are the means of their respective partitions, i.e.,

$$\hat{z}_j = \mathbb{E}[z|z \in \Omega_j],\ j = 1, \ldots, M, \tag{15}$$

e.g., see (2.6) in Pagès and Printems (2003). Since partitions, $\Omega_j,\ j = 1, \ldots, M$, are exhaustive and disjoint, i.e., each outcome has only one quantizer that is closest, it follows from the law of total expectation that

$$\sum_{j=1}^{M} q_j \mathbb{E}_z[c(x,z)|z \in \Omega_j] = \mathbb{E}_z[c(x,z)],\ x \in X. \tag{16}$$

If the objective function $c(x,z)$ is convex in $z$, by Jensen's inequality it holds that

$$c(x, \mathbb{E}[z|z \in \Omega_j]) \leq \mathbb{E}_z[c(x,z)|z \in \Omega_j],\ x \in X,\ j \in 1, \ldots, M. \tag{17}$$

With $x^* \in \operatorname{argmin}_{x \in X} C(x)$, it follows that

$$\begin{aligned} \min_x \hat{C}(x) &= \min_x \sum_{j=1}^{M} q_j c(x, \hat{z}_j) \\ &\leq \sum_{j=1}^{M} q_j c(x^*, \hat{z}_j) = \sum_{j=1}^{M} q_j^* c(x^*, \mathbb{E}[z|z \in \Omega_j]) \\ &\leq \sum_{j=1}^{M} q_j^* \mathbb{E}_z[c(x^*,z)|z \in \Omega_j] = \mathbb{E}_z[c(x^*,z)] = \min_x C(x). \ \square \end{aligned} \tag{18}$$

**Remark** The function $c(x,z)$ is convex in $z$, if $z$ enters the right-hand side of a convex minimization problem, or equivalently, if $z$ enters the objective of a concave maximization problem.

### 2.3.2 Competitive Learning

A major drawback of k-means is that that it does not converge to an optimal quantizer, not even locally, unless some form of randomization is used. To reduce the error of the Monte Carlo estimate in each partition, large sample sizes are required, which is computationally expensive as the algorithm scales polynomially in sample size (Bally et al., 2003).

A much more attractive approach for optimal quantization is based on stochastic gradient descent to optimize (11), a solution approach also known as competitive learning (Pagès, 1998).

Denote $(\alpha_n)_{n=1}^N$ as a sequence of stepsizes with $0 \leq \alpha_n \leq 1, n = 1, \ldots, N$. The idea is to draw

random sequences $(z_n)_{n=1}^N$ and $(\hat{z}_j^0)_{j=1}^M$ from the continuous distribution $F$ and recursively evaluate

$$\hat{z}_j^n = \begin{cases} \hat{z}_j^{n-1} + \alpha_n(z_n - \hat{z}_j^{n-1}) & \text{if } j = \text{argmin}_k\{||z_n - \hat{z}_k^{n-1}||^2\}, \\ \hat{z}_j^{n-1} & \text{otherwise,} \end{cases} \tag{19}$$

for $j = 1, \ldots, M,\ n = 1, \ldots, N$. The first condition holds if a quantizer $j$ is found to be closest to $z_n$, in which case the quantizer is moved by an amount $\alpha_n$ in the direction of $z_n$.

In Pagès and Printems (2003), Theorem 2.4, it is shown that if the sequence $(\alpha_n)_{n=1}^N$ satisfies $\sum_{n=1}^\infty \alpha_n = \infty$ and $\sum_{n=1}^\infty \alpha_n^2 < \infty$ and if $F(Z)$ is continuously differentiable, then variables $\hat{z}_1^N, \ldots, \hat{z}_M^N$ converge to local optimizers of (11),

$$\hat{z}_j^N \sim \hat{z}_j^* \text{ as } N \to \infty,\ j = 1, \ldots, M. \tag{20}$$

The probabilities of the mass points can be obtained in a similar fashion, for example, by simultaneously evaluating

$$\hat{q}_j^n = \begin{cases} \frac{n-1}{n}\hat{q}_j^{n-1} + \frac{1}{n} & \text{if } j = \text{argmin}_k\{||z_n - \hat{z}_k^{n-1}||^2\}, \\ \frac{n-1}{n}\hat{q}_j^{n-1} & \text{otherwise,} \end{cases} \tag{21}$$

for $j = 1, \ldots, M,\ n = 1, \ldots, N$, with $\hat{q}_j^0 = 0$.

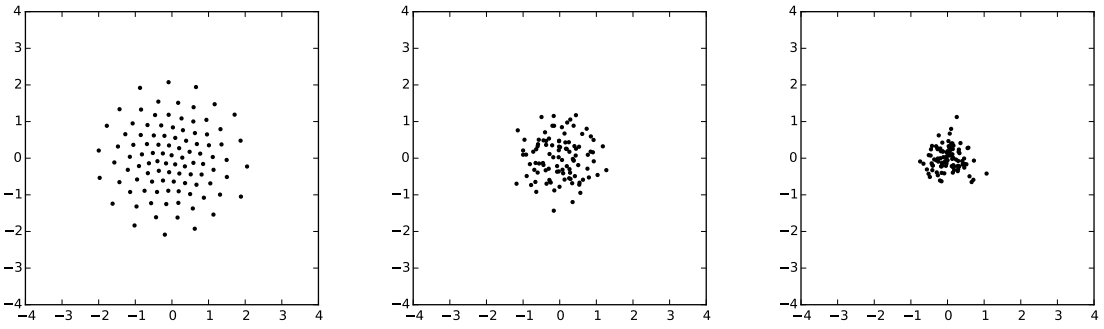
Competitive learning yields a sequential version of the k-means clustering algorithm, but, as opposed to k-means, scales linearly in sample size (Bottou, 1998). The fact that drawing samples from continuous distributions via Monte Carlo simulation is relatively inexpensive turns competitive learning into an attractive computational strategy for optimal quantization.

## 2.4 Voronoi Cell Sampling

A known problem with optimal quantization is that even if $\hat{z}_1^*, ..., \hat{z}_M^*$ are minimizers of (11), the variance of the set of scenarios is smaller than that of the sample distribution (Gersho and Gray, 1992; Hochreiter and Pflug, 2007; Dempster et al., 2011).

Figure 1 shows scenarios obtained by optimal quantization of the multivariate standard normal distribution with 2 (a), 8 (b), and 32 (c) dimensions. For 8 and 32 dimensions only the first two dimensions are plotted. The figure illustrates that the variance of the marginal distributions decreases in the dimensionality of the joint distribution.

The contraction of scenarios towards the mean can be mitigated by combining optimal quantization with stratified sampling (Corlay and Pagès, 2010). The purpose of stratified sampling is to



(a) $d = 2$ (b) $d = 8$ (c) $d = 32$

Figure 1: Optimal quantization of a multivariate normal distribution

reduce the variance of the Monte Carlo estimate by partitioning the outcome space into a number of non-intersecting regions called strata. The optimal quantizers provide such a partitioning as a by-product. Since these partitions are also known as Voronoi cells (Graf and Luschgy, 2000), let us refer to this method as *Voronoi cell sampling*.

