Monte Carlo Sampling-Based Methods in Stochastic Programming

Güzin Bayraksan

Integrated Systems Engineering
The Ohio State University

20 May 2013

IIE Annual Conference & Expo
ISERC 2013
1 Introduction
   - Stochastic Programming
   - Need for Monte Carlo Sampling

2 Sample Average Approximation
   - Optimized Sample Averages and Bias
   - Consistency
   - Rates of Convergence

3 Monte Carlo Sampling-Based Solution Methods
   - Introduction
   - Stochastic Approximation
   - Sampling-Based Cutting-Plane Algorithms

4 Assessing Solution Quality and Stopping Rules
   - Solution Quality Estimation
   - Stopping Rules

5 Alternative Sampling Techniques

6 Conclusions
Outline

1 **Introduction**
   - Stochastic Programming
   - Need for Monte Carlo Sampling

2 **Sample Average Approximation**
   - Optimized Sample Averages and Bias
   - Consistency
   - Rates of Convergence

3 **Monte Carlo Sampling-Based Solution Methods**
   - Introduction
   - Stochastic Approximation
   - Sampling-Based Cutting-Plane Algorithms

4 **Assessing Solution Quality and Stopping Rules**
   - Solution Quality Estimation
   - Stopping Rules

5 **Alternative Sampling Techniques**

6 **Conclusions**
A General Stochastic Program

\[
\begin{align*}
\text{(SP)} & \quad \min_x \quad \mathbb{E} f(x, \xi) \\
\text{s.t.} & \quad \mathbb{E} g_i(x, \xi) \leq 0, \quad i = 1, 2, \ldots, m, \\
& \quad x \in X
\end{align*}
\]

\(X \subseteq \mathbb{R}^{d_x}\), set of deterministic constraints
(assume nonempty and compact)

\(f : \mathbb{R}^{d_x} \times \mathbb{R}^{d_\xi} \to \mathbb{R}\), objective function

\(g_i : \mathbb{R}^{d_x} \times \mathbb{R}^{d_\xi} \to \mathbb{R}\), stochastic constraints

\(\xi\) is a random vector on probability space \((\Omega, \mathcal{F}, \mathbb{P})\)

Expectations are taken with respect to the distribution of \(\xi\)
(assume all expectations are finite for all \(x \in X\))
A General Stochastic Program

\[
\begin{align*}
\text{(SP)} & \quad \min_x \mathbb{E}f(x, \xi) \\
\text{s.t.} & \quad \mathbb{E}g_i(x, \xi) \leq 0, \ i = 1, 2, \ldots, m, \\
& \quad x \in X
\end{align*}
\]

Many problems can be cast in this form:

- Simulation (e.g., min. average waiting time)
- Statistics (e.g., M-estimators of Huber, with \( m = 0 \))
- Mathematical Programming
  - one-stage models (e.g., in Risk Management)
  - two-stage or multi-stage stochastic programs with recourse

Our focus is on mathematical programming. Assume distribution of \( \xi \) is independent of \( x \). The type of problem depends on \( f, X, m, \) and \( g_i \), for \( i = 1, 2, \ldots, m \).
A one-stage model — e.g., max probability of exceeding a threshold:

\[ m = 0, \quad \text{and} \quad f(x, \xi) = -\mathbb{I}\left(\sum_{i=1}^{d_x} \xi_i x_i \geq \tau\right), \]

where \( \mathbb{I}(\cdot) \) denotes an indicator function.

Stochastic program with a probabilistic constraint:

\[ m = 1, \quad f(x, \xi) = cx \quad \text{and} \quad g_1(x, \xi) = \mathbb{I}(\tilde{A}'x \geq \tilde{b}') - \alpha, \]

\( \mathbb{I}(\cdot) \): indicator function, \( \alpha \in (0, 1), \xi = (\tilde{A}', \tilde{b}') \)
A simple recourse model — e.g., newsvendor problem:

- $x$ – amount of inventory to obtain
- $\xi$ – demand
- $v < c < r$
- $c$ – unit cost
- $r$ – unit revenue
- $v$ – salvage value
A simple recourse model — e.g., newsvendor problem:

- \( x \) — amount of inventory to obtain
- \( \xi \) — demand
- \( v < c < r \)
- \( c \) — unit cost
- \( r \) — unit revenue
- \( v \) — salvage value

Maximizing expected profit:

\[
f(x, \xi) = - [cx - r \min\{x, \xi\} - v \max\{x - \xi, 0\}]
\]

\( m = 0 \), and \( X = \{x : x \geq 0\} \).
A General Stochastic Program: Special Cases

• A simple recourse model — e.g., newsvendor problem:
  - \( x \) — amount of inventory to obtain
  - \( \xi \) — demand
  - \( v < c < r \)
  - \( c \) — unit cost
  - \( r \) — unit revenue
  - \( v \) — salvage value

Maximizing expected profit:

\[
f(x, \xi) = -[cx - r \min\{x, \xi\} - v \max\{x - \xi, 0\}]
\]

\( m = 0 \), and \( X = \{x : x \geq 0\} \).

Set of optimal solutions is the \( \gamma \)-quantiles (\( \gamma = (r - c)/(r - v) \)) of \( \xi \),

\[
X^* = \{z \in \mathbb{R} : P(\xi \geq z) \geq 1 - \gamma \text{ and } P(\xi \leq z) \geq \gamma\}
\]
Two-stage stochastic linear program with recourse:

\[ m = 0, \ X = \{ Ax = b, x \geq 0 \}, \text{ and } f(x, \xi) = cx + h(x, \xi), \text{ where } \]

\[ h(x, \xi) = \min_y \tilde{q} y \]

\[ \text{s.t. } \tilde{W} y = \tilde{r} - \tilde{T} x, \ y \geq 0. \]

\[ \xi = (\tilde{q}, \tilde{W}, \tilde{R}, \tilde{T}) \]
Two-stage stochastic linear program with recourse:

\[ m = 0, \ X = \{ Ax = b, x \geq 0 \}, \ \text{and} \ f(x, \xi) = cx + h(x, \xi), \ \text{where} \]
\[
h(x, \xi) = \min_y \tilde{q}y
\]
\[
\text{s.t.} \ \tilde{W}y = \tilde{r} - \tilde{T}x, \ y \geq 0.
\]
\[
\xi = (\tilde{q}, \tilde{W}, \tilde{R}, \tilde{T})
\]

Example: Power network with uncertain demand (Infanger, 1992):

1st-stage: What capacities to build the power plants?

