CSCI 1951-G – Optimization Methods in Finance
Part 10:
Robust Optimization

April 29, 2016
Additional optional readings

Uncertainty

The future is uncertain.
Uncertainty

The future is uncertain. The present too.
Uncertainty

The future is uncertain. The present too.

- Uncertainty is not an occasional deviation from every day norm.
- It is a basic *structural* feature of the environment.
Uncertainty

The future is uncertain. The present too.

- Uncertainty is not an occasional deviation from every day norm.

- It is a basic *structural* feature of the environment.

- The best way to handle uncertainty is to:
  - accept it;
  - structure it and understand it;
  - make it part of the decision making reasoning;
Stochastic optimization

Stochastic optimization considers uncertainty in a very specific sense.

- It assumes complete knowledge of a probability distribution over all possible scenarios.
  - Difficult to estimate (e.g.: rare scenarios with high impact, or behavior of other agents)
- The solution is optimal in expectation, i.e., in the long run, and only in the long run.
  - Assumes "repeated decisions"
- Decision makers are often risk-adverse:
  - Hedge against the risk of poor performances in some scenarios;
  - Do not optimize expectation over all scenarios;
  - What is important: performance of a decision across all potential scenarios.
Stochastic optimization

Stochastic optimization considers uncertainty in a very specific sense.
- It assumes complete knowledge of a probability distribution over all possible scenarios.
Stochastic optimization

Stochastic optimization considers uncertainty in a very specific sense.

- It assumes complete knowledge of a probability distribution over all possible scenarios.
  
  Difficult to estimate (e.g.: rare scenarios with high impact, or behavior of other agents)
Stochastic optimization

Stochastic optimization considers uncertainty in a very specific sense.

- It assumes complete knowledge of a probability distribution over all possible scenarios.
  - Difficult to estimate (e.g.: rare scenarios with high impact, or behavior of other agents)
- The solution is optimal in expectation, i.e., in the long run, and only in the long run.
  - Assumes “repeated decisions”
Stochastic optimization

Stochastic optimization considers uncertainty in a very specific sense.

- It assumes complete knowledge of a probability distribution over all possible scenarios.
  Difficult to estimate (e.g.: rare scenarios with high impact, or behavior of other agents)
- The solution is optimal in expectation, i.e., in the long run, and only in the long run.
  Assumes “repeated decisions”

Decision makers are often risk-adverse:
- hedge against the risk of poor performances in some scenarios;
- do not optimize expectation over all scenarios;
**Stochastic optimization**

Stochastic optimization considers uncertainty in a very specific sense.
- It assumes complete knowledge of a probability distribution over all possible scenarios.
  Difficult to estimate (e.g.: rare scenarios with high impact, or behavior of other agents)
- The solution is optimal in expectation, i.e., in the long run, and only in the long run.
  Assumes “repeated decisions”

Decision makers are often risk-adverse:
- hedge against the risk of poor performances in some scenarios;
- do not optimize expectation over all scenarios;

What is important: performance of a decision across all potential scenarios.
Robust optimization

Robust optimization considers environments where knowledge about the random state of nature is incomplete.

• It is a complementary alternative to stochastic optimization.
• It can be useful when:
  • parameters are estimated and carry estimation risk;
  • constraints with uncertain parameters must be satisfied regardless of the values of the parameters;
  • the objective function is sensitive to perturbations;
  • the decision maker cannot afford to take low probability but high-magnitude risks.
• Goal: produce decisions with reasonable objective value under any scenario.
Robust optimization

Robust optimization considers environments where knowledge about the random state of nature is incomplete.

• It is a complementary alternative to stochastic optimization.
Robust optimization considers environments where knowledge about the random state of nature is incomplete.

- It is a complementary alternative to stochastic optimization.
- It can be useful when:
  - parameters are estimated and carry estimation risk;
  - constraints with uncertain parameters must be satisfied regardless of the values of the parameters;
  - the objective function is sensitive to perturbations;
  - the decision maker cannot afford to take low probability but high-magnitude risks.

Goal: produce decisions with reasonable objective value under any scenario.
Robust optimization considers environments where knowledge about the random state of nature is incomplete.

- It is a complementary alternative to stochastic optimization.

- It can be useful when:
  - parameters are estimated and carry estimation risk;
  - constraints with uncertain parameters must be satisfied regardless of the values of the parameters;
Robust optimization considers environments where knowledge about the random state of nature is incomplete.

- It is a complementary alternative to stochastic optimization.

- It can be useful when:
  - parameters are estimated and carry estimation risk;
  - constraints with uncertain parameters must be satisfied regardless of the values of the parameters;
  - the objective function is sensitive to perturbations;
Robust optimization considers environments where knowledge about the random state of nature is incomplete.

- It is a complementary alternative to stochastic optimization.

- It can be useful when:
  - parameters are estimated and carry estimation risk;
  - constraints with uncertain parameters must be satisfied regardless of the values of the parameters;
  - the objective function is sensitive to perturbations;
  - the decision maker cannot afford to take low probability but high-magnitude risks.

Goal: produce decisions with reasonable objective value under any scenario.
Robust optimization

Robust optimization considers environments where knowledge about the random state of nature is incomplete.

- It is a complementary alternative to stochastic optimization.

- It can be useful when:
  - parameters are estimated and carry estimation risk;
  - constraints with uncertain parameters must be satisfied regardless of the values of the parameters;
  - the objective function is sensitive to perturbations;
  - the decision maker cannot afford to take low probability but high-magnitude risks.

- Goal: produce decisions with reasonable objective value under any scenario.
Representing the uncertainty

We use *uncertainty sets* that contain the possible *scenarios*. 

