Stochastic Programming: A tutorial – Part I

DORS Tutorials 14/02/2023

Giovanni Pantuso

Department of Mathematical Sciences
University of Copenhagen
Copenhagen, Denmark
gp@math.ku.dk

Table of Contents

At the end of this tutorial you will know

How to formulate general stochastic programs

At the end of this tutorial you will know

How to formulate general stochastic programs

What makes them difficult

At the end of this tutorial you will know

How to formulate general stochastic programs

What makes them difficult

How to solve them

At the end of this tutorial you will know

How to formulate general stochastic programs:two-stage linear (mixed-integer) recourse problems

What makes them difficult

How to solve them

At the end of this tutorial you will know

How to formulate general stochastic programs: two-stage linear (mixed-integer) recourse problems for a risk neutral decision maker

What makes them difficult

How to solve them

At the end of this tutorial you will know

How to formulate general stochastic programs:two-stage linear (mixed-integer) recourse problems for a risk neutral decision maker

What makes them difficult and how to address the difficulty

How to solve them

At the end of this tutorial you will know

How to formulate general stochastic programs:two-stage linear (mixed-integer) recourse problems for a risk neutral decision maker

What makes them difficult and how to address the difficulty

How to solve (approximations of) two-stage stochastic programs

Limitations ...

- Risk-aversion
- Chance constraints
- Stochastic dominance
- Multi-stage problems
- Endogenous uncertainty
- Robust/Distributionally robust
- ...

Plan

- Introduction (45 min)
- break (15 min)
- Exercise 1 (10 min)
- More info on approximations (15 min)
- Exercise 2 (30 min)
- break (10 min)
- L-Shaped method (45 min)
- Exercise 3 (50 min)

Table of Contents

Uncertainty

Something is uncertain

$$(\Omega, \mathcal{F}, \mathbb{P})$$

Uncertainty

Something is uncertain

$$(\Omega, \mathcal{F}, \mathbb{P})$$

$$\omega \in \Omega \to \xi(\omega) \in \Xi$$

Uncertainty

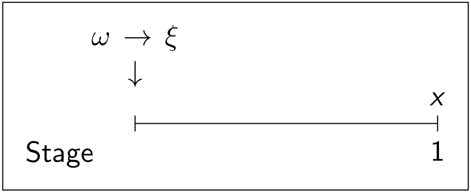
Something is uncertain

$$(\Omega, \mathcal{F}, \mathbb{P})$$

$$\omega \in \Omega \to \xi(\omega) \in \Xi$$

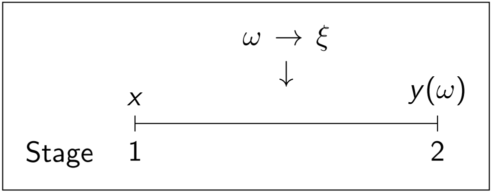
Notation can vary

Decision stages

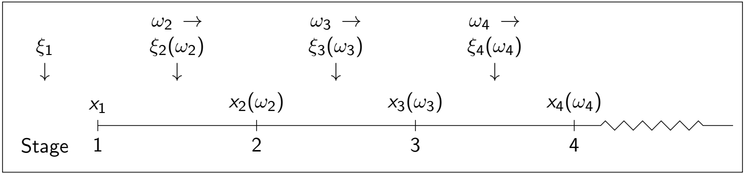


Decision stages

Decision stages



Decision stages



Two-Stage Stochastic Programs with Recourse

The decision maker:

1. Makes decisions $x \in \mathcal{X} \subseteq \mathbb{R}^{n_1}$ (first decision stage)

Two-Stage Stochastic Programs with Recourse

The decision maker:

1. Makes decisions $x \in \mathcal{X} \subseteq \mathbb{R}^{n_1}$ (first decision stage)
2. Waits for the outcome $\omega \in \Omega$ of some random experiment.
3. ω determines $\boldsymbol{\xi}(\omega)$ (our random data)

Two-Stage Stochastic Programs with Recourse

The decision maker:

1. Makes decisions $x \in \mathcal{X} \subseteq \mathbb{R}^{n_1}$ (first decision stage)
2. Waits for the outcome $\omega \in \Omega$ of some random experiment.
3. ω determines $\boldsymbol{\xi}(\omega)$ (our random data)
4. Makes decisions $y(\omega) \in \mathcal{Y} \subseteq \mathbb{R}^{n_2}$, given ξ and x (second decision stage)

Two-Stage Stochastic Programs with Recourse

$$\min \ c^\top x + \mathbb{E}_{\boldsymbol{\xi}}[\min \boldsymbol{q}(\omega)^T y(\omega)]$$

Two-Stage Stochastic Programs with Recourse

$$
\begin{aligned}
\min z = & c^\top x + \mathbb{E}_{\boldsymbol{\xi}}[\min \boldsymbol{q}(\omega)^T y(\omega)] \\
\text{s.t. } & Ax = b
\end{aligned}
$$

Two-Stage Stochastic Programs with Recourse

$$
\begin{aligned}
\min z = & c^\top x + \mathbb{E}_{\boldsymbol{\xi}}[\min \boldsymbol{q}(\omega)^T y(\omega)] \\
\text{s.t. } & Ax = b \\
& \boldsymbol{T}(\omega)x + \boldsymbol{W}(\omega)y(\omega) = \boldsymbol{h}(\omega) \quad a.s. \\
& x \in \mathcal{X}, y(\omega) \in \mathcal{Y}
\end{aligned}
$$

Two-Stage Stochastic Programs with Recourse

Parameters $c \in \mathbb{R}^{n_1}$, $b \in \mathbb{R}^{m_1}$, and $A \in \mathbb{R}^{n_1 \times m_1}$ are known.

Two-Stage Stochastic Programs with Recourse

Parameters $c \in \mathbb{R}^{n_1}$, $b \in \mathbb{R}^{m_1}$, and $A \in \mathbb{R}^{n_1 \times m_1}$ are known.

Parameters $\boldsymbol{q}(\omega) \in \mathbb{R}^{n_2}$, $\boldsymbol{h}(\omega) \in \mathbb{R}^{m_2}$, $\boldsymbol{W}(\omega) \in \mathbb{R}^{m_2 \times n_2}$ and $\boldsymbol{T}(\omega) \in \mathbb{R}^{m_2 \times n_1}$ are uncertain.

Two-Stage Stochastic Programs with Recourse

Parameters $c \in \mathbb{R}^{n_1}$, $b \in \mathbb{R}^{m_1}$, and $A \in \mathbb{R}^{n_1 \times m_1}$ are known.

Parameters $\boldsymbol{q}(\omega) \in \mathbb{R}^{n_2}$, $\boldsymbol{h}(\omega) \in \mathbb{R}^{m_2}$, $\boldsymbol{W}(\omega) \in \mathbb{R}^{m_2 \times n_2}$ and $\boldsymbol{T}(\omega) \in \mathbb{R}^{m_2 \times n_1}$ are uncertain.

$$\boldsymbol{\xi}(\omega) = \left(\boldsymbol{q}(\omega)^\top, \boldsymbol{h}(\omega)^\top, \boldsymbol{W}^{\mathbf{1}}(\omega), \ldots, \boldsymbol{W}(\omega)^{m_2}, \boldsymbol{T}(\omega)^1, \ldots, \boldsymbol{T}(\omega)^{m2}\right).$$

Two-Stage Stochastic Programs with Recourse

Parameters $c \in \mathbb{R}^{n_1}$, $b \in \mathbb{R}^{m_1}$, and $A \in \mathbb{R}^{n_1 \times m_1}$ are known.

Parameters $\boldsymbol{q}(\omega) \in \mathbb{R}^{n_2}$, $\boldsymbol{h}(\omega) \in \mathbb{R}^{m_2}$, $\boldsymbol{W}(\omega) \in \mathbb{R}^{m_2 \times n_2}$ and $\boldsymbol{T}(\omega) \in \mathbb{R}^{m_2 \times n_1}$ are uncertain.