2nd-stage: Purchase additional capacities to fulfill unmet demands
Multi-stage stochastic linear program with recourse:

\[
\begin{align*}
\min_{x_1} & \quad c_1 x_1 + \mathbb{E} \left[ Q_2(x_1, \xi^2) | \xi^1 \right] \\
\text{s.t.} & \quad A_1 x_1 = b_1, \\
& \quad x_1 \geq 0,
\end{align*}
\]

where, for \( t = 2, \ldots, T \),

\[
Q_t(x_{t-1}, \xi^t) = \min_{x_t} \tilde{c}_t x_t + \mathbb{E} \left[ Q_t(x_t, \xi^{t+1}) | \xi^t \right]
\]

\[
\text{s.t.} \quad \tilde{A}_t x_t = \tilde{b}_t - \tilde{B}_t x_{t-1}, \\
& \quad x_t \geq 0
\]

and \( Q_{T+1} = 0 \). Here,

\( \xi_t = (\tilde{c}_t, \tilde{A}_t, \tilde{B}_t, \tilde{b}_t) \) at stage \( t \), \( \xi_1 \) is deterministic, and

\( \xi^t \) denotes the history of the process till time \( t \).
Multi-stage stochastic linear program with recourse: $\xi$ modeled or approximated as a “tree”
Example: Lower Colorado River Basin Water Allocation

How to best allocate water among different users (agriculture, hydroelectric, municipal, etc.) while meeting uncertain water demand and not exceeding uncertain water supply over the next 50 years?

*Modeled as a multistage generalized stochastic network*
A major challenge:

- Calculating the multidimensional expectations that appear in (SP)—even for a fixed $x$. 

A major challenge:

- Calculating the multidimensional expectations that appear in (SP)—even for a fixed $x$.

Calculating Multidimensional Integrals:

- Analytical integration:
  Exact method but often impossible

- Quadrature methods:
  Deterministic error bounds but limited in dimension $d_{\xi}$

- Monte Carlo sampling-based integration:
  Statistical error bounds, can be applied to high dimensions $d_{\xi}$
Another major challenge:

- Even if the expectations can be calculated for a fixed $x$, optimization of $x \in X$ can be challenging: nonconvex, discontinuous.
Another major challenge:

- Even if the expectations can be calculated for a fixed $x$, optimization of $x \in X$ can be challenging: nonconvex, discontinuous.

Solution Methods:

- *Exact solution methods*: Exact but often impossible

- *Bounding approximations*: Deterministic error bounds but limited

- *Monte Carlo sampling-based methods*: Statistical error bounds, intuitive to apply, can be external/internal to an optimization algorithm
Monte Carlo Sampling

Replace expectations in (SP) with Monte Carlo sampling-based estimators:

\[(SP) \quad \min_x \mathbb{E}f(x, \xi) \]
\[\text{s.t. } \mathbb{E}g_i(x, \xi) \leq 0, \quad i = 1, 2, \ldots, m, \]
\[x \in X\]

While other sampling schemes are possible, let \(\{\xi^1, \xi^2, \ldots, \xi^N\}\) be a random sample from the distribution of \(\xi\). Then,

\[(SP_N) \quad \min_x \frac{1}{N} \sum_{j=1}^{N} f(x, \xi^j) \]
\[\text{s.t. } \frac{1}{N} \sum_{j=1}^{N} g_i(x, \xi^j) \leq 0, \quad i = 1, 2, \ldots, m, \]
\[x \in X\]
MC Framework ("Algorithm"):

Step 1: Choose an initial solution $x^0$; let $k := 1$

Step 2: Generate a sample of size $N_k \{\xi^1, \xi^2, \ldots, \xi^{N_k}\}$

Step 3: Perform some optimization steps on the Monte Carlo estimator $(SP_{N_k})$ of (SP) (perhaps using information from previous iterations) to obtain $x^k$

Step 4: Check some stopping criteria; if not satisfied, set $k := k + 1$ and go back to Step 2.
MC Framework ("Algorithm"):

Step 1: Choose an initial solution $x^0$; let $k := 1$

Step 2: Generate a sample of size $N_k \{\xi_1, \xi_2, \ldots, \xi_{N_k}\}$

Step 3: Perform some optimization steps on the Monte Carlo estimator $(SP_{N_k})$ of $SP$ (perhaps using information from previous iterations) to obtain $x^k$

Step 4: Check some stopping criteria; if not satisfied, set $k := k + 1$ and go back to Step 2.

"Algorithm" to be interpreted as a framework. A specific example:
MC Framework ("Algorithm"):

Step 1: Choose an initial solution $x^0$; let $k := 1$

Step 2: Generate a sample of size $N_k \{\xi_1^1, \xi_2^2, \ldots, \xi_{N_k}^{N_k}\}$

Step 3: Perform some optimization steps on the Monte Carlo estimator $(SP_{N_k})$ of $(SP)$ (perhaps using information from previous iterations) to obtain $x^k$

Step 4: Check some stopping criteria; if not satisfied, set $k := k + 1$ and go back to Step 2.

"Algorithm" to be interpreted as a framework. A specific example:

Sample Average Approximation (SAA):

In Step 3: Solve the resulting problem $(SP_{N_k})$

In Step 4: Stop at iteration $k = 1$
Monte Carlo Sampling: Questions on Implementing Framework

- What happens as the sample size increases?
- What can be said about the quality of the solution obtained?
- What sample size $N_k$ to use at each iteration?
- How should this sample be generated? Should one use crude Monte Carlo or can other methods—for instance, aimed to reduce variability—be used?
- How to perform an optimization step in Step 3 and how many steps should be taken?
- How to test the stopping criteria in Step 4 in the presence of sampling-based estimators?
Suppose there are no stochastic constraints \((m = 0)\)
and
consider a random (i.i.d.) sample, unless otherwise noted.
Optimized vs. Non-Optimized Sample Means

Regular Sample Mean:

- Let $Y := f(x, \xi)$ for a fixed $x \in X$ (No optimization involved)
  
  Mean $\mu_Y = \mathbb{E}f(x, \xi)$, Variance $\sigma_Y^2 = \text{Var}(f(x, \xi)) < \infty$

- Let $\xi^1, \xi^2, \ldots, \xi^N$ be a random sample and let $Y^1, \ldots, Y^N$ be the corresponding random sample

- Let
  
  $$\bar{Y}_N = \frac{1}{N} \sum_{j=1}^{N} Y^j = \frac{1}{N} \sum_{j=1}^{N} f(x, \xi^j)$$
Optimized vs. Non-Optimized Sample Means

Regular Sample Mean:

- Let $Y := f(x, \xi)$ for a fixed $x \in X$ (No optimization involved)
  Mean $\mu_Y = \mathbb{E}f(x, \xi)$, Variance $\sigma^2_Y = \text{Var}(f(x, \xi)) < \infty$