• Finite set of scenarios: $\mathcal{U} = \{p_1, \ldots, p_k\}$

• Convex hull of a finite set of scenarios: $\mathcal{U} = \text{conv}(p_1, \ldots, p_k)$

• Intervals: $\mathcal{U} = \{p: \ell \leq p \leq u\}$

• Ellipsoidal uncertainty sets: $\mathcal{U} = \{p: p = p_0 + Mu, \|u\| \leq 1\}$
Representing the uncertainty

We use *uncertainty sets* that contain the possible *scenarios*.

Examples:
- Finite set of scenarios: $\mathcal{U} = \{p_1, \ldots, p_k\}$
Representing the uncertainty

We use *uncertainty sets* that contain the possible *scenarios*.

Examples:
- Finite set of scenarios: \( \mathcal{U} = \{p_1, \ldots, p_k\} \)
- Convex hull of a finite set of scenarios: \( \mathcal{U} = \text{conv}(p_1, \ldots, p_k) \)
Representing the uncertainty

We use *uncertainty sets* that contain the possible *scenarios*.

Examples:
- Finite set of scenarios: \( \mathcal{U} = \{ p_1, \ldots, p_k \} \)
- Convex hull of a finite set of scenarios: \( \mathcal{U} = \text{conv}(p_1, \ldots, p_k) \)
- Intervals: \( \mathcal{U} = \{ p : \ell \leq p \leq u \} \)
Representing the uncertainty

We use *uncertainty sets* that contain the possible *scenarios*.

Examples:
- Finite set of scenarios: $\mathcal{U} = \{p_1, \ldots, p_k\}$
- Convex hull of a finite set of scenarios: $\mathcal{U} = \text{conv}(p_1, \ldots, p_k)$
- Intervals: $\mathcal{U} = \{p : \ell \leq p \leq u\}$
- Ellipsoidal uncertainty sets:

  $$\mathcal{U} = \{p : p = p_0 + Mu, \|u\| \leq 1\}$$
Where is the uncertainty?

- **constraints uncertainty**:  
  - feasibility of potential solution is at risk (some solutions may be feasible in some scenarios but not in others);

- **objective uncertainty**:  
  - feasibility constraints are fixed 
  - the optimality of the generated solutions is at risk (they may be distant from the actual optimal)
Where is the uncertainty?

- **constraints uncertainty:**
  - feasibility of potential solution is at risk (some solutions may be feasible in some scenarios but not in others);
  - we are interested in solutions that are feasible in all scenarios

- **objective uncertainty:**
  - feasibility constraints are fixed
Where is the uncertainty?

- **constraints uncertainty:**
  - feasibility of potential solution is at risk (some solutions may be feasible in some scenarios but not in others);
  - we are interested in solutions that are feasible in all scenarios

- **objective uncertainty:**
  - feasibility constraints are fixed
  - the optimality of the generated solutions is at risk (they may be distant from the actual optimal)
Constraints robustness

• The uncertainty is in the constraints:

\[
\begin{align*}
\min & \quad f(x) \\
G(x, p) & \in K
\end{align*}
\]

where \( f \) and \( K \) are fixed, and \( p \) are the uncertain parameters.
Constraints robustness

- The uncertainty is in the constraints:

\[
\min f(x) \\
G(x, p) \in K
\]

where \( f \) and \( K \) are fixed, and \( p \) are the uncertain parameters

- We seek solutions that remain feasible for all values of the uncertain inputs
Constraints robustness

• The uncertainty is in the constraints:

\[
\min f(x) \\
G(x, p) \in K
\]

where \( f \) and \( K \) are fixed, and \( p \) are the uncertain parameters.

• We seek solutions that remain feasible for all values of the uncertain inputs.

• We reformulate the problem as:

\[
\min f(x) \\
G(x, p) \in K, \forall p \in U
\]
Constraints robustness

- The uncertainty is in the constraints:

\[
\begin{align*}
\min f(x) \\
G(x, p) &\in K
\end{align*}
\]

where \( f \) and \( K \) are fixed, and \( p \) are the uncertain parameters
- We seek solutions that remain feasible for all values of the uncertain inputs
- We reformulate the problem as:

\[
\begin{align*}
\min f(x) \\
G(x, p) &\in K, \forall p \in U
\end{align*}
\]

If \( U \) is infinite, this is a *semi-infinite problem*, for which specific resolution techniques exist.
Different Flavors of Robustness

\[ R^\Omega \] Robust feasible set

\[ -5 \quad -4 \quad -3 \quad -2 \quad -1 \quad 0 \quad 1 \quad 2 \quad 3 \quad 4 \quad 5 \]

\[ S(p_1) \quad S(p_2) \quad S(p_3) \quad S(p_4) \]

Figure 19.1: Constrain robustness

Exercise 19.1

Show that if \( S(p) = \{ x : G(x, p) \in K \} \) is convex for all \( p \), then the robust feasible set \( S := \bigcap_{p \in U} S(p) \) is also convex.

If \( S(p) \) is polyhedral for all \( p \), is \( S \) necessarily polyhedral?

19.3.2 Objective Robustness

Another important robustness concept is objective robustness. This refers to solutions that will remain close to optimal for all possible realizations of the uncertain problem parameters. Since such solutions may be difficult to obtain, especially when uncertain sets are relatively large, an alternative goal for objective robustness is to find solutions whose worst-case behavior is optimized. The worst-case behavior of a solution corresponds to the value of the objective function for the worst possible realization of the uncertain data for that particular solution.

