Non-linear Programming and Solver

Vahan Mkrtchyan

1Lane Department of Computer Science and Electrical Engineering
West Virginia University

February 17, 2015
Motivating Examples
Motivating Examples

Problem Formulation
Motivating Examples
Problem Formulation
Constrained Optimization

The generalized reduced gradient method
Outline

1 Motivating Examples
2 Problem Formulation
3 Constrained Optimization
4 The generalized reduced gradient method
5 Non-smooth Optimization: Subgradient methods
Motivating Examples
Problem Formulation
Constrained Optimization
Non-smooth Optimization: Subgradient methods

Portfolio Optimization

Problem formulation
Suppose we have a sum of money $M$ to split among three managed investment funds, which claim to offer percentage rates of return $r_1, r_2,$ and $r_3$.

If we invest amounts $y_1, y_2,$ and $y_3$, we can expect our overall return to be $R = r_1 \cdot y_1 + r_2 \cdot y_2 + r_3 \cdot y_3$.

Assume that the management charge associated with the $i$-th fund is calculated as $c_i \cdot y_i$.

Then the total cost of investment is $C = c_1 \cdot y_1 + c_2 \cdot y_2 + c_3 \cdot y_3$.

Now, assume that we are aiming for a return $R_\rho\%$, and that we want to pay the least charges to achieve this.

Mkrtchyan
Optimization Methods in Finance
Suppose we have a sum of money $M$ to split among three managed investment funds, which claim to offer percentage rates of return $r_1, r_2,$ and $r_3$. If we invest amounts $y_1, y_2,$ and $y_3$, we can expect our overall return to be $R = r_1 \cdot y_1 + r_2 \cdot y_2 + r_3 \cdot y_3 \%$.

Assume that the management charge associated with the $i$-th fund is calculated as $c_i \cdot y_i$. Then the total cost of investment is $C = c_1 \cdot y_1 + c_2 \cdot y_2 + c_3 \cdot y_3$.

Now, assume that we are aiming for a return $R_{\rho} \%$, and that we want to pay the least charges to achieve this.
Suppose we have a sum of money $M$ to split among three managed investment funds, which claim to offer percentage rates of return $r_1$, $r_2$ and $r_3$. If we invest amounts $y_1$, $y_2$ and $y_3$, we can expect our overall return to be $R = r_1 \cdot y_1 + r_2 \cdot y_2 + r_3 \cdot y_3 \%$.

Assume that the management charge associated with the $i$-th fund is calculated as $c_i \cdot y_i$. Then the total cost of investment is $C = c_1 \cdot y_1 + c_2 \cdot y_2 + c_3 \cdot y_3$.

Now, assume that we are aiming for a return $R_{\rho} \%$, and that we want to pay the least charges to achieve this.
Suppose we have a sum of money $M$ to split among three managed investment funds, which claim to offer percentage rates of return $r_1$, $r_2$ and $r_3$.

If we invest amounts $y_1$, $y_2$ and $y_3$, we can expect our overall return to be

$$R = \frac{r_1 \cdot y_1 + r_2 \cdot y_2 + r_3 \cdot y_3}{M} \%.$$
Problem formulation

Suppose we have a sum of money $M$ to split among three managed investment funds, which claim to offer percentage rates of return $r_1$, $r_2$ and $r_3$.

If we invest amounts $y_1$, $y_2$ and $y_3$, we can expect our overall return to be

$$R = \frac{r_1 \cdot y_1 + r_2 \cdot y_2 + r_3 \cdot y_3}{M} \%. $$

Assume that the management charge associated with the $i$-th fund is calculated as $c_i \cdot y_i$. 
Motivating Examples

Portfolio Optimization

Problem formulation

Suppose we have a sum of money $M$ to split among three managed investment funds, which claim to offer percentage rates of return $r_1$, $r_2$ and $r_3$.

If we invest amounts $y_1$, $y_2$ and $y_3$, we can expect our overall return to be

$$R = \frac{r_1 \cdot y_1 + r_2 \cdot y_2 + r_3 \cdot y_3}{M} \%.$$ 

Assume that the management charge associated with the $i$-th fund is calculated as $c_i \cdot y_i$.

Then the total cost of investment is

$$C = c_1 \cdot y_1 + c_2 \cdot y_2 + c_3 \cdot y_3.$$
Motivating Examples

Problem Formulation

Constrained Optimization

The generalized reduced gradient method

Non-smooth Optimization: Subgradient methods

Portfolio Optimization

Problem formulation

Suppose we have a sum of money $M$ to split among three managed investment funds, which claim to offer percentage rates of return $r_1$, $r_2$ and $r_3$.

If we invest amounts $y_1$, $y_2$ and $y_3$, we can expect our overall return to be

$$R = \frac{r_1 \cdot y_1 + r_2 \cdot y_2 + r_3 \cdot y_3}{M} \%.$$  

Assume that the management charge associated with the $i$-th fund is calculated as $c_i \cdot y_i$.

Then the total cost of investment is

$$C = c_1 \cdot y_1 + c_2 \cdot y_2 + c_3 \cdot y_3.$$  

Now, assume that we are aiming for a return $R_\rho \%$, and that we want to pay the least charges to achieve this.
Motivating Examples
Problem Formulation
Constrained Optimization
Non-smooth Optimization: Subgradient methods

Portfolio Optimization

The amounts $y_1$, $y_2$, and $y_3$ need to be chosen so that the following conditions are satisfied:

\[
\begin{align*}
\min & \quad c_1 \cdot y_1 + c_2 \cdot y_2 + c_3 \cdot y_3 \\
& \quad r_1 \cdot y_1 + r_2 \cdot y_2 + r_3 \cdot y_3 = M \cdot R \cdot \rho \\
y_1 + y_2 + y_3 & = M \\
y_1 & \geq 0 \\
y_2 & \geq 0 \\
y_3 & \geq 0.
\end{align*}
\]

Remark
The last inequalities are included because investments must obviously be positive. If we tried to solve the problem without them, an optimization algorithm would attempt to reduce costs by making one or more of the $y_i$ large and negative.
The amounts $y_1, y_2$ and $y_3$ need to be chosen so that the following conditions are satisfied:

$$
\begin{align*}
\min & \quad c_1 \cdot y_1 + c_2 \cdot y_2 + c_3 \cdot y_3 \\
& \quad r_1 \cdot y_1 + r_2 \cdot y_2 + r_3 \cdot y_3 = M \cdot R \rho, \\
& \quad y_1 + y_2 + y_3 = M \\
& \quad y_1 \geq 0, \\
& \quad y_2 \geq 0, \\
& \quad y_3 \geq 0.
\end{align*}
$$

Remark: The last inequalities are included because investments must obviously be positive. If we tried to solve the problem without them, an optimization algorithm would attempt to reduce costs by making one or more of the $y_i$ large and negative.
Problem modeling

The amounts $y_1$, $y_2$ and $y_3$ need to be chosen so that the following conditions are satisfied:
Motivating Examples

Problem Formulation

Constrained Optimization

The generalized reduced gradient method

Non-smooth Optimization: Subgradient methods

Portfolio Optimization

Problem modeling

The amounts $y_1$, $y_2$ and $y_3$ need to be chosen so that the following conditions are satisfied:

$$\min c_1 \cdot y_1 + c_2 \cdot y_2 + c_3 \cdot y_3$$

Remark

The last inequalities are included because investments must obviously be positive. If we tried to solve the problem without them, an optimization algorithm would attempt to reduce costs by making one or more of the $y_i$ large and negative.
Portfolio Optimization

Problem modeling

The amounts \( y_1, y_2 \) and \( y_3 \) need to be chosen so that the following conditions are satisfied:

\[
\begin{align*}
\min & \quad c_1 \cdot y_1 + c_2 \cdot y_2 + c_3 \cdot y_3 \\
\text{subject to} & \quad r_1 \cdot y_1 + r_2 \cdot y_2 + r_3 \cdot y_3 = M \cdot R_p,
\end{align*}
\]

Remark

The last inequalities are included because investments must obviously be positive. If we tried to solve the problem without them, an optimization algorithm would attempt to reduce costs by making one or more of the \( y_i \) large and negative.
Problem modeling

The amounts $y_1$, $y_2$ and $y_3$ need to be chosen so that the following conditions are satisfied:

$$\begin{align*}
\min & \quad c_1 \cdot y_1 + c_2 \cdot y_2 + c_3 \cdot y_3 \\
r_1 \cdot y_1 + r_2 \cdot y_2 + r_3 \cdot y_3 &= M \cdot R_p, \\
y_1 + y_2 + y_3 &= M
\end{align*}$$

Remark

The last inequalities are included because investments must obviously be positive. If we tried to solve the problem without them, an optimization algorithm would attempt to reduce costs by making one or more of the $y_i$ large and negative.
The amounts $y_1$, $y_2$, and $y_3$ need to be chosen so that the following conditions are satisfied:

$$\begin{align*}
\min & \quad c_1 \cdot y_1 + c_2 \cdot y_2 + c_3 \cdot y_3 \\
& \quad r_1 \cdot y_1 + r_2 \cdot y_2 + r_3 \cdot y_3 = M \cdot R_p, \\
& \quad y_1 + y_2 + y_3 = M \\
& \quad y_1 \geq 0, y_2 \geq 0, y_3 \geq 0.
\end{align*}$$
Problem modeling

The amounts $y_1$, $y_2$ and $y_3$ need to be chosen so that the following conditions are satisfied:

\[
\begin{align*}
\min & \quad c_1 \cdot y_1 + c_2 \cdot y_2 + c_3 \cdot y_3 \\
\text{subject to} & \quad r_1 \cdot y_1 + r_2 \cdot y_2 + r_3 \cdot y_3 = M \cdot R_p, \\
& \quad y_1 + y_2 + y_3 = M \\
& \quad y_1 \geq 0, y_2 \geq 0, y_3 \geq 0.
\end{align*}
\]

Remark

The last inequalities are included because investments must obviously be positive. If we tried to solve the problem without them, an optimization algorithm would attempt to reduce costs by making one or more of the $y_i$ large and negative.
Portfolio Optimization

Problem modeling

The amounts $y_1$, $y_2$ and $y_3$ need to be chosen so that the following conditions are satisfied:

$$\begin{align*}
\min & \quad c_1 \cdot y_1 + c_2 \cdot y_2 + c_3 \cdot y_3 \\
& \quad r_1 \cdot y_1 + r_2 \cdot y_2 + r_3 \cdot y_3 = M \cdot R_p, \\
& \quad y_1 + y_2 + y_3 = M \\
& \quad y_1 \geq 0, y_2 \geq 0, y_3 \geq 0.
\end{align*}$$

Remark

The last inequalities are included because investments must obviously be positive.
Problem modeling

The amounts $y_1$, $y_2$ and $y_3$ need to be chosen so that the following conditions are satisfied:

$$
\begin{align*}
\min & \quad c_1 \cdot y_1 + c_2 \cdot y_2 + c_3 \cdot y_3 \\
& \quad r_1 \cdot y_1 + r_2 \cdot y_2 + r_3 \cdot y_3 = M \cdot R_\rho, \\
& \quad y_1 + y_2 + y_3 = M \\
& \quad y_1 \geq 0, y_2 \geq 0, y_3 \geq 0.
\end{align*}
$$

Remark

The last inequalities are included because investments must obviously be positive.

If we tried to solve the problem without them, an optimization algorithm would attempt to reduce costs by making one or more of the $y_i$ large and negative.
Motivating Examples
Problem Formulation
Constrained Optimization
The generalized reduced gradient method
Non-smooth Optimization: Subgradient methods

Portfolio Optimization

Problem formulation

Now, assume that an attempt to make a negative investment would be penalized by a very high management charge. Assume that the charge is given by the function $K \cdot \Psi(x)$, where

$$\Psi(x) = \begin{cases} x^2, & \text{if } x < 0 \\ 0, & \text{otherwise} \end{cases}$$

and $K$ is a large positive constant.