- Let $\xi^1, \xi^2, \ldots, \xi^N$ be a random sample and let $Y^1, \ldots, Y^N$ be the corresponding random sample

- Let

$$\bar{Y}_N = \frac{1}{N} \sum_{j=1}^{N} Y^j = \frac{1}{N} \sum_{j=1}^{N} f(x, \xi^j)$$

Observations:

1. **Unbiased Estimator:** $\mathbb{E}\bar{Y}_N = \mu_Y$
Optimized vs. Non-Optimized Sample Means

Regular Sample Mean:

- Let \( Y := f(x, \xi) \) for a fixed \( x \in X \) (No optimization involved)
  - Mean \( \mu_Y = E f(x, \xi) \), Variance \( \sigma_Y^2 = \text{Var}(f(x, \xi)) < \infty \)
- Let \( \xi_1, \xi_2, \ldots, \xi_N \) be a random sample and let \( Y^1, \ldots, Y^N \) be the corresponding random sample
- Let
  \[
  \bar{Y}_N = \frac{1}{N} \sum_{j=1}^{N} Y^j = \frac{1}{N} \sum_{j=1}^{N} f(x, \xi^j)
  \]

Observations:

1. **Unbiased Estimator:** \( E \bar{Y}_N = \mu_Y \)
2. **Strong Consistency:** \( \bar{Y}_N \to \mu_Y \), with probability one (wp1) as \( N \to \infty \)
Regular Sample Mean:

- Let $Y := f(x, \xi)$ for a fixed $x \in X$ (No optimization involved)
  Mean $\mu_Y = \mathbb{E}f(x, \xi)$, Variance $\sigma_Y^2 = \text{Var}(f(x, \xi)) < \infty$

- Let $\xi^1, \xi^2, \ldots, \xi^N$ be a random sample and let $Y^1, \ldots, Y^N$ be the corresponding random sample

- Let

$$\bar{Y}_N = \frac{1}{N} \sum_{j=1}^{N} Y^j = \frac{1}{N} \sum_{j=1}^{N} f(x, \xi^j)$$

Observations:

1. **Unbiased Estimator:** $\mathbb{E} \bar{Y}_N = \mu_Y$

2. **Strong Consistency:** $\bar{Y}_N \rightarrow \mu_Y$, with probability one (wp1) as $N \rightarrow \infty$

3. **Normal Errors:** $\sqrt{N}(\bar{Y}_N - \mu_Y) \Rightarrow \text{Normal}(0, \sigma_Y^2)$
Example of an Optimized Sample Mean:

\[ z^* = \min_x \mathbb{E} [\xi x] \]

s.t. \[-1 \leq x \leq 1\]

with \(\xi \sim \text{Normal}(0, 1)\)
Example of an Optimized Sample Mean:

\[ z^* = \min_x \mathbb{E} [\xi x] \quad \text{s.t.} \quad -1 \leq x \leq 1 \]

with \( \xi \sim \text{Normal}(0, 1) \)

- Optimal value
  \[ z^* = 0 \]

- Set of optimal solutions
  \[ X^* = [-1, 1] \]
Example of an Optimized Sample Mean:

\[ z^* = \min_x \mathbb{E} [\xi x] \]

s.t. \(-1 \leq x \leq 1\)

with \(\xi \sim \text{Normal}(0, 1)\)

- Optimal value \(z^* = 0\)

- Set of optimal solutions \(X^* = [-1, 1]\)

Optimal value \(z_N^* = \min_x \frac{1}{N} \sum_{j=1}^{N} \xi_j x\)

s.t. \(-1 \leq x \leq 1\)
Example of an Optimized Sample Mean:

\[ z^* = \min_x E[\xi x] \]

s.t. \(-1 \leq x \leq 1\)

with \(\xi \sim \text{Normal}(0, 1)\)

- Optimal value
  \(z^* = 0\)

- Set of optimal solutions
  \(X^* = [-1, 1]\)

\[ z^*_N = \min_x \frac{1}{N} \sum_{j=1}^{N} \xi^j x \]

s.t. \(-1 \leq x \leq 1\)

- Optimal value
  \(z^*_N = -\left| \frac{1}{N} \sum_{j=1}^{N} \xi^j \right|\)

- Optimal solution
  \(x^* = 1\) or \(x^* = -1\)
Example of an Optimized Sample Mean — Observations:

1. **Negative Bias:** \( \mathbb{E} z_N^* \leq z^* , \forall N \)

2. **Strong Consistency:** \( z_N^* \rightarrow z^* , \text{ wp1} \)

3. **Non-Normal Errors:** \( \sqrt{N}(z_N^* - z^*) = -|\text{Normal}(0,1)| \)
Example of an Optimized Sample Mean — Observations:

1. **Negative Bias:** \[ \mathbb{E} z^*_N \leq z^*, \forall N \]

2. **Strong Consistency:** \[ z^*_N \rightarrow z^*, \text{ wp}1 \]

3. **Non-Normal Errors:** \[ \sqrt{N}(z^*_N - z^*) = -|Normal(0, 1)| \]

4. In the above problem, if we change the feasible region from \( X = \{x: -1 \leq x \leq 1\} \) to \( X = \mathbb{R} \), we lose 2 and 3 above

So, optimization changes the nature of sample means!
Example of an Optimized Sample Mean — Observations:

1. **Negative Bias:** \( \mathbb{E} z^*_N \leq z^*, \forall N \)

2. **Strong Consistency:** \( z^*_N \rightarrow z^*, \text{ wp1} \)

3. **Non-Normal Errors:** \( \sqrt{N}(z^*_N - z^*) = -|\text{Normal}(0,1)| \)

4. In the above problem, if we change the feasible region from \( X = \{x: -1 \leq x \leq 1\} \) to \( X = \mathbb{R} \), we lose 2 and 3 above

So, optimization changes the nature of sample means!
1 Negative Bias: Let $\xi^1, \xi^2, \ldots, \xi^N$ satisfy
\[ E \left[ \frac{1}{N} \sum_{j=1}^{N} f(x, \xi^j) \right] = E f(x, \xi), \ \forall x \in X \]
Then,
\[ E z_N^* \leq z^*. \]

2 Monotonically Decreasing Bias: Let $\xi^1, \xi^2, \ldots, \xi^N$ be a random sample from the distribution of $\xi$. Then,
\[ E z_N^* \leq E z_{N+1}^* \leq z^*. \]
That is, the bias of the estimator $z_N^*$ shrinks as the sample size increases.

(Norkin, Pflug, Ruszczyński 1998) and (Mak, Morton, Wood, 1999)
Now, what happens as $N$ increases?

Do the estimators $z_N^*$ and $x_N^*$ "converge" to their counterparts $z^*$ and $x^*$, wp1?