We now develop a mathematical model that addresses objective robustness. Consider an optimization problem of the form:

\[
\min_{x} f(x, p) \quad x \in S.
\] (19.5)

Here, \( S \) is the (certain) feasible set and \( f \) is the objective function that depends on uncertain parameters \( p \). As before, \( U \) denotes the uncertainty set that contains all possible values of the uncertain parameters \( p \). Then, an objective robust solution can be obtained by solving:

\[
\min_{x \in S} \max_{p \in U} f(x, p).
\] (19.6)

We illustrate objective robustness problem (19.6) in Figure 19.2. In this example, the feasible set \( S \) is the real line, the uncertainty set is \( U = [-1, 1] \),

\[ S_p \]: set of feasible solutions for \( p \in U \)

\[ S = \bigcap_{p \in U} S_p \]: solutions that are feasible in all scenarios
Constraints uncertainty

We can envision multi-stage problems with constraint uncertainty:
Constraints uncertainty

We can envision multi-stage problems with constraint uncertainty:

- the uncertain outcomes of earlier stages impact the decision of later stages
We can envision multi-stage problems with constraint uncertainty:

- the uncertain outcomes of earlier stages impact the decision of later stages
- decision variable must satisfy balance constraints in all scenarios (e.g., inputs to a state cannot exceed outputs of the previous stage)
Constraints uncertainty

\[
\min_x \ f(x) \\
G(x, p) \in K
\]

We considered \( f(x) \) to have no uncertainty. This is not restrictive.

We can rewrite the above as

\[
\min_{x,t} \ t \\
t - f(x, p) \geq 0 \\
G(x, p) \in K
\]

All uncertainty is in the constraints
Objective robustness

- The uncertainty is only in the objective function (not in the constraints)
- We can move the uncertainty to the constraints, but not (always) advantageous
- Instead we want a minimax formulation:
  - Let $Q(x, p)$ be a quality measure of solution $x$ in scenario $p$.
  - We want to find
  $$\min_{x \in S} \max_{p \in U} Q(x, p)$$
Flavors of robustness

\[ S = \bigcap_{p \in \mathcal{U}} S_p \] : solutions that are feasible in all scenarios
Flavors of robustness

\[ S = \cap_{p \in U} S_p \] : solutions that are feasible in all scenarios

- **absolute robustness**: \( \min_{X \in S} \max_{p \in U} f(X, p) \)
  
  For each solution, consider the worst objective value over all scenarios. Pick the solution with the best worst case.

- **deviation robustness**: \( \min_{X \in S} \max_{p \in U} f(X, p) - f(X^*, p) \)
  
  For each solution, in each scenario, compare (difference) obj. value with optimal obj. value in that scenario, and then consider the worst case for that solution. Pick the solution with the best worst case.

- **relative robustness**: \( \min_{X \in S} \max_{p \in U} \frac{f(X, p)}{f(X^*, p)} \)
  
  As before, but use the ratio.

Different criteria lead to different robust solutions.
Flavors of robustness

\[ S = \cap_{p \in U} S_p \] : solutions that are feasible in all scenarios

- **absolute robustness**: \( \min_{X \in S} \max_{p \in U} f(X, p) \)

  For each solution, consider the worst objective value over all scenarios. Pick the solution with the best worst case.

- **deviation robustness**: \( \min_{X \in S} \max_{p \in U} f(X, p) - f(X_p^*, p) \)

  For each solution, in each scenario, compare (difference) obj. value with optimal obj. value in that scenario, and then consider the worst case for that solution. Pick the solution with the best worst case.
Flavors of robustness

\[ S = \cap_{p \in U} S_p : \text{solutions that are feasible in all scenarios} \]

- **Absolute robustness**: \[ \min_{X \in S} \max_{p \in U} f(X, p) \]

  For each solution, consider the worst objective value over all scenarios. Pick the solution with the best worst case.

- **Deviation robustness**: \[ \min_{X \in S} \max_{p \in U} f(X, p) - f(X^*_p, p) \]

  For each solution, in each scenario, compare (difference) obj. value with optimal obj. value in that scenario, and then consider the worst case for that solution. Pick the solution with the best worst case.

- **Relative robustness**: \[ \min_{X \in S} \max_{p \in U} \frac{f(X, p)}{f(X^*_p, p)} \]

  As before, but use the ratio.
Flavors of robustness

\[ S = \bigcap_{p \in U} S_p : \text{solutions that are feasible in all scenarios} \]

- **absolute robustness**: \[ \min_{X \in S} \max_{p \in U} f(X, p) \]

  For each solution, consider the worst objective value over all scenarios. Pick the solution with the best worst case.

- **deviation robustness**: \[ \min_{X \in S} \max_{p \in U} f(X, p) - f(X^*_p, p) \]

  For each solution, in each scenario, compare (difference) obj. value with optimal obj. value in that scenario, and then consider the worst case for that solution. Pick the solution with the best worst case.

- **relative robustness**: \[ \min_{X \in S} \max_{p \in U} \frac{f(X, p)}{f(X^*_p, p)} \]

  As before, but use the ratio.

Different criteria lead to different robust solutions.
Absolute robustness

Objective Robustness

max_i f(x, p_i)

f(x, p_2)
f(x, p_1)
f(x, p_4)
f(x, p_3)

Robust minimizer

x_1 = x_2

x^*_1, x^*_2, x^*_5, x^*_4, x^*_3
Example: a scheduling problem

Different criteria lead to different robust solutions.

- A set of jobs must be processed on a single machine;
- The machine can process one job at a time;
- The jobs arrive all together at time 0;
- The jobs have random processing times, with different distributions;
- We need to schedule the jobs;
- We want to minimize the sum of the completion times;
Example: a scheduling problem

Different criteria lead to different robust solutions.