Then the problem becomes:

$$\min c_1 \cdot y_1 + c_2 \cdot y_2 + c_3 \cdot y_3 + K \cdot \sum_{i=1}^{3} \Psi(y_i)$$

subject to

$$r_1 \cdot y_1 + r_2 \cdot y_2 + r_3 \cdot y_3 = M \cdot R$$

and

$$y_1 + y_2 + y_3 = M$$
Motivating Examples

Problem Formulation

Constrained Optimization

The generalized reduced gradient method

Non-smooth Optimization: Subgradient methods

Portfolio Optimization

Problem formulation

Now, assume that an attempt to make a negative investment would be penalized by a very high management charge. Assume that the charge is given by the function

$$K \cdot \Psi(x)$$

where

$$\Psi(x) = \begin{cases} x^2, & \text{if } x < 0 \\ 0, & \text{otherwise.} \end{cases}$$

and

$$K$$ is a large positive constant.

Then the problem becomes:

$$\min c_1 \cdot y_1 + c_2 \cdot y_2 + c_3 \cdot y_3 + K \cdot \sum_{i=1}^{3} \Psi(y_i)$$

Subject to:

$$r_1 \cdot y_1 + r_2 \cdot y_2 + r_3 \cdot y_3 = M \cdot R \cdot \rho,$$

$$y_1 + y_2 + y_3 = M.$$
Problem formulation

Now, assume that an attempt to make a negative investment would be penalized by a very high management charge.
Problem formulation

Now, assume that an attempt to make a negative investment would be penalized by a very high management charge.

Assume that the charge is given by the function $K \cdot \Psi(x)$, where

$$
\Psi(x) = \begin{cases} 
x^2, & \text{if } x < 0 \\
0, & \text{otherwise.}
\end{cases}
$$

and $K$ is a large positive constant.
Portfolio Optimization

Problem formulation

Now, assume that an attempt to make a negative investment would be penalized by a very high management charge.

Assume that the charge is given by the function $K \cdot \Psi(x)$, where

$$\Psi(x) = \begin{cases} 
  x^2, & \text{if } x < 0 \\
  0, & \text{otherwise}.
\end{cases}$$

and $K$ is a large positive constant.

Then the problem becomes:

$$\min c_1 \cdot y_1 + c_2 \cdot y_2 + c_3 \cdot y_3 + K \cdot \sum_{i=1}^{3} \Psi(y_i)$$
Now, assume that an attempt to make a negative investment would be penalized by a very high management charge.

Assume that the charge is given by the function $K \cdot \Psi(x)$, where

$$\Psi(x) = \begin{cases} x^2, & \text{if } x < 0 \\ 0, & \text{otherwise.} \end{cases}$$

and $K$ is a large positive constant.

Then the problem becomes:

$$\min c_1 \cdot y_1 + c_2 \cdot y_2 + c_3 \cdot y_3 + K \cdot \sum_{i=1}^{3} \Psi(y_i)$$

$$r_1 \cdot y_1 + r_2 \cdot y_2 + r_3 \cdot y_3 = M \cdot R_\rho,$$
Motivating Examples

Problem Formulation

Constrained Optimization

The generalized reduced gradient method

Non-smooth Optimization: Subgradient methods

Portfolio Optimization

Problem formulation

Now, assume that an attempt to make a negative investment would be penalized by a very high management charge.

Assume that the charge is given by the function $K \cdot \psi(x)$, where

$$
\psi(x) = \begin{cases} 
  x^2, & \text{if } x < 0 \\
  0, & \text{otherwise.}
\end{cases}
$$

and $K$ is a large positive constant.

Then the problem becomes:

$$
\min c_1 \cdot y_1 + c_2 \cdot y_2 + c_3 \cdot y_3 + K \cdot \sum_{i=1}^{3} \psi(y_i)
$$

$$
r_1 \cdot y_1 + r_2 \cdot y_2 + r_3 \cdot y_3 = M \cdot R \rho,
$$

$$
y_1 + y_2 + y_3 = M
$$
Portfolio return and risk

Suppose we have a history of percentage returns, over $m$ time periods, for each of a group of $n$ assets (such as shares, bonds etc.). We can use this information as a guide to future investments. As an example, consider the following data for three assets over six months.

<table>
<thead>
<tr>
<th>Monthly rates of return on three assets</th>
<th>Assets 1</th>
<th>Assets 2</th>
<th>Assets 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>January</td>
<td>1.2</td>
<td>1.3</td>
<td>0.9</td>
</tr>
<tr>
<td>February</td>
<td>1.4</td>
<td>0.8</td>
<td>1.1</td>
</tr>
<tr>
<td>March</td>
<td>1.5</td>
<td>0.3</td>
<td>1.0</td>
</tr>
<tr>
<td>April</td>
<td>1.6</td>
<td>0.4</td>
<td>1.1</td>
</tr>
<tr>
<td>May</td>
<td>1.7</td>
<td>0.9</td>
<td>1.2</td>
</tr>
<tr>
<td>June</td>
<td>1.8</td>
<td>1.0</td>
<td>1.3</td>
</tr>
</tbody>
</table>

Mkrtchyan

Optimization Methods in Finance
Suppose we have a history of percentage returns, over $m$ time periods, for each of a $n$ assets (such as shares, bonds etc.). We can use this information as a guide to future investments. As an example, consider the following data for three assets over six months.

<table>
<thead>
<tr>
<th>Monthly rates of return on three assets</th>
<th>Assets/Months</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>January</td>
</tr>
<tr>
<td></td>
<td>February</td>
</tr>
<tr>
<td></td>
<td>March</td>
</tr>
<tr>
<td></td>
<td>April</td>
</tr>
<tr>
<td></td>
<td>May</td>
</tr>
<tr>
<td></td>
<td>June</td>
</tr>
<tr>
<td>Assets 1</td>
<td>1.2</td>
</tr>
<tr>
<td></td>
<td>1.3</td>
</tr>
<tr>
<td></td>
<td>1.4</td>
</tr>
<tr>
<td></td>
<td>1.5</td>
</tr>
<tr>
<td></td>
<td>1.1</td>
</tr>
<tr>
<td></td>
<td>1.2</td>
</tr>
<tr>
<td>Assets 2</td>
<td>1.3</td>
</tr>
<tr>
<td></td>
<td>1.0</td>
</tr>
<tr>
<td></td>
<td>0.8</td>
</tr>
<tr>
<td></td>
<td>0.9</td>
</tr>
<tr>
<td></td>
<td>1.4</td>
</tr>
<tr>
<td></td>
<td>1.3</td>
</tr>
<tr>
<td>Assets 3</td>
<td>0.9</td>
</tr>
<tr>
<td></td>
<td>1.1</td>
</tr>
<tr>
<td></td>
<td>1.0</td>
</tr>
<tr>
<td></td>
<td>1.1</td>
</tr>
<tr>
<td></td>
<td>1.1</td>
</tr>
<tr>
<td></td>
<td>1.3</td>
</tr>
</tbody>
</table>
Portfolio return and risk

Main concepts

Suppose we have a history of percentage returns, over $m$ time periods, for each of a group of $n$ assets (such as shares, bonds etc.).
Portfolio return and risk

Main concepts

Suppose we have a history of percentage returns, over \( m \) time periods, for each of a group of \( n \) assets (such as shares, bonds etc.).

We can use this information as a guide to future investments.

<table>
<thead>
<tr>
<th>Assets/Months</th>
<th>January</th>
<th>February</th>
<th>March</th>
<th>April</th>
<th>May</th>
<th>June</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assets 1</td>
<td>1.2</td>
<td>1.3</td>
<td>1.4</td>
<td>1.5</td>
<td>1.1</td>
<td>1.2</td>
</tr>
<tr>
<td>Assets 2</td>
<td>1.3</td>
<td>1.0</td>
<td>0.8</td>
<td>0.9</td>
<td>1.4</td>
<td>1.3</td>
</tr>
<tr>
<td>Assets 3</td>
<td>0.9</td>
<td>1.1</td>
<td>1.0</td>
<td>1.1</td>
<td>1.1</td>
<td>1.3</td>
</tr>
</tbody>
</table>
Main concepts

Suppose we have a history of percentage returns, over \( m \) time periods, for each of a group of \( n \) assets (such as shares, bonds etc.).

We can use this information as a guide to future investments.

As an example, consider the following data for three assets over six months.
Main concepts

Suppose we have a history of percentage returns, over $m$ time periods, for each of a group of $n$ assets (such as shares, bonds etc.).

We can use this information as a guide to future investments.

As an example, consider the following data for three assets over six months.

### Monthly rates of return on three assets

<table>
<thead>
<tr>
<th>Assets/Months</th>
<th>January</th>
<th>February</th>
<th>March</th>
<th>April</th>
<th>May</th>
<th>June</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assets 1</td>
<td>1.2</td>
<td>1.3</td>
<td>1.4</td>
<td>1.5</td>
<td>1.1</td>
<td>1.2</td>
</tr>
<tr>
<td>Assets 2</td>
<td>1.3</td>
<td>1.0</td>
<td>0.8</td>
<td>0.9</td>
<td>1.4</td>
<td>1.3</td>
</tr>
<tr>
<td>Assets 3</td>
<td>0.9</td>
<td>1.1</td>
<td>1.0</td>
<td>1.1</td>
<td>1.1</td>
<td>1.3</td>
</tr>
</tbody>
</table>
Main concepts

In general, we can calculate the mean return $\bar{r}_i$ for each asset as

$$\bar{r}_i = \sum_{m=1}^{m} r_{ij}$$

where $r_{ij}$ denotes the return on asset $i$ in period $j$. 

Portfolio return and risk
Portfolio return and risk

Main concepts

In general, we can calculate the mean return $\bar{r}_i$ for each asset as

$$\bar{r}_i = \frac{\sum_{j=1}^{m} r_{ij} m}{m},$$

where $r_{ij}$ denotes the return on asset $i$ in period $j$. 
Main concepts

In general, we can calculate the mean return $\bar{r}_i$ for each asset as

$$\bar{r}_i = \frac{\sum_{j=1}^{m} r_{ij}}{m},$$
In general, we can calculate the mean return $\bar{r}_i$ for each asset as

$$
\bar{r}_i = \frac{\sum_{j=1}^{m} r_{ij}}{m},
$$

where $r_{ij}$ denotes the return on asset $i$ in period $j$. 

Main concepts

Portfolio return and risk
Portfolio return and risk

If we spread an investment across the $n$ assets and if $y_i$ denotes the fraction invested in asset $i$ then the values of the $y_i$ define a portfolio. Since all investment must be split between the $n$ assets, the invested fractions must satisfy

$$S = n \sum_{i=1}^{n} y_i = 1$$

The expected portfolio return is given by

$$R = \sum_{i=1}^{n} \bar{r}_i \cdot y_i.$$
Motivating Examples
Problem Formulation
Constrained Optimization
The generalized reduced gradient method
Non-smooth Optimization: Subgradient methods

Portfolio return and risk

If we spread an investment across the $n$ assets and if $y_i$ denotes the fraction invested in asset $i$, then the values of the $y_i$ define a portfolio. Since all investment must be split between the $n$ assets, the invested fractions must satisfy

$$S = n \sum_{i=1}^{n} y_i = 1$$

The expected portfolio return is given by

$$R = \sum_{i=1}^{n} \bar{r}_i \cdot y_i.$$
Portfolio and expected Portfolio return

If we spread an investment across the $n$ assets and if $y_i$ denotes the fraction invested in asset $i$ then the values of the $y_i$ define a portfolio.
If we spread an investment across the $n$ assets and if $y_i$ denotes the fraction invested in asset $i$ then the values of the $y_i$ define a portfolio.

Since all investment must be split between the $n$ assets, the invested fractions must satisfy

$$S = \sum_{i=1}^{n} y_i = 1$$
Portfolio return and risk

**Portfolio and expected Portfolio return**

If we spread an investment across the $n$ assets and if $y_i$ denotes the fraction invested in asset $i$ then the values of the $y_i$ define a portfolio.