$\boldsymbol{\xi}(\omega) = \left(\boldsymbol{q}(\omega)^\top, \boldsymbol{h}(\omega)^\top, \ \boldsymbol{W}^{\mathbf{1}}(\omega), \ldots, \boldsymbol{W}(\omega)^{m_2}, \boldsymbol{T}(\omega)^1, \ldots, \boldsymbol{T}(\omega)^{m2}\right)$.

ξ is a realization of $\boldsymbol{\xi}(\omega)$.

Two-Stage Stochastic Programs with Recourse

$$\begin{aligned} \min \;& c^\top x + Q(x) \\ \text{s.t. } & Ax = b \\ & x \in \mathcal{X} \end{aligned}$$

Two-Stage Stochastic Programs with Recourse

$$\begin{aligned} \min \;\; & c^\top x + Q(x) \\ \text{s.t.} \;\; & Ax = b \\ & x \in \mathcal{X} \end{aligned}$$

where

$$Q(x) = \mathbb{E}_{\boldsymbol{\xi}}[Q(x, \xi)]$$

Two-Stage Stochastic Programs with Recourse

$$\begin{aligned} \min \ & c^\top x + Q(x) \\ \text{s.t. } & Ax = b \\ & x \in \mathcal{X} \end{aligned}$$

where

$$Q(x) = \mathbb{E}_{\boldsymbol{\xi}}[Q(x, \xi)]$$

$$Q(x, \xi) = \min_y \{q^\top y | Wy = h - Tx, y \in \mathcal{Y}\}.$$

Two-Stage Stochastic Programs with Recourse with Discrete $\boldsymbol{\xi}$

Consider $\Xi = \{\xi_1, \ldots, \xi_S\}$ with probabilities π_s, $s = 1 \ldots, S$.

Two-Stage Stochastic Programs with Recourse with Discrete $\boldsymbol{\xi}$

Consider $\Xi = \{\xi_1, \ldots, \xi_S\}$ with probabilities π_s, $s = 1 \ldots, S$.

$\xi_s \implies q_s, T_s, W_s, h_s$.

Two-Stage Stochastic Programs with Recourse with Discrete $\boldsymbol{\xi}$

Consider $\Xi = \{\xi_1, \dots, \xi_S\}$ with probabilities π_s, $s = 1 \dots, S$.

$\xi_s \implies q_s, T_s, W_s, h_s$.

$y(\omega)$ becomes $y_1, \dots, y_S$.

Two-Stage Stochastic Programs with Recourse with Discrete $\boldsymbol{\xi}$

$$
\begin{aligned}
\min\ & c^\top x + \sum_{s=1}^{S} \pi_s q_s^T y_s \\
\text{s.t. } & Ax = b \\
& T_s x + W_s y_s = h_s && s = 1, \ldots, S \\
& x \in \mathcal{X} \\
& y_s \in \mathcal{Y} && s = 1, \ldots, S.
\end{aligned}
$$

Table of Contents

A closer look at ξ

Where do we get ξ?

A closer look at ξ

Where do we get ξ?

- Number of failures $\approx$ Weibull
- Wind speed $\approx$ Weibull,Rayleigh
- Forecast error (linear regression) $\approx$ Normal
- Hospitalization in certain epidemics $\approx$ LogNormal
- Repair times $\approx$ LogNormal
- Choice model $\approx$ Gumbel, Normal, EV Type I
- Waiting times $\approx$ Beta
- ...

A closer look at ξ

Some randomness is discrete

A closer look at ξ

Some randomness is discrete

- Number of occurrences $\approx$ Poisson
- Number of trials before success $\approx$ Geometric
- Number of successes $\approx$ HyperGeometric
- ...

A closer look at ξ

High dimensions and co-dependencies are problematic

A closer look at the constraints

If ξ is continuous...

A closer look at the constraints

If ξ is continuous...

Constraints must hold a.s. ...

A closer look at the constraints

If ξ is continuous...

Constraints must hold a.s. ...