Denote $p_j$ as the proportion of samples allocated to partition $\Omega_j$. Then, the variance of the Monte Carlo estimate using stratified sampling is given by

$$\hat{\sigma}^2_{SS} = \frac{1}{M} \sum_{j=1}^{M} \frac{p_j}{q_j} q_j \mathbb{V}ar_z[c(x,z)|z \in \Omega_j]. \tag{22}$$

With proportional allocation of samples to partitions, i.e., $p_j = q_j$, variance reduction follows from the law of total variance,

$$\begin{aligned} \frac{1}{M}\mathbb{V}ar_z[c(x,z)] &= \frac{1}{M}\left(\mathbb{E}_z[c(x,z)^2] - \mathbb{E}_z[c(x,z)]^2\right) \\ &= \frac{1}{M}\left(\mathbb{E}_z[\mathbb{E}_z[c(x,z)^2|z \in \Omega_j]] - \mathbb{E}_z[\mathbb{E}_z[c(x,z)|z \in \Omega_j]]^2\right) \\ &\geq \frac{1}{M}\mathbb{E}_z[\mathbb{V}ar_z[c(x,z)|z \in \Omega_j]] \\ &= \frac{1}{M}\sum_{j=1}^{M} q_j \mathbb{V}ar_z[c(x,z)|z \in \Omega_j], \end{aligned} \tag{23}$$

e.g., see Glasserman (2004). Further variance reduction is possible by using the optimal quantizers.

**Proposition 2.2** *Voronoi cell sampling provides optimal variance reduction under stratified sampling with proportional allocation.*

**Proof** With $\Omega_j^* \equiv \Omega_j(\hat{z}_j^*)$, for any stationary quantizer $\hat{z}_j$, it holds that

$$\frac{1}{M}\sum_{j=1}^{M} q_j \mathbb{V}ar_z[c(x,z)|z \in \Omega_j] \geq \frac{1}{M}\sum_{j=1}^{M} q_j^* \mathbb{V}ar_z[c(x,z)|z \in \Omega_j^*]. \ \square \tag{24}$$

Unless all weights are equal, i.e., $q_j^* = \frac{1}{M},\ j = 1, \ldots, M$, proportional allocation requires that there are fewer partitions than scenarios to be able to make multiple draws from partitions with large weights. However, instead of drawing multiple times from fewer partitions, we could also draw once from more partitions, since (23) also ensures that further stratification of a partition would result in further variance reduction.

Let us therefore assume that Voronoi cell sampling draws only one random realization from each partition and assigns the resulting scenario the probability of the nearest quantizer. Denote $(\hat{z}_1^{'n}, \ldots, \hat{z}_M^{'n})_{n=1}^N$ as another sequence of candidate scenarios and $\hat{z}_j^{'0} = \hat{z}_j^0,\ j = 1, \ldots, M$. Then, scenarios can be obtained as a by-product of competitive learning,

$$\hat{z}_j^{'n} = \begin{cases} z_n & \text{if } j = \text{argmin}_k\{||z_n - \hat{z}_k^{n-1}||^2\}, \\ \hat{z}_j^{'n-1} & \text{otherwise,} \end{cases} \tag{25}$$

for $j = 1, \ldots, M,\ n = 1, \ldots, N$.

Under this allocation scheme, variance reduction can only be shown asymptotically by exploiting a property of the partitions defined by the optimal quantizers.

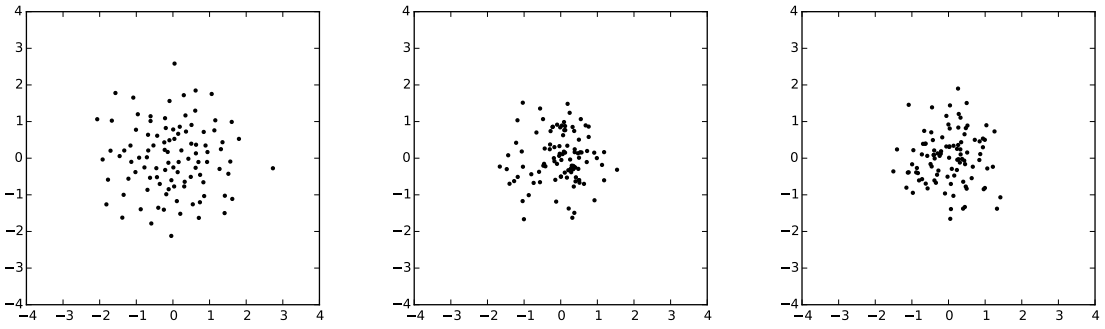
**Proposition 2.3** *The sampling method in (25) reduces the variance of the cost function estimate for sufficiently large $M$.*

**Proof** According to the partial distortion theorem, Theorem 6.3.1 of Gersho and Gray (1992), the weighted conditional variance of any partition $\Omega_j^*$ makes an equal contribution to the minimum distortion as $M$ approaches infinity,

$$D^* \approx q_j^* \mathbb{V}ar_z[c(x,z)|z \in \Omega_j^*],\ j = 1, \ldots, M. \tag{26}$$

If we randomly draw a realization once from each partition, so that $p_j = \frac{1}{M}$, then the variance of the cost function estimate is approximately given by

$$\frac{1}{M}\sum_{j=1}^{M} \frac{q_j^*}{p_j} q_j^* \mathbb{V}ar_z[c(x,z)|z \in \Omega_j^*] \approx \frac{1}{M}\sum_{j=1}^{M} \frac{q_j^*}{\frac{1}{M}} D^* = D^*, \tag{27}$$



(a) $d = 2$ (b) $d = 8$ (c) $d = 32$

Figure 2: Voronoi cell sampling of a multivariate normal distribution

with $\sum_{j=1}^{M} q_j^* = 1$. From (23), it follows that

$$\begin{aligned} \frac{1}{M}\mathbb{V}ar_z[c(x,z)] &\leq \frac{1}{M}\sum_{j=1}^{M} q_j^* \mathbb{V}ar_z[c(x,z)|z \in \Omega_j^*] \\ &\approx \frac{1}{M}\sum_{j=1}^{M} D^* = D^*. \ \square \end{aligned} \tag{28}$$

As can be seen from Figure 2, Voronoi cell sampling does not avoid it but reduces the concentration of scenarios towards the mean that is observed with the optimal quantizers.

## 2.5 Moment Matching

Another successful heuristic to generate a reduced set of scenarios is moment matching. The method has been originally proposed by Fleishman (1978) for the univariate case and extended by Høyland et al. (2003) for the multivariate case. The idea is to generate a set of scenarios that possesses the same first four moments as the desired distribution. The basic procedure works as follows:

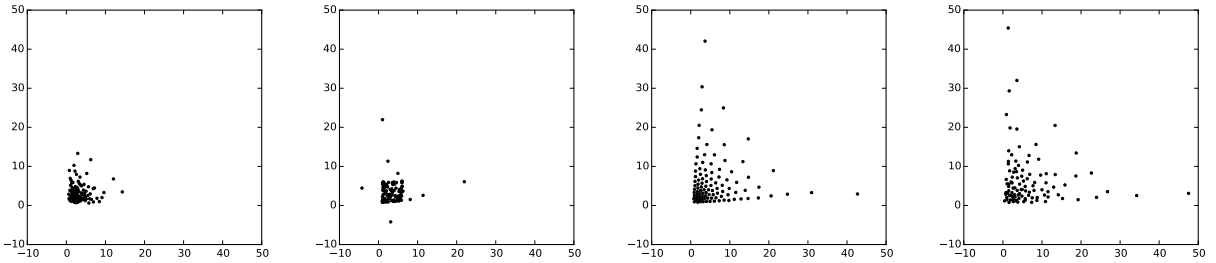
1. Generate $z_1, \ldots, z_M$ random vectors from the standard normal distribution using Monte Carlo sampling.

2. Transform the vectors to exhibit a given correlation by pre-multiplying the vectors with the lower triangular matrix $L$ of covariance matrix $\Sigma$,

$$z_j' = Lz_j, \ \Sigma = LL', \ j = 1, \ldots, M, \tag{29}$$

   where $L$ can be obtained by applying Cholesky decomposition (Glasserman, 2004).