Since all investment must be split between the $n$ assets, the invested fractions must satisfy

\[ S = \sum_{i=1}^{n} y_i = 1 \]

The expected portfolio return is given by

\[ R = \sum_{i=1}^{n} \bar{r}_i \cdot y_i. \]
Portfolio return and risk

The risk associated with a particular portfolio is determined from variances and covariances that can be calculated from the history of returns $r_{ij}$. The variance of asset $i$ is $\sigma^2_i = \sum_{m \in m} (r_{ij} - \bar{r}_i)^2 m$, while the covariance of assets $i$ and $k$ is $\sigma_{ik} = \sum_{m \in m} (r_{ij} - \bar{r}_i) \cdot (r_{kj} - \bar{r}_k) m$. 
Portfolio return and risk

The risk associated with a particular portfolio is determined from variances and covariances that can be calculated from the history of returns $r_{ij}$. The variance of asset $i$ is $\sigma_i^2 = \sum_{j=1}^{m} (r_{ij} - \bar{r}_i)^2$, while the covariance of assets $i$ and $k$ is $\sigma_{ik} = \sum_{j=1}^{m} (r_{ij} - \bar{r}_i)(r_{kj} - \bar{r}_k)$. 
Portfolio return and risk

The risk associated with a particular portfolio is determined from variances and covariances that can be calculated from the history of returns $r_{ij}$. 

The variance of asset $i$ is $\sigma^2_i = \sum_{j=1}^{m} (r_{ij} - \bar{r}_i)^2/m$, while the covariance of assets $i$ and $k$ is $\sigma_{ik} = \sum_{j=1}^{m} (r_{ij} - \bar{r}_i) \cdot (r_{kj} - \bar{r}_k)/m$. 
Portfolio risk

The risk associated with a particular portfolio is determined from variances and covariances that can be calculated from the history of returns $r_{ij}$.

The variance of asset $i$ is

$$
\sigma^2_i = \frac{\sum_{j=1}^{m} (r_{ij} - \bar{r}_i)^2}{m},
$$
Portfolio return and risk

Portfolio risk

The risk associated with a particular portfolio is determined from variances and covariances that can be calculated from the history of returns $r_{ij}$.

The variance of asset $i$ is

$$\sigma_i^2 = \frac{\sum_{j=1}^m (r_{ij} - \bar{r}_i)^2}{m},$$

while the covariance of assets $i$ and $k$ is

$$\sigma_{ik} = \frac{\sum_{j=1}^m (r_{ij} - \bar{r}_i) \cdot (r_{kj} - \bar{r}_k)}{m}.$$
Motivating Examples
Problem Formulation
Constrained Optimization
The generalized reduced gradient method
Non-smooth Optimization: Subgradient methods

Portfolio return and risk

Portfolio variance

The variance of the portfolio defined by the investment fractions $y_1, \ldots, y_n$ is

$$V = \sum_{i=1}^{n} \sigma_i^2 \cdot y_i^2 + 2 \sum_{i=1}^{n} \sigma_{ij} \cdot y_i \cdot y_j,$$

which can be used as a measure of portfolio risk.
Portfolio return and risk

Portfolio variance

The variance of the portfolio defined by the investment fractions $y_1, \ldots, y_n$ is

$$V = \sum_{i=1}^{n} \sigma_i^2 \cdot y_i^2 + 2 \sum_{i=1}^{n} \sum_{j=i+1}^{n} \sigma_{ij} \cdot y_i \cdot y_j,$$

which can be used as a measure of portfolio risk.
Portfolio return and risk

**Portfolio variance**

The variance of the portfolio defined by the investment fractions $y_1, ..., y_n$ is

$$V = \sum_{i=1}^{n} \sigma_i^2 \cdot y_i^2 + 2 \cdot \sum_{i=1}^{n} \sum_{j=i+1}^{n} \sigma_{ij} \cdot y_i \cdot y_j,$$

which can be used as a measure of portfolio risk.
Portfolio return and risk: simple notation

Matrix-vector notation

The return and risk functions can be written more conveniently using matrix-vector notation.

The expression for expected return can be written as:

$$ R = \bar{r} \cdot y, $$

and

$$ V = y^T \cdot Q \cdot y, $$

where $\bar{r}$ denotes the vector of mean returns $\bar{r}_i$, and $Q = \|\sigma_{ij}\|$.

The constraint on partitions can be written as

$$ S = e \cdot y, $$

where $e = (1, 1, \ldots, 1)$.
The return and risk functions can be written more conveniently using matrix-vector notation.

The expression for expected return can be written as:

\[ R = \bar{r} \cdot y, \]

and

\[ V = y^T \cdot Q \cdot y, \]

where \( \bar{r} \) denotes the vector of mean returns \( \bar{r}_i \), and \( Q = \| \sigma_{ij} \| \).

The constraint on partitions can be written as

\[ S = e \cdot y, \]

where \( e = (1, 1, \ldots, 1) \).
## Portfolio return and risk: simple notation

### Matrix-vector notation

The return and risk functions can be written more conveniently using matrix-vector notation.

\[
R = \bar{r} \cdot y, \\
V = y^T \cdot Q \cdot y,
\]

where \(\bar{r}\) denotes the vector of mean returns \(\bar{r}_i\), and \(Q = \|\sigma_{ij}\|\).
The return and risk functions can be written more conveniently using matrix-vector notation.

The expression for expected return can be written as:

$$ R = \bar{r} \cdot y, $$
Matrix-vector notation

The return and risk functions can be written more conveniently using matrix-vector notation.

The expression for expected return can be written as:

\[ R = \bar{r} \cdot y, \]

and

\[ V = y^T \cdot Q \cdot y, \]
The return and risk functions can be written more conveniently using matrix-vector notation.

The expression for expected return can be written as:

\[ R = \bar{r} \cdot y, \]

and

\[ V = y^T \cdot Q \cdot y, \]

where \( \bar{r} \) denotes the vector of mean returns \( \bar{r}_i \), and \( Q = ||\sigma_{ij}|| \).
The return and risk functions can be written more conveniently using matrix-vector notation.

The expression for expected return can be written as:

\[ R = \bar{r} \cdot y, \]

and

\[ V = y^T \cdot Q \cdot y, \]

where \( \bar{r} \) denotes the vector of mean returns \( \bar{r}_i \), and \( Q = \|\sigma_{ij}\| \).

The constraint on partitions can be written as

\[ S = e \cdot y, \]

where \( e = (1, 1, ..., 1) \).
The basic minimum risk problem

A major concern in portfolio selection is the minimization of risk. In its simplest form, this means finding invested fractions $y_1, \ldots, y_n$, to solve the problem

$$\min_{y} \mathbf{y}^T \cdot \mathbf{Q} \cdot \mathbf{y} \quad \text{subject to} \quad \mathbf{e} \cdot \mathbf{y} = 1,$$

where $\mathbf{Q}$ is a positive semi-definite matrix representing the covariance of asset returns, and $\mathbf{e}$ is a vector of ones representing the constraint on the total investment.
The basic minimum risk problem

Problem formulation

A major concern in portfolio selection is the minimization of risk. In its simplest form, this means finding invested fractions \( y_1, \ldots, y_n \), to solve the problem

\[
\min \ y^\top \cdot Q \cdot y \quad \text{subject to} \quad e^\top \cdot y = 1,
\]

where \( Q \) is a matrix representing the risk associated with each investment.
A major concern in portfolio selection is the minimization of risk.
A major concern in portfolio selection is the minimization of risk.

In its simplest form, this means finding invested fractions $y_1, \ldots, y_n$, to solve the problem
A major concern in portfolio selection is the minimization of risk. In its simplest form, this means finding invested fractions $y_1, \ldots, y_n$, to solve the problem

$$\min y^T \cdot Q \cdot y$$
The basic minimum risk problem

Problem formulation

A major concern in portfolio selection is the minimization of risk.

In its simplest form, this means finding invested fractions $y_1, \ldots, y_n$, to solve the problem

$$\min y^T \cdot Q \cdot y$$

$$e \cdot y = 1,$$
The solution to the basic problem can sometimes be useful. But in practice we will normally be interested in both risk and return rather than risk on its own. In a rather general way, we can say that an optimal portfolio is one which gives “biggest return at lowest risk”. One way of trying to determine such a portfolio is to consider a composite function such as

$$F = -R + \rho \cdot V = -\bar{r} \cdot y + \rho \cdot y^T \cdot Q \cdot y.$$ 

The first term is the negative of the expected return and the second term is a multiple of the risk. If we choose invested fractions $y_i$ to minimize $F$, we can expect to obtain a large value for return coupled with a small value for risk. The positive constant $\rho$ controls the balance between return and risk.
Motivating Examples

Problem Formulation

Constrained Optimization

The generalized reduced gradient method

Non-smooth Optimization: Subgradient methods

Optimizing return and risk

Problem formulation

The solution to the basic problem can sometimes be useful. But in practice we will normally be interested in both risk and return rather than risk on its own.

In a rather general way, we can say that an optimal portfolio is one which gives "biggest return at lowest risk".

One way of trying to determine such a portfolio is to consider a composite function such as

\[ F = -\bar{r} \cdot y + \rho \cdot y^T \cdot Q \cdot y. \]

The first term is the negative of the expected return and the second term is a multiple of the risk.

If we choose invested fractions \( y \) to minimize \( F \), we can expect to obtain a large value for return coupled with a small value for risk.

The positive constant \( \rho \) controls the balance between return and risk.
Problem formulation

The solution to the basic problem can sometimes be useful.
Optimizing return and risk

**Problem formulation**

The solution to the basic problem can sometimes be useful.

But in practice we will normally be interested in both risk and return rather than risk on its own.
Optimizing return and risk

Problem formulation

The solution to the basic problem can sometimes be useful.

But in practice we will normally be interested in both risk and return rather than risk on its own.

In a rather general way, we can say that an optimal portfolio is one which gives "biggest return at lowest risk".
Optimizing return and risk

Problem formulation

The solution to the basic problem can sometimes be useful.

But in practice we will normally be interested in both risk and return rather than risk on its own.

In a rather general way, we can say that an optimal portfolio is one which gives "biggest return at lowest risk".

One way of trying to determine such a portfolio is to consider a composite function such as

\[ F = -R + \rho \cdot V = -\bar{r} \cdot y + \rho \cdot y^T \cdot Q \cdot y. \]
Problem formulation

The solution to the basic problem can sometimes be useful.

But in practice we will normally be interested in both risk and return rather than risk on its own.

In a rather general way, we can say that an optimal portfolio is one which gives "biggest return at lowest risk".

One way of trying to determine such a portfolio is to consider a composite function such as

\[ F = -R + \rho \cdot V = -\bar{r} \cdot y + \rho \cdot y^T \cdot Q \cdot y. \]

The first term is the negative of the expected return and the second term is a multiple of the risk.
Optimizing return and risk

Problem formulation

The solution to the basic problem can sometimes be useful.

But in practice we will normally be interested in both risk and return rather than risk on its own.

In a rather general way, we can say that an optimal portfolio is one which gives "biggest return at lowest risk".

One way of trying to determine such a portfolio is to consider a composite function such as

\[ F = -R + \rho \cdot V = -\bar{r} \cdot y + \rho \cdot y^T \cdot Q \cdot y. \]

The first term is the negative of the expected return and the second term is a multiple of the risk.

If we choose invested fractions \( y_i \) to minimize \( F \), we can expect to obtain a large value for return coupled with a small value for risk.
The solution to the basic problem can sometimes be useful.

But in practice we will normally be interested in both risk and return rather than risk on its own.

In a rather general way, we can say that an optimal portfolio is one which gives "biggest return at lowest risk".

One way of trying to determine such a portfolio is to consider a composite function such as

\[ F = -R + \rho \cdot V = -\bar{r} \cdot y + \rho \cdot y^T \cdot Q \cdot y. \]

The first term is the negative of the expected return and the second term is a multiple of the risk.

If we choose invested fractions \( y_i \) to minimize \( F \), we can expect to obtain a large value for return coupled with a small value for risk.