• A set of jobs must be processed on a single machine;
• The machine can process one job at a time;
• The jobs arrive all together at time 0;
• The jobs have random processing times, with different distributions;
• We need to schedule the jobs;
• We want to minimize the sum of the completion times;

Question

Is this a constraint robustness problem or a objective robustness problem?
## Example

<table>
<thead>
<tr>
<th>Job</th>
<th>Processing time distribution</th>
<th>Expected processing time</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$U(23, 24)$</td>
<td>23.5</td>
</tr>
<tr>
<td>2</td>
<td>$U(21, 27)$</td>
<td>24</td>
</tr>
<tr>
<td>3</td>
<td>$U(20, 29)$</td>
<td>24.5</td>
</tr>
<tr>
<td>4</td>
<td>$U(4, 45)$</td>
<td>25</td>
</tr>
</tbody>
</table>

Stochastic optimization: minimize the expected sum of completion times

Answer: schedule jobs in non-decreasing order of expected processing time.

Schedule: $X_{E} = 1 - 2 - 3 - 4$, with expected sum of completion times:

$$E[f(X_{E}, p)] = 23.5 + (23.5 + 24) + (23.5 + 24 + 25.5) + (23.5 + 24 + 24.5 + 25) = \frac{24016}{37}$$
### Example

<table>
<thead>
<tr>
<th>Job</th>
<th>Processing time distribution</th>
<th>Expected processing time</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$U(23, 24)$</td>
<td>23.5</td>
</tr>
<tr>
<td>2</td>
<td>$U(21, 27)$</td>
<td>24</td>
</tr>
<tr>
<td>3</td>
<td>$U(20, 29)$</td>
<td>24.5</td>
</tr>
<tr>
<td>4</td>
<td>$U(4, 45)$</td>
<td>25</td>
</tr>
</tbody>
</table>

Stochastic optimization: minimize the expected sum of completion times
## Example

<table>
<thead>
<tr>
<th>Job</th>
<th>Processing time distribution</th>
<th>Expected processing time</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$U(23, 24)$</td>
<td>23.5</td>
</tr>
<tr>
<td>2</td>
<td>$U(21, 27)$</td>
<td>24</td>
</tr>
<tr>
<td>3</td>
<td>$U(20, 29)$</td>
<td>24.5</td>
</tr>
<tr>
<td>4</td>
<td>$U(4, 45)$</td>
<td>25</td>
</tr>
</tbody>
</table>

Stochastic optimization: minimize the expected sum of completion times

Answer: schedule jobs in non-decreasing order of expected processing time.
## Example

<table>
<thead>
<tr>
<th>Job</th>
<th>Processing time distribution</th>
<th>Expected processing time</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$U(23, 24)$</td>
<td>23.5</td>
</tr>
<tr>
<td>2</td>
<td>$U(21, 27)$</td>
<td>24</td>
</tr>
<tr>
<td>3</td>
<td>$U(20, 29)$</td>
<td>24.5</td>
</tr>
<tr>
<td>4</td>
<td>$U(4, 45)$</td>
<td>25</td>
</tr>
</tbody>
</table>

Stochastic optimization: minimize the expected sum of completion times

Answer: schedule jobs in non-decreasing order of expected processing time.

Schedule: $X_E = 1 - 2 - 3 - 4$, with expected sum of completion times:

$$E[f(X_E, p)] = 23.5 + (23.5 + 24) + (23.5 + 24 + 25.5) + (23.5 + 24 + 24.5 + 25) = 240$$
Example

- Assume the realization of processing times, i.e., the scenario $p_1$, is such that

$$p_1 = (t_1, t_2, t_3, t_4) = (24, 27, 20, 5)$$
Example

• Assume the realization of processing times, i.e., the scenario \( p_1 \), is such that

\[
p_1 = (t_1, t_2, t_3, t_4) = (24, 27, 20, 5)
\]

• \( X_E = 1 - 2 - 3 - 4 \) has \( f(X_E, p) = 222 \)

• The optimal schedule (for \( p_1 \), and in general for any scenario) is the one that goes in increasing order of processing time.

• For \( p_1 \), it is \( X^*_{p_1} = 4 - 3 - 1 - 2 \), with \( f(X^*_{p_1}, p_1) = 155 \)

• The price of perfect information is 67

• In other words, the stochastically optimal solution is 43\% worse than the optimal for the realized scenario. Still, \( X_E \) is the absolute robust solution, i.e., the one with best worst case objective value.
Example

• Assume the realization of processing times, i.e., the scenario $p_1$, is such that

\[ p_1 = (t_1, t_2, t_3, t_4) = (24, 27, 20, 5) \]

• $X_E = 1 - 2 - 3 - 4$ has $f(X_E, p) = 222$

• The optimal schedule (for $p_1$, and in general for any scenario) is the one that...
Example

• Assume the realization of processing times, i.e., the scenario \( p_1 \), is such that

\[
p_1 = (t_1, t_2, t_3, t_4) = (24, 27, 20, 5)
\]

• \( X_E = 1 - 2 - 3 - 4 \) has \( f(X_E, p) = 222 \)

• The optimal schedule (for \( p_1 \), and in general for any scenario) is the one that...goes in increasing order of processing time.

• For \( p_1 \), it is \( X^*_p = 4 - 3 - 1 - 2 \), with \( f(X^*_p, p_1) = 155 \)
Example

• Assume the realization of processing times, i.e., the scenario $p_1$, is such that

$$p_1 = (t_1, t_2, t_3, t_4) = (24, 27, 20, 5)$$

• $X_E = 1 - 2 - 3 - 4$ has $f(X_E, p) = 222$

• The optimal schedule (for $p_1$, and in general for any scenario) is the one that...goes in increasing order of processing time.