3. Apply the cubic transformation of Fleishman (1978) to obtain sequences that satisfy the specification for the first four moments.



(a) quasi-Monte Carlo (b) moment matching (c) optimal quantization (d) Voronoi cell sampling

Figure 3: Scenarios for a bivariate log-normal distribution

Since the vectors $z_j$ are likely to violate the assumption of independence, zero sample mean, and unit sample variance, the transformed sample will most likely not satisfy the specified moments. Høyland et al. (2003) therefore propose an iterative procedure that produces a set of scenarios where the first four moments deviate from their target moments at most by a given epsilon.

A major drawback of moment matching is that theoretically there exist infinitely many distributions with identical moments, so that a discrete distribution found by moment matching may not fit to the data at hand. The potential problem is illustrated in Figure 3 which shows scenarios from a bivariate log-normal distribution, for which moment matching selects scenarios with negative components.

# 3 The Newsvendor Problem

To assess the SAA error of the above scenario generation methods, we resort to one of the most simple stochastic optimization models – the newsvendor problem, which describes the classic dilemma of a decision-maker who has to purchase newspapers before demand for that purchase becomes known.

## 3.1 The Classic Newsvendor Problem

Denote $p$ as the sales price, $c < p$ as the cost and $x$ as the purchase quantity. Further assume that demand can be described by a univariate probability distribution $F$ with density $f$. The expected profit is given by

$$\Pi(x) = p \int_{-\infty}^{x} z f(z) dz + px \int_{x}^{\infty} f(z) dz - cx. \tag{30}$$

Note that this formulation allows for demand to be negative. While this model may render the model useless for decision support, it makes analytical solutions more accessible, which facilitates using the model to evaluate the approximation quality of SAA.

An exact solution that maximizes $\Pi(x; p, c, f)$ is given by

$$x^* = \text{argmax}_x \Pi(x) = F^{-1}\left(\frac{p-c}{p}\right), \tag{31}$$

with $F^{-1}$ as the inverse of the distribution function. While an optimal solution can be found for any invertible distribution, solving (30) requires evaluation of the partial expectation as given by the first integral, which can be done analytically for the following three distributions:

**Normal Distribution:** With $\mu$ and $\sigma$ as mean and standard deviation, the expected profit under normally distributed demand is given by

$$(p-c)\mu - p\sigma f_{0,1}(x) \tag{32}$$

where $F_{0,1}$ denotes the standard normal distribution with density function $f_{0,1}$.

**Log-normal Distribution:** With $\mu$ and $\sigma$ as mean and standard deviation of the log-normal distribution, the expected profit under log-normally distributed demand is given by

$$p\mu - p\mu f_{0,1}\left(\sqrt{\log(1+\frac{\sigma^2}{\mu^2})} - x\right) \tag{33}$$

with $F_{0,1}$ and $f_{0,1}$ defined as above.

**Uniform Distribution:** With $a$ and $b$ as lower and upper bound of the uniform distribution, the expected profit under uniformly distributed demand is given by

$$(p-c)x - p\frac{(x-a)^2}{2(b-a)} \tag{34}$$

For all other distributions, we must resort to numerical integration. To approximate the expected profit, simply replace $F$ by an appropriate discrete distribution, e.g., by drawing a large sample from $F$. The expected profit of a newsvendor under discrete demand is then given by

$$p\sum_{j=1}^{M} q_j \min\{x, \hat{z}_{jk}\} - cx. \tag{35}$$

Let us assume for a moment that demand realizations are ordered increasingly, $\hat{z}_{0k} < \hat{z}_{1k} < \cdots < z_{Mk}$, in dimension $k$. To compute the corresponding optimal decision $\hat{x}^*$, now choose $\hat{x}^* = z_{jk}$ such that the probability of demand being smaller than $q_j$ is greater than $\frac{p-c}{p}$, i.e.,

$$\hat{x}^* = z_{jk} : j = \text{argmin}_j \left\{\sum_{k=1}^{j} q_k \geq \frac{p-c}{p}\right\}. \tag{36}$$

## 3.2 The Multidimensional Newsvendor Problem

As we are interested in evaluating the SAA error under multivariate randomness, we assume that the newsvendor purchases several different newspapers at once. Since purchase decisions are independent, the problem is separable.

Denote $f_k$ as the marginal of the multivariate distribution $F$ in dimension $k$. Then, the optimal expected joint profit is given by

$$\max_{x_1,\ldots,x_d} \left\{ \sum_{k=1}^{d} \left( p_k \int_{-\infty}^{x_k} z f_k(z) dz + p_k x_k \int_{x_k}^{\infty} f_k(z) dz - c_k x_k \right) \right\}, \tag{37}$$

for the continuous case, and

$$\max_{x_1,\ldots,x_d} \left\{ \sum_{k=1}^{d} p_k \sum_{j=1}^{M} q_j \min\{x_k, \hat{z}_{jk}\} - c_k x_k \right\}, \tag{38}$$

for the discrete case.

## 3.3 The Risk-Averse Newsvendor Problem

The biggest advantage of the multivariate newsvendor problem, separability, is also its biggest shortcoming for evaluating the SAA error, because only the marginals of the multivariate distribution are relevant to find an optimal solution, while dependency is not. To overcome this problem, let us consider a more general objective function that cannot be separated. To achieve this, we replace the expectation in (30) with the conditional value-at-risk (CVaR).

The CVaR was first proposed in (Rockafellar and Uryasev, 2002) and has been used to model risk-aversion in the newsvendor problem, e.g., Gotoh and Takano (2007) or Jammernegg and Kischka (2007). An important aspect for large-scale stochastic optimization is that the CVaR preserves concavity of the profit function.

In line with Gotoh and Takano (2007), assume that the newsvendor aims at maximizing the expected profit that falls below some $\alpha$-quantile, $\theta$, of the profit distribution. Instead of maximizing (30), the risk-averse newsvendor now seeks to minimize its expected shortfall. For univariate continuous demand, the expected profit below $\theta$ is given by

$$\Pi_\alpha(x, \theta) = \theta - \frac{1}{\alpha} \int_{-\infty}^{x} [\theta - pz + cx]^+ f(z) dz + \int_{x}^{\infty} [\theta - px + cx]^+ f(z) dz, \tag{39}$$

whereby the maximizers can be found analytically (Gotoh and Takano, 2007; Jammernegg and Kischka, 2007).

To solve the multidimensional problem under multivariate demand, we must resort to numerical

solutions that need the distribution either to be discrete or to be approximated by a discrete distribution. Denote $\boldsymbol{y} = (x_1, \ldots, x_d, \theta)$ and $\boldsymbol{z} = (z_1, \ldots, z_d)$. The deterministic equivalent formulation that uses a discrete distribution with mass points at $\hat{\boldsymbol{z}}_{\boldsymbol{j}} = (\hat{z}_{1j}, \ldots, \hat{z}_{dj}),\ j = 1, \ldots, M$, is given by

$$\max_{\boldsymbol{y},\boldsymbol{\pi}} \theta - \frac{1}{\alpha} \sum_{j=1}^{M} q_j \pi_j \tag{40}$$

$$\text{s.t. } \pi_j \geq \theta - \sum_{k=1}^{d} (p_k \min(x_k, \hat{z}_{jk}) - c_k x_k) \qquad j = 1, \ldots, M \tag{41}$$

$$\pi_j \geq 0 \qquad j = 1, \ldots, M. \tag{42}$$

To compare the SAA error of the scenario generation methods across different instances of the problem, we must be able to solve the problem for a large number of Monte Carlo samples. These samples must be several orders of magnitude larger than the number of scenarios obtained by any of the scenario generation methods. Although the problem can be solved via linear programming, computation times are substantial for large sample sizes.