At what rate?
Back to Newsvendor (NV) Problem: SAA problems with $N = 10, 30, 90, 270$ observations

**NV with Exponential Demand:**
- $r = 6, c = 5, v = 1$
- Exponentially distributed $\xi$ with mean 10
- Unique optimal solution

**NV with Discrete Uniform Demand:**
- $r = 6, c = 5, v = 1$
- Discrete Uniform $\xi$ on $\{1, 2, \ldots, 10\}$
- Multiple optimal solutions
Back to Newsvendor (NV) Problem: SAA problems with $N = 10, 30, 90, 270$ observations

**NV with Exponential Demand:**
- $r = 6, c = 5, \nu = 1$
- Exponentially distributed $\xi$ with mean 10
- Unique optimal solution

**NV with Discrete Uniform Demand:**
- $r = 6, c = 5, \nu = 1$
- Discrete Uniform $\xi$ on $\{1, 2, \ldots, 10\}$
- Multiple optimal solutions
Back to Newsvendor (NV) Problem:
SAA problems with $N = 10, 30, 90, 270$ observations

**NV with Exponential Demand:**
- $r = 6$, $c = 5$, $v = 1$
- Exponentially distributed $\xi$ with mean 10
- Unique optimal solution

**NV with Discrete Uniform Demand:**
- $r = 6$, $c = 5$, $v = 1$
- Discrete Uniform $\xi$ on $\{1, 2, \ldots, 10\}$
- Multiple optimal solutions
Back to Newsvendor (NV) Problem:
SAA problems with \( N = 10, 30, 90, 270 \) observations

**NV with Exponential Demand:**

<table>
<thead>
<tr>
<th>( N )</th>
<th>10</th>
<th>30</th>
<th>90</th>
<th>270</th>
<th>( \infty )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( x^*_N )</td>
<td>1.46</td>
<td>1.44</td>
<td>1.54</td>
<td>2.02</td>
<td>2.23</td>
</tr>
<tr>
<td>( z^*_N )</td>
<td>-1.11</td>
<td>-0.84</td>
<td>-0.98</td>
<td>-1.06</td>
<td>-1.07</td>
</tr>
</tbody>
</table>

**NV with Discrete Uniform Demand:**

<table>
<thead>
<tr>
<th>( N )</th>
<th>10</th>
<th>30</th>
<th>90</th>
<th>270</th>
<th>( \infty )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( x^*_N )</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>[2,3]</td>
</tr>
<tr>
<td>( z^*_N )</td>
<td>-2.00</td>
<td>-2.50</td>
<td>-1.67</td>
<td>-1.35</td>
<td>-1.50</td>
</tr>
</tbody>
</table>
Consistency of $z_N^*$

- **Strong Law of Large Numbers (SLLN)**

$$\lim_{N \to \infty} \left| \frac{1}{N} \sum_{j=1}^{N} f(x, \xi^j) - \mathbb{E}f(x, \xi) \right| = 0, \text{ wp1 for each } x \in X$$

is not enough to guarantee $z_N^* \to z^*$, wp1.

- **Strong Consistency of $z_N^*$**: However, under a Uniform SLLN

$$\lim_{N \to \infty} \sup_{x \in X} \left| \frac{1}{N} \sum_{j=1}^{N} f(x, \xi^j) - \mathbb{E}f(x, \xi) \right| = 0$$

we have

$$z_N^* \to z^*, \text{ wp1.}$$
Consistency of $z_N^*$

- **Strong Law of Large Numbers (SLLN)**

$$\lim_{N \to \infty} \left| \frac{1}{N} \sum_{j=1}^{N} f(x, \xi^j) - \mathbb{E} f(x, \xi) \right| = 0, \text{ wp1 for each } x \in X$$

is **not** enough to guarantee $z_N^* \to z^*$, wp1.

- **Strong Consistency of $z_N^*$**: However, under a Uniform SLLN

$$\lim_{N \to \infty} \sup_{x \in X} \left| \frac{1}{N} \sum_{j=1}^{N} f(x, \xi^j) - \mathbb{E} f(x, \xi) \right| = 0$$

we have

$$z_N^* \to z^*, \text{ wp1.}$$
Consistency of $z_N^*$

- **Strong Law of Large Numbers (SLLN)**

$$\lim_{N \to \infty} \left| \frac{1}{N} \sum_{j=1}^{N} f(x, \xi_j) - \mathbb{E}f(x, \xi) \right| = 0, \text{ wp1 for each } x \in X$$

is not enough to guarantee $z_N^* \to z^*, \text{ wp1}$.

- **Strong Consistency of $z_N^*$:** However, under a **Uniform SLLN**

$$\lim_{N \to \infty} \sup_{x \in X} \left| \frac{1}{N} \sum_{j=1}^{N} f(x, \xi_j) - \mathbb{E}f(x, \xi) \right| = 0$$

we have

$$z_N^* \to z^*, \text{ wp1}.$$
Strong Consistency of $x^*_N$: Assume USSLN holds, $\mathbb{E} f(\cdot, \xi)$ is continuous, and $X$ is compact. Then

$$\lim_{N \to \infty} \text{dist} (x^*_N, X^*) = 0, \wp 1,$$

where distance between a point $x$ and a set $A$ is defined as $\text{dist}(x, A) := \inf_{a \in A} \|x - a\|$.

In the unique optimum case; i.e., $X^* = \{x^*\}$,

$$x^*_N \to x^*, \wp 1.$$
Under finite variances assumption, we have seen earlier that by the Central Limit Theorem

\[
\sqrt{N} \left( \frac{1}{N} \sum_{j=1}^{N} f(x, \xi^j) - \mathbb{E}f(x, \xi) \right) \Rightarrow Y(x) \sim \text{Normal}(0, \sigma^2(x)),
\]

where \( \sigma^2(x) = \text{Var}(f(x, \xi)) \).
Rates of Convergence: $z_N^*$

- Under finite variances assumption, we have seen earlier that by the Central Limit Theorem

$$\sqrt{N} \left( \frac{1}{N} \sum_{j=1}^{N} f(x, \xi^j) - \mathbb{E} f(x, \xi) \right) \Rightarrow Y(x) \sim \text{Normal}(0, \sigma^2(x)),$$

where $\sigma^2(x) = \text{Var}(f(x, \xi))$.