Possibly ∞ constraints

A closer look at $Q(x)$

The recourse function ...

$$Q(x) = \mathbb{E}_{\boldsymbol{\xi}}\left[Q(x, \xi)\right] = \int_{\Omega} Q(x, \xi(\omega))\mathbb{P}(d\omega)$$

Why is this difficult?

A closer look at $Q(x)$

The recourse function ...

$$Q(x) = \mathbb{E}_{\boldsymbol{\xi}}\big[Q(x,\xi)\big] = \int_{\Omega} Q(x,\xi(\omega))\mathbb{P}(d\omega)$$

Why is this difficult?

Ingredient 1: a closed form expression $Q(x,\xi)$

A closer look at $Q(x)$

The recourse function ...

$$Q(x) = \mathbb{E}_{\boldsymbol{\xi}}\big[Q(x,\xi)\big] = \int_{\Omega} Q(x,\xi(\omega))\mathbb{P}(d\omega)$$

Why is this difficult?

Ingredient 1: a closed form expression $Q(x,\xi)$

Ingredient 2: an antiderivative

A closer look at $Q(x)$

The recourse function ...

$$Q(x) = \mathbb{E}_{\boldsymbol{\xi}}\left[Q(x,\xi)\right] = \int_{\Omega} Q(x,\xi(\omega))\mathbb{P}(d\omega)$$

Why is this difficult?

Ingredient 1: a closed form expression $Q(x,\xi)$

Ingredient 2: an antiderivative

Observe: $Q(x) = \int\int\cdots\int Q(x,\xi)\mathbb{D}(\xi)d\xi_1 d\xi_2\cdots d\xi_N$

A closer look at $Q(x)$

The recourse function ...

$$Q(x) = \mathbb{E}_{\boldsymbol{\xi}}\big[Q(x, \xi)\big] = \int_{\Omega} Q(x, \xi(\omega))\mathbb{P}(d\omega)$$

Idea! Numerical integration!

A closer look at $Q(x)$

In one dimension (i.e., $N = 1$): Riemann sums, Trapezoidal rule, Simpson's rule

Ex. Riemann Sums

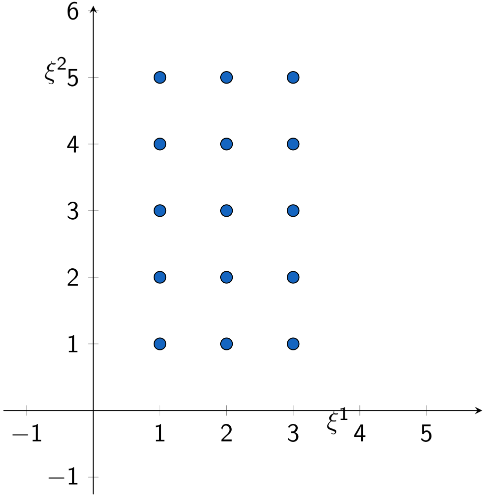
$$\int_a^b f(x)dx$$

- Partition $[a, b]$ using K points $x_0 = a$, $x_1, \ldots, x_K = b$ equally spaced Δx
- $\int_a^b f(x)dx \approx \sum_k f(x_k)\Delta x$
- As K increases we improve the approximation.

A closer look at $Q(x)$

In multiple dimensions: *Quadrature* methods.

Same principle, harder partition



A closer look at $Q(x)$

Numerical integration: it is already an approximation

A closer look at $Q(x)$

Numerical integration: it is already an approximation

all this work for one x...

A closer look at $Q(x)$

Numerical integration: it is already an approximation

all this work for one x...

we still have to solve the stochastic program..

A closer look at $Q(x)$

When $\boldsymbol{\xi}$ is discrete...

$$Q(x) = \sum_{s=1}^{S} \pi_s Q(x, \xi_s)$$

A closer look at $Q(x)$

When $\boldsymbol{\xi}$ is discrete...

$$Q(x) = \sum_{s=1}^{S} \pi_s Q(x, \xi_s)$$

Finitely many linear constraints

A closer look at $Q(x)$

When $\boldsymbol{\xi}$ is discrete...

$$Q(x) = \sum_{s=1}^{S} \pi_s Q(x, \xi_s)$$

Finitely many linear constraints

When $\boldsymbol{\xi}$ is discrete ... but large ...