• For $p_1$, it is $X^*_{p_1} = 4 - 3 - 1 - 2$, with $f(X^*_{p_1}, p_1) = 155$

• The price of perfect information is 67
Example

• Assume the realization of processing times, i.e., the scenario $p_1$, is such that

\[ p_1 = (t_1, t_2, t_3, t_4) = (24, 27, 20, 5) \]

• $X_E = 1 - 2 - 3 - 4$ has $f(X_E, p) = 222$

• The optimal schedule (for $p_1$, and in general for any scenario) is the one that...goes in increasing order of processing time.

• For $p_1$, it is $X^*_{p_1} = 4 - 3 - 1 - 2$, with $f(X^*_{p_1}, p_1) = 155$

• The price of perfect information is 67

• In other words, the stochastically optimal solution is 43.2% worse than the optimal for the realized scenario.
Example

• Assume the realization of processing times, i.e., the scenario $p_1$, is such that

$$p_1 = (t_1, t_2, t_3, t_4) = (24, 27, 20, 5)$$

• $X_E = 1 - 2 - 3 - 4$ has $f(X_E, p) = 222$

• The optimal schedule (for $p_1$, and in general for any scenario) is the one that... goes in increasing order of processing time.

• For $p_1$, it is $X^*_p = 4 - 3 - 1 - 2$, with $f(X^*_p, p_1) = 155$

• The price of perfect information is 67

• In other words, the stochastically optimal solution is 43.2% worse than the optimal for the realized scenario.

Still, $X_E$ is the absolute robust solution, i.e., the one with best worst case objective value.
Example

Using a different robustness criteria leads to different solution, with different schedule can mitigate the exposure to the risk of poor performances.

• The difference $222 - 155 = 67$ is the max. over all scenarios for $X_E$, i.e.,

$$\max_{p \in U} (f(X_E, p) - f(X^*_p, p)) = 222 - 155 = 67$$
Example

Using a different robustness criteria leads to different solution, with different schedule can mitigate the exposure to the risk of poor performances.

- The difference $222 - 155 = 67$ is the max. over all scenarios for $X_E$, i.e.,

$$\max_{p \in U} (f(X_E, p) - f(X^*_p, p)) = 222 - 155 = 67$$

- Consider now $p_2$ s.t.

$$p_2 = t_1, t_2, t_3, t_4) = (23, 27, 20, 45)$$
Example

Using a different robustness criteria leads to different solution, with different schedule can mitigate the exposure to the risk of poor performances.

• The difference $222 - 155 = 67$ is the max. over all scenarios for $X_E$, i.e.,

$$\max_{p \in \mathcal{U}} (f(X_E, p) - f(X^*_p, p)) = 222 - 155 = 67$$

• Consider now $p_2$ s.t.

$$p_2 = t_1, t_2, t_3, t_4) = (23, 27, 20, 45)$$

• The schedule $X_A = 2 - 4 - 3 - 1$ has $f(X_A, p_2) = 306$
Example

Using a different robustness criteria leads to different solution, with different schedule can mitigate the exposure to the risk of poor performances.

- The difference $222 - 155 = 67$ is the max. over all scenarios for $X_E$, i.e.,
  \[
  \max_{p \in U} (f(X_E, p) - f(X^*_p, p)) = 222 - 155 = 67
  \]

- Consider now $p_2$ s.t.
  \[p_2 = t_1, t_2, t_3, t_4) = (23, 27, 20, 45)\]

- The schedule $X_A = 2 - 4 - 3 - 1$ has $f(X_A, p_2) = 306$
- The optimal (for $p_2$) schedule has $f(X^*_{p_2}, p_2) = 248$. 
Example

Using a different robustness criteria leads to different solution, with different schedule can mitigate the exposure to the risk of poor performances.

- The difference $222 - 155 = 67$ is the max. over all scenarios for $X_E$, i.e.,

$$\max_{p \in U}(f(X_E, p) - f(X_p^*, p)) = 222 - 155 = 67$$

- Consider now $p_2$ s.t.

$$p_2 = t_1, t_2, t_3, t_4) = (23, 27, 20, 45)$$

- The schedule $X_A = 2 - 4 - 3 - 1$ has $f(X_A, p_2) = 306$
- The optimal (for $p_2$) schedule has $f(X_p^*, p_2) = 248$.
- The difference $307 - 248 = 52$ is the max. over all scenario for $X_A$, i.e.,

$$\max_{p \in U}(f(X_A, p) - f(X_p^*, p)) = 306 - 248 = 52$$
Example

- for $X_E$, $\max_{p \in U}(f(X_E, p) - f(X^*, p)) = 67$
- for $X_A$, $\max_{p \in U}(f(X_A, p) - f(X^*, p)) = 52$

By choosing $X_A$ over $X_E$ we can greatly reduce the worst case loss w.r.t. the optimal choice for the realized scenario.
Example

- for $X_E$, $\max_{p \in U} (f(X_E, p) - f(X_p^*, p)) = 67$
- for $X_A$, $\max_{p \in U} (f(X_A, p) - f(X_p^*, p)) = 52$

By choosing $X_A$ over $X_E$ we can greatly reduce the worst case loss w.r.t. the optimal choice for the realized scenario.