- Under additional assumptions of $X$ compact, and Lipschitz continuity of $f(\cdot, \xi)$, we have

$$\sqrt{N} (z_N^* - z^*) \Rightarrow \inf_{x \in X^*} Y(x)$$

Note that $\inf_{x \in X^*} Y(x)$ is not Normally distributed unless $X^*$ is a singleton. In the unique optimality case, we have $X^* = \{x^*\}$ and

$$\sqrt{N} (z_N^* - z^*) \Rightarrow Y(x^*) \sim \text{Normal}(0, \sigma^2(x^*))$$
Back to Newsvendor (NV) Problem: 1,000 SAA problems with \( N = 270 \) observations

**NV with Exponential Demand:**

- Theory suggests for large \( N \), \( z^*_N \) is approximately \( \text{Normal}(z^*, \sigma^2(x^*)/N) \)
- With \( N = 270 \), this is \( \text{Normal}(-1.07, 0.166) \)
- Fitted distribution is \( \text{Normal}(-1.1, 0.169) \)

**NV with Discrete Uniform Demand:**

- Although the histogram may look close to Normal...
- Does not pass KS goodness-of-fit test at a significance level of 0.01 for Normal distribution
- In fact, none of the classical distributions do
Additional assumptions are needed to obtain limiting distributions. See, e.g., (King and Rockefellar 1993, Shapiro 1993, Rubinstein and Shapiro 1993)

**Exponential Rates of Convergence:** Suppose (i) $f(\cdot, \xi)$ is piecewise linear and convex, (ii) $X$ is polyhedral, and (iii) the distribution of $\xi$ has finite support. Under further boundedness assumptions,

- $X^*$ is polyhedral and $X_N^*$ is a face of $X^*$, w.p.1 for $N$ large enough
- The probability that $X_N^*$ is a face of $X^*$ converges to one exponentially fast with $N$. That is, as $N$ gets large

$$P(X_N^* \text{ is not a face of } X^*) \leq Ce^{-\beta N}$$

for some constants $C$, $\beta > 0$. 
Additional assumptions are needed to obtain limiting distributions. See, e.g., (King and Rockefellar 1993, Shapiro 1993, Rubinstein and Shapiro 1993)

**Exponential Rates of Convergence:** Suppose (i) $f(\cdot, \xi)$ is piecewise linear and convex, (ii) $X$ is polyhedral, and (iii) the distribution of $\xi$ has finite support. Under further boundedness assumptions,

- $X^*$ is polyhedral and $X_N^*$ is a face of $X^*$, w.p.1 for $N$ large enough
- The probability that $X_N^*$ is a face of $X^*$ converges to one exponentially fast with $N$. That is, as $N$ gets large

$$P(X_N^* \text{ is not a face of } X^*) \leq Ce^{-\beta N}$$

for some constants $C$, $\beta > 0$. 
Exponential Rates of Convergence, Continued:
The above can be satisfied for two-stage stochastic linear programs

Consider the case $|X|$ is finite, as in stochastic integer programs. Then, we again have

$$1 - P(X_N^* \subseteq X^*) \leq Ce^{-\beta N}$$

(Shapiro and Homem-de-Mello 2000), (Shapiro et al. 2002), and (Kleywegt et al. 2002)
Exponential Rates of Convergence, Continued:
The above can be satisfied for two-stage stochastic linear programs

Consider the case $|X|$ is finite, as in stochastic integer programs. Then, we again have

$$1 - P(X^*_N \subseteq X^*) \leq Ce^{-\beta N}$$

(Shapiro and Homem-de-Mello 2000), (Shapiro et al. 2002), and (Kleywegt et al. 2002)

Note: Exponential rate of converge is desired but not does always mean fast convergence. Depends on $\beta$. 
Outline

1 Introduction
   • Stochastic Programming
   • Need for Monte Carlo Sampling

2 Sample Average Approximation
   • Optimized Sample Averages and Bias
   • Consistency
   • Rates of Convergence

3 Monte Carlo Sampling-Based Solution Methods
   • Introduction
   • Stochastic Approximation
   • Sampling-Based Cutting-Plane Algorithms

4 Assessing Solution Quality and Stopping Rules
   • Solution Quality Estimation
   • Stopping Rules

5 Alternative Sampling Techniques

6 Conclusions
Consider a deterministic optimization problem:

\[
(P) \quad \min_{x} F(x) \quad \text{s.t.} \quad x \in X
\]

Can use different algorithms to solve these problems:
- Steepest Descent,
- Cutting Plane Methods, etc.

These algorithms require evaluations of \( F(\cdot) \) and \( \nabla F(\cdot) \) at each iteration.
Monte Carlo Sampling-Based Solution Methods

- Consider a deterministic optimization problem:

  \[
  (P) \quad \min_x F(x) \\
  \text{s.t. } x \in X
  \]

- Can use different algorithms to solve these problems:
  - Steepest Descent,
  - Cutting Plane Methods, etc.

- These algorithms require evaluations of \( F(\cdot) \) and \( \nabla F(\cdot) \) at each iteration

- Note that (SP) is a (P) with \( F(x) := \mathbb{E}f(x, \xi) \)

- However, in (SP), evaluations of \( F(x) := \mathbb{E}f(x, \xi) \) and \( \nabla \mathbb{E}f(x, \xi) \) are difficult

- Use their Monte Carlo sampling-based estimators instead
Consider a deterministic optimization problem:

\[
\begin{align*}
(P) & \quad \min_{x} F(x) \\
& \text{s.t. } x \in X
\end{align*}
\]

Can use different algorithms to solve these problems:
- Steepest Descent,
- Cutting Plane Methods, etc.

These algorithms require evaluations of \( F(\cdot) \) and \( \nabla F(\cdot) \) at each iteration.

Note that (SP) is a (P) with \( F(x) := \mathbb{E}f(x, \xi) \)

However, in (SP), evaluations of \( F(x) := \mathbb{E}f(x, \xi) \) and \( \nabla \mathbb{E}f(x, \xi) \) are difficult.

Use their Monte Carlo sampling-based estimators instead.
Monte Carlo sampling-based methods typically consist of adapting a deterministic algorithm by replacing the function values and their (sub)gradients by their Monte Carlo estimators.

- Can either generate new set of observations at each iteration or augment existing ones.

- Want good asymptotic properties, generating high-quality solutions with high probability.

- Desire fast convergence.
Stochastic Approximation Methods

Variants of Steepest Descent:

\[ x^{k+1} := x^k - \alpha_k \eta^k, \quad k \geq 0, \]

where

- \( -\eta^k \) is a random direction satisfying some properties. For example, the expectation of such a direction should be a descent direction for the true function \( \mathbb{E} f(x^k, \xi) \)
- \( \alpha_k \) is the step-size at iteration \( k \)
- The sequence \( \{\alpha_k\} \) should go to zero but not too fast, which is usually formalized as \( \sum_{k=0}^{\infty} \alpha_k = \infty \), \( \sum_{k=0}^{\infty} \alpha_k^2 < \infty \)
- For constrained problems, \( x^{k+1} \) is defined as the projection of \( x^k - \alpha_k \eta^k \) onto the feasibility set \( X \)
Stochastic Approximation Methods

Have a long history...

