Stochastic Optimization
IDA PhD course 2011ht

Stefanie Kosuch
PostDok at TCSLab
www.kosuch.eu/stefanie/

7. Lecture: Inner and Outer Approximation Methods
24. November 2011

Linköping University

1 Multi-Stage Programming
- Modeling Multi-Stage Stochastic Programming Problems
- Deterministic Reformulation in case of discrete distributions (Main Idea)

1 Multi-Stage Programming
- Modeling Multi-Stage Stochastic Programming Problems
- Deterministic Reformulation in case of discrete distributions (Main Idea)

2 Outer and Inner Approximation Approaches
- Sample Average Approximation
- Stochastic Gradient method

Linköping University

Outline

1. Multi-Stage Programming
   - Modeling Multi-Stage Stochastic Programming Problems
   - Deterministic Reformulation in case of discrete distributions (Main Idea)
2. Outer and Inner Approximation Approaches
   - Sample Average Approximation
   - Stochastic Gradient method

Basics of Multi-Stage Programming

Basics of Multi-Stage Programming

- Information obtained in several "stages"
- Recourse action can be taken in each stage
- Recourse action depends on:
  - $\rightarrow$ Information available
  - $\rightarrow$ Previous decisions
- Recourse action does not depend on:
  - $\rightarrow$ Information to be released later ($\neq$ det. multi-period optimization)
- Objective: Minimize total expected cost
- Crucial: Discretization of time and random distribution(s)

Main Differences to Online Optimization

Main Differences to Online Optimization

- Statistic Information used
- Parameters modeled as random variables
- Decision(s) made "offline"
- Actual decision computed (for all scenarios)
- Objective: Good solution on average

Linköping University

Which approach to choose?

Questions to be answered:

- Do I have a finite, infinite or undefined number of periods/stages?
- Do I have statistical information about the uncertain parameters?
- Do I have the infrastructure and/or time to make decisions online?
- Will I use my solution once or several times?

Linköping University

Outline

Linköping University

General Linear Multi-Stage Model

Linköping University

General Linear Multi-Stage Model

$$\min_{x_0 \geq 0} \quad c_0^T x_0$$

s.t.

Linköping University

General Linear Multi-Stage Model

$$\min_{x_0 \geq 0} \quad c_0^T x_0$$

$$\text{s.t.} \quad A_{00} x_0 \geq b_0,$$

Linköping University

General Linear Multi-Stage Model

$$\min_{x_0 \geq 0} \quad c_0^\intercal x_0 + \mathbb{E}_{\zeta_1}\left[\mathcal{Q}^1(x_0, \zeta_1)\right]$$

$$\text{s.t.} \quad A_{00} x_0 \geq b_0,$$

General Linear Multi-Stage Model

$$\min_{x_0 \geq 0} \quad c_0^\intercal x_0 + \mathbb{E}_{\zeta_1}\left[\mathcal{Q}^1(x_0, \zeta_1)\right]$$

$$\text{s.t.} \quad A_{00} x_0 \geq b_0,$$

$$\mathcal{Q}(x_0, \zeta_1) = \min_{x_1 \geq 0} \quad c_1^\intercal x_1$$

$x_1 = x_1(\zeta_1) \ \forall t \geq 1$

Linköping University

General Linear Multi-Stage Model

$$\min_{x_0 \geq 0} \quad c_0^\intercal x_0 + \mathbb{E}_{\zeta_1}\left[\mathcal{Q}^1(x_0, \zeta_1)\right]$$

$$\text{s.t.} \quad A_{00} x_0 \geq b_0,$$

$$\mathcal{Q}(x_0, \zeta_1) = \min_{x_1 \geq 0} \quad c_1^\intercal x_1$$

$$\text{s.t.} \quad \sum_{s=0}^{1} A_{1s}(\zeta_1) x_s \geq b_1(\zeta_1).$$

$$x_1 = x_1(\zeta_1) \ \forall t \geq 1$$

Linköping University

General Linear Multi-Stage Model

$$\min_{x_0 \geq 0} \quad c_0^\intercal x_0 + \mathbb{E}_{\zeta_1}\left[\mathcal{Q}^1(x_{[0]}, \zeta_{[1]})\right]$$
$$\text{s.t.} \quad A_{00} x_0 \geq b_0,$$

$$\mathcal{Q}(x_{[T-1]}, \zeta_{[T]}) = \min_{x_T \geq 0} \quad c_T^\intercal x_T$$
$$\text{s.t.} \quad \sum_{s=0}^{T} A_{Ts}(\zeta_{[T]}) x_s \geq b_T(\zeta_{[T]}).$$

$\zeta_{[t]} = (\zeta_1, \ldots, \zeta_t)$
$x_{[t-1]} = (x_0, x_1, \ldots, x_{t-1})$
$x_t = x_t(\zeta_{[t]}) \ \forall t \geq 1$