Let us therefore resort to the *L-shaped* method as solution approach (Van Slyke and Wets, 1969). As in Benders decomposition, the L-shaped method requires a problem to be decomposed into a master problem and a number of small subproblems. Since detailed descriptions of the L-shaped method can be found in Birge and Louveaux (1997) and elsewhere, let us skip the computational steps involved, and focus on the solution of the decomposed problem.

Denote $Q(\boldsymbol{y}, \boldsymbol{z})$ as the maximum value of the subproblem for given $\boldsymbol{y}$ under outcome $\boldsymbol{z}$,

$$Q(\boldsymbol{y}, \boldsymbol{z}) = \max_{\pi} \ -\pi \tag{43}$$

$$\text{s.t. } \pi \geq \theta - \sum_{k=1}^{d} (p_k \min(x_k, z_k) - c_k x_k) \tag{44}$$

$$\pi \geq 0. \tag{45}$$

The corresponding master problem that is solved in place of (40) to (42) is then given by

$$\max_{\boldsymbol{y}} \ \theta + \frac{1}{\alpha} \sum_{j=1}^{M} q_j Q(\boldsymbol{y}, \hat{\boldsymbol{z}}_{\boldsymbol{j}}). \tag{46}$$

The single-cut version of the L-shaped method solves problem (46) by iteratively constructing an outer approximation of $\sum_{j=1}^{M} q_j Q(\boldsymbol{y}, \hat{\boldsymbol{z}}_{\boldsymbol{j}})$. In each iteration, the master problem is first solved to obtain an incumbent decision $\hat{\boldsymbol{y}}_{\boldsymbol{t}} = (\hat{x}_{1t}, \ldots, \hat{x}_{dt}, \hat{\theta}_t)$. Then, supporting hyperplanes of $Q(\cdot, \hat{\boldsymbol{z}}_{\boldsymbol{j}})$, $j = 1, \ldots, M$ going through $\hat{\boldsymbol{y}}_{\boldsymbol{t}}$ are obtained from the dual solutions of (43) to (45) for each scenario and aggregated into the master problem. It can be shown that, as the L-shaped method adds more

supporting hyperplanes to the outer approximation, the solution converges to an optimal solution (Van Slyke and Wets, 1969).

To solve the subproblem defined by (43) to (45), linear programming is not required, since the expression on the right-hand-side of (44) is constant,

$$Q(\boldsymbol{y}, \boldsymbol{z}) = \max \left( \sum_{k=1}^{d} (p_k \min(x_k, z_k) - c_k x_k) - \theta,\ 0 \right). \tag{47}$$

Nevertheless, the expression in (47) is not differentiable, so that we must consider the superdifferential, $\partial Q(\boldsymbol{y}, \boldsymbol{z})$, which is the set of supergradients of $Q(\cdot, \boldsymbol{z})$ going through $\boldsymbol{y}$. A supergradient can be obtained at low computational cost by looking at the partial derivatives,

$$\frac{\partial Q(\boldsymbol{y}, \boldsymbol{z})}{\partial x_k} = \begin{cases} (p_k - c_k) & \text{if } z_k > x_k \text{ and } \theta > \sum_{l=1}^{d} (p_l - c_l) x_l, \\ -c_k & \text{if } z_k \leq x_k \text{ and } \theta > \sum_{l=1}^{d} (p_l z_l - c_l x_l), \\ 0 & \text{otherwise,} \end{cases} \tag{48}$$

for $k = 1, \ldots, d$, and

$$\frac{\partial Q(\boldsymbol{y}, \boldsymbol{z})}{\partial \theta} = \begin{cases} -1 & \text{if } \theta > \sum_{k=1}^{d} (p_k \min(x_k, z_k) - c_k x_k), \\ 0 & \text{otherwise.} \end{cases} \tag{49}$$

This supergradient defines the slopes of a supporting hyperplane of $Q(\cdot, \boldsymbol{z})$ at $\boldsymbol{y}$,

$$\eta(\boldsymbol{y}, \boldsymbol{z}) = \left( \frac{\partial Q(\boldsymbol{y}, \boldsymbol{z})}{\partial x_1}, \ldots, \frac{\partial Q(\boldsymbol{y}, \boldsymbol{z})}{\partial x_d}, \frac{\partial Q(\boldsymbol{y}, \boldsymbol{z})}{\partial \theta} \right)^{\top}. \tag{50}$$

The expected value of $\hat{Q}(\boldsymbol{y}, \hat{\boldsymbol{z}}_{\boldsymbol{j}})$ can now be approximated from above by the minimum of a set of such supporting hyperplanes going through $\hat{\boldsymbol{y}}_{\boldsymbol{t}} = (\hat{x}_{1t}, \ldots, \hat{x}_{dt}, \hat{\theta}_t)$, $t = 1, \ldots, \tau$,

$$\sum_{j=1}^{M} q_j Q(\boldsymbol{y}, \hat{\boldsymbol{z}}_{\boldsymbol{j}}) \leq \min \left\{ \sum_{j=1}^{M} q_j \left( Q(\hat{\boldsymbol{y}}_{\boldsymbol{t}}, \hat{\boldsymbol{z}}_{\boldsymbol{j}}) + \eta(\hat{\boldsymbol{y}}_{\boldsymbol{t}}, \hat{\boldsymbol{z}}_{\boldsymbol{j}})^{\top} (\boldsymbol{y} - \hat{\boldsymbol{y}}_{\boldsymbol{t}}) \right),\ t = 1, \ldots, \tau \right\}. \tag{51}$$

Since (51) can be expressed as a linear program, let us include the corresponding set of constraints into (46), so that the master problem is now given by

$$\max_{\boldsymbol{y}, v} \theta + \frac{1}{\alpha} v, \tag{52}$$

$$\text{s.t. } v \leq \sum_{j=1}^{M} q_j \left( Q(\hat{\boldsymbol{y}}_{\boldsymbol{t}}, \hat{\boldsymbol{z}}_{\boldsymbol{j}}) + \eta(\hat{\boldsymbol{y}}_{\boldsymbol{t}}, \hat{\boldsymbol{z}}_{\boldsymbol{j}})^{\top} (\boldsymbol{y} - \hat{\boldsymbol{y}}_{\boldsymbol{t}}) \right), \qquad t = 1, \ldots, \tau. \tag{53}$$

Finally, without loss of generality, let us assume that the L-shaped method is stopped as soon as the absolute change in the objective value between two successive iterations has dropped below a given $\varepsilon$.

Since the subproblems of the risk-averse newsvendor problem can be evaluated analytically, the L-shaped method provides an efficient approach to solve the SAA problem with a large sample of scenarios, as it is the case for the computational study that is described in the next section.

# 4 Results

## 4.1 Measuring Scenario Quality

To measure the SAA error across multiple problem instances, we are going to measure the error with respect to the optimum, which is equivalent to the mean absolute percentage error (MAPE) that is often used in forecasting, e.g., Hyndman and Koehler (2006).

Let us introduce two different criteria to measure the SAA error (here for the classic newsvendor problem): the first criterion measures the difference in objective values,

$$e^{obj}(\Pi, \hat{\Pi}) = \left| 1 - \frac{\max_x \hat{\Pi}(x))}{\max_x \Pi(x)} \right|, \tag{54}$$

and will be referred to as the *objective error*.