The positive constant \( \rho \) controls the balance between return and risk.
Optimizing return and risk

Mathematical formulation

\[ \min -\bar{r} \cdot y + \rho \cdot y^T \cdot Q \cdot y \]
\[ e \cdot y = 1 \]
Optimizing return and risk

Mathematical formulation

\[
\min -\bar{r} \cdot y + \rho \cdot y^T \cdot Q \cdot y
\]
Optimizing return and risk

Mathematical formulation

\[
\min -\bar{r} \cdot y + \rho \cdot y^T \cdot Q \cdot y \\
e \cdot y = 1,
\]
Problem formulation

The previous problem allows us to balance risk and return according to the choice of the parameter $\rho$.

Another approach could be to fix a target value for return, say $R_{\rho \%}$, and to consider the problem

$$\min_y y^T \cdot Q \cdot y - \bar{r} \cdot y = R_{\rho \%}, \quad e^T \cdot y = 1.$$
Minimum risk for specified return

Problem formulation
Problem formulation

The previous problem allows us to balance risk and return according to the choice of the parameter $\rho$. 

$$\min_y \mathbf{y}^T \mathbf{Q} \mathbf{y} = \bar{r} \cdot \mathbf{y} = R_{\rho}, \quad e \cdot \mathbf{y} = 1.$$
Problem formulation

The previous problem allows us to balance risk and return according to the choice of the parameter $\rho$.

Another approach could be to fix a target value for return, say $R_\rho\%$, and to consider the problem

$$\min y^T \cdot Q \cdot y - \bar{r} \cdot y = R_\rho, \quad e \cdot y = 1.$$
Minimum risk for specified return

Problem formulation

The previous problem allows us to balance risk and return according to the choice of the parameter $\rho$.

Another approach could be to fix a target value for return, say $R_\rho\%$, and to consider the problem

$$\min y^T \cdot Q \cdot y$$
Minimum risk for specified return

**Problem formulation**

The previous problem allows us to balance risk and return according to the choice of the parameter $\rho$.

Another approach could be to fix a target value for return, say $R_\rho\%$, and to consider the problem

$$
\min y^T \cdot Q \cdot y \\
\tilde{r} \cdot y = R_\rho,
$$

where $y$ is the vector of decision variables, $Q$ is the covariance matrix, $\tilde{r}$ is the vector of returns, and $R_\rho$ is the target return.
Minimum risk for specified return

Problem formulation

The previous problem allows us to balance risk and return according to the choice of the parameter $\rho$.

Another approach could be to fix a target value for return, say $R_\rho \%$, and to consider the problem

$$\min y^T \cdot Q \cdot y \quad \bar{r} \cdot y = R_\rho, \quad e \cdot y = 1.$$
Problem formulation

Suppose we want to fix an acceptable level of risk (as $V_a$, say) and then to maximize the expected return. This can be posed as the constrained minimization problem:

$$\max \bar{r} \cdot y - y^T \cdot Q \cdot y = V_a,$$

$$e \cdot y = 1.$$
Motivating Examples
Problem Formulation
Constrained Optimization
The generalized reduced gradient method
Non-smooth Optimization: Subgradient methods

Maximum return problem

Problem formulation

Suppose we want to fix an acceptable level of risk (as $V$, say) and then to maximize the expected return. This can be posed as the constrained minimization problem:

$$\max \bar{r} \cdot y$$
$$y \cdot Q \cdot y = V,$$
$$e \cdot y = 1.$$
Suppose we want to fix an acceptable level of risk (as $V_a$, say) and then to maximize the expected return.
**Problem formulation**

Suppose we want to fix an acceptable level of risk (as $V_a$, say) and then to maximize the expected return.

This can be posed as the constrained minimization problem

$$\max \bar{r} \cdot y$$
Motivating Examples
Problem Formulation
Constrained Optimization
The generalized reduced gradient method
Non-smooth Optimization: Subgradient methods

Maximum return problem

Problem formulation

Suppose we want to fix an acceptable level of risk (as \( V_a \), say) and then to maximize the expected return.

This can be posed as the constrained minimization problem

\[
\max \tilde{r} \cdot y
\]

\[
y^T \cdot Q \cdot y = V_a,
\]
Maximum return problem

Problem formulation

Suppose we want to fix an acceptable level of risk (as $V_a$, say) and then to maximize the expected return.

This can be posed as the constrained minimization problem

$$\text{max} \; \bar{r} \cdot y$$

$$y^T \cdot Q \cdot y = V_a, \quad e \cdot y = 1.$$
Fundamentals

Given a function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ and a set $S \subseteq \mathbb{R}^n$, the problem of finding an $x^* \in \mathbb{R}^n$ that solves

$$\min_{x \in S} f(x)$$

is called an optimization problem.

Features of an optimization problem

- Decision variables.
- Objective function.
- Feasible region (Infeasibility, Unboundedness, Discrete).
- Global minimizer (strict).
- Local minimizer.
Motivating Examples
Problem Formulation
Constrained Optimization
The generalized reduced gradient method
Non-smooth Optimization: Subgradient methods

Optimization Theory

Fundamentals
Optimization Theory

Fundamentals

Given a function \( f : \mathbb{R}^n \rightarrow \mathbb{R} \) and a set \( S \subseteq \mathbb{R}^n \), the problem of finding an \( x^* \in \mathbb{R}^n \) that solves

\[
\min_{x} f(x) \\
\text{s.t. } x \in S
\]

is called an optimization problem.
Motivating Examples
Problem Formulation
Constrained Optimization
The generalized reduced gradient method
Non-smooth Optimization: Subgradient methods

Optimization Theory

Fundamentals

Given a function \( f : \mathbb{R}^n \rightarrow \mathbb{R} \) and a set \( S \subseteq \mathbb{R}^n \), the problem of finding an \( \mathbf{x}^* \in \mathbb{R}^n \) that solves

\[
\min_{\mathbf{x}} f(\mathbf{x}) \\
\mathbf{x} \in S
\]

is called an optimization problem.

Features of an optimization problem
Fundamentals

Given a function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ and a set $S \subseteq \mathbb{R}^n$, the problem of finding an $x^* \in \mathbb{R}^n$ that solves

$$
\min_{x} f(x)
$$

$x \in S$

is called an optimization problem.

Features of an optimization problem

- Decision variables.
Given a function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ and a set $S \subseteq \mathbb{R}^n$, the problem of finding an $x^* \in \mathbb{R}^n$ that solves

$$\min_{x} f(x)$$

$$x \in S$$

is called an optimization problem.

**Features of an optimization problem**
- Decision variables.
- Objective function.
### Optimization Theory

#### Fundamentals

Given a function $f : \mathbb{R}^n \to \mathbb{R}$ and a set $S \subseteq \mathbb{R}^n$, the problem of finding an $x^* \in \mathbb{R}^n$ that solves

$$\min_x f(x) \quad x \in S$$

is called an optimization problem.

#### Features of an optimization problem

- Decision variables.
- Objective function.
- Feasible region
Optimization Theory

Fundamentals

Given a function $f : R^n \rightarrow R$ and a set $S \subseteq R^n$, the problem of finding an $x^* \in R^n$ that solves

$$\min_{x} f(x) \quad x \in S$$

is called an optimization problem.

Features of an optimization problem

- Decision variables.
- Objective function.
- Feasible region (Infeasibility,
Motivating Examples

Problem Formulation

Constrained Optimization

The generalized reduced gradient method

Non-smooth Optimization: Subgradient methods

Optimization Theory

Fundamentals

Given a function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ and a set $S \subseteq \mathbb{R}^n$, the problem of finding an $x^* \in \mathbb{R}^n$ that solves

$$\min_x f(x)$$

$$x \in S$$

is called an optimization problem.

Features of an optimization problem

- Decision variables.
- Objective function.
- Feasible region (Infeasibility, Unboundedness,
Given a function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ and a set $S \subseteq \mathbb{R}^n$, the problem of finding an $x^* \in \mathbb{R}^n$ that solves

$$
\min_{x} f(x) \\
\quad x \in S
$$

is called an optimization problem.

**Features of an optimization problem**

- Decision variables.
- Objective function.
- Feasible region (Infeasibility, Unboundedness, Discrete).
Given a function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ and a set $S \subseteq \mathbb{R}^n$, the problem of finding an $x^* \in \mathbb{R}^n$ that solves

$$\min_{x} f(x)$$

$$x \in S$$

is called an optimization problem.

Features of an optimization problem

- Decision variables.
- Objective function.
- Feasible region (Infeasibility, Unboundedness, Discrete).
- Global minimizer (strict).
Fundamentals

Given a function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ and a set $S \subseteq \mathbb{R}^n$, the problem of finding an $x^* \in \mathbb{R}^n$ that solves

$$\min_{x} f(x)$$
$$x \in S$$

is called an optimization problem.

Features of an optimization problem

- Decision variables.
- Objective function.
- Feasible region (Infeasibility, Unboundedness, Discrete).
- Global minimizer (strict).
- Local minimizer.
Motivating Examples

Problem Formulation

Constrained Optimization

The generalized reduced gradient method

Non-smooth Optimization: Subgradient methods

Non-linear programming

\[
\begin{align*}
\text{min} & \quad f(x) \\
\text{subject to} & \quad g_i(x) = 0, \quad i \in E \\
& \quad g_i(x) \geq 0, \quad i \in I
\end{align*}
\]

Such optimization problems usually are called non-linear programming problems or non-linear programs.

Constrained or Unconstrained Optimization

A non-linear program in which the set \(E \cup I\) is empty, is called an unconstrained program. Otherwise, it is constrained.
Non-linear programming

**Problem Formulation**

\[
\begin{align*}
\min & \quad f(x) \\
\text{s.t.} & \quad g_i(x) = 0, \quad i \in E \\
& \quad g_i(x) \geq 0, \quad i \in I
\end{align*}
\]

Such optimization problems are usually called **non-linear programming problems** or **non-linear programs**.

**Constrained or Unconstrained Optimization**

A non-linear program in which the set \(E \cup I\) is empty, is called an **unconstrained program**. Otherwise, it is **constrained**.
Non-linear programming

Problem Formulation

\[ \min f(x) \]
Non-linear programming

Problem Formulation

\[
\begin{align*}
\min f(x) \\
g_i(x) &= 0, \ i \in \mathcal{E}
\end{align*}
\]
Problem Formulation

\[
\begin{align*}
\min & \quad f(x) \\
\text{s.t.} & \quad g_i(x) = 0, \quad i \in \mathcal{E} \\
& \quad g_i(x) \geq 0, \quad i \in \mathcal{I}
\end{align*}
\]
Non-linear programming

Problem Formulation

\[
\begin{align*}
\min f(x) \\
g_i(x) &= 0, \; i \in \mathcal{E} \\
g_i(x) \geq 0, \; i \in \mathcal{I}
\end{align*}
\]

Non-linear programs
Non-linear programming

Problem Formulation

\[ \min f(x) \]
\[ g_i(x) = 0, \ i \in \mathcal{E} \]
\[ g_i(x) \geq 0, \ i \in \mathcal{I} \]

Non-linear programs

Such optimization problems usually are called *non-linear programming problems*.
### Problem Formulation

\[
\begin{align*}
\min f(x) \\
g_i(x) &= 0, \ i \in \mathcal{E} \\
g_i(x) &\geq 0, \ i \in \mathcal{I}
\end{align*}
\]

### Non-linear programs

Such optimization problems usually are called *non-linear programming problems* or *non-linear programs*. 
## Problem Formulation

\[
\min f(x) \\
g_i(x) = 0, \; i \in \mathcal{E} \\
g_i(x) \geq 0, \; i \in \mathcal{I}
\]

## Non-linear programs

Such optimization problems usually are called *non-linear programming problems* or *non-linear programs*.

## Constrained or Unconstrained Optimization
Non-linear programming

Problem Formulation

\[
\begin{align*}
\min & \quad f(x) \\
g_i(x) &= 0, \quad i \in \mathcal{E} \\
g_i(x) &\geq 0, \quad i \in \mathcal{I}
\end{align*}
\]

Non-linear programs

Such optimization problems usually are called *non-linear programming problems* or *non-linear programs*.