- *Robbins and Monro (1951)*: unbiased gradient estimators, finding a zero
- *Keifer and Wolfowitz (1952)*: finite-difference gradient estimators

Focus on simulation, construction of gradient estimates, accelerating the method, etc.

- Stochastic Quasi-Gradient Methods: Surveys can be found in *Ermoliev (1983)* and *Pflug (1988)*: biased estimator, with bias shrinking to zero typically for solving convex but nondifferentiable constrained optimization problems.
Stochastic Approximation Methods

Have a long history...

- Robbins and Monro (1951): unbiased gradient estimators, finding a zero
- Keifer and Wolfowitz (1952): finite-difference gradient estimators
- Focus on simulation, construction of gradient estimates, accelerating the method, etc.
- Stochastic Quasi-Gradient Methods: Surveys can be found in Ermoliev (1983) and Pflug (1988): biased estimator, with bias shrinking to zero typically for solving convex but nondifferentiable constrained optimization problems.

Issue: Even though “optimal” rate of convergence is good $O(1/k)$, in implementation, slow to converge
An important development that led to speed-ups in convergence: Polyak averaging

- Rather than looking at the individual iterates \( \{x_k\} \), one should analyze the average iterates

\[
\bar{x}^k := \frac{1}{k} \sum_{j=0}^{k-1} x^j
\]

(Polyak and Juditsky 1992)

- Such a method also achieves the theoretical rate of convergence, but the averaging allows for more robustness with respect to stepsizes
Another new line of Stochastic Approximation (SA) for convex problems: Find \( x^{k+1} \) as a suitable projection using proximal-type functions

- **Nemirovski et al. (2009): Mirror-Descent SA**
  - Generalizes the basic SA
  - Using appropriate prox-function provides more flexibility, robustness in choosing the parameters, and better performance

- **Lan (2010)** provides further enhancements *Accelerated SA*

- **Nesterov (2009):** primal-dual algorithm
A Cutting-Plane Algorithm

For two-stage or multistage stochastic linear programs with recourse, can make use of the special structures.

Solve a sequence of master problems:

$$\min_{x, \theta} \quad cx + \theta$$
$$\text{s.t.} \quad \theta \geq \mathbb{E}f(x^k, \xi) + \nabla \mathbb{E}f(x^k, \xi)(x - x^k), \quad k = 1, 2, \ldots, \kappa,$$
$$x \in X$$

Cannot calculate $\mathbb{E}f(x^k, \xi)$ and $\nabla \mathbb{E}f(x^k, \xi)$!

What to do?
Step 1: Choose an initial solution $x^0$; let $k := 1$

Step 2: Let $N_k = k$ and generate $\xi^{N_k}$ from the distribution of $\xi$, independent of other observations

Step 3: Compute Monte Carlo sampling-based estimators of $\mathbb{E}f(x^k, \xi)$ and $\nabla \mathbb{E}f(x^k, \xi)$;

For earlier generated cut coefficients, wash them out by multiplying with $\frac{k-1}{k}$;

Solve the master problem with all the generated “sampled” cuts

Step 4: Check some stopping criteria; if not satisfied, set $k := k + 1$ and go back to Step 2
- **Issue:** sampling-based cuts can “cut” optimal solutions
**Issue:** sampling-based cuts can “cut” optimal solutions

To remedy this, weaker cuts are formed, and also

Olds cuts are washed out in Step 3

Under appropriate conditions, there exists a subsequence of solutions that solve (SP)
In multistage stochastic programs, sampled trees are used to form sampling-based nested Benders decomposition.

- Stochastic Dual Dynamic Programming (SDDP) (Pereira and Pinto 1991), (Shapiro 2011)
- CUPPS (Chen and Powell 1999)
- Abridged Nested Decomposition (Donehue and Birge 2006)
- ReSa (Hindberger and Philpott 2001)
- Converge analysis of these algorithms are provided in (Philpott and Guan 2008)
Outline

1 Introduction
   - Stochastic Programming
   - Need for Monte Carlo Sampling

2 Sample Average Approximation
   - Optimized Sample Averages and Bias
   - Consistency
   - Rates of Convergence

3 Monte Carlo Sampling-Based Solution Methods
   - Introduction
   - Stochastic Approximation
   - Sampling-Based Cutting-Plane Algorithms

4 Assessing Solution Quality and Stopping Rules
   - Solution Quality Estimation
   - Stopping Rules

5 Alternative Sampling Techniques

6 Conclusions
Assume a candidate solution, $x \in X$, is given. How to obtain such a solution?
Assume a candidate solution, $x \in X$, is given. *How to obtain such a solution?*

1. Solve a sample average approximation problem:

   $$ x \in \arg \min_{y \in X} \left[ \frac{1}{N} \sum_{j=1}^{N} f(y, \xi_j) \right] $$

2. Stochastic Approximation (Robbins-Monro, 1951, Nemirovski et al., 2009)

3. Stochastic Decomposition (Higle and Sen, 1996)

4. or any other method . . .
Solution Quality

- Assume a candidate solution, \( x \in X \), is given. How to obtain such a solution?

1. Solve a sample average approximation problem:

\[
x \in \arg\min_{y \in X} \left[ \frac{1}{N} \sum_{j=1}^{N} f(y, \xi_j) \right]
\]

2. Stochastic Approximation (Robbins-Monro, 1951, Nemirovski et al., 2009)

3. Stochastic Decomposition (Higle and Sen, 1996)

4. or any other method . . .

- We are interested in determining its quality
Define the quality of $x$ by its optimality gap:

$$\mu_x = \mathbb{E}f(x, \xi) - z^*$$

Issue: $G_N(x) \geq 0$ is not asymptotically normal due to constrained optimization of sample means (how to deal with this?)
Define the quality of $x$ by its optimality gap:

$$\mu_x = \mathbb{E} f(x, \xi) - z^*$$

Using the bias ($\mathbb{E} z_N^* \leq z^*$), we can bound the optimality gap by:

$$\mu_x = \mathbb{E} f(x, \xi) - z^* \leq \mathbb{E} f(x, \xi) - \mathbb{E} z_N^*$$
Solution Quality Estimation

- Define the quality of $x$ by its optimality gap:

$$
\mu_x = \mathbb{E}f(x, \xi) - z^*
$$

- Using the bias ($\mathbb{E}z_N^* \leq z^*$), we can bound the optimality gap by:

$$
\mu_x = \mathbb{E}f(x, \xi) - z^* \leq \mathbb{E}f(x, \xi) - \mathbb{E}z_N^*
$$

- We can estimate an upper bound on optimality gap of $x \in X$ via:

$$
G_N(x) = \frac{1}{N} \sum_{j=1}^{N} f(x, \xi^j) - \min_{x \in X} \frac{1}{N} \sum_{j=1}^{N} f(x, \xi^j) - z_N^*
$$

- **Issue:** $G_N(x) \geq 0$ is not asymptotically normal due to constrained optimization of sample means (how to deal with this?)
Multiple Replications Procedure (MRP)

(Mak, Morton, and Wood 1999)

- Generate $N_G$ batches of $N$ observations to estimate $N_G$ gap estimators $G_{N,j}(x)$, $j = 1, 2, \ldots, N_G$ and use

$$\bar{G}(x) = \frac{1}{N_G} \sum_{j=1}^{N_G} G_{n,j}(x)$$

as a point estimator of the optimality gap.