Linköping University

General Linear Multi-Stage Model

$$\min_{x_0 \geq 0} \quad c_0^\intercal x_0 + \mathbb{E}_{\zeta_1}\left[\mathcal{Q}^1(x_{[0]}, \zeta_{[1]})\right]$$

$$\text{s.t.} \quad A_{00} x_0 \geq b_0,$$

$$\left.\begin{aligned} \mathcal{Q}^t(x_{[t-1]}, \zeta_{[t]}) = \min_{x_t \geq 0} \quad & c_t^\intercal x_t + \mathbb{E}_{\zeta_{t+1}|\zeta_{[t]}}\left[\mathcal{Q}^{t+1}(x_{[t]}, \zeta_{[t+1]})\right] \\ \text{s.t.} \quad & \sum_{s=0}^{t} A_{ts}(\zeta_{[t]}) x_s \geq b_t(\zeta_{[t]}). \end{aligned}\right\} \quad \forall t \in \{1, \dots, t-1\}$$

$$\mathcal{Q}(x_{[T-1]}, \zeta_{[T]}) = \min_{x_T \geq 0} \quad c_T^\intercal x_T$$

$$\text{s.t.} \quad \sum_{s=0}^{T} A_{Ts}(\zeta_{[T]}) x_s \geq b_T(\zeta_{[T]}).$$

$\zeta_{[t]} = (\zeta_1, \dots, \zeta_t)$
$x_{[t-1]} = (x_0, x_1, \dots, x_{t-1})$
$x_t = x_t(\zeta_{[t]}) \ \forall t \geq 1$

Linköping University

Outline

Definition

Scenario

Definition

Scenario

- Sequence of outcomes $\zeta_1^k, \ldots, \zeta_T^k$ of $\zeta_1, \ldots, \zeta_T$

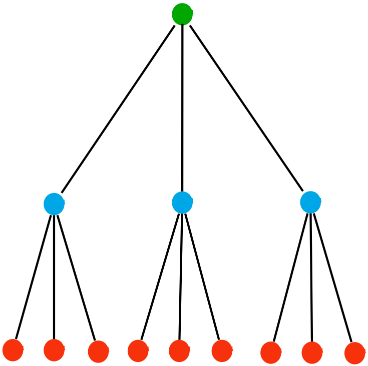
Linköping University

Definition

Scenario

- Sequence of outcomes $\zeta_1^k, \ldots, \zeta_T^k$ of $\zeta_1, \ldots, \zeta_T$
- Outcome $\zeta_{[T]}^k$ of $\zeta_{[T]} = (\zeta_1, \ldots, \zeta_T)$

Scenario Tree



Definition

Scenario

- Sequence of outcomes $\zeta_1^k, \ldots, \zeta_T^k$ of $\zeta_1, \ldots, \zeta_T$
- Outcome $\zeta_{[T]}^k$ of $\zeta_{[T]} = (\zeta_1, \ldots, \zeta_T)$

Definition

Scenario

- Sequence of outcomes $\zeta_1^k, \ldots, \zeta_T^k$ of $\zeta_1, \ldots, \zeta_T$
- Outcome $\zeta_{[T]}^k$ of $\zeta_{[T]} = (\zeta_1, \ldots, \zeta_T)$

Linköping University

Definition

Scenario

- Sequence of outcomes $\zeta_1^k, \ldots, \zeta_T^k$ of $\zeta_1, \ldots, \zeta_T$
- Outcome $\zeta_{[T]}^k$ of $\zeta_{[T]} = (\zeta_1, \ldots, \zeta_T)$

Idea

Definition

Scenario

- Sequence of outcomes $\zeta_1^k, \dots, \zeta_T^k$ of $\zeta_1, \dots, \zeta_T$
- Outcome $\zeta_{[T]}^k$ of $\zeta_{[T]} = (\zeta_1, \dots, \zeta_T)$

Idea

Introduce sequence of decision vectors $(x_0^k, x_1^k, \dots, x_T^k)$ ($\forall$ scenarios $\zeta_{[T]}^k$)

Definition

Scenario

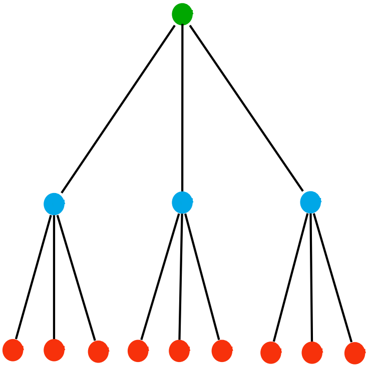
- Sequence of outcomes $\zeta_1^k, \ldots, \zeta_T^k$ of $\zeta_1, \ldots, \zeta_T$
- Outcome $\zeta_{[T]}^k$ of $\zeta_{[T]} = (\zeta_1, \ldots, \zeta_T)$

Idea

Introduce sequence of decision vectors $(x_0^k, x_1^k, \ldots, x_T^k)$ ($\forall$ scenarios $\zeta_{[T]}^k$)

Problem

Scenario Tree



Definition

Scenario

- Sequence of outcomes $\zeta_1^k, \ldots, \zeta_T^k$ of $\zeta_1, \ldots, \zeta_T$
- Outcome $\zeta_{[T]}^k$ of $\zeta_{[T]} = (\zeta_1, \ldots, \zeta_T)$

Idea

Introduce sequence of decision vectors $(x_0^k, x_1^k, \ldots, x_T^k)$ ($\forall$ scenarios $\zeta_{[T]}^k$)

Problem

Linköping University

Definition

Scenario

- Sequence of outcomes $\zeta_1^k, \dots, \zeta_T^k$ of $\zeta_1, \dots, \zeta_T$
- Outcome $\zeta_{[T]}^k$ of $\zeta_{[T]} = (\zeta_1, \dots, \zeta_T)$

Idea

Introduce sequence of decision vectors $(x_0^k, x_1^k, \dots, x_T^k)$ ($\forall$ scenarios $\zeta_{[T]}^k$)