Constrained or Unconstrained Optimization

A non-linear program in which the set \( \mathcal{E} \cup \mathcal{I} \) is empty, is called an unconstrained program.
## Problem Formulation

\[
\begin{align*}
\min f(x) \\
g_i(x) &= 0, \quad i \in \mathcal{E} \\
g_i(x) &\geq 0, \quad i \in \mathcal{I}
\end{align*}
\]

## Non-linear programs

Such optimization problems usually are called *non-linear programming problems* or *non-linear programs*.

## Constrained or Unconstrained Optimization

A non-linear program in which the set \(\mathcal{E} \cup \mathcal{I}\) is empty, is called an unconstrained program.

Otherwise, it is constrained.
Non-linear programs arise

Probabilistic elements
Non-linear programs arise

Probabilistic elements

Nonlinearities may arise when some of the coefficients in the model are random variables.
Non-linear programs arise

Probabilistic elements

Nonlinearities may arise when some of the coefficients in the model are random variables.

For example, consider a linear program, where the right-hand sides are random.
Non-linear programs arise

Probabilistic elements

Nonlinearities may arise when some of the coefficients in the model are random variables.

For example, consider a linear program, where the right-hand sides are random.

We consider the case when we have two constraints.
Probabilistic elements in linear programming
Problem Formulation

\[ \text{max } c_1 \cdot x_1 + \ldots + c_n \cdot x_n \]
\[ \text{subject to } a_{11} \cdot x_1 + \ldots + a_{1n} \cdot x_n \leq b_1 \]
\[ a_{21} \cdot x_1 + \ldots + a_{2n} \cdot x_n \leq b_2 \]

where the coefficients \( b_1 \) and \( b_2 \) are independently distributed and \( G_i(y) \) represents the probability that the random variable \( b_i \) is at least as large as \( y \).
Motivating Examples

Problem Formulation

Constrained Optimization

The generalized reduced gradient method

Non-smooth Optimization: Subgradient methods

Probabilistic elements in linear programming

Problem Formulation

\[ \max c_1 \cdot x_1 + \ldots + c_n \cdot x_n \]
Problem Formulation

\[ \max c_1 \cdot x_1 + ... + c_n \cdot x_n \]

\[ a_{11} \cdot x_1 + ... + a_{1n} \cdot x_n \leq b_1 \]
Problem Formulation

\[
\begin{align*}
\text{max } & \quad c_1 \cdot x_1 + \ldots + c_n \cdot x_n \\
\text{subject to } & \quad a_{11} \cdot x_1 + \ldots + a_{1n} \cdot x_n \leq b_1 \\
& \quad a_{21} \cdot x_1 + \ldots + a_{2n} \cdot x_n \leq b_2
\end{align*}
\]
Problem Formulation

\[ \max c_1 \cdot x_1 + \ldots + c_n \cdot x_n \]
\[ a_{11} \cdot x_1 + \ldots + a_{1n} \cdot x_n \leq b_1 \]
\[ a_{21} \cdot x_1 + \ldots + a_{2n} \cdot x_n \leq b_2 \]

where the coefficients \( b_1 \) and \( b_2 \) are independently distributed and \( G_i(y) \) represents the probability that the random variable \( b_i \) is at least as large as \( y \).
The joint probability of both of the constraints being satisfied is at least \( \beta \).

The last constraint mathematically can be written as follows:

\[
\Pr\left[ a_{11} \cdot x_1 + \ldots + a_{1n} \cdot x_n \leq b_1 \right] \times \Pr\left[ a_{21} \cdot x_1 + \ldots + a_{2n} \cdot x_n \leq b_2 \right] \geq \beta.
\]

Then this condition can be written as the following set of constraints:

\[
-y_1 + a_{11} \cdot x_1 + \ldots + a_{1n} \cdot x_n = 0
\]
\[
-y_2 + a_{21} \cdot x_1 + \ldots + a_{2n} \cdot x_n = 0
\]

\( G_1(y_1) \times G_2(y_2) \geq \beta. \)
Problem Formulation

Assume that we would like to choose the variables $x_1, \ldots, x_n$, so that the joint probability of both of the constraints being satisfied is at least $\beta$.

The last constraint mathematically can be written as follows:

$$\Pr\left[ a_{11} \cdot x_1 + \ldots + a_{1n} \cdot x_n \leq b_1 \right] \times \Pr\left[ a_{21} \cdot x_1 + \ldots + a_{2n} \cdot x_n \leq b_2 \right] \geq \beta.$$ 

Then this condition can be written as the following set of constraints:

$$-y_1 + a_{11} \cdot x_1 + \ldots + a_{1n} \cdot x_n = 0$$
$$-y_2 + a_{21} \cdot x_1 + \ldots + a_{2n} \cdot x_n = 0$$

$$G_1(y_1) \times G_2(y_2) \geq \beta.$$
## Problem Formulation

Assume that we would like to choose the variables $x_1, \ldots, x_n$, so that the joint probability of both of the constraints being satisfied is at least $\beta$. 

\[
\begin{align*}
\Pr[a_{11} \cdot x_1 + \ldots + a_{1n} \cdot x_n \leq b_1] \times \Pr[a_{21} \cdot x_1 + \ldots + a_{2n} \cdot x_n \leq b_2] & \geq \beta.
\end{align*}
\]

\[
\begin{align*}
G_1(y_1) \times G_2(y_2) & \geq \beta.
\end{align*}
\]
Problem Formulation

Assume that we would like to choose the variables $x_1, \ldots, x_n$, so that the joint probability of both of the constraints being satisfied is at least $\beta$.

The last constraint mathematically can be written as follows:

$$Pr[a_{11} \cdot x_1 + \ldots + a_{1n} \cdot x_n \leq b_1] \times Pr[a_{21} \cdot x_1 + \ldots + a_{2n} \cdot x_n \leq b_2] \geq \beta.$$
Problem Formulation

Assume that we would like to choose the variables $x_1, \ldots, x_n$, so that the joint probability of both of the constraints being satisfied is at least $\beta$.

The last constraint mathematically can be written as follows:

$$Pr[a_{11} \cdot x_1 + \ldots + a_{1n} \cdot x_n \leq b_1] \times Pr[a_{21} \cdot x_1 + \ldots + a_{2n} \cdot x_n \leq b_2] \geq \beta.$$ 

Then this condition can be written as the following set of constraints:

$$-y_1 + a_{11} \cdot x_1 + \ldots + a_{1n} \cdot x_n = 0$$
Problem Formulation

Assume that we would like to choose the variables $x_1, \ldots, x_n$, so that the joint probability of both of the constraints being satisfied is at least $\beta$.

The last constraint mathematically can be written as follows:

$$Pr[a_{11} \cdot x_1 + \ldots + a_{1n} \cdot x_n \leq b_1] \times Pr[a_{21} \cdot x_1 + \ldots + a_{2n} \cdot x_n \leq b_2] \geq \beta.$$ 

Then this condition can be written as the following set of constraints:

$$-y_1 + a_{11} \cdot x_1 + \ldots + a_{1n} \cdot x_n = 0$$
$$-y_2 + a_{21} \cdot x_1 + \ldots + a_{2n} \cdot x_n = 0$$
Problem Formulation

Assume that we would like to choose the variables $x_1, \ldots, x_n$, so that the joint probability of both of the constraints being satisfied is at least $\beta$.

The last constraint mathematically can be written as follows:

$$Pr[a_{11} \cdot x_1 + \ldots + a_{1n} \cdot x_n \leq b_1] \times Pr[a_{21} \cdot x_1 + \ldots + a_{2n} \cdot x_n \leq b_2] \geq \beta.$$ 

Then this condition can be written as the following set of constraints:

$$-y_1 + a_{11} \cdot x_1 + \ldots + a_{1n} \cdot x_n = 0$$

$$-y_2 + a_{21} \cdot x_1 + \ldots + a_{2n} \cdot x_n = 0$$

$$G_1(y_1) \times G_2(y_2) \geq \beta.$$
Motivating Examples
Problem Formulation
Constrained Optimization
The generalized reduced gradient method
Non-smooth Optimization: Subgradient methods

Constrained Optimization

\[
\begin{align*}
\min & \quad f(x) \\
\text{subject to} & \quad g_i(x) = 0, i \in E \\
                  & \quad g_i(x) \geq 0, i \in I
\end{align*}
\]

Here we assume that we have at least one constraint, i.e., the set \( E \cup I \) is not empty. Moreover, we assume that the functions \( f \) and \( g_i \) are continuously differentiable.
Motivating Examples

Non-smooth Optimization: Subgradient methods

Constrained Optimization

The generalized reduced gradient method

Problem Formulation

Here we assume that we have at least one constraint, i.e., the set $E \cup I$ is not empty. Moreover, we assume that the functions $f$ and $g_i$ are continuously differentiable.

$$\min f(x)$$

$$g_i(x) = 0, \quad i \in E$$

$$g_i(x) \geq 0, \quad i \in I$$
Constrained Optimization

Problem Formulation

\[ \min f(x) \]

Here we assume that we have at least one constraint, i.e., the set \( E \cup I \) is not empty. Moreover, we assume that the functions \( f \) and \( g_i \) are continuously differentiable.
Motivating Examples
Problem Formulation
Constrained Optimization
The generalized reduced gradient method
Non-smooth Optimization: Subgradient methods

Constrained Optimization

Problem Formulation

\[
\min f(x) \\
g_i(x) = 0, \ i \in \mathcal{E}
\]
Constrained Optimization

Problem Formulation

\[ \min f(x) \]
\[ g_i(x) = 0, \quad i \in \mathcal{E} \]
\[ g_i(x) \geq 0, \quad i \in \mathcal{I} \]
Constrained Optimization

Problem Formulation

\[ \min f(x) \]
\[ g_i(x) = 0, \ i \in \mathcal{E} \]
\[ g_i(x) \geq 0, \ i \in \mathcal{I} \]

Here we assume that we have at least one constraint, i.e., the set \( \mathcal{E} \cup \mathcal{I} \) is not empty.
Motivating Examples
Problem Formulation
Constrained Optimization
The generalized reduced gradient method
Non-smooth Optimization: Subgradient methods

Constrained Optimization

Problem Formulation

\[
\begin{align*}
\min f(x) \\
g_i(x) &= 0, \quad i \in \mathcal{E} \\
g_i(x) &\geq 0, \quad i \in \mathcal{I}
\end{align*}
\]

Here we assume that we have at least one constraint, i.e., the set \( \mathcal{E} \cup \mathcal{I} \) is not empty.

Moreover, we assume that the functions \( f \) and \( g_i \) are continuously differentiable.
Non-smooth Optimization: Subgradient methods

The Lagrangian function (or Lagrangian) is defined as follows:

\[ L(x, \lambda) = f(x) - \sum_{i \in E \cup I} \lambda_i \cdot g_i(x). \]

Why Lagrangian function?
It turns out that for suitably chosen values of \( \lambda_i \), minimizing the unconstrained Lagrangian function \( L(x, \lambda) \) is equivalent to solving the above constrained non-linear program.
Lagrangian function of the problem

The Lagrangian function (or Lagrangian) is defined as follows:

\[ L(x, \lambda) = f(x) - \sum_{i \in E \cup I} \lambda_i \cdot g_i(x). \]

Why Lagrangian function?
It turns out that for suitably chosen values of \( \lambda_i \), minimizing the unconstrained Lagrangian function \( L(x, \lambda) \) is equivalent to solving the above constrained non-linear program.
Motivating Examples

Problem Formulation

Constrained Optimization

The generalized reduced gradient method

Non-smooth Optimization: Subgradient methods

Constrained Optimization

Lagrangian function of the problem

The Lagrangian function (or Lagrangian) is defined as follows:

\[ L(x, \lambda) = f(x) - \sum_{i \in \mathcal{E} \cup \mathcal{I}} \lambda_i \cdot g_i(x). \]
Motivating Examples
Problem Formulation
Constrained Optimization
The generalized reduced gradient method
Non-smooth Optimization: Subgradient methods

Constrained Optimization

Lagrangian function of the problem

The Lagrangian function (or Lagrangian) is defined as follows:

\[ L(x, \lambda) = f(x) - \sum_{i \in E \cup I} \lambda_i \cdot g_i(x). \]

Why Lagrangian function?