The second criterion is based on the SAA error as defined in (6),

$$e^{pol}(\Pi, \hat{\Pi}) = \left| 1 - \frac{\Pi(\text{argmax}_x \hat{\Pi}(x))}{\max_x \Pi(x)} \right|, \tag{55}$$

and will be referred to as the *policy error*, as it measures the error that is made when implementing the decision obtained by solving the deterministic equivalent problem using SAA.

While one may argue that the policy error is the true SAA error, the quality of the objective function can be equally important, in particular when a stochastic optimization model is used for valuation, for example in portfolio management or asset pricing (Dempster et al., 2011).

## 4.2 Implementation

The computational study, the simulation, and the algorithms have been implemented in Java. Independent standard uniform pseudo-random numbers are generated using the $2^{1024}$ period Xorshift random number generator of Marsaglia (2003).

Correlation among random variables is enforced by applying the appropriate transformations to vectors of independent pseudo-random numbers. For the normal and the t-distribution, cor-

relation among elements is enforced by multiplying the random vectors with the lower triangular matrix of the covariance matrix. Since this transformation only works for elliptical distributions, the transformation described in Dias et al. (2008) is used to generate correlated uniform random variables.

For competitive learning, parameters have been tuned to trade off computational time and quality of the quantization: $a = 100M$, $N = 10000M$, and $\alpha_i = \frac{a}{a+i}$, such that $\alpha_N < 0.01$. For scenario sets with $d = \{2, 10\}$ and $M = \{5, 25, 50, 250\}$, scenarios generated via optimal quantization with the above parameters exhibit a less than 0.5% higher Wasserstein distance than the grids provided by Corlay et al. (2005). Nevertheless, to ensure that experimental results for optimal quantization are as good as possible, the grids provided by Corlay et al. (2005) are used as scenarios for all instances with a normal distribution with $d \leq 10$ and $\rho = 0.0$.

Scenarios generated by moment matching are computed with the method described in Høyland et al. (2003) using the source code provided by Kaut (2003).

Quasi-Monte Carlo scenarios are generated by using randomized $d$-dimensional Sobol sequences, as experimental evidence points to Sobol sequences as providing the most effective uniform point sets for Monte Carlo integration (Glasserman, 2004; Koivu, 2005). Sobol sequences are generated using the SSJ library of L'Ecuyer et al. (2002) with a left-matrix scramble as proposed by Hong and Hickernell (2003) for randomization of the sequence.

If the optimal expected profit cannot be computed exactly, i.e., for the t-distribution and the risk-averse case, $N = 10^5$ random vectors are drawn from the respective multivariate distribution and the optimal solution is computed numerically. Under 20-dimensional randomness, the resulting stochastic optimization problem has 2 million random parameters. The L-shaped method that is used to solve these problems for the CVaR case is stopped when the absolute change in objective values of the master problem between successive iterations is below $\varepsilon < 10^{-3}$. The master problem is solved using CLP 1.16 via Google Or-Tools' Java interface.

## 4.3 Experimental Setup

The aim of the experiment is to study the SAA error of the methods from Section (2) for a diverse set of problem instances and multivariate distributions.

Let us make the following simplifying assumptions:

1. All products have unit price $p_k = 1$ and cost are defined by an all-identical contribution margin $h$, so that $c_k = p(1 - h)$.
2. Mean demand across all products and for all distributions is $\mu_k = 1$, and the variability is defined by an all-identical coefficient of variation $cv$, so that $\sigma_k = cv$.

3. The correlation between product demand of different products, $\rho$, is identical among all products.

The parameters selected for the experiment are summarized in Table 1.

| | | |
|---|---|---|
| Distribution | | normal, uniform, log-normal, t-dist with df=5 |
| Contribution margin | $h$ | $0.1, \ldots, 0.9$ |
| Coefficient of variation | $cv$ | $0.3, 0.7$ |
| Correlation | $\rho$ | $0.0, 0.5$ |
| Dimensions, scenarios | $(d, M)$ | $(2, 5), (2, 50), (10, 25), (10, 250), (20, 50), (20, 500)$ |
| CVaR | $\alpha$ | 0.05 |

Table 1: Design parameters

Results are computed for all possible parameter combinations and grouped by distribution type, number of dimensions, and number of scenarios. For each of the three categories, the MAPE is computed for all figures within one group. Each group contains $10 \cdot 2 \cdot 2 = 40$ values.

The difference in MAPEs between the method with the lowest MAPE and all others is tested for statistical significance via pairwise t-tests. Since values in each group are derived from objective values of different model instances, they are not sampled from the same distribution and a direct application of a pairwise t-test would lead to biased significance levels. Instead, the pairwise differences in each group are bootstrapped and the p-values of the bootstrap distribution are computed. To test whether the lowest MAPE is significant across all pairwise comparisons, we use the Bonferroni correction which lowers the significance level of each pairwise t-test to obtain the family-wise significance level, e.g., see (Law, 2007, p. 337). The significance levels of the lowest MAPE in one group are indicated as follows: ¡0.01 (***), ¡0.05 (**), and ¡0.1 (*), or not significant (ns). The values give the probability of spuriously rejecting the null hypothesis of two MAPEs being equal.

The computational results are summarized in Tables 2 to 5, with a separate table for each distribution and a separate set of columns for the two objectives, maximizing expected profit and maximizing the profit's conditional value-at-risk. The different scenario reduction methods are abbreviated as follows: moment matching (MM), competitive learning quantization (CLQ), quasi-Monte Carlo (QMC), and Voronoi cell sampling (VCS).

Moment matching does not always return a set of scenarios, especially when the ratio $M/d$ is small. When this happens, the respective cells are left blank and the instance is excluded from all column averages that are shown in the last two rows. In this way, we avoid that the column averages of moment matching exclude difficult cases with a low number of scenarios while the other column averages do not.

| | | | Maximize Expected Profit | | | | Minimize Expected Shortfall | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| $d$ | $M$ | Error | MM | CLQ | QMC | VCS | MM | CLQ | QMC | VCS |
| 2 | 5 | Objective | $0.28^{*}$ | 0.65 | 2.64 | 1.03 | 1.39 | $1.37^{\text{ns}}$ | 2.37 | 1.51 |
| | | Policy | 0.29 | $0.12^{\text{ns}}$ | 0.48 | 0.43 | $1.00^{*}$ | 1.06 | 2.59 | 1.37 |
| | 50 | Objective | 0.06 | 0.05 | 0.43 | $0.02^{\text{ns}}$ | 0.18 | 0.14 | 0.84 | $0.06^{***}$ |
| | | Policy | 0.01 | 0.01 | $0.01^{\text{ns}}$ | 0.02 | 0.06 | $0.02^{***}$ | 0.43 | 0.05 |
| 10 | 25 | Objective | $0.06^{\text{ns}}$ | 1.15 | 0.96 | 0.22 | 0.83 | 1.37 | 1.47 | $0.73^{***}$ |
| | | Policy | $0.03^{\text{ns}}$ | 0.12 | 0.07 | 0.13 | $0.19^{*}$ | 0.29 | 0.62 | 0.36 |
| | 250 | Objective | $0.00^{*}$ | 0.67 | 0.19 | 0.01 | $0.09^{\text{ns}}$ | 0.88 | 0.65 | 0.10 |
| | | Policy | 0.00 | 0.03 | $0.00^{\text{ns}}$ | 0.01 | $0.05^{\text{ns}}$ | 0.08 | 0.08 | 0.07 |
| 20 | 50 | Objective | $0.04^{*}$ | 1.38 | 0.63 | 0.05 | 0.45 | 1.23 | 1.04 | $0.35^{***}$ |
| | | Policy | $0.02^{\text{ns}}$ | 0.22 | 0.04 | 0.05 | $0.15^{*}$ | 0.23 | 0.37 | 0.22 |
| | 500 | Objective | $0.01^{\text{ns}}$ | 1.05 | 0.13 | 0.02 | $0.06^{\text{ns}}$ | 0.93 | 0.42 | 0.06 |
| | | Policy | 0.00 | 0.12 | $0.00^{\text{ns}}$ | 0.01 | $0.03^{*}$ | 0.13 | 0.05 | 0.04 |
| Mean | | Objective | $0.08^{**}$ | 0.83 | 0.83 | 0.23 | 0.50 | 1.00 | 1.15 | $0.47^{**}$ |
| | | Policy | $0.06^{\text{ns}}$ | 0.10 | 0.10 | 0.11 | $0.26^{***}$ | 0.31 | 0.72 | 0.36 |