Problem

Some decision vectors should be equal (e.g. $x_0^k = x_0^{k'}$ $\forall k, k'$)

Definition

Scenario

- Sequence of outcomes $\zeta_1^k, \ldots, \zeta_T^k$ of $\zeta_1, \ldots, \zeta_T$
- Outcome $\zeta_{[T]}^k$ of $\zeta_{[T]} = (\zeta_1, \ldots, \zeta_T)$

Idea

Introduce sequence of decision vectors $(x_0^k, x_1^k, \ldots, x_T^k)$ ($\forall$ scenarios $\zeta_{[T]}^k$)

Problem

Some decision vectors should be equal (e.g. $x_0^k = x_0^{k'} \ \forall k, k'$)

Solution

Definition

Scenario

- Sequence of outcomes $\zeta_1^k, \ldots, \zeta_T^k$ of $\zeta_1, \ldots, \zeta_T$
- Outcome $\zeta_{[T]}^k$ of $\zeta_{[T]} = (\zeta_1, \ldots, \zeta_T)$

Idea

Introduce sequence of decision vectors $(x_0^k, x_1^k, \ldots, x_T^k)$ ($\forall$ scenarios $\zeta_{[T]}^k$)

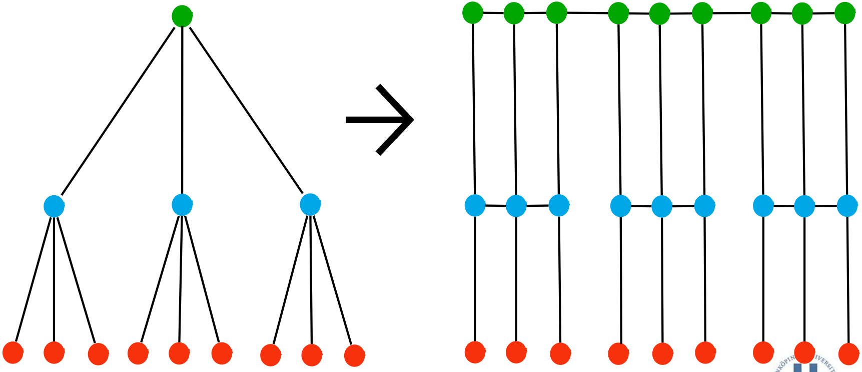
Problem

Some decision vectors should be equal (e.g. $x_0^k = x_0^{k'}$ $\forall k, k'$)

Solution

Non-anticipativity constraints: $x_t^k = x_t^{k'}$ if $\zeta_{[t]}^k = \zeta_{[t]}^{k'}$

Scenario Tree



Definition

Scenario

- Sequence of outcomes $\zeta_1^k, \ldots, \zeta_T^k$ of $\zeta_1, \ldots, \zeta_T$
- Outcome $\zeta_{[T]}^k$ of $\zeta_{[T]} = (\zeta_1, \ldots, \zeta_T)$

Idea

Introduce sequence of decision vectors $(x_0^k, x_1^k, \ldots, x_T^k)$ ($\forall$ scenarios $\zeta_{[T]}^k$)

Problem

Some decision vectors should be equal (e.g. $x_0^k = x_0^{k'}$ $\forall k, k'$)

Solution

Non-anticipativity constraints: $x_t^k = x_t^{k'}$ if $\zeta_{[t]}^k = \zeta_{[t]}^{k'}$

Outline

Aim

Solve Stochastic Programming problems approximately...

Aim

Solve Stochastic Programming problems approximately...

1) ...whose objective function is computationally hard to evaluate.

Linköping University

Aim

Solve Stochastic Programming problems approximately...

1) ...whose objective function is computationally hard to evaluate.
2) ...that involve continuously distributed random parameters.

Aim

Solve Stochastic Programming problems approximately...

1) ...whose objective function is computationally hard to evaluate.
2) ...that involve continuously distributed random parameters.
3) ...that involve too many scenarios.

Aim

Solve Stochastic Programming problems approximately...

1) ...whose objective function is computationally hard to evaluate.
2) ...that involve continuously distributed random parameters.
3) ...that involve too many scenarios.
4) ...whose underlying distribution is unknown ("Black-Box model").

Linköping University

Aim

Solve Stochastic Programming problems approximately...

1) ...whose objective function is computationally hard to evaluate.
2) ...that involve continuously distributed random parameters.
3) ...that involve too many scenarios.
4) ...whose underlying distribution is unknown ("Black-Box model").
5) ...are structural difficult (lack of convexity, linearity, etc.).

Outer Approximation

Outer Approximation

- Problem Approximated

Outer Approximation

- Problem Approximated
- Underlying distribution replaced by finite random sample

Outer Approximation

- Problem Approximated
- Underlying distribution replaced by finite random sample
- Deterministic reformulation

Linköping University

Outer Approximation

- Problem Approximated
- Underlying distribution replaced by finite random sample
- Deterministic reformulation
- Famous example: Sample Average Approach

Linköping University

Inner Approximation

Inner Approximation

- Randomized Solution Algorithm

Inner Approximation

- Randomized Solution Algorithm
- Sampling during solution process

Inner Approximation

- Randomized Solution Algorithm
- Sampling during solution process
- Either: Find good solution over iterations

Linköping University

Inner Approximation

- Randomized Solution Algorithm
- Sampling during solution process
- Either: Find good solution over iterations
- Or: Problem approximated over iterations

Inner Approximation

- Randomized Solution Algorithm
- Sampling during solution process
- Either: Find good solution over iterations
- Or: Problem approximated over iterations
- Famous examples:

Linköping University

Inner Approximation

- Randomized Solution Algorithm
- Sampling during solution process
- Either: Find good solution over iterations
- Or: Problem approximated over iterations
- Famous examples:

  $\rightarrow$ Stochastic gradient algorithm (Stochastic approximation)

Linköping University

Inner Approximation

- Randomized Solution Algorithm
- Sampling during solution process
- Either: Find good solution over iterations
- Or: Problem approximated over iterations
- Famous examples:

  → Stochastic gradient algorithm (Stochastic approximation)
  → Stochastic Decomposition

Linköping University

Outline

1. Multi-Stage Programming
   - Modeling Multi-Stage Stochastic Programming Problems
   - Deterministic Reformulation in case of discrete distributions (Main Idea)
2. Outer and Inner Approximation Approaches
   - Sample Average Approximation
   - Stochastic Gradient method

Basic Idea

Basic Idea

- Sample uniformly N random vectors from distribution

Basic Idea

- Sample uniformly N random vectors from distribution
- Assign probability $p_k = \frac{1}{N}$ $(k = 1, \ldots, N)$

Linköping University

Basic Idea

- Sample uniformly N random vectors from distribution
- Assign probability $p_k = \frac{1}{N}$ $(k = 1, \ldots, N)$
- Replace underlying distribution by obtained discrete finite distribution

Basic Idea

- Sample uniformly N random vectors from distribution
- Assign probability $p_k = \frac{1}{N}$ $(k = 1, \ldots, N)$
- Replace underlying distribution by obtained discrete finite distribution
- Solve approximated problem $\mathcal{P}(\chi^1, \ldots, \chi^N)$

Linköping University

Basic Idea

- Sample uniformly N random vectors from distribution
- Assign probability $p_k = \frac{1}{N}$ $(k = 1, \dots, N)$
- Replace underlying distribution by obtained discrete finite distribution
- Solve approximated problem $\mathcal{P}(\chi^1, \dots, \chi^N)$
- Hopefully:

Linköping University

Basic Idea

- Sample uniformly N random vectors from distribution
- Assign probability $p_k = \frac{1}{N}$ $(k = 1, \ldots, N)$
- Replace underlying distribution by obtained discrete finite distribution
- Solve approximated problem $\mathcal{P}(\chi^1, \ldots, \chi^N)$
- Hopefully:

1) $x(\mathcal{P}(\chi^1, \ldots, \chi^N)) \to x^*$ as $N \to \infty$

Basic Idea

- Sample uniformly N random vectors from distribution
- Assign probability $p_k = \frac{1}{N}$ $(k = 1, \dots, N)$
- Replace underlying distribution by obtained discrete finite distribution
- Solve approximated problem $\mathcal{P}(\chi^1, \dots, \chi^N)$
- Hopefully:

1) $x(\mathcal{P}(\chi^1, \dots, \chi^N)) \to x^*$ as $N \to \infty$

2) $\mathbb{P}\{x(\mathcal{P}(\chi^1, \dots, \chi^N)) = x^*\}$ increases as $N \to \infty$

Problem type (in general)

Problem type (in general)

$$\begin{aligned} \text{min} \quad & \mathbb{E}[f(x,\chi)] \\ \text{s.t.} \quad & x \in X \end{aligned}$$

Problem type (in general)

$$\begin{aligned} \min \quad & \mathbb{E}[f(x,\chi)] \\ \text{s.t.} \quad & x \in X \end{aligned}$$

X independent of distribution of random vector

Problem type (in general)

$$\begin{aligned} \min \quad & \mathbb{E}[f(x,\chi)] \\ \text{s.t.} \quad & x \in X \end{aligned}$$

X independent of distribution of random vector

Includes:

Problem type (in general)

$$\begin{aligned} \min \quad & \mathbb{E}[f(x,\chi)] \\ \text{s.t.} \quad & x \in X \end{aligned}$$

X independent of distribution of random vector

Includes:

- Minimization of expected value

Problem type (in general)

$$\begin{aligned} \min \quad & \mathbb{E}[f(x,\chi)] \\ \text{s.t.} \quad & x \in X \end{aligned}$$

X independent of distribution of random vector

Includes:

- Minimization of expected value
- Simple Recourse problems

Linköping University

Problem type (in general)

$$\begin{aligned} \min \quad & \mathbb{E}[f(x,\chi)] \\ \text{s.t.} \quad & x \in X \end{aligned}$$

Simple-Recourse problem

$$\min_{x \in X} \quad f(x) + \sum_{i=1}^{m} d_i \cdot \mathbb{E}\left[[g_i(x,\chi)]^+\right]$$

- $d_i > 0 \ \forall i \in \{1, \ldots, m\}$
- $[x]^+ = max(0, x)$

Problem type (in general)

$$\begin{aligned} \min \quad & \mathbb{E}[f(x,\chi)] \\ \text{s.t.} \quad & x \in X \end{aligned}$$

X independent of distribution of random vector

Includes:

- Minimization of expected value
- Simple Recourse problems

Problem type (in general)

$$\begin{aligned} \min \quad & \mathbb{E}[f(x,\chi)] \\ \text{s.t.} \quad & x \in X \end{aligned}$$

X independent of distribution of random vector

Includes:

- Minimization of expected value
- Simple Recourse problems
- Two-Stage Programming problems