It turns out that for suitably chosen values of \( \lambda_i \), minimizing the unconstrained Lagrangian function \( L(x, \lambda) \) is equivalent to solving the above constrained non-linear program.
The Lagrangian function (or Lagrangian) is defined as follows:

\[ L(x, \lambda) = f(x) - \sum_{i \in E \cup I} \lambda_i \cdot g_i(x). \]

Why Lagrangian function?

It turns out that for suitably chosen values of \( \lambda_i \), minimizing the unconstrained Lagrangian function \( L(x, \lambda) \) is equivalent to solving the above constrained non-linear program.
Some definitions

**Definition**
A point $x$ satisfying $g_i(x) = 0, \ i \in E$ and $g_i(x) \geq 0, \ i \in I$ is called a feasible solution to the non-linear program.

**Definition**
Let $x$ be a feasible solution to the non-linear program, and let $J \subseteq I$ be the set of indices for which $g_i(x) \geq 0$ is satisfied with equality. Then $x$ is a regular point of the program, if the gradient vectors $\nabla g_i(x)$ for $i \in E \cup J$ are linearly independent.
Some definitions

<table>
<thead>
<tr>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>A point $x$ satisfying $g_i(x) = 0$, $i \in E$ and $g_i(x) \geq 0$, $i \in I$ is called a feasible solution to the non-linear program.</td>
</tr>
</tbody>
</table>

Definition

Let $x$ be a feasible solution to the non-linear program, and let $J \subseteq I$ be the set of indices for which $g_i(x) \geq 0$ is satisfied with equality. Then $x$ is a regular point of the program, if the gradient vectors $\nabla g_i(x)$ for $i \in E \cup J$ are linearly independent.
Some definitions

Definition

A point $\mathbf{x}$ satisfying $g_i(\mathbf{x}) = 0$, $i \in \mathcal{E}$ and $g_i(\mathbf{x}) \geq 0$, $i \in \mathcal{I}$ is called a feasible solution to the non-linear program.
**Some definitions**

<table>
<thead>
<tr>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>A point $\mathbf{x}$ satisfying $g_i(\mathbf{x}) = 0$, $i \in \mathcal{E}$ and $g_i(\mathbf{x}) \geq 0$, $i \in \mathcal{I}$ is called a feasible solution to the non-linear program.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Let $\mathbf{x}$ be a feasible solution to the non-linear program,</td>
</tr>
</tbody>
</table>
Some definitions

**Definition**
A point \( \mathbf{x} \) satisfying \( g_i(\mathbf{x}) = 0, \ i \in \mathcal{E} \) and \( g_i(\mathbf{x}) \geq 0, \ i \in \mathcal{I} \) is called a feasible solution to the non-linear program.

**Definition**
Let \( \mathbf{x} \) be a feasible solution to the non-linear program, and let \( \mathcal{J} \subseteq \mathcal{I} \) be the set of indices for which \( g_i(\mathbf{x}) \geq 0 \) is satisfied with equality.
Some definitions

Definition
A point $\mathbf{x}$ satisfying $g_i(\mathbf{x}) = 0$, $i \in \mathcal{E}$ and $g_i(\mathbf{x}) \geq 0$, $i \in \mathcal{I}$ is called a feasible solution to the non-linear program.

Definition
Let $\mathbf{x}$ be a feasible solution to the non-linear program, and let $\mathcal{J} \subseteq \mathcal{I}$ be the set of indices for which $g_i(\mathbf{x}) \geq 0$ is satisfied with equality.

Then $\mathbf{x}$ is a regular point of the program, if the gradient vectors $\nabla g_i(\mathbf{x})$ for $i \in \mathcal{E} \cup \mathcal{J}$ are linearly independent.
Some examples

### Motivating Examples

#### Problem Formulation

The generalized reduced gradient method

#### Constrained Optimization

Non-smooth Optimization: Subgradient methods

---

#### Some examples

Problem Formulation: Regular points

\[
\begin{align*}
\max (x^2 + y^2) \\
x \geq 0 \\
y \geq 0 \\
x + y \leq 1
\end{align*}
\]

In this example any feasible point is regular, since the gradients of the constraints are

\[
\left(1, 0\right), \quad \left(0, 1\right), \quad \left(-1, -1\right)
\]
Some examples

Problem Formulation: Regular points

\[
\begin{align*}
\text{max} & \quad (x^2 + y^2) \\
\text{s.t.} & \quad x \geq 0, y \geq 0, x + y \leq 1
\end{align*}
\]

In this example any feasible point is regular, since the gradients of the constraints are \((1,0), (0,1), (-1,-1)\).
First example: regular points

\[
\max(x^2 + y^2) \quad \text{subject to} \quad x^2 + y^2 \leq 1, \quad x \geq 0, \quad y \geq 0.
\]
Some examples

Problem Formulation: Regular points

\[
\max (x^2 + y^2) \\
x \geq 0
\]
Problem Formulation: Regular points

\[ \max(x^2 + y^2) \]

\[ x \geq 0 \]

\[ y \geq 0 \]
Problem Formulation: Regular points

$$\max(x^2 + y^2)$$

$$x \geq 0$$

$$y \geq 0$$

$$x + y \leq 1$$
Some examples

Problem Formulation: Regular points

\[
\max (x^2 + y^2)
\]
\[
x \geq 0
\]
\[
y \geq 0
\]
\[
x + y \leq 1
\]

In this example any feasible point is regular, since the gradients of the constraints are \((1, 0), (0, 1)\) and \((-1, -1)\).
Motivating Examples
Problem Formulation
Constrained Optimization
Non-smooth Optimization: Subgradient methods

Some examples

Problem Formulation: Non-regular points

\[
\begin{align*}
\max & \quad x^2 + y^2 \\
\text{s.t.} & \quad x \geq 0, y \geq 0, x^3 + 3 \cdot y^2 \leq 0
\end{align*}
\]

In this example the feasible solution \((0,0)\) is not a regular point since the gradients of the constraints are \((1,0), (0,1), (0,0)\).
Some examples

Problem Formulation: Non-regular points

\[
\begin{align*}
\text{max} & \quad (x^2 + y^2) \\
\text{subject to} & \quad x \geq 0, \quad y \geq 0, \quad x^3 + 3 \cdot y^2 \leq 0
\end{align*}
\]

In this example, the feasible solution \((0, 0)\) is not a regular point since the gradients of the constraints are \((1, 0)\), \((0, 1)\), and \((0, 0)\).
Some examples

Problem Formulation: Non-regular points

\[
\max(x^2 + y^2)
\]
Motivating Examples

Problem Formulation

Constrained Optimization

The generalized reduced gradient method

Non-smooth Optimization: Subgradient methods

Some examples

Problem Formulation: Non-regular points

\[
\text{max}(x^2 + y^2) \\
\quad x \geq 0
\]
Some examples

Problem Formulation: Non-regular points

\[
\max(x^2 + y^2)
\]
\[
x \geq 0
\]
\[
y \geq 0
\]
Some examples

Problem Formulation: Non-regular points

\[
\begin{align*}
\max (x^2 + y^2) \\
x &\geq 0 \\
y &\geq 0 \\
x^3 + 3 \cdot y^2 &\leq 0
\end{align*}
\]
Some examples

Problem Formulation: Non-regular points

\[
\begin{align*}
\max (x^2 + y^2) \\
x & \geq 0 \\
y & \geq 0 \\
x^3 + 3 \cdot y^2 & \leq 0
\end{align*}
\]

In this example the feasible solution \((0, 0)\) is not a regular point since the gradients of the constraints are \((1, 0)\), \((0, 1)\) and \((0, 0)\).
The conditions that will be presented in the upcoming three theorems are called Karush-Kuhn-Tucker (KKT) conditions after their inventors.

Theorem
Let \( x^* \) be a local minimizer of the non-linear problem, and assume that \( x^* \) is a regular point for the constraints of the problem. Then there exists \( \lambda_i, i \in E \cup I \) such that

\[
\nabla f(x^*) - \sum_{i \in E \cup I} \lambda_i \cdot \nabla g_i(x^*) = 0,
\]

\[
\lambda_i \geq 0, \quad i \in I,
\]

\[
\lambda_i \cdot g_i(x^*) = 0, \quad i \in I.
\]
The conditions that will be presented in the upcoming three theorems are called Karush-Kuhn-Tucker (KKT) conditions after their inventors.

**Theorem**

Let $x^*$ be a local minimizer of the non-linear problem, and assume that $x^*$ is a regular point for the constraints of the problem. Then there exists $\lambda_i, i \in E \cup I$ such that

$$\nabla f(x^*) - \sum_{i \in E \cup I} \lambda_i \cdot \nabla g_i(x^*) = 0,$$

$$\lambda_i \geq 0, i \in I,$$

$$\lambda_i \cdot g_i(x^*) = 0, i \in I.$$
First order necessary conditions

KKT conditions

The conditions that will be presented in the upcoming three theorems are called Karush-Kuhn-Tucker (KKT) conditions after their inventors.
First order necessary conditions

KKT conditions
The conditions that will be presented in the upcoming three theorems are called Karush-Kuhn-Tucker (KKT) conditions after their inventors.

Theorem
First order necessary conditions

KKT conditions
The conditions that will be presented in the upcoming three theorems are called Karush-Kuhn-Tucker (KKT) conditions after their inventors.

Theorem

Let $x^*$ be a local minimizer of the non-linear problem,
First order necessary conditions

**KKT conditions**

The conditions that will be presented in the upcoming three theorems are called Karush-Kuhn-Tucker (KKT) conditions after their inventors.

**Theorem**

Let \( x^* \) be a local minimizer of the non-linear problem, and assume that \( x^* \) is a regular point for the constraints of the problem.
The conditions that will be presented in the upcoming three theorems are called Karush-Kuhn-Tucker (KKT) conditions after their inventors.

**Theorem**

Let $x^*$ be a local minimizer of the non-linear problem, and assume that $x^*$ is a regular point for the constraints of the problem.

Then there exists $\lambda_i, i \in \mathcal{E} \cup \mathcal{I}$ such that

$$
\nabla f(x^*) - \sum_{i \in \mathcal{E} \cup \mathcal{I}} \lambda_i \cdot \nabla g_i(x^*) = 0,
$$

with $\lambda_i \geq 0$ for $i \in \mathcal{I}$. 

Here, $\mathcal{E}$ represents the set of equality constraints and $\mathcal{I}$ represents the set of inequality constraints.
KKT conditions

The conditions that will be presented in the upcoming three theorems are called Karush-Kuhn-Tucker (KKT) conditions after their inventors.

Theorem

Let $x^*$ be a local minimizer of the non-linear problem, and assume that $x^*$ is a regular point for the constraints of the problem.

Then there exists $\lambda_i, i \in \mathcal{E} \cup \mathcal{I}$ such that

$$
\nabla f(x^*) - \sum_{i \in \mathcal{E} \cup \mathcal{I}} \lambda_i \cdot \nabla g_i(x^*) = 0,
$$

$$
\lambda_i \geq 0, \ i \in \mathcal{I},
$$
First order necessary conditions

KKT conditions

The conditions that will be presented in the upcoming three theorems are called Karush-Kuhn-Tucker (KKT) conditions after their inventors.

Theorem

Let \( x^* \) be a local minimizer of the non-linear problem, and assume that \( x^* \) is a regular point for the constraints of the problem.