Table of Contents

Approximations

Continuous/discrete but large $\rightarrow$ discrete and small

Approximations

Continuous/discrete but large $\rightarrow$ discrete and small

Three categories of methods (loosely speaking)

- Probability Metrics
- Property Matching
- Monte Carlo

Method based on Probability Metrics in a nutshell

Results from research on stability, see,e.g.,
[Dup90, Pfl01, RR02, Röm03].

$$|z(\mathbb{P}) - z(\mathbb{Q})| \leq Ld(\mathbb{P}, \mathbb{Q})$$

Method based on Probability Metrics in a nutshell

- Start from large N scenarios (e.g., sampled)

Method based on Probability Metrics in a nutshell

- Start from large N scenarios (e.g., sampled)
- Remove one scenario at a time to minimize the distance between the new and old distribution

Method based on Probability Metrics in a nutshell

- Start from large N scenarios (e.g., sampled)
- Remove one scenario at a time to minimize the distance between the new and old distribution
- Add the probability of the deleted scenarios to the closest scenarios (in the sense of the probability metric)

Scenario reduction/generation, see, e.g., [HR03, DGKR03].

Property Matching in a nutshell

Idea: Replicate only the statistical properties that are important for the problem [HW01].

Create a small distribution that replicates only those properties.

Property Matching in a nutshell

Not always necessary to increase the size of the distribution.

Property Matching in a nutshell

Not always necessary to increase the size of the distribution.

Problem driven.

Property Matching in a nutshell

Not always necessary to increase the size of the distribution.

Problem driven.

Observe: requires an NLP (heuristics exist [HKW03])

Property Matching in a nutshell

Not always necessary to increase the size of the distribution.

Problem driven.

Observe: requires an NLP (heuristics exist [HKW03])

Which properties?

Monte Carlo in a nutshell

Make K identical copies $\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_K$ of $\boldsymbol{\xi}$.

Monte Carlo in a nutshell

Make K identical copies $\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_K$ of $\boldsymbol{\xi}$.

From each take, independently, a realization ξ_k.

Monte Carlo in a nutshell

Make K identical copies $\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_K$ of $\boldsymbol{\xi}$.

From each take, independently, a realization ξ_k.

Write the *Sample Average Approximation* (SAA)

$$
\begin{aligned}
z^K = \min f(x) :=& c^\top x + \sum_{k=1}^{K} \frac{1}{K} q_k^T y_k \\
\text{s.t. } & Ax = b, \\
& T_k x + W_k y_k = h_k, \qquad k = 1, \ldots, K \\
& x \in \mathcal{X}, \\
& y_k \in \mathcal{Y}, \qquad k = 1, \ldots, K.
\end{aligned}
$$

Sample Average Approximation (SAA)

$$
\begin{aligned}
z^K = \min f^K(x) :=& c^\top x + \sum_{k=1}^{K} \frac{1}{K} q_k^T y_k \\
\text{s.t. } & Ax = b, \\
& T_k x + W_k y_k = h_k, \qquad k = 1, \dots, K \\
& x \in \mathcal{X}, \\
& y_k \in \mathcal{Y}, \qquad k = 1, \dots, K.
\end{aligned}
$$

or

$$
\begin{aligned}
z^K = \min f^K(x) :=& c^\top x + \sum_{k=1}^{K} \frac{1}{K} Q(x, \xi_k) \\
\text{s.t. } & Ax = b, \\
& x \in \mathcal{X}
\end{aligned}
$$

SAA: some observations

z^K is a stochastic estimator of z^*

SAA: some observations

$$f^K(\bar{x}) = c^\top \bar{x} + \sum_{k=1}^{K} \frac{1}{K} Q(\bar{x}, \xi_k)$$

is an **unbiased** estimator (pointwise $x = \bar{x}$) of

$$f(\bar{x}) = c^\top \bar{x} + \mathbb{E}_{\boldsymbol{\xi}}[Q(\bar{x}, \xi)]$$

SAA: some observations

Finally, $f^K(\bar{x})$ a *consistent estimator* of $f(\bar{x})$, that is

$$\lim_{K\to\infty} \frac{1}{K}\sum_{k=1}^{K} Q(\bar{x},\xi_k) \to \mathbb{E}_{\boldsymbol{\xi}}\left[Q(\bar{x},\xi)\right] = Q(\bar{x})$$

We say that $f^K(\bar{x})$ *converges pointwise*.