Linköping University

Problem type (in general)

$$\begin{aligned} \min \quad & \mathbb{E}[f(x,\chi)] \\ \text{s.t.} \quad & x \in X \end{aligned}$$

Linear Two-Stage Problem

$$\begin{aligned} \min_{x \geq 0} \quad & c^T x + \mathbb{E}[\mathcal{Q}(x,\chi)] \\ \text{s.t.} \quad & Ax \geq b, \\ & \mathcal{Q}(x,\chi) = \min_{y \geq 0} \quad d^T y \\ & \qquad\qquad\qquad \text{s.t.} \quad T(\chi)x + W(\chi)y \geq h(\chi). \end{aligned}$$

Problem type (in general)

$$\begin{aligned} \min \quad & \mathbb{E}[f(x,\chi)] \\ \text{s.t.} \quad & x \in X \end{aligned}$$

X independent of distribution of random vector

Includes:

- Minimization of expected value
- Simple Recourse problems
- Two-Stage Programming problems

Linköping University

Problem type (in general)

$$\begin{array}{ll} \min & \mathbb{E}[f(x,\chi)] \\ \text{s.t.} & x \in X \end{array}$$

Problem type (in general)

$$\begin{aligned} \min \quad & \mathbb{E}[f(x,\chi)] \\ \text{s.t.} \quad & x \in X \end{aligned}$$

Does not (in general) include:

Linköping University

Problem type (in general)

$$\begin{aligned} \min \quad & \mathbb{E}[f(x,\chi)] \\ \text{s.t.} \quad & x \in X \end{aligned}$$

Does not (in general) include:

- Chance-Constrained programming problems (Feasible set described by CDF)

Chance-Constrained Stochastic Optimization Problem

$$\begin{aligned} \min_{x \in X} \quad & f(x) \\ \text{s.t.} \quad & \mathbb{P}\{G(x,\chi) \leq 0\} \geq p \end{aligned}$$

Problem type (in general)

$$\begin{aligned} \min \quad & \mathbb{E}[f(x,\chi)] \\ \text{s.t.} \quad & x \in X \end{aligned}$$

Does not (in general) include:

- Chance-Constrained programming problems (Feasible set described by CDF)

Problem type (in general)

$$\begin{aligned} \min \quad & \mathbb{E}[f(x,\chi)] \\ \text{s.t.} \quad & x \in X \end{aligned}$$

Does not (in general) include:

- Chance-Constrained programming problems (Feasible set described by CDF)
- Multi-Stage Programming (conditional sampling needed)

Deterministic Opt. Model → Stochastic Programming Model

$$\begin{aligned} \max_{x \in X} \quad & f(x) \\ \text{s.t.} \quad & G(x) \leq 0 \end{aligned} \quad \rightarrow \quad \begin{aligned} \min_{x \in X} \quad & f(x, \chi) \\ \text{s.t.} \quad & G(x, \chi) \leq 0 \end{aligned}$$

$\chi \in \Omega \subseteq \mathbb{R}^s$: random vector

Stochastic Programming Model → Deterministic Equivalent Model

Stochastic Programming Model → Deterministic Equivalent Model

$$\min_{x \in X} \quad \mathbb{E}[f(x,\chi)]$$

$\chi \in \Omega \subseteq \mathbb{R}^s$: random vector

Stochastic Programming Model → Deterministic Equivalent Model

$$\min_{x \in X} \quad \mathbb{E}[f(x,\chi)]$$

$\chi \in \Omega \subseteq \mathbb{R}^s$: random vector

$$\xrightarrow{SAA}$$

Stochastic Programming Model → Deterministic Equivalent Model

$$\min_{x \in X} \quad \mathbb{E}[f(x, \chi)] \quad \xrightarrow{SAA} \quad \min_{x \in X} \quad \frac{1}{N} \sum_{k=1}^{N} f(x, \chi^k)$$

$\chi \in \Omega \subseteq \mathbb{R}^s$: random vector

Stochastic Programming Model → Deterministic Equivalent Model

$$\min_{x \in X} \quad \mathbb{E}[f(x,\chi)] \quad \xrightarrow{SAA} \quad \min_{x \in X} \quad \frac{1}{N}\sum_{k=1}^{N} f(x,\chi^k)$$

$\chi \in \Omega \subseteq \mathbb{R}^s$: random vector

$\chi^1, \ldots, \chi^N$: random sample

Linköping University

Convergence results

Convergence results

- $\hat{x}_N$: optimal solution of SAA with N samples ($\hat{x}_N$ is random variable)

Convergence results

- $\hat{x}_N$: optimal solution of SAA with N samples ($\hat{x}_N$ is random variable)
- A: set of optimal solutions of the exact problem

Convergence results

- $\hat{x}_N$: optimal solution of SAA with N samples ($\hat{x}_N$ is random variable)
- A: set of optimal solutions of the exact problem

Linköping University

Convergence results

- $\hat{x}_N$: optimal solution of SAA with N samples ($\hat{x}_N$ is random variable)
- A: set of optimal solutions of the exact problem

General distributions

Under some mild (technical) assumptions:

Convergence results

- $\hat{x}_N$: optimal solution of SAA with N samples ($\hat{x}_N$ is random variable)
- A: set of optimal solutions of the exact problem

General distributions

Under some mild (technical) assumptions:

- $\text{dist}_A(\hat{x}_N) \to 0$ as $N \to \infty$ (w.p.1)

Linköping University

Convergence results

- $\hat{x}_N$: optimal solution of SAA with N samples ($\hat{x}_N$ is random variable)
- A: set of optimal solutions of the exact problem