Then there exists \( \lambda_i, i \in \mathcal{E} \cup \mathcal{I} \) such that

\[
\nabla f(x^*) - \sum_{i \in \mathcal{E} \cup \mathcal{I}} \lambda_i \cdot \nabla g_i(x^*) = 0,
\]

\[
\lambda_i \geq 0, i \in \mathcal{I},
\]

\[
\lambda_i \cdot g_i(x^*) = 0, i \in \mathcal{I}.
\]
Motivating Examples

Problem Formulation

Constrained Optimization

The generalized reduced gradient method

Non-smooth Optimization: Subgradient methods

First order necessary conditions

Problem Formulation: Example

\[
\begin{align*}
\text{max} & \quad (x^2 + y^2) \\
\text{s.t.} & \quad x \geq 0 \\
& \quad y \geq 0 \\
& \quad x + y \leq 1
\end{align*}
\]

Lagrangian function

\[
L(x, y, \lambda) = - (x^2 + y^2) - \lambda_1 x - \lambda_2 y - \lambda_3 (1 - x - y)
\]
Motivating Examples
Problem Formulation
Constrained Optimization
Non-smooth Optimization: Subgradient methods

First order necessary conditions

Problem Formulation: Example

\[
\begin{align*}
\text{max} & \quad (x^2 + y^2) \\
\text{s.t.} & \quad x \geq 0, \quad y \geq 0, \quad x + y \leq 1
\end{align*}
\]

Lagrangian function
\[
L(x, y, \lambda) = - (x^2 + y^2) - \lambda_1 \cdot x - \lambda_2 \cdot y - \lambda_3 \cdot (1 - x - y)
\]
First order necessary conditions

Problem Formulation: Example

\[ \max (x^2 + y^2) \]
First order necessary conditions

Problem Formulation: Example

\[
\max (x^2 + y^2) \\
\text{subject to } x \geq 0
\]
Motivating Examples

Problem Formulation

Constrained Optimization

The generalized reduced gradient method

Non-smooth Optimization: Subgradient methods

First order necessary conditions

Problem Formulation: Example

\[
\text{max}(x^2 + y^2) \\
\quad x \geq 0 \\
\quad y \geq 0
\]
First order necessary conditions

Problem Formulation: Example

\[
\begin{align*}
\max (x^2 + y^2) \\
x \geq 0 \\
y \geq 0 \\
x + y \leq 1
\end{align*}
\]
First order necessary conditions

Problem Formulation: Example

\[
\text{max}(x^2 + y^2)
\]
\[
x \geq 0
\]
\[
y \geq 0
\]
\[
x + y \leq 1
\]

Lagrangian function
First order necessary conditions

**Problem Formulation: Example**

\[
\begin{align*}
\max & (x^2 + y^2) \\
\text{s.t.} & \quad x \geq 0 \\
& \quad y \geq 0 \\
& \quad x + y \leq 1
\end{align*}
\]

**Lagrangian function**

\[
L(x, y, \lambda) = -(x^2 + y^2) - \lambda_1 \cdot x - \lambda_2 \cdot y - \lambda_3 \cdot (1 - x - y).
\]
First order necessary conditions

If $\lambda_3 = 0$, it can be shown that $x = y = 0$. On the other hand, if $\lambda_3 \neq 0$, then $x + y = 1$, hence the problem is reduced to one-dimensional case, which by standard methods lead to points $(x = 0, y = 1)$, $(x = 1, y = 0)$, and $(x = y = 1/2)$. 
First order necessary conditions

The constraints

\begin{align*}
-2 \cdot x - \lambda_1 + \lambda_3 &= 0 \\
-2 \cdot y - \lambda_2 + \lambda_3 &= 0 \\
\lambda_1 &= 0, \quad \lambda_2 = 0, \quad \lambda_3 \geq 0 \\
\lambda_1 \cdot x &= 0 \\
\lambda_2 \cdot y &= 0 \\
\lambda_3 \cdot (1 - x - y) &= 0
\end{align*}

Stationary points

If \( \lambda_3 = 0 \), it can be shown that \( x = y = 0 \).

On the other hand, if \( \lambda_3 \neq 0 \), then \( x + y = 1 \), hence the problem is reduced to one dimensional case, which by standard methods lead to points \((x = 0, y = 1)\), \((x = 1, y = 0)\) and \((x = y = 1/2)\).
First order necessary conditions

The constraints

\[-2 \cdot x - \lambda_1 + \lambda_3 = 0\]
First order necessary conditions

The constraints

\[-2 \cdot x - \lambda_1 + \lambda_3 = 0\]
\[-2 \cdot y - \lambda_2 + \lambda_3 = 0\]
First order necessary conditions

The constraints

\[-2 \cdot x - \lambda_1 + \lambda_3 = 0\]
\[-2 \cdot y - \lambda_2 + \lambda_3 = 0\]
\[\lambda_1, \lambda_2, \lambda_3 \geq 0\]
First order necessary conditions

The constraints

\[-2 \cdot x - \lambda_1 + \lambda_3 = 0\]
\[-2 \cdot y - \lambda_2 + \lambda_3 = 0\]
\[\lambda_1, \lambda_2, \lambda_3 \geq 0\]
\[\lambda_1 \cdot x = 0\]
First order necessary conditions

The constraints

\[-2 \cdot x - \lambda_1 + \lambda_3 = 0\]
\[-2 \cdot y - \lambda_2 + \lambda_3 = 0\]
\[\lambda_1, \lambda_2, \lambda_3 \geq 0\]
\[\lambda_1 \cdot x = 0\]
\[\lambda_2 \cdot y = 0\]
First order necessary conditions

The constraints

\[-2 \cdot x - \lambda_1 + \lambda_3 = 0\]
\[-2 \cdot y - \lambda_2 + \lambda_3 = 0\]
\[\lambda_1, \lambda_2, \lambda_3 \geq 0\]
\[\lambda_1 \cdot x = 0\]
\[\lambda_2 \cdot y = 0\]
\[\lambda_3 \cdot (1 - x - y) = 0\]
First order necessary conditions

The constraints

\[-2 \cdot x - \lambda_1 + \lambda_3 = 0\]

\[-2 \cdot y - \lambda_2 + \lambda_3 = 0\]

\[\lambda_1, \lambda_2, \lambda_3 \geq 0\]

\[\lambda_1 \cdot x = 0\]

\[\lambda_2 \cdot y = 0\]

\[\lambda_3 \cdot (1 - x - y) = 0\]

Stationary points

If \(\lambda_3 = 0\), it can be shown that \(x = y = 0\).

On the other hand, if \(\lambda_3 \neq 0\), then \(x + y = 1\), hence the problem is reduced to one-dimensional case, which by standard methods lead to points \((x = 0, y = 1)\), \((x = 1, y = 0)\) and \((x = y = 1/2)\).
First order necessary conditions

The constraints

\[-2 \cdot x - \lambda_1 + \lambda_3 = 0\]
\[-2 \cdot y - \lambda_2 + \lambda_3 = 0\]
\[\lambda_1, \lambda_2, \lambda_3 \geq 0\]
\[\lambda_1 \cdot x = 0\]
\[\lambda_2 \cdot y = 0\]
\[\lambda_3 \cdot (1 - x - y) = 0\]

Stationary points

If \(\lambda_3 = 0\), it can be shown that \(x = y = 0\).
First order necessary conditions

The constraints

\[
-2 \cdot x - \lambda_1 + \lambda_3 = 0 \\
-2 \cdot y - \lambda_2 + \lambda_3 = 0 \\
\lambda_1, \lambda_2, \lambda_3 \geq 0 \\
\lambda_1 \cdot x = 0 \\
\lambda_2 \cdot y = 0 \\
\lambda_3 \cdot (1 - x - y) = 0
\]

Stationary points

If \( \lambda_3 = 0 \), it can be shown that \( x = y = 0 \).

On the other hand, if \( \lambda_3 \neq 0 \), then \( x + y = 1 \),
First order necessary conditions

The constraints

\[-2 \cdot x - \lambda_1 + \lambda_3 = 0\]
\[-2 \cdot y - \lambda_2 + \lambda_3 = 0\]
\[\lambda_1, \lambda_2, \lambda_3 \geq 0\]
\[\lambda_1 \cdot x = 0\]
\[\lambda_2 \cdot y = 0\]
\[\lambda_3 \cdot (1 - x - y) = 0\]

Stationary points

If \(\lambda_3 = 0\), it can be shown that \(x = y = 0\).

On the other hand, if \(\lambda_3 \neq 0\), then \(x + y = 1\), hence the problem is reduced to one dimensional case, which by standard methods lead to points \((x = 0, y = 1)\), \((x = 1, y = 0)\) and \((x = y = \frac{1}{2})\).
Second order necessary conditions
Second order necessary conditions

Theorem

Assume that \( f \) and \( g_i \), \( i \in E \cup I \) are all twice continuously differentiable functions. Let \( x^* \) be a local minimizer of the non-linear problem, and assume that \( x^* \) is a regular point for the constraints of the problem. Then there exists \( \lambda_i, i \in E \cup I \) satisfying the conditions of the previous theorem as well as the following condition:

\[
\nabla^2 f(x^*) - \sum_{i \in E \cup I} \lambda_i \cdot \nabla^2 g_i(x^*)
\]

Is positive semidefinite on the tangent subspace of active constraints at \( x^* \).

Active Constraint

Recall that a constraint is said to be active at a point \( x^* \), if it satisfies the constraint with equality.
Assume that $f$ and $g_i$, $i \in E \cup I$ are all twice continuously differentiable functions.
Second order necessary conditions

Theorem

Assume that $f$ and $g_i$, $i \in \mathcal{E} \cup \mathcal{I}$ are all twice continuously differentiable functions.

Let $x^*$ be a local minimizer of the non-linear problem,

$$\nabla^2 f(x^*) - \sum_{i \in \mathcal{E} \cup \mathcal{I}} \lambda_i \cdot \nabla^2 g_i(x^*)$$

is positive semidefinite on the tangent subspace of active constraints at $x^*$. 

Active Constraint

Recall that a constraint is said to be active at a point $x^*$, if it satisfies the constraint with equality.
Theorem

Assume that $f$ and $g_i$, $i \in E \cup I$ are all twice continuously differentiable functions.

Let $x^*$ be a local minimizer of the non-linear problem, and assume that $x^*$ is a regular point for the constraints of the problem.

Then there exists $\lambda_i$, $i \in E \cup I$ satisfying the conditions of the previous theorem as well as the following condition:

$$\nabla^2 f(x^*) - \sum_{i \in E \cup I} \lambda_i \cdot \nabla^2 g_i(x^*)$$

is positive semidefinite on the tangent subspace of active constraints at $x^*$. 

Active Constraint

Recall that a constraint is said to be active at a point $x^*$, if it satisfies the constraint with equality.
Theorem

Assume that $f$ and $g_i, \ i \in \mathcal{E} \cup \mathcal{I}$ are all twice continuously differentiable functions.

Let $x^*$ be a local minimizer of the non-linear problem, and assume that $x^*$ is a regular point for the constraints of the problem.

Then there exists $\lambda_i, \ i \in \mathcal{E} \cup \mathcal{I}$ satisfying the conditions of the previous theorem as well as the following condition:
Theorem

Assume that $f$ and $g_i$, $i \in \mathcal{E} \cup \mathcal{I}$ are all twice continuously differentiable functions.

Let $x^*$ be a local minimizer of the non-linear problem, and assume that $x^*$ is a regular point for the constraints of the problem.

Then there exists $\lambda_i$, $i \in \mathcal{E} \cup \mathcal{I}$ satisfying the conditions of the previous theorem as well as the following condition:

$$\nabla^2 f(x^*) - \sum_{i \in \mathcal{E} \cup \mathcal{I}} \lambda_i \cdot \nabla^2 g_i(x^*)$$

Is positive semidefinite on the tangent subspace of active constraints at $x^*$. 

Active Constraint
Recall that a constraint is said to be active at a point $x^*$, if it satisfies the constraint with equality.
Theorem

Assume that $f$ and $g_i$, $i \in \mathcal{E} \cup \mathcal{I}$ are all twice continuously differentiable functions.

Let $x^*$ be a local minimizer of the non-linear problem, and assume that $x^*$ is a regular point for the constraints of the problem.