- $N_G$ is typically taken to be $\sim 30$
An asymptotically valid \((1 - \alpha)\)-level confidence interval estimator is formed via

\[
\left[ 0, \bar{G}(x) + \frac{z_\alpha s_G(x)}{\sqrt{n_G}} \right],
\]

where

- \(z_\alpha\) is the \(1 - \alpha\) quantile of a standard normal distribution and

\[
s^2_G(x) = \frac{1}{N_G - 1} \sum_{j=1}^{N_G} (G_{N,j}(x) - \bar{G}(x))^2.
\]
(Bayraksan and Morton, 2006)

- Use **only one** optimality gap estimator of $x \in X$:

\[
G_N(x) = \frac{1}{N} \sum_{i=1}^{N} f(x, \xi^i) - \min_{x \in X} \frac{1}{N} \sum_{i=1}^{N} f(x, \xi^i).
\]

Recall $x^*_N \in \arg \min_{x \in X} \frac{1}{N} \sum_{i=1}^{N} f(x, \xi^i)$

- Estimate variance by:

\[
s^2_N(x) = \frac{1}{N - 1} \sum_{i=1}^{N} \left( (f(x, \xi^i) - f(x^*_N, \xi^i)) - G_N(x) \right)^2
\]
Form a \((1 - \alpha)\)-level approximate confidence interval on optimality gap of \(x \in X\) by:
\[
\left[ 0, \ G_N(x) + \frac{z_\alpha s_N(x)}{\sqrt{N}} \right],
\]

**NOTE:** When using only one gap estimator, variance needs to be calculated differently
Averaged 2 Replication Procedure (A2RP)

To achieve better small-sample behavior:

- Calculate gap and variance estimators on two samples of size n,

  On Partition 1: \( G_{N,1}(x) \quad s^2_{N,1}(x) \)
  On Partition 2: \( G_{N,2}(x) \quad s^2_{N,2}(x) \)

- Average to obtain A2RP estimators:

  \[
  G'(x) = \frac{1}{2} G_{N,1}(x) + \frac{1}{2} G_{N,2}(x) \\
  s'^2(x) = \frac{1}{2} s^2_{N,1}(x) + \frac{1}{2} s^2_{N,2}(x)
  \]

- Interval estimator formed same way.
### Optimality Gap Estimation: Summary

**Input:** $x \in X$; a method to generate observations; a method to solve sampling problems

**Output:** A point estimator and a $(1 - \alpha)$-level approximate confidence interval estimator of optimality gap of $x$

<table>
<thead>
<tr>
<th></th>
<th>Observations</th>
<th>Point Estimator</th>
<th>Variance Estimator</th>
<th>Sampling Error</th>
</tr>
</thead>
</table>
| **MRP**              | $\xi^{j,1}, \xi^{j,2}, \ldots, \xi^{j,N}$  
  
  $j = 1, 2, \ldots, N_G$ | $\bar{G}(x) = \frac{1}{N_G} \sum_{j=1}^{N_G} G_{N,j}(x)$ | $s_G^2(x) = \frac{1}{N_G - 1} \sum_{j=1}^{N_G} (G_{N,j}(x) - \bar{G}(x))^2$ | $\frac{z_\alpha s_G(x)}{\sqrt{N_G}}$ |
| **SRP**              | $\xi^1, \xi^2, \ldots, \xi^N$ | $G_{N,j}(x)$ | $s_N^2(x) = \frac{1}{N-1} \sum_{i=1}^{N} \left( f(x, \xi^i) - f(x_N^*, \xi^i) - G_N(x) \right)^2$ | $\frac{z_\alpha s_N(x)}{\sqrt{N}}$ |
| **A2RP**             | $\xi^{1,1}, \xi^{1,2}, \ldots, \xi^{1,N}$  
  
  $\xi^{2,1}, \xi^{2,2}, \ldots, \xi^{2,N}$ | $G'(x) = \frac{1}{2} \sum_{j=1}^{2} G_{N,j}(x)$ | $s_N'^2(x) = \frac{1}{2} \sum_{j=1}^{2} s_{N,j}^2(x)$ | $\frac{z_\alpha s_N'(x)}{\sqrt{2N}}$ |
I2RP is similar to A2RP but has uses gap estimator from one sample, and variance estimator from the other.

TRUE uses the true variance value instead of sample variance and has SRP gap estimator.
Notation:

\( k \): iteration number

\( x^k \): candidate solution at iteration \( k \)

\( G_k \): optimality gap estimator of \( x^k \) at iteration \( k \)

\( s_k^2 \): associated variance estimator

\( D_N(x) = \frac{1}{n} \sum_{j=1}^{N} [f(x, \xi) - f(x_{\min}^*)], \) where

\( x_{\min}^* \in \text{arg min}_{y \in X^*} \text{var}[f(x, \xi) - f(y, \xi)] \)
Assumptions

A1. The sequence of feasible candidate solutions \( \{x_k\} \) has at least one limit point in \( X^* \), w.p.1.

A2. Let \( \{x^k\} \) be a feasible sequence (i.e., \( x^k \in X \)) with \( x \) as one if its limit points. Let sample size \( N_k \) satisfy \( N_k \to \infty \) as \( k \to \infty \). Then, \( \lim \inf_{k \to \infty} P(|G_{N_k}(x_k) - \mu_x| > \delta) = 0 \) for any \( \delta > 0 \).

A3. \( G_N(x) \geq D_N(x) \), w.p.1, for all \( x \in X \) and \( N \geq 1 \).

A4. \( \lim \inf_{N \to \infty} s_N^2(x) \geq \sigma^2(x) \), w.p.1, for all \( x \in X \).