Table 2: Results for the normal distribution grouped by dimensions and number of scenarios

| | | | Maximize Expected Profit | | | | Minimize Expected Shortfall | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| $d$ | $M$ | Error | MM | CLQ | QMC | VCS | MM | CLQ | QMC | VCS |
| 2 | 5 | Objective | $0.39^{**}$ | 0.72 | 1.96 | 0.56 | $2.26^{**}$ | 2.46 | 4.46 | 2.39 |
| | | Policy | $0.07^{\text{ns}}$ | 0.10 | 0.62 | 0.30 | $1.05^{**}$ | 1.46 | 3.35 | 2.03 |
| | 50 | Objective | 0.04 | 0.05 | 0.16 | $0.01^{**}$ | 0.48 | 0.41 | 0.72 | $0.11^{\text{ns}}$ |
| | | Policy | $0.00^{\text{ns}}$ | 0.01 | 0.00 | 0.01 | 0.11 | 0.21 | 0.20 | $0.11^{\text{ns}}$ |
| 10 | 25 | Objective | $0.09^{*}$ | 1.10 | 0.48 | 0.15 | $1.00^{**}$ | 3.17 | 1.67 | 1.20 |
| | | Policy | $0.03^{**}$ | 0.30 | 0.07 | 0.09 | $0.57^{\text{ns}}$ | 0.95 | 1.38 | 0.66 |
| | 250 | Objective | $0.01^{*}$ | 0.79 | 0.06 | 0.02 | $0.18^{\text{ns}}$ | 2.32 | 0.59 | 0.20 |
| | | Policy | 0.00 | 0.14 | $0.00^{\text{ns}}$ | 0.01 | 0.12 | 0.40 | $0.10^{\text{ns}}$ | 0.16 |
| 20 | 50 | Objective | 0.05 | 1.24 | 0.30 | $0.04^{\text{ns}}$ | 0.95 | 4.43 | 1.70 | $0.76^{\text{ns}}$ |
| | | Policy | $0.01^{\text{ns}}$ | 0.40 | 0.04 | 0.05 | $0.51^{***}$ | 1.14 | 0.78 | 0.64 |
| | 500 | Objective | 0.01 | 1.02 | 0.04 | $0.01^{**}$ | 0.19 | 3.27 | 0.43 | $0.12^{\text{ns}}$ |
| | | Policy | 0.00 | 0.24 | $0.00^{\text{ns}}$ | 0.00 | $0.09^{\text{ns}}$ | 0.54 | 0.09 | 0.09 |
| Mean | | Objective | $0.10^{***}$ | 0.82 | 0.50 | 0.13 | 0.84 | 2.68 | 1.59 | $0.80^{\text{ns}}$ |
| | | Policy | $0.02^{***}$ | 0.20 | 0.12 | 0.08 | $0.41^{***}$ | 0.78 | 0.98 | 0.62 |

Table 3: Results for the uniform distribution grouped by dimensions and number of scenarios

### 4.4 Discussion

Let us begin by looking at what we can see in Table 2 which summarizes the results for the normal distribution. We see that CLQ yields the lowest errors on average for 2-dimensional randomness, while moment matching yields the lowest errors in higher dimensions.

With increasing dimensionality, the errors from using scenarios generated by CLQ increase considerably. The objective error is predicted by Proposition 2.1, however, we also observe an increase in policy error, for example, with $M = 50$ scenarios, from 1% in 2 dimensions to 22% in 20 dimensions.

In contrast to quantization (CLQ), the Monte Carlo methods appear to be more sensitive to the number of scenarios than to the dimensionality of randomness. Let us again take a look at the profit maximizing newsvendor, only this time at the policy errors of QMC (VCS). For 50 scenarios, the errors increase from 1% (2%) in $d = 2$ dimensions to 4% (5%) in 20 dimensions on average.

| | | | Maximize Expected Profit | | | | Minimize Expected Shortfall | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| $d$ | $M$ | Error | MM | CLQ | QMC | VCS | MM | CLQ | QMC | VCS |
| 1 | 5 | Objective | | 0.38 | $0.15^{**}$ | 0.31 | | 3.52 | 1.63 | 3.01 |
| | | Policy | | 0.22 | 0.10 | 0.33 | | 8.84 | 3.59 | 7.36 |
| | 50 | Objective | | $0.00^{***}$ | 0.03 | 0.00 | | $0.05^{\text{ns}}$ | 0.18 | 0.05 |
| | | Policy | | 0.00 | 0.00 | 0.00 | | $0.01^{***}$ | 0.15 | 0.03 |
| 10 | 50 | Objective | | 0.54 | $0.01^{***}$ | 0.08 | | 2.11 | 0.17 | 0.76 |
| | | Policy | | 0.32 | 0.00 | 0.07 | | 3.42 | 0.13 | 0.94 |
| | 500 | Objective | 0.32 | 0.36 | $0.00^{***}$ | 0.01 | 18.37 | 1.22 | $0.01^{***}$ | 0.16 |
| | | Policy | 0.32 | 0.19 | $0.00^{*}$ | 0.01 | 5.44 | 1.33 | $0.01^{*}$ | 0.16 |
| 25 | 50 | Objective | | 0.75 | $0.01^{***}$ | 0.09 | | 2.15 | 0.22 | 0.73 |
| | | Policy | | 0.61 | 0.00 | 0.10 | | 3.60 | 0.15 | 1.05 |
| | 500 | Objective | 0.35 | 0.60 | $0.00^{***}$ | 0.01 | 12.13 | 1.34 | $0.02^{***}$ | 0.22 |
| | | Policy | 0.27 | 0.49 | $0.00^{*}$ | 0.01 | 2.79 | 1.62 | $0.02^{*}$ | 0.24 |
| Mean | | Objective | 0.23 | 0.30 | $0.01^{\text{ns}}$ | 0.01 | 8.90 | 0.87 | $0.07^{***}$ | 0.17 |
| | | Policy | 0.16 | 0.19 | $0.00^{**}$ | 0.01 | 2.46 | 0.96 | $0.10^{***}$ | 0.20 |