SAA: some observations

Finally, $f^K(\bar{x})$ a *consistent estimator* of $f(\bar{x})$, that is

$$\lim_{K\to\infty} \frac{1}{K}\sum_{k=1}^{K} Q(\bar{x},\xi_k) \to \mathbb{E}_{\boldsymbol{\xi}}\left[Q(\bar{x},\xi)\right] = Q(\bar{x})$$

We say that $f^K(\bar{x})$ *converges pointwise*.

But observe

$$\mathbb{S}td[\frac{1}{K}\sum_{k=1}^{K} Q(\bar{x},\xi_k)] = \frac{\mathbb{S}td[Q(\bar{x},\xi)]}{\sqrt{K}}$$

SAA: some observations

Exercise 1: 15 minutes

`https://tinyurl.com/sptutorial1`

SAA: some observations

Under certain conditions $z^K \to z^*$ as $k \to \infty$ (exponentially fast!)

SAA: some observations

Under certain conditions $z^K \to z^*$ as $k \to \infty$ (exponentially fast!)

$\mathbb{E}[z^K]$ gives a statistical lower bound! (obs! z^K is biased)

Proof

SAA: some observations

Under certain conditions $z^K \to z^*$ as $k \to \infty$ (exponentially fast!)

$\mathbb{E}[z^K]$ gives a statistical lower bound! (obs! z^K is biased)

Proof

$\mathbb{E}\left[c^\top \bar{x} + \frac{1}{K}\sum_{k=1}^{K}[Q(\bar{x}, \xi_k)]\right]$ gives a statistical upper bound!

See, e.g., [Sha91, MMW99, Sha03].

SAA: some observations

Exercise 2: 30 minutes

`https://tinyurl.com/sptutorial2`

Approximations

Getting a good solution vs Estimating its value

Table of Contents

Mathematical properties of discrete stochastic programs

$$\mathcal{K}_2(\xi) = \{x | \exists y \geq 0, \text{s.t.} W(\omega)y = h(\omega) - T(\omega)x\}$$

Mathematical properties of discrete stochastic programs

$$\mathcal{K}_2(\xi) = \{x | \exists y \geq 0, \text{s.t.} W(\omega) y = h(\omega) - T(\omega) x\}$$

Convex and polyhedral!

Mathematical properties of discrete stochastic programs

$$\mathcal{K}_2 = \bigcap_{\xi \in \Xi} \mathcal{K}_2(\xi)$$

Mathematical properties of discrete stochastic programs

$$\mathcal{K}_2 = \bigcap_{\xi \in \Xi} \mathcal{K}_2(\xi)$$

Convex and polyhedral!

Mathematical properties of discrete stochastic programs

Useful jargon

Complete recourse

$$\mathcal{K}_2 = \mathbb{R}^{n_1}$$

Mathematical properties of discrete stochastic programs

Useful jargon

Complete recourse

$$\mathcal{K}_2 = \mathbb{R}^{n_1}$$

Relatively complete recourse

$$\mathcal{K}_2 \subseteq \mathcal{K}_1 = \{x | Ax = b, x \geq 0\}$$

.

Mathematical properties of discrete stochastic programs

$Q(x, \xi)$ is:

a. piece-wise linear convex in h, T and x,

b. piece-wise linear concave in q.

Mathematical properties of discrete stochastic programs

$$Q(x) = \mathbb{E}_{\boldsymbol{\xi}} Q(x, \xi) = \sum_{s=1}^{S} \pi_s Q(x, \xi_s)$$

piece-wise linear convex in x.