A5. \( \sqrt{N}(D_N(x) - \mu_x) \Rightarrow Normal(0, \sigma^2(x)) \) as \( N \to \infty \) for all \( x \in X \).
Some Remarks on the Assumptions

- No need to know $x_{\min}^*$ or $D_N(x)$. Used for theoretical properties. But can view $D_N(x)$ as the “nominal” sample average estimate of $\mu_x$ with minimal variance. (Note that $\mathbb{E}D_N(x) = \mu_x$)

- We need a good way to generate the solutions (A1)

- We need good properties for point estimators (A2, A3, A4)

- Need sampling to be done via a method that satisfies the Central Limit Theorem (A5). For instance, i.i.d. sampling, antithetic variates, bootstrapping (under certain conditions) work.
**Stopping Rule:** Terminate at iteration

\[ T = \inf_{k \geq 1} \{ k : G_k \leq h' s_k + \epsilon' \}, \]

i.e., stop the first time \( G_k \)’s width relative to \( s_k \) falls below \( h' > 0 \) plus a small positive number \( \epsilon' \). Let \( h > h' \).


**Rules to Stop and to Increase the Sample Sizes**

**Stopping Rule:** Terminate at iteration

\[ T = \inf_{k \geq 1} \{ k : G_k \leq h's_k + \epsilon' \}, \]

i.e., stop the first time \( G_k \)'s width relative to \( s_k \) falls below \( h' > 0 \) plus a small positive number \( \epsilon' \). Let \( h > h' \).

**Rules to Increase the Sample Size:** Select the sample size at iteration \( k \) according to

\[ N_k \geq \left( \frac{1}{h - h'} \right)^2 \left( c_q + 2q \ln^2 k \right), \]

where \( c_q = \max\{2 \ln \left( \sum_{k=1}^{\infty} k^{-q \ln k} / \sqrt{2\pi \alpha} \right), 1\} \). Here, \( q > 0 \) is a parameter we can choose, which affects the number of samples we generate.
**Quality Statement:** When stopped at iteration $T$, the sequential sampling procedure provides an approximate solution, $x^T$, and a confidence interval on its optimality gap, $\mu_T$, as

$$[0, hs_T + \epsilon],$$
Quality Statement: When stopped at iteration $T$, the sequential sampling procedure provides an approximate solution, $x^T$, and a confidence interval on its optimality gap, $\mu_T$, as

$$[0, hs_T + \epsilon],$$

Theorem:

Let $\epsilon > \epsilon' > 0$ and $h > h' > 0$ and $0 < \alpha < 1$. Consider the above sequential sampling procedure.

(i) Assume A1 and A2 are satisfied. Then, $P(T < \infty) = 1$.

(ii) Assume A3-A5 and a moment generating function assumption are satisfied. Then,

$$\liminf_{h \downarrow h'} P(\mu_T \leq hs_T + \epsilon) \geq 1 - \alpha.$$
For problems that exhibit exponential rate of convergence, using the Single Replication Procedure, a fixed-width stopping rule that tries to determine \( \epsilon \)-optimal solutions is:

\[
T = \inf_{k \geq 1} \left\{ k : G_k + \frac{z\alpha s_k}{\sqrt{N_k}} + \frac{1}{N_k} \leq \epsilon \right\}
\]

Under appropriate sample size increases, this procedure asymptotically finds an \( \epsilon \)-optimal solution with probability one.

*(Bayraksan and Pierre-Luis 2012)*
Outline

1 Introduction
   - Stochastic Programming
   - Need for Monte Carlo Sampling

2 Sample Average Approximation
   - Optimized Sample Averages and Bias
   - Consistency
   - Rates of Convergence

3 Monte Carlo Sampling-Based Solution Methods
   - Introduction
   - Stochastic Approximation
   - Sampling-Based Cutting-Plane Algorithms

4 Assessing Solution Quality and Stopping Rules
   - Solution Quality Estimation
   - Stopping Rules

5 Alternative Sampling Techniques

6 Conclusions
**Alternative Sampling Techniques**

Samples can be generated differently:

- Crude (or standard) Monte Carlo
- Antithetic Variates
- Latin Hypercube Sampling
- Randomized Quasi-Monte Carlo Sampling
Idea: use pairs of negatively correlated random variates to reduce variance.

- Sample observations \( \{u^1, \ldots, u^{\frac{N}{2}}\} \) from a \( U(0, 1)^{d\xi} \) distribution.
- Calculate the antithetic pairs \( \{u^1', \ldots, u^\frac{N}{2}'\} = \{1 - u^1, \ldots, 1 - u^\frac{N}{2}\} \).
- Apply the inverse cumulative distribution function (CDF) to the observations to obtain \( \{\xi^1, \ldots, \xi^N\} \).
Idea: perform stratified sampling in each component of $\xi$. 

(a) Stratified

(b) LHS

Images: (Lemieux, 2009)
For each component of $\xi$:
- Sample observations $u^i \sim U\left(\frac{i-1}{N}, \frac{i}{N}\right)$ for $i = 1, \ldots, N$.
- Randomly permute these $n$ observations.

Apply the inverse CDF to the observations to obtain $\{\xi^1, \ldots, \xi^N\}$. 
Idea: use a low-discrepancy sequence to sample more uniformly.

(Images: (a) (Lemieux, 2009), (b) (Homem de Mello and Bayraksan, 2013)
The alternative sampling techniques are often used to reduce variance. However, in stochastic programming, they have been observed to reduce bias as well.

- AV easy to generate, modest effect
- LHS relatively easy to generate, very effective
- RQMC effective at low dimensions, hard to generate and loses effectiveness at high dimensions $d_\xi$
Outline

1 Introduction
   - Stochastic Programming
   - Need for Monte Carlo Sampling

2 Sample Average Approximation
   - Optimized Sample Averages and Bias
   - Consistency
   - Rates of Convergence

3 Monte Carlo Sampling-Based Solution Methods
   - Introduction
   - Stochastic Approximation
   - Sampling-Based Cutting-Plane Algorithms

4 Assessing Solution Quality and Stopping Rules
   - Solution Quality Estimation
   - Stopping Rules

5 Alternative Sampling Techniques

6 Conclusions
Conclusions

- Many things we didn’t cover; e.g., how to allocate sample sizes
- Also, we didn’t cover any stochastic constraints
- However, we discussed
  1. Sample Average Approximation and its Properties,
  2. Some Common Sampling-Based Solution Methods,
  3. Assessing Solution Quality,
  4. Stopping Rules,
  5. Briefly went over Alternative Sampling Techniques
Acknowledgments:

I am grateful to David P. Morton for having introduced me to the topic and collaborations and discussions over the years. Parts of this tutorial are based on his teachings.

I am also grateful to Tito Homem-de-Mello. Most of this tutorial is based on a survey paper

- Homem-de-Mello, Tito and G. Bayraksan, “Monte Carlo Sampling-Based Methods for Stochastic Programming,”

Available upon request: bayraksan.1@osu.edu
Thank you... Questions?

(Please contact bayraksan.1@osu.edu)