The increase in expectation is minimal:

$$E[f(X_A, p)] = 243.5 \quad \text{(vs. 240.5 for } X_E)$$
Comparison between criteria

Absolute robustness tends to be very conservative

- main concern is protecting against worst possible happening

- the absolute robust schedule $X_E$ will never take more than 52 secs

- It gives no information to how much better we could do with less uncertainty
Comparison between criteria

Robust deviation and relative robustness are less conservative

- they take into account the best that could have been done in each scenario (benchmark)

- the robust solution is the one that tend to stay close to the benchmark (in a ranking, close to the top)

- There is a concept of regret, measuring the “benefit of hindsight”

- It express how much one could improve if some uncertainty was removed
Comparison between criteria

Use relative robustness (vs. robust deviation) when:
• the performance of the benchmark fluctuates across a wide range; or
• the performances of a decision across scenarios is highly variable.
• Otherwise, gives solutions that are pretty similar to robust deviation.
Comparison between criteria

Robust deviation and relative robustness are more difficult to compute than absolute robustness:

- The problem

\[
\min_{x \in S} \max_{p \in U} \left( f(x, p) - f(x^*_p, p) \right)
\]

is a three-level optimization problem.
Comparison between criteria

Robust deviation and relative robustness are more difficult to compute than absolute robustness:

• The problem

$$\min_{x \in S} \max_{p \in U} f(x, p) - f(x^*_p, p)$$

is a three-level optimization problem.

• $f(x^*_p, p) = g(p)$ is often non-smooth and non available in analytic form
Comparison between criteria

Robust deviation and relative robustness are more difficult to compute than absolute robustness:

• The problem

$$\min_{x \in S} \max_{p \in \mathcal{U}} f(x, p) - f(x^*_p, p)$$

is a three-level optimization problem.

• $f(x^*_p, p) = g(p)$ is often non-smooth and non available in analytic form

• If $f$ is linear in $p$, then $g(p)$ is a concave function, hence the inner optimization problem is a convex maximization problem

• easier variant: fix a maximum level of regret $R$ and solve the constraint program requiring a feasible $x$ with regret at most $R$. 
Distance on solutions

Instead of the difference in objective value, consider the distance in the solutions:

- Given a scenario $p$, let $X^*_p = \arg\min_x f(x, p)$ s.t. $x \in S$ ($X^*_p$ is a set)
- For a given $x$ and $p$, let $d(x, p) = \inf_{x^* \in X^*_p} \|x - x^*\|
- We want to find $x$ such that $\min_{x \in S} \min_{p \in U} d(x, p)$
- This is attractive if we can slightly revise our decision after $p$ has been revealed, but there is a cost in the perturbation.
- We want a solution that will not need much perturbation in any case
Distance on solutions

Instead of the difference in objective value, consider the distance in the solutions:

- Given a scenario $p$, let

  $$X_p^* = \arg \min_x f(x, p) \text{ s.t. } x \in S$$

  ($X_p^*$ is a set)
Distance on solutions

Instead of the difference in objective value, consider the distance in the solutions:

- Given a scenario $p$, let
  \[
  X^*_p = \arg \min_x f(x, p) \text{ s.t. } x \in S
  \]
  ($X^*_p$ is a set)

- For a given $x$ and $p$, let
  \[
  d(x, p) = \inf_{x^* \in X^*_p} \|x - x^*\|
  \]
Distance on solutions

Instead of the difference in objective value, consider the distance in the solutions:

- Given a scenario $p$, let

$$X_p^* = \arg \min_x f(x, p) \text{ s.t. } x \in S$$

($X_p^*$ is a set)

- For a given $x$ and $p$, let

$$d(x, p) = \inf_{x^* \in X_p^*} \| x - x^* \|$$

- We want to find $x$ such that

$$\min_{x \in S} \min_{p \in U} d(x, p)$$
Distance on solutions

Instead of the difference in objective value, consider the distance in the solutions:

• Given a scenario $p$, let

$$X_p^* = \arg \min_x f(x, p) \text{ s.t. } x \in S$$

($X_p^*$ is a set)

• For a given $x$ and $p$, let

$$d(x, p) = \inf_{x^* \in X_p^*} \|x - x^*\|$$

• We want to find $x$ such that

$$\min_x \min_p d(x, p)$$

• This is attractive if we can slightly revise our decision after $p$ has been revealed, but there is a cost in the perturbation.

• We want a solution that will not need much perturbation in any case
Adjustable robust optimization

- Consider multi-period decision models with uncertain parameters
- Some parameters are revealed during the decision process
- After observing the parameters, later stage decision can recourse
- closely related to multi-stage stochastic programming
Example

Two stages linear problem:

\[
\min_{x_1, x_2} \{ c^T x_1 : A_1 x_1 + A_2 x_2 \leq b \}
\]

- decision variables \(x_1\) must be determined immediately
- second-stage variables \(x_2\) are chosen after the parameters \(A_1, A_2\)
  and \(b\) are realized.
- \(x_2\) does not appear in the objective (not restrictive)
- \(\mathcal{U}\): uncertainty set for \(A_1, A_2, b\).
Example

\( \mathcal{U} \): uncertainty set for \( A^1, A^2, b \).

\[
\min_{x^1, x^2} \{ c^T x^1 : A^1 x^1 + A^2 x^2 \leq b \}
\]

In the standard robust formulation, both \( x^1 \) and \( x^2 \) are chosen before the uncertain parameters are observed:

\[
\min_{x^1, x^2} \{ c^T x^1 : A^1 x^1 + A^2 x^2 \leq b, \forall (A^1, A^2, b) \in \mathcal{U} \}
\]
Example

\( \mathcal{U} \): uncertainty set for \( A^1, A^2, b \).

\[
\min_{x^1, x^2} \{ c^T x^1 : A^1 x^1 + A^2 x^2 \leq b \}
\]