Table 4: Results for the log-normal distribution grouped by dimensions and number of scenarios

| | | | Maximize Expected Profit | | | | Minimize Expected Shortfall | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| $d$ | $M$ | Error | MM | CLQ | QMC | VCS | MM | CLQ | QMC | VCS |
| 2 | 5 | Objective | | 2.69 | 8.84 | $2.52^{\text{ns}}$ | | 3.11 | 5.66 | $2.96^{*}$ |
| | | Policy | | $0.48^{\text{ns}}$ | 1.37 | 1.21 | | $0.64^{***}$ | 2.25 | 1.28 |
| | 50 | Objective | 2.30 | $0.27^{\text{ns}}$ | 1.90 | 0.29 | 4.38 | $0.19^{\text{ns}}$ | 2.92 | 0.20 |
| | | Policy | 0.18 | 0.08 | 0.06 | $0.03^{\text{ns}}$ | 1.47 | $0.05^{**}$ | 0.49 | 0.10 |
| 10 | 25 | Objective | | 2.97 | 2.00 | $0.33^{*}$ | | 2.00 | 2.66 | 0.92 |
| | | Policy | | 0.49 | $0.09^{\text{ns}}$ | 0.13 | | $0.28^{\text{ns}}$ | 0.79 | 0.46 |
| | 250 | Objective | 0.92 | 1.48 | 0.45 | $0.04^{\text{ns}}$ | 1.95 | 1.07 | 1.46 | $0.06^{***}$ |
| | | Policy | $0.02^{\text{ns}}$ | 0.23 | 0.05 | 0.04 | 0.37 | 0.21 | 0.17 | $0.07^{**}$ |
| 20 | 50 | Objective | 1.14 | 2.74 | 1.09 | $0.24^{*}$ | 1.58 | 2.20 | 2.20 | $0.81^{**}$ |
| | | Policy | 0.12 | 0.42 | 0.06 | $0.04^{\text{ns}}$ | 0.94 | $0.39^{\text{ns}}$ | 0.67 | 0.40 |
| | 500 | Objective | 0.89 | 2.30 | 0.36 | $0.09^{*}$ | 2.44 | 1.53 | 1.41 | $0.10^{**}$ |
| | | Policy | $0.01^{\text{ns}}$ | 0.29 | 0.05 | 0.05 | 0.37 | 0.34 | 0.15 | $0.06^{\text{ns}}$ |
| Mean | | Objective | 1.42 | 2.01 | 1.06 | $0.18^{***}$ | 2.41 | 1.40 | 1.99 | $0.40^{***}$ |
| | | Policy | 0.11 | 0.32 | 0.05 | $0.05^{\text{ns}}$ | 0.74 | 0.26 | 0.35 | $0.18^{*}$ |

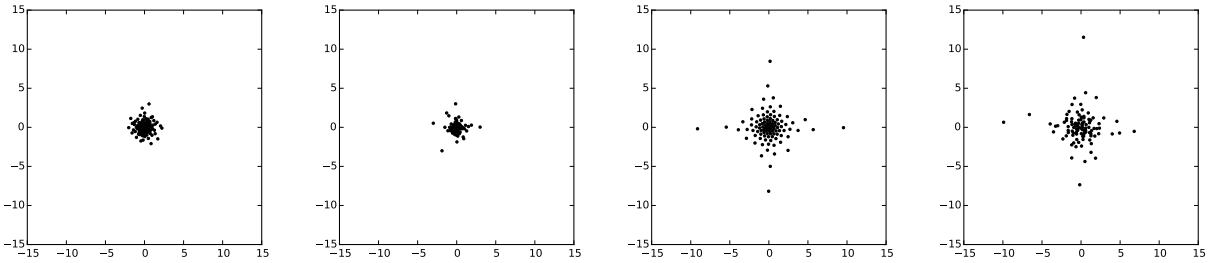
Table 5: Results for the heavy-tailed t-distribution grouped by dimensions and number of scenarios

In 2 dimensions, however, the same figure decreases from 48% (43%) with 5 scenarios to 1% (2%) with 50 scenarios. It appears that the errors decrease more quickly in sample size as they increase in the number of dimensions.

But what causes the deterioration of CLQ scenario quality? It seems that the bias that is introduced by the contraction of scenarios towards the distribution mean, as illustrated in Figure 1, not only decreases the variance of the marginal distributions, but also leads to increasingly suboptimal decisions in the number of dimensions when compared with Monte Carlo methods.

This shows that the Wasserstein distance as the only optimality criterion is insufficient to choose an optimal set of scenarios, since generating scenarios with minimal distance to the true distribution does not necessarily provide the lowest SAA error. Finding an alternative metric that overcomes this problem may provide an interesting area of future work.

As can be seen from Table 3, results for the uniform distribution are also in favor of moment



(a) quasi-Monte Carlo (b) moment matching (c) optimal quantization (d) Voronoi cell sampling

Figure 4: Scenarios for a bivariate t-distribution with df=5

matching, which provides the lowest errors on average. Irrespective of its focus on moments as opposed to quantiles, moment matching shows steady results even when the CVaR is used as objective criterion. Given its superior performance when compared with the other methods, moment matching sets the bar for generating scenarios of the normal as well as the uniform distribution.

But can we infer from these results to the general case? The answer is – no. The picture already changes, once we look at the log-normal distribution, for which results are provided in Table 4. For the log-normal distribution, quasi-Monte Carlo provides the lowest errors on average which are in many cases significantly lower than the errors of all other methods. Moreover, even for those instances where moment matching is able to provide a set of scenarios, which is in less than half of the cases, moment matching exhibits errors that are an order of magnitude higher than those of the Monte Carlo methods.

Results for the heavy-tailed t-distribution are shown in Table 5. For the t-distribution, VCS yields the lowest errors overall, although only the difference in objective errors is highly significant, whereas the difference in policy errors is either not significant or only significant at a low level of confidence. The low confidence levels are due to the high probability of extreme outcomes that affects the variability of the distribution of errors. Again, moment matching yields higher errors than the Monte Carlo methods, although the gap is not as large as with the log-normal distribution.

A graphical explanation why VCS performs particularly well when applied to the t-distribution is given in Figure 4. In contrast to moment matching and quasi-Monte Carlo, the competitive learning algorithm also generates quantizers for the tails, which forces VCS to draw scenarios from regions with a low probability density. Including these extreme scenarios appears to have an impact on the quality of the policy.

Table 6 summarizes the result of the experiment by taking the average across all problem instances, for which all methods were able to provide a set of scenarios. The table identifies VCS as the method which provides solutions that have the lowest objective and policy errors on average. While for many instances, one or the other method provides lower errors than VCS, the differences

are often small. Although moment matching and quasi-Monte Carlo perform best for particular distributions, they also showed a lot more variability in performance across all instances. VCS, by contrast, often presented itself as the second best choice, without any apparent weaknesses for a particular set of problem instances.

| Error | MM | CLQ | QMC | VCS |
|---|---|---|---|---|
| Objective | 0.81 | 1.2 | 0.94 | $0.31^{***}$ |
| Policy | 0.30 | 0.32 | 0.33 | $0.21^{***}$ |

Table 6: Errors averaged across all problem instances

# 5 Conclusion

This work presents the results of an empirical analysis of popular scenario generation methods for stochastic optimization, including quasi-Monte Carlo, moment matching, and methods based on probability metrics. Additionally, a new method referred as Voronoi cell sampling is proposed - which was discovered as a by-product of the experimental work.

To assess the effect on solution quality of using scenarios in place of the true distribution, the empirical analysis resorts to the well-known newsvendor problem, as solutions can still be obtained analytically even under high-dimensional randomness. The experimental study computes errors for the normal, uniform, log-normal, as well as a heavy-tailed t-distribution in 2, 10, and 20 dimensions for varying distribution and model parameters. The analysis considers not only maximization of the expected value, but also minimization of the expected shortfall using a formulation that is based on the conditional value-at-risk (CVaR), for which a tailored solution strategy based on Benders decomposition is proposed. Solution quality is assessed using two different measures: the *objective error*, which measures the deviation between approximate and true objective value, and the *policy error*, which measures the deviation between the objective value of the approximate solution and the true objective value.

The results of the empirical analysis indicate that there is not one scenario reduction method that is best in all cases. For instance, if scenarios from a low-dimensional distribution are needed, optimal quantization as a method that is based on probability metrics is likely to deliver the best results. For the multivariate normal and the multivariate uniform distribution, moment matching generates the best scenarios, but performs poorly for the log-normal distribution. Quasi-Monte Carlo, on the other hand, provides the best method for the log-normal distribution. The lowest errors on average are observed with Voronoi cell sampling, with particularly good results for a heavy-tailed t-distribution.