General distributions

Under some mild (technical) assumptions:

- $\text{dist}_A(\hat{x}_N) \to 0$ as $N \to \infty$ (w.p.1)
- Convergence rate: order $N^{-\frac{1}{2}}$

Linköping University

Convergence results

- $\hat{x}_N$: optimal solution of SAA with N samples ($\hat{x}_N$ is random variable)
- A: set of optimal solutions of the exact problem

General distributions

Under some mild (technical) assumptions:

- $\text{dist}_A(\hat{x}_N) \to 0$ as $N \to \infty$ (w.p.1)
- Convergence rate: order $N^{-\frac{1}{2}}$
- $\forall \epsilon > 0$: $\mathbb{P}\{\text{dist}_A(\hat{x}_N) \le \epsilon\}$ increases exponentially

Convergence results

- $\hat{x}_N$: optimal solution of SAA with N samples ($\hat{x}_N$ is random variable)
- A: set of optimal solutions of the exact problem

Discrete distributions

Convergence results

- $\hat{x}_N$: optimal solution of SAA with N samples ($\hat{x}_N$ is random variable)
- A: set of optimal solutions of the exact problem

Discrete distributions

- $\forall \chi \in \Omega \; f(\cdot, \chi)$ is convex

Convergence results

- $\hat{x}_N$: optimal solution of SAA with N samples ($\hat{x}_N$ is random variable)
- A: set of optimal solutions of the exact problem

Discrete distributions

- $\forall \chi \in \Omega \; f(\cdot, \chi)$ is convex
- $\mathbb{E}[f(\cdot, \chi)]$ is well defined and is finite valued

Convergence results

- $\hat{x}_N$: optimal solution of SAA with N samples ($\hat{x}_N$ is random variable)
- A: set of optimal solutions of the exact problem

Discrete distributions

- $\forall \chi \in \Omega \; f(\cdot, \chi)$ is convex
- $\mathbb{E}[f(\cdot, \chi)]$ is well defined and is finite valued
- X is closed and convex

Convergence results

- $\hat{x}_N$: optimal solution of SAA with N samples ($\hat{x}_N$ is random variable)
- A: set of optimal solutions of the exact problem

Discrete distributions

- $\forall \chi \in \Omega \ f(\cdot, \chi)$ is convex
- $\mathbb{E}[f(\cdot, \chi)]$ is well defined and is finite valued
- X is closed and convex
- the exact problem has a unique optimal solution x^* s.t.

$$\exists c > 0 : \forall x \in X : \quad \mathbb{E}[f(x, \chi)] - \mathbb{E}[f(x^*, \chi)] \geq c \cdot \text{dist}(x, x^*)$$

Convergence results

- $\hat{x}_N$: optimal solution of SAA with N samples ($\hat{x}_N$ is random variable)
- A: set of optimal solutions of the exact problem

Discrete distributions

- $\forall \chi \in \Omega \; f(\cdot, \chi)$ is convex
- $\mathbb{E}[f(\cdot, \chi)]$ is well defined and is finite valued
- X is closed and convex
- the exact problem has a unique optimal solution x^* s.t.

$$\exists c > 0 : \forall x \in X : \quad \mathbb{E}[f(x, \chi)] - \mathbb{E}[f(x^*, \chi)] \geq c \cdot \mathrm{dist}(x, x^*)$$

Convergence results

- $\hat{x}_N$: optimal solution of SAA with N samples ($\hat{x}_N$ is random variable)
- A: set of optimal solutions of the exact problem

Discrete distributions

- $\forall \chi \in \Omega \; f(\cdot, \chi)$ is convex
- $\mathbb{E}[f(\cdot, \chi)]$ is well defined and is finite valued
- X is closed and convex
- the exact problem has a unique optimal solution x^* s.t.

$$\exists c > 0 : \forall x \in X : \quad \mathbb{E}[f(x, \chi)] - \mathbb{E}[f(x^*, \chi)] \geq c \cdot \text{dist}(x, x^*)$$

N large enough $\overset{wp1}{\Longrightarrow}$ SAA has a **unique optimal solution $\mathbf{\hat{x}_N = x^*}$**

Further results

Further results

- Sample size (e.g. for ϵ-exact solutions) (with given probability bound)

Further results

- Sample size (e.g. for ϵ-exact solutions) (with given probability bound)
- Error bounds

Outline

Linköping University

Basic Idea

Basic Idea

- Basically: Gradient method

Linköping University

Basic Idea

- Basically: Gradient method
- At each iteration: Sample random parameters

Basic Idea

- Basically: Gradient method
- At each iteration: Sample random parameters
- Compute new solution based on this sample

Linköping University

Basic Idea

- Basically: Gradient method
- At each iteration: Sample random parameters
- Compute new solution based on this sample
- Use gradient of function **inside** expectation

Basic Idea

- Basically: Gradient method
- At each iteration: Sample random parameters
- Compute new solution based on this sample
- Use gradient of function **inside** expectation
- Hopefully:

Linköping University

Basic Idea

- Basically: Gradient method
- At each iteration: Sample random parameters
- Compute new solution based on this sample
- Use gradient of function **inside** expectation
- Hopefully:

1) $x^k \to x^*$ as $k \to \infty$ w.h.p.