In the standard robust formulation, both \( x^1 \) and \( x^2 \) are chosen before the uncertain parameters are observed:

\[
\min_{x^1, x^2} \{ c^T x^1 : A^1 x^1 + A^2 x^2 \leq b, \forall (A^1, A^2, b) \in \mathcal{U} \}
\]

We can rewrite this as:

\[
\min_{x^1} \{ c^T x^1 : \exists x^2 \forall (A^1, A^2, b) \in \mathcal{U} : A^1 x^1 + A^2 x^2 \leq b \}
\]
Example

\( \mathcal{U} \): uncertainty set for \( A^1, A^2, b \).

\[
\min_{x^1, x^2} \{ c^T x^1 : A^1 x^1 + A^2 x^2 \leq b \}
\]

In the standard robust formulation, both \( x^1 \) and \( x^2 \) are chosen before the uncertain parameters are observed:

\[
\min_{x^1, x^2} \{ c^T x^1 : A^1 x^1 + A^2 x^2 \leq b, \forall (A^1, A^2, b) \in \mathcal{U} \}
\]

We can rewrite this as:

\[
\min_{x^1} \{ c^T x^1 : \exists x^2 \forall (A^1, A^2, b) \in \mathcal{U} : A^1 x^1 + A^2 x^2 \leq b \}
\]
Example

$$\min_{x^1, x^2} \{ c^T x^1 : \exists x^2 \forall (A^1, A^2, b) \in U : A^1 x^1 + A^2 x^2 \leq b \}$$
Example

\[ \min_{x^1, x^2} \{ c^T x^1 : \exists x^2 \forall (A^1, A^2, b) \in U : A^1 x^1 + A^2 x^2 \leq b \} \]

In the adjustable robust optimization formulation, the second-period variables \( x^2 \) may depend on the realized value of the parameters:

\[ \min_{x^1} \{ c^T x^1 : \forall (A^1, A^2, b) \in U, \exists x^2 = x^2(A^1, A^2, b) : A^1 x^1 + A^2 x^2 \leq b \} \]
Example

\[
\min_{x^1, x^2} \{c^T x^1 : \exists x^2 \forall (A^1, A^2, b) \in U : A^1 x^1 + A^2 x^2 \leq b\}
\]

In the adjustable robust optimization formulation, the second-period variables \(x^2\) may depend on the realized value of the parameters:

\[
\min_{x^1} \{c^T x^1 : \forall (A^1, A^2, b) \in U, \exists x^2 = x^2(A^1, A^2, b) : A^1 x^1 + A^2 x^2 \leq b\}
\]

The feasible set of this problem is larger than that of the top problem, hence the model is more flexible.

The price to pay is that ARO models are much harder to solve (often NP-Hard).
Tools and strategies

• We mostly focus on (simpler) reformulation strategy

• The goal is to rewrite the robust optimization problem as deterministic optimization problems with no uncertainty

• We aim at economy (small programs) and tractability (efficiency)

• There is no real single approach, because of the different flavours of robust optimization
Sampling

$$\min_x f(x, p)$$

$$G(x, p) \in K$$

- Very simple strategy: sample uncertain parameters from the uncertainty set.
Sampling

\[ \min_{x} f(x, p) \]
\[ G(x, p) \in K \]

- Very simple strategy: sample uncertain parameters from the uncertainty set.
- Assumes sampling is possible ...
- Sampling can be done with or without distributional assumptions
- We obtain a robust optimization formulation on a finite uncertainty set

\[ \min_{t, x} t \]
\[ t - f(x, p_i) \geq 0, i = 1, \ldots, k \]
\[ G(x, p_i) \in K, i = 1, \ldots, k. \]

- Apart from the larger number of constraints, solving the formulation is not more difficult than considering the non-robust formulation.
Conic optimization

• When the uncertainty set is infinite, the robust formulation is a *semi-infinite* problem: finite number of variables, infinite constraints.

• It is possible to formulate some semi-infinite optimization problems using a finite set of conic constraints.
Example

\[
\begin{align*}
\min_{x} c^T x \\
& a^T x + b \geq 0
\end{align*}
\]

where \((a, b)\) belong to the ellipsoidal uncertainty set

\[
\mathcal{U} = \{(a, b) = (a^0, b^0) + \sum_{j=1}^{k} u_j (a^j, b^j), \|u\| \leq 1\}
\]
Example

\[ \min_x c^T x \]
\[ a^T x + b \geq 0 \]

where \((a, b)\) belong to the ellipsoidal uncertainty set

\[ \mathcal{U} = \{(a, b) = (a^0, b^0) + \sum_{j=1}^{k} u_j (a^j, b^j), \|u\| \leq 1\} \]

It is equivalent to:

\[ \min_{x,z} c^T x \]
\[ a_j^T x + b_j = z_j, \quad j = 0, \ldots, k \]
\[ (z_0, z_1, \ldots, z_k) \in C_q \]

where \(C_q\) is the second-order cone:

\[ C_q = \{ x = (x_1, x_2, \ldots, x_k) \in \mathbb{R}^k : x_1 \geq \|(x_2, \ldots, x_k)\| \} \]
Saddle point characterizations

• The robust solution of some problems arising from objective uncertainty can be characterized using saddle-point conditions

• Requires that the original problem satisfies certain convexity assumptions

• There are specific efficient algorithms for saddle-point problems (e.g., interior point methods)
Consider

$$\min_{x \in S} \max_{p \in U} f(x, p)$$

The dual of this problem can be obtained by switching the order:

$$\max_{p \in U} \min_{x \in S} f(x, p)$$

Now we have:

**Lemma**

- If $f(x, p)$ is a convex function of $x$ and a concave function of $p$
- and if $S$ and $U$ are nonempty, and at least one of them is bounded

Then

- the optimal values of the two problems coincide
- and there is a saddle point $(x^*, p^*)$ such that

$$f(x^*, p^*) \leq f(x^*, p) \leq f(x, p^*)$$

$\forall x \in S, p \in U$
Consider

\[
\min_{x \in S} \max_{p \in U} f(x, p)
\]

The *dual* of this problem can be obtained by switching the order:

\[
\max_{p \in U} \min_{x \in S} f(x, p)
\]
Consider

$$\min_{x \in S} \max_{p \in U} f(x, p)$$

The dual of this problem can be obtained by switching the order:

$$\max_{p \in U} \min_{x \in S} f(x, p)$$

Now we have:

**Lemma**

*If*

$$f(x, p)$$ is a convex function of $x$ and a concave function of $p$.