The most controversial result concerns the finding that the Wasserstein distance as probability

metric appears to be an ineffective criterion for choosing scenarios in higher dimensions, as these clearly provide sub-optimal solutions. A possible explanation is that the variance of the marginals of the discrete distribution decreases when compared to the marginals of the true distribution as the dimensionality of the random variable increases. This mismatch in variance potentially introduces a bias into the scenario-based solution. Future work on optimal scenario reduction should therefore take this bias into account, for example, by developing methods that minimize the Wasserstein distance but additionally penalize the difference in variance, as it has been proposed in De Oliveira et al. (2010).

## Acknowledgements

The author would like to thank Lena Silbermayr for a fruitful discussion on risk-averse newsvendors as well as the three anonymous referees for their helpful comments and suggestions which helped to significantly improve the paper.

## References

Bally, V., G. Pages, et al. (2003). A quantization algorithm for solving multidimensional discrete-time optimal stopping problems. *Bernoulli 9*(6), 1003–1049.

Birge, J. R. and F. Louveaux (1997). *Introduction to stochastic programming*. Springer.

Bottou, L. (1998). On-line learning in neural networks. Chapter On-line Learning and Stochastic Approximations, pp. 9–42. New York, NY, USA: Cambridge University Press.

Corlay, S. and G. Pagès (2010). Functional quantization-based stratified sampling methods. *arXiv preprint arXiv:1008.4441*.

Corlay, S., G. Pagès, and J. Printems (2005). The optimal quantization website. `http://www.quantize.maths-fi.com`. Last checked: 2016-05-04.

De Oliveira, W. L., C. Sagastizábal, D. D. J. Penna, M. E. P. Maceira, and J. M. Damázio (2010). Optimal scenario tree reduction for stochastic streamflows in power generation planning problems. *Optimisation Methods & Software 25*(6), 917–936.

Dempster, M. A., E. A. Medova, and Y. S. Yong (2011). Comparison of sampling methods for dynamic stochastic programming. In *Stochastic Optimization Methods in Finance and Energy*, pp. 389–425. Springer.

Dias, C. T. d. S., A. Samaranayaka, and B. Manly (2008). On the use of correlated beta random variables with animal population modelling. *Ecological Modelling 215*(4), 293–300.

Fleishman, A. I. (1978). A method for simulating non-normal distributions. *Psychometrika 43*(4), 521–532.

Freimer, M. B., J. T. Linderoth, and D. J. Thomas (2012). The impact of sampling methods on bias and variance in stochastic linear programs. *Computational Optimization and Applications 51*(1), 51–75.

Gersho, A. and R. M. Gray (1992). *Vector Quantization and Signal Compression*. Springer.

Glasserman, P. (2004). *Monte Carlo methods in financial engineering*, Volume 53. Springer.

Gotoh, J. and Y. Takano (2007). Newsvendor solutions via conditional value-at-risk minimization. *European Journal of Operational Research 179*(1), 80–96.

Graf, S. and H. Luschgy (2000). *Foundations of quantization for probability distributions*. Lecture notes in mathematics. Secaucus, NJ, USA: Springer-Verlag New York, Inc.

Heitsch, H. and W. Römisch (2003). Scenario reduction algorithms in stochastic programming. *Computational Optimization and Applications 24*(2-3), 187–206.

Heitsch, H. and W. Römisch (2007). A note on scenario reduction for two-stage stochastic programs. *Operations Research Letters 35*(6), 731–738.

Higle, J. L. (1998). Variance reduction and objective function evaluation in stochastic linear programs. *INFORMS Journal on Computing 10*(2), 236.

Hochreiter, R. and G. C. Pflug (2007). Financial scenario generation for stochastic multi-stage decision processes as facility location problems. *Annals of Operations Research 152*(1), 257–272.

Homem-de Mello, T., V. L. de Matos, and E. C. Finardi (2011). Sampling strategies and stopping criteria for stochastic dual dynamic programming: a case study in long-term hydrothermal scheduling. *Energy Systems 2*(1), 1–31.

Hong, H. S. and F. J. Hickernell (2003). Algorithm 823: Implementing scrambled digital sequences. *ACM Transactions on Mathematical Software (TOMS) 29*(2), 95–109.

Høyland, K., M. Kaut, and S. W. Wallace (2003). A heuristic for moment-matching scenario generation. *Computational Optimization and Applications 24*(2-3), 169–185.

Hyndman, R. J. and A. B. Koehler (2006). Another look at measures of forecast accuracy. *International journal of forecasting 22*(4), 679–688.

Jammernegg, W. and P. Kischka (2007). Risk-averse and risk-taking newsvendors: a conditional expected value approach. *Review of Managerial Science 1*(1), 93–110.

Judd, K. L. (1998). *Numerical Methods in Economics*. MIT press.

Kaut, M. (2003). A heuristic for moment-matching scenario generation. `http://work.michalkaut.net/files/scen-gen_HKW_src.zip`. Last checked: 2016-05-04.

Koivu, M. (2005). Variance reduction in sample approximations of stochastic programs. *Mathematical Programming 103*(3), 463–485.

Law, A. M. (2007). *Simulation modeling and analysis*. McGraw-Hill.

L'Ecuyer, P., L. Meliani, and J. Vaucher (2002). SSJ: A framework for stochastic simulation in Java. In *Proceedings of the 2002 Winter Simulation Conference*, pp. 234–242. IEEE Press.

Linderoth, J., A. Shapiro, and S. Wright (2006). The empirical behavior of sampling methods for stochastic programming. *Annals of Operations Research 142*(1), 215–241.

MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations.

Marsaglia, G. (2003). Xorshift RNGs. *Journal of Statistical Software 8*(14), 1–6.

Mehrotra, S. and D. Papp (2013). Generating moment matching scenarios using optimization techniques. *SIAM Journal on Optimization 23*(2), 963–999.

Niederreiter, H. (1992). *Random number generation and quasi-Monte Carlo methods*, Volume 63. SIAM.

Pagès, G. (1998). A space quantization method for numerical integration. *Journal of Computational and Applied Mathematics 89*(1), 1–38.

Pagès, G. and J. Printems (2003). Optimal quadratic quantization for numerics: the gaussian case. *Monte Carlo Methods and Applications 9*, 135–166.

Pagès, G. and J. Printems (2008). Optimal quantization for finance: from random vectors to stochastic processes. *Handbook of Numerical Analysis 15*.

Pennanen, T. and M. Koivu (2005). Epi-convergent discretizations of stochastic programs via integration quadratures. *Numerische Mathematik 100*(1), 141–163.

Pflug, G. C. (2001). Scenario tree generation for multiperiod financial optimization by optimal discretization. *Mathematical Programming 89*(2), 251–271.

Rockafellar, R. T. and S. Uryasev (2002). Conditional value-at-risk for general loss distributions. *Journal of banking & finance 26*(7), 1443–1471.

Shapiro, A. (2003). Monte Carlo sampling methods. In *Stochastic programming*, Volume 10 of *Handbooks in Operations Research and Management Science*, pp. 353–425. Elsevier.

Shapiro, A. (2013). Sample average approximation. In *Encyclopedia of Operations Research and Management Science*, pp. 1350–1355. Springer.

Van Slyke, R. M. and R. Wets (1969). L-shaped linear programs with applications to optimal control and stochastic programming. *SIAM Journal on Applied Mathematics 17*(4), 638–663.