Problem type (mostly)

Problem type (mostly)

$$\begin{aligned} \min \quad & \mathbb{E}[f(x,\chi)] \\ \text{s.t.} \quad & x \in X \end{aligned}$$

Linköping University

Problem type (mostly)

$$\begin{aligned} \min \quad & \mathbb{E}[f(x,\chi)] \\ \text{s.t.} \quad & x \in X \end{aligned}$$

Problem type (mostly)

$$\begin{aligned} \min \quad & \mathbb{E}[f(x,\chi)] \\ \text{s.t.} \quad & x \in X \end{aligned}$$

Where:

Linköping University

Problem type (mostly)

$$\begin{aligned} \text{min} \quad & \mathbb{E}[f(x,\chi)] \\ \text{s.t.} \quad & x \in X \end{aligned}$$

Where:

- X independent of distribution of random vector

Linköping University

Problem type (mostly)

$$\begin{aligned} \min \quad & \mathbb{E}[f(x,\chi)] \\ \text{s.t.} \quad & x \in X \end{aligned}$$

Where:

- X independent of distribution of random vector
- X convex set

Problem type (mostly)

$$\begin{aligned} \min \quad & \mathbb{E}[f(x,\chi)] \\ \text{s.t.} \quad & x \in X \end{aligned}$$

Where:

- X independent of distribution of random vector
- X convex set
- $f(\cdot,\chi)$ convex

Problem type (mostly)

$$\begin{aligned} \min \quad & \mathbb{E}[f(x,\chi)] \\ \text{s.t.} \quad & x \in X \end{aligned}$$

Where:

- X independent of distribution of random vector
- X convex set
- $f(\cdot,\chi)$ convex
- $f(\cdot,\chi)$ differentiable (nearly everywhere)

Linköping University

Stochastic Gradient Algorithm

Stochastic Gradient Algorithm

$k \leftarrow 0$
Choose x^0 in X

Stochastic Gradient Algorithm

$k \leftarrow 0$
Choose x^0 in X
while $k < K_{max}$ do
  $k \leftarrow k + 1$

end while

Stochastic Gradient Algorithm

$k \leftarrow 0$
Choose x^0 in X
while $k < K_{max}$ do
  $k \leftarrow k + 1$
  Draw $\chi^k = (\chi_1^k, ..., \chi_n^k)$
end while

Stochastic Gradient Algorithm

$k \leftarrow 0$
Choose x^0 in X
while $k < K_{max}$ do
  $k \leftarrow k+1$
  Draw $\chi^k = (\chi_1^k, ..., \chi_n^k)$
  Update x^k as follows:

$$x^{k+1} \leftarrow x^k + \epsilon^k r^k$$

end while

- $r^k = \nabla_x f(x, \chi^k)$

Stochastic Gradient Algorithm

$k \leftarrow 0$
Choose x^0 in X
while $k < K_{max}$ do
  $k \leftarrow k + 1$
  Draw $\chi^k = (\chi_1^k, ..., \chi_n^k)$
  Update x^k as follows:

$$x^{k+1} \leftarrow x^k + \epsilon^k r^k$$

end while

- $r^k = \nabla_x f(x, \chi^k)$
- $(\epsilon^k)_{k \in \mathbb{N}}$ is a σ-sequence

Stochastic Gradient Algorithm

```
k ← 0
Choose x^0 in X
while k < K_max do
  k ← k + 1
  Draw χ^k = (χ_1^k, ..., χ_n^k)
  Update x^k as follows:
```

$$x^{k+1} \leftarrow x^k + \epsilon^k r^k$$

```
end while
```

- $r^k = \nabla_x f(x, \chi^k)$
- $(\epsilon^k)_{k \in \mathbb{N}}$ is a σ-sequence

Stochastic Gradient Algorithm

```
k ← 0
Choose x^0 in X
while k < K_max do
  k ← k + 1
  Draw χ^k = (χ_1^k, ..., χ_n^k)
  Update x^k as follows:
```

$$x^{k+1} \leftarrow x^k + \epsilon^k r^k$$

```
  Project x^{k+1} on X
end while
```

Linköping University

- $r^k = \nabla_x f(x, \chi^k)$
- $(\epsilon^k)_{k \in \mathbb{N}}$ is a σ-sequence

Stochastic Gradient Algorithm

```
k ← 0
Choose x^0 in X
while k < K_max do
  k ← k + 1
  Draw χ^k = (χ_1^k, ..., χ_n^k)
  Update x^k as follows:
```

$$x^{k+1} \leftarrow x^k + \epsilon^k r^k$$

```
  Project x^{k+1} on X
end while
return  Best found or last solution
```

Linköping University

Assumptions

Linköping University

Assumptions

1) $\forall x \in X$: $f(x, \chi)$ is a random variable with finite expectation

Assumptions

1) $\forall x \in X$: $f(x, \chi)$ is a random variable with finite expectation
2) $\forall \chi \in \Omega$: $f(\cdot, \chi)$ is convex, proper, differentiable

Assumptions

1) $\forall x \in X$: $f(x, \chi)$ is a random variable with finite expectation
2) $\forall \chi \in \Omega$: $f(\cdot, \chi)$ is convex, proper, differentiable
3) $\exists m > 0$ s.t.