And if $S$ and $U$ are nonempty, and at least one of them is bounded.

Then

- the optimal values of the two problems coincide
- there is a saddle point $(x^*, p^*)$ such that
  $$f(x^*, p^*) \leq f(x, p^*) \leq f(x, p), \quad \forall x \in S, p \in U$$
Consider

\[ \min_{x \in S} \max_{p \in U} f(x, p) \]

The *dual* of this problem can be obtained by switching the order:

\[ \max_{p \in U} \min_{x \in S} f(x, p) \]

Now we have:

**Lemma**

If

- \( f(x, p) \) is a convex function of \( x \) and a concave function of \( p \)
Consider
\[
\min_{x \in S} \max_{p \in U} f(x, p)
\]

The dual of this problem can be obtained by switching the order:
\[
\max_{p \in U} \min_{x \in S} f(x, p)
\]

Now we have:

Lemma

If
- \( f(x, p) \) is a convex function of \( x \) and a concave function of \( p \)
- and if \( S \) and \( U \) are nonempty, and at least one of them is bounded

Then

\[
\text{the optimal values of the two problems coincide}
\]

\[
\text{and there is a saddle point } (x^*, p^*) \text{ such that}
\]

\[
f(x^*, p^*) \leq f(x, p^*) \leq f(x^*, p), \quad \forall x \in S, p \in U
\]
Consider
\[ \min_{x \in S} \max_{p \in U} f(x, p) \]

The *dual* of this problem can be obtained by switching the order:
\[ \max_{p \in U} \min_{x \in S} f(x, p) \]

Now we have:

**Lemma**

*If*

- \( f(x, p) \) is a convex function of \( x \) and a concave function of \( p \)
- and if \( S \) and \( U \) are nonempty, and at least one of them is bounded

*Then*

- the optimal values of the two problems coincide
Consider
\[
\min_{x \in S} \max_{p \in U} f(x, p)
\]
The *dual* of this problem can be obtained by switching the order:
\[
\max_{p \in U} \min_{x \in S} f(x, p)
\]
Now we have:

**Lemma**

If
- \( f(x, p) \) is a convex function of \( x \) and a concave function of \( p \)
- and if \( S \) and \( U \) are nonempty, and at least one of them is bounded

Then
- the optimal values of the two problems coincide
- and there is a saddle point \( (x^*, p^*) \) such that
  \[
  f(x^*, p) \leq f(x^*, p^*) \leq f(x, p^*), \forall x \in S, p \in U
  \]
Robust portfolio selection

A possible formulation of Markovitz mean-variance optimization is:

$$\max_{x \in X} \mu^T x - \lambda x^T \Sigma x$$

- The set $X$ of feasible portfolios is predetermined without uncertainty (constraints on investments)
- There is uncertainty in estimating $\mu$ and $\Sigma$
- Small changes in the estimations lead to large changes in allocations.
- Errors in the estimation leads to inefficient portfolios
- A conservative investor shouldn’t trust point estimates
- She would be more comfortable with a portfolio that performs well under a number of different scenarios
Robust portfolio selection

- We must incorporate robustness in the portfolio optimization formulation.
- The model allows to give intervals for the return and the covariance matrix:

\[ U = \{ (\mu, \Sigma) : \mu^L \leq \mu \leq \mu^U, \Sigma^L \leq \Sigma \leq \Sigma^U, \Sigma \succeq 0 \} \]
Robust portfolio selection

- We must incorporate robustness in the portfolio optimization formulation.
- The model allows to give intervals for the return and the covariance matrix:

\[ U = \{ (\mu, \Sigma) : \mu^L \leq \mu \leq \mu^U, \Sigma^L \leq \Sigma \leq \Sigma^U, \Sigma \succeq 0 \} \]

- The robust optimization problem is

\[
\max \left\{ \min_{x \in X} \mu^T x - \lambda x^T \Sigma x \right\} \quad (\mu, \Sigma) \in U
\]
Robust portfolio selection

- We must incorporate robustness in the portfolio optimization formulation.
- The model allows to give intervals for the return and the covariance matrix:

\[ U = \{(\mu, \Sigma) : \mu^L \leq \mu \leq \mu^U, \Sigma^L \leq \Sigma \leq \Sigma^U, \Sigma \succeq 0\} \]

- The robust optimization problem is

\[ \max_{\mu, \Sigma} \min_{x \in X} \{ \mu^T x - \lambda x^T \Sigma x \} \]

- \( U \) is bounded, the function is convex in \( x \) and concave in \( (\mu, \sigma) \), so the saddle-point lemma holds.
- Hence we can solve

\[ \min_{(\mu, \Sigma) \in U} \{ \max_{x \in X} \mu^T x - \lambda x^T \Sigma x \}, \]

where the internal problem is convex in \( x \).
That's all!

- Final out later tonight
- Due 5/10
- No collaboration
- Send questions to TAs, answers on course ML.

Thank you!