Proof of lower bound

The expectation of the optimal objective value of SAAs of size K is a lower bound for the true optimal objective value, that is:

$$\mathbb{E}[z^K] \leq z^*$$

Proof.
First, observe that $z^* = \min_{x \in \mathcal{K}_1} f(x) = \min_{x \in \mathcal{K}_1} c^\top x + \mathbb{E}\left[K^{-1} \sum_{k=1}^{K} Q(x, \xi_k)\right]$. We can use a direct proof. We know that for all $x' \in \mathcal{K}_1$

$$\min_{x \in \mathcal{K}_1} K^{-1} \sum_{k=1}^{K} c^\top x + Q(x, \xi_k) \leq K^{-1} \sum_{k=1}^{K} c^\top x' + Q(x', \xi_k)$$

by taking the expectation of both sides we obtain continues... □

Proof of lower bound

The expectation of the optimal objective value of SAAs of size K is a lower bound for the true optimal objective value, that is:

$$\mathbb{E}[z^K] \leq z^*$$

Proof.

continuing...

$$\mathbb{E}\left[\min_{x \in \mathcal{K}_1} K^{-1} \sum_{k=1}^{K} c^\top x + Q(x, \xi_k)\right] \leq \mathbb{E}\left[K^{-1} \sum_{k=1}^{K} c^\top x' + Q(x', \xi_k)\right]$$

which corresponds to

$$\mathbb{E}\left[z^K\right] \leq \mathbb{E}\left[K^{-1} \sum_{k=1}^{K} c^\top x' + Q(x', \xi_k)\right]$$

continues... □

Proof of lower bound

The expectation of the optimal objective value of SAAs of size K is a lower bound for the true optimal objective value, that is:

$$\mathbb{E}[z^K] \leq z^*$$

Proof.
continuing... Now, since the inequality holds for all x', it holds also for the solution x' that minimizes expectation on the right-hand-side and which yields z^*, that is

$$\mathbb{E}\left[z^K\right] \leq \min_{x' \in \mathcal{K}_1} \mathbb{E}\left[K^{-1} \sum_{k=1}^{K} c^\top x' + Q(x', \xi_k)\right] = z^*$$

□

Table of Contents

References I

[DGKR03] Jitka Dupačová, Nicole Gröwe-Kuska, and Werner Römisch. Scenario reduction in stochastic programming. *Mathematical programming*, 95(3):493–511, 2003.

[Dup90] Jitka Dupačová. Stability and sensitivity-analysis for stochastic programming. *Annals of operations research*, 27(1):115–142, 1990.

[HKW03] Kjetil Høyland, Michal Kaut, and Stein W Wallace. A heuristic for moment-matching scenario generation. *Computational optimization and applications*, 24(2):169–185, 2003.

[HR03] Holger Heitsch and Werner Römisch. Scenario reduction algorithms in stochastic programming. *Computational optimization and applications*, 24(2):187–206, 2003.

References II

[HW01] Kjetil Høyland and Stein W Wallace. Generating scenario trees for multistage decision problems. *Management science*, 47(2):295–307, 2001.

[MMW99] Wai-Kei Mak, David P Morton, and R Kevin Wood. Monte carlo bounding techniques for determining solution quality in stochastic programs. *Operations research letters*, 24(1-2):47–56, 1999.

[Pfl01] G Ch Pflug. Scenario tree generation for multiperiod financial optimization by optimal discretization. *Mathematical programming*, 89(2):251–271, 2001.

[Röm03] Werner Römisch. Stability of stochastic programming problems. *Handbooks in operations research and management science*, 10:483–554, 2003.

References III

[RR02] Svetlozar T Rachev and Werner Römisch. Quantitative stability in stochastic programming: The method of probability metrics. *Mathematics of Operations Research*, 27(4):792–818, 2002.

[Sha91] Alexander Shapiro. Asymptotic analysis of stochastic programs. *Annals of Operations Research*, 30(1):169–186, 1991.

[Sha03] Alexander Shapiro. Monte carlo sampling methods. In *Stochastic Programming*, volume 10 of *Handbooks in Operations Research and Management Science*, pages 353–425. Elsevier, 2003.