$$\forall x \in X, \forall \chi \in \Omega : \|\nabla_x f(x, \chi)\| \leq m$$

Linköping University

Assumptions

1) $\forall x \in X$: $f(x, \chi)$ is a random variable with finite expectation
2) $\forall \chi \in \Omega$: $f(\cdot, \chi)$ is convex, proper, differentiable
3) $\exists m > 0$ s.t.
$$\forall x \in X, \forall \chi \in \Omega : \|\nabla_x f(x, \chi)\| \leq m$$
4) $\exists$ set of optimal solutions X^* and $c > 0$ s.t.:
$$\forall x \in X, x^* \in X^* : \quad \mathbb{E}[f(x, \chi)] - \mathbb{E}[f(x^*, \chi)] \geq c \cdot (\text{dist}_{X^*}(x))^2$$

Assumptions

1) $\forall x \in X$: $f(x, \chi)$ is a random variable with finite expectation
2) $\forall \chi \in \Omega$: $f(\cdot, \chi)$ is convex, proper, differentiable
3) $\exists m > 0$ s.t.
$$\forall x \in X, \forall \chi \in \Omega : \|\nabla_x f(x, \chi)\| \leq m$$
4) $\exists$ set of optimal solutions X^* and $c > 0$ s.t.:
$$\forall x \in X, x^* \in X^* : \quad \mathbb{E}[f(x, \chi)] - \mathbb{E}[f(x^*, \chi)] \geq c \cdot (\text{dist}_{X^*}(x))^2$$

Theorem

Under assumptions 1) - 4) we have:

$$\lim_{k \to \infty} \mathbb{E}\left[(dist_{X^*}(x^k))^2\right] = 0$$

Linköping University

Assumptions

1) $\forall x \in X$: $f(x,\chi)$ is a random variable with finite expectation
2) $\forall \chi \in \Omega$: $f(\cdot,\chi)$ is convex, proper, differentiable
3) $\exists m > 0$ s.t.
$$\forall x \in X, \forall \chi \in \Omega : \|\nabla_x f(x,\chi)\| \leq m$$
4) $\exists$ set of optimal solutions X^* and $c > 0$ s.t.:
$$\forall x \in X, x^* \in X^* : \quad \mathbb{E}[f(x,\chi)] - \mathbb{E}[f(x^*,\chi)] \geq c \cdot (\text{dist}_{X^*}(x))^2$$

Assumptions

1) $\forall x \in X$: $f(x,\chi)$ is a random variable with finite expectation
2) $\forall \chi \in \Omega$: $f(\cdot,\chi)$ is convex, proper, differentiable
3) $\exists m > 0$ s.t.

$$\forall x \in X, \forall \chi \in \Omega : \|\nabla_x f(x,\chi)\| \leq m$$

4) $\exists$ set of optimal solutions X^* and $c > 0$ s.t.:

$$\forall x \in X, x^* \in X^* : \quad \mathbb{E}[f(x,\chi)] - \mathbb{E}[f(x^*,\chi)] \geq c \cdot (\text{dist}_{X^*}(x))^2$$

Theorem

Let $d^0 = (dist_{X^*}(x^0))^2$ *and* $\epsilon^k = \frac{1}{ck+\frac{m^2}{cd^0}}$.

ersity

- $r^k = \nabla_x f(x, \chi^k)$
- $(\epsilon^k)_{k \in \mathbb{N}}$ is a σ-sequence

Stochastic Gradient Algorithm

$k \leftarrow 0$
Choose x^0 in X
while $k < K_{max}$ do
  $k \leftarrow k + 1$
  Draw $\chi^k = (\chi_1^k, ..., \chi_n^k)$
  Update x^k as follows:

$$x^{k+1} \leftarrow x^k + \epsilon^k r^k$$

  Project x^{k+1} on X
end while
return Best found or last solution

Assumptions

1) $\forall x \in X$: $f(x, \chi)$ is a random variable with finite expectation
2) $\forall \chi \in \Omega$: $f(\cdot, \chi)$ is convex, proper, differentiable
3) $\exists m > 0$ s.t.
$$\forall x \in X, \forall \chi \in \Omega : \|\nabla_x f(x, \chi)\| \leq m$$
4) $\exists$ set of optimal solutions X^* and $c > 0$ s.t.:
$$\forall x \in X, x^* \in X^* : \quad \mathbb{E}[f(x, \chi)] - \mathbb{E}[f(x^*, \chi)] \geq c \cdot (\text{dist}_{X^*}(x))^2$$

Theorem

Let $d^0 = (dist_{X^*}(x^0))^2$ *and* $\epsilon^k = \frac{1}{ck + \frac{m^2}{cd^0}}$.

ersity

Assumptions

1) $\forall x \in X$: $f(x,\chi)$ is a random variable with finite expectation
2) $\forall \chi \in \Omega$: $f(\cdot,\chi)$ is convex, proper, differentiable
3) $\exists m > 0$ s.t.

$$\forall x \in X, \forall \chi \in \Omega : \|\nabla_x f(x,\chi)\| \leq m$$

4) $\exists$ set of optimal solutions X^* and $c > 0$ s.t.:

$$\forall x \in X, x^* \in X^* : \quad \mathbb{E}[f(x,\chi)] - \mathbb{E}[f(x^*,\chi)] \geq c \cdot (\text{dist}_{X^*}(x))^2$$

Theorem

Let $d^0 = (dist_{X^*}(x^0))^2$ and $\epsilon^k = \frac{1}{ck+\frac{m^2}{cd^0}}$.
Under assumptions 1) - 4) we have:

$$\mathbb{E}\left[(dist_{X^*}(x^k))^2\right] \leq \frac{1}{\frac{c^2}{m^2}k + \frac{1}{d^0}}$$

ersity

QUESTIONS?

What about next week?