Then there exists $\lambda_i$, $i \in \mathcal{E} \cup \mathcal{I}$ satisfying the conditions of the previous theorem as well as the following condition:

$$
\nabla^2 f(x^*) - \sum_{i \in \mathcal{E} \cup \mathcal{I}} \lambda_i \cdot \nabla^2 g_i(x^*)
$$

Is positive semidefinite on the tangent subspace of active constraints at $x^*$. 

Active Constraint

Recall that a constraint is said to be active at a point $x^*$, if it satisfies the constraint with equality.
Second order necessary conditions

**Theorem**

Assume that $f$ and $g_i$, $i \in \mathcal{E} \cup \mathcal{I}$ are all twice continuously differentiable functions.

Let $x^*$ be a local minimizer of the non-linear problem, and assume that $x^*$ is a regular point for the constraints of the problem.

Then there exists $\lambda_i$, $i \in \mathcal{E} \cup \mathcal{I}$ satisfying the conditions of the previous theorem as well as the following condition:

$$\nabla^2 f(x^*) - \sum_{i \in \mathcal{E} \cup \mathcal{I}} \lambda_i \cdot \nabla^2 g_i(x^*)$$

Is positive semidefinite on the tangent subspace of active constraints at $x^*$.

**Active Constraint**

Recall that a constraint is said to be active at a point $x^*$, if it satisfies the constraint with equality.
Second order necessary conditions

Let $A(x^*)$ denote the Jacobian of the active constraints at $x^*$, and let $N(x^*)$ be a null-space basis for $A(x^*)$. Then, the last condition of the previous theorem, is equivalent to the following condition:

$$N^T(x^*) \cdot (\nabla^2 f(x^*) - \sum_{i \in E \cup I} \lambda_i \cdot \nabla^2 g_i(x^*)) \cdot N(x^*)$$

is positive semidefinite.
Let $A(x^*)$ denote the Jacobian of the active constraints at $x^*$, and let $N(x^*)$ be a null-space basis for $A(x^*)$. Then, the last condition of the previous theorem, is equivalent to the following condition:

$$N^T(x^*) \cdot (\nabla^2 f(x^*) - \sum_{i \in E \cup I} \lambda_i \cdot \nabla^2 g_i(x^*)) \cdot N(x^*)$$

is positive semidefinite.
Restatement of the last condition

Let $A(x^*)$ denote the Jacobian of the active constraints at $x^*$,
Restatement of the last condition

Let \( \mathbf{A}(\mathbf{x}^*) \) denote the Jacobian of the active constraints at \( \mathbf{x}^* \), and let \( \mathbf{N}(\mathbf{x}^*) \) be a null-space basis for \( \mathbf{A}(\mathbf{x}^*) \).
Restatement of the last condition

Let $A(x^*)$ denote the Jacobian of the active constraints at $x^*$, and let $N(x^*)$ be a null-space basis for $A(x^*)$.

Then, the last condition of the previous theorem, is equivalent to the following condition:
Motivating Examples
Problem Formulation
Constrained Optimization
The generalized reduced gradient method
Non-smooth Optimization: Subgradient methods

Second order necessary conditions

Restatement of the last condition

Let $A(x^*)$ denote the Jacobian of the active constraints at $x^*$, and let $N(x^*)$ be a null-space basis for $A(x^*)$.

Then, the last condition of the previous theorem, is equivalent to the following condition:

$$N^T(x^*) \cdot \left( \nabla^2 f(x^*) - \sum_{i \in \mathcal{E} \cup \mathcal{I}} \lambda_i \cdot \nabla^2 g_i(x^*) \right) \cdot N(x^*)$$

Is positive semidefinite.
Second order sufficient conditions
Theorem

Assume that $f$ and $g_i$, $i \in E \cup I$, are all twice continuously differentiable functions. Let $x^*$ be a feasible solution of the non-linear problem, and assume that it is a regular point for the constraints of the problem. Let $A(x^*)$ denote the Jacobian of the active constraints at $x^*$, and let $N(x^*)$ be a null-space basis for $A(x^*)$.

If there exists $\lambda_i$, $i \in E \cup I$ satisfying the conditions of the first order necessary theorem as well as the following condition:

$g_i(x^*) = 0$, $i \in I$ implies $\lambda_i > 0$,

and $N^T(x^*) \cdot \left( \nabla^2 f(x^*) - \sum_{i \in E \cup I} \lambda_i \cdot \nabla^2 g_i(x^*) \right) \cdot N(x^*)$ is positive semidefinite,

then $x^*$ is local minimizer of the non-linear program.
Theorem

Assume that $f$ and $g_i$, $i \in \mathcal{E} \cup \mathcal{I}$ are all twice continuously differentiable functions.
Theorem

Assume that $f$ and $g_i, i \in \mathcal{E} \cup \mathcal{I}$ are all twice continuously differentiable functions.

Let $x^*$ be a feasible solution of the non-linear problem,
Second order sufficient conditions

**Theorem**

Assume that \( f \) and \( g_i, i \in \mathcal{E} \cup \mathcal{I} \) are all twice continuously differentiable functions.

Let \( x^* \) be a feasible solution of the non-linear problem, and assume that it is a regular point for the constraints of the problem.
Theorem

Assume that $f$ and $g_i$, $i \in \mathcal{E} \cup \mathcal{I}$ are all twice continuously differentiable functions.  

Let $x^*$ be a feasible solution of the non-linear problem, and assume that it is a regular point for the constraints of the problem.

Let $A(x^*)$ denote the Jacobian of the active constraints at $x^*$, and let $N(x^*)$ be a null-space basis for $A(x^*)$. 

If there exists $\lambda_i$, $i \in \mathcal{E} \cup \mathcal{I}$ satisfying the conditions of the first order necessary theorem as well as the following condition:

$$g_i(x^*) = 0, i \in \mathcal{I} \Rightarrow \lambda_i > 0,$$

and

$$N^T(x^*) \cdot (\nabla^2 f(x^*) - \sum_{i \in \mathcal{E} \cup \mathcal{I}} \lambda_i \cdot \nabla^2 g_i(x^*)) \cdot N(x^*)$$

is positive semidefinite,

then $x^*$ is local minimizer of the non-linear program.
Assume that $f$ and $g_i, i \in \mathcal{E} \cup \mathcal{I}$ are all twice continuously differentiable functions.

Let $x^*$ be a feasible solution of the non-linear problem, and assume that it is a regular point for the constraints of the problem.

Let $A(x^*)$ denote the Jacobian of the active constraints at $x^*$, and let $N(x^*)$ be a null-space basis for $A(x^*)$.

If there exists $\lambda_i, i \in \mathcal{E} \cup \mathcal{I}$ satisfying the conditions of the first order necessary theorem as well as the following condition:
Theorem

Assume that $f$ and $g_i$, $i \in E \cup I$ are all twice continuously differentiable functions.

Let $x^*$ be a feasible solution of the non-linear problem, and assume that it is a regular point for the constraints of the problem.

Let $A(x^*)$ denote the Jacobian of the active constraints at $x^*$, and let $N(x^*)$ be a null-space basis for $A(x^*)$.

If there exists $\lambda_i$, $i \in E \cup I$ satisfying the conditions of the first order necessary theorem as well as the following condition:

$$g_i(x^*) = 0, \ i \in I \text{ implies } \lambda_i > 0,$$

then $x^*$ is local minimizer of the non-linear program.
Motivating Examples
Problem Formulation
Constrained Optimization
The generalized reduced gradient method
Non-smooth Optimization: Subgradient methods

Second order sufficient conditions

Theorem

Assume that $f$ and $g_i$, $i \in \mathcal{E} \cup \mathcal{I}$ are all twice continuously differentiable functions.

Let $x^*$ be a feasible solution of the non-linear problem, and assume that it is a regular point for the constraints of the problem.

Let $A(x^*)$ denote the Jacobian of the active constraints at $x^*$, and let $N(x^*)$ be a null-space basis for $A(x^*)$.

If there exists $\lambda_i$, $i \in \mathcal{E} \cup \mathcal{I}$ satisfying the conditions of the first order necessary theorem as well as the following condition:

$$g_i(x^*) = 0, i \in \mathcal{I} \text{ implies } \lambda_i > 0,$$

and

$$N^T(x^*) \cdot (\nabla^2 f(x^*) - \sum_{i \in \mathcal{E} \cup \mathcal{I}} \lambda_i \cdot \nabla^2 g_i(x^*)) \cdot N(x^*)$$

is positive semidefinite,
Second order sufficient conditions

**Theorem**

Assume that \( f \) and \( g_i, i \in \mathcal{E} \cup \mathcal{I} \) are all twice continuously differentiable functions.

Let \( \mathbf{x}^* \) be a feasible solution of the non-linear problem, and assume that it is a regular point for the constraints of the problem.

Let \( \mathbf{A}(\mathbf{x}^*) \) denote the Jacobian of the active constraints at \( \mathbf{x}^* \), and let \( \mathbf{N}(\mathbf{x}^*) \) be a null-space basis for \( \mathbf{A}(\mathbf{x}^*) \).

If there exists \( \lambda_i, i \in \mathcal{E} \cup \mathcal{I} \) satisfying the conditions of the first order necessary theorem as well as the following condition:

\[
g_i(\mathbf{x}^*) = 0, i \in \mathcal{I} \text{ implies } \lambda_i > 0,
\]

and

\[
\mathbf{N}^T(\mathbf{x}^*) \cdot (\nabla^2 f(\mathbf{x}^*) - \sum_{i \in \mathcal{E} \cup \mathcal{I}} \lambda_i \cdot \nabla^2 g_i(\mathbf{x}^*)) \cdot \mathbf{N}(\mathbf{x}^*)
\]

is positive semidefinite, then \( \mathbf{x}^* \) is local minimizer of the non-linear program.
Constrained non-linear programs

Below, we introduce an approach for solving non-linear programs. It relies on the method of steepest decent method. The idea is to reduce the number of variables using the constraints and to solve this reduced and unconstrained problem using the steepest decent method.
An approach for solving non-linear programs relies on the method of steepest descent. The idea is to reduce the number of variables using the constraints and then solve the reduced and unconstrained problem using the steepest descent method.
An approach

Below, we introduce an approach for solving non-linear programs.
An approach

Below, we introduce an approach for solving non-linear programs.

It relies on the method of steepest decent method.
An approach

Below, we introduce an approach for solving non-linear programs.

It relies on the method of steepest decent method.

The idea is to reduce the number of variables using the constraints and to solve this reduced and unconstrained problem using the steepest decent method.
Motivating Examples

Problem Formulation

Constrained Optimization

The generalized reduced gradient method

Non-smooth Optimization: Subgradient methods

Linear equality constraints
Motivating Examples
Problem Formulation
Constrained Optimization

The generalized reduced gradient method
Non-smooth Optimization: Subgradient methods

Linear equality constraints

Problem Formulation
Problem Formulation

\[ \min f(x) := x_1^3 + x_2 + x_3^3 + x_4 \]
Linear equality constraints

Problem Formulation

\[ \min f(x) := x_1^3 + x_2 + x_3^3 + x_4 \]

\[ g_1(x) := x_1 + x_2 + 4 \cdot x_3 + 4 \cdot x_4 - 4 = 0, \]
Problem Formulation

\[ \min f(x) := x_1^3 + x_2 + x_3^3 + x_4 \]
\[ g_1(x) := x_1 + x_2 + 4 \cdot x_3 + 4 \cdot x_4 - 4 = 0, \]
\[ g_2(x) := -x_1 + x_2 + 2 \cdot x_3 - 2 \cdot x_4 + 2 = 0, \]
Linear equality constraints

- Solving the linear equations:
  \[ x_2 = 3 \cdot x_1 + 8 \cdot x_4 - 8, \]
  \[ x_3 = -x_1 - 3 \cdot x_4 + 3. \]
Linear equality constraints

Solving the linear equations

\[ x_2 = 3 \cdot x_1 + 8 \cdot x_4 - 8, \]
\[ x_3 = -x_1 - 3 \cdot x_4 + 3, \]
Linear equality constraints

Solving the linear equations

\[ x_2 = 3 \cdot x_1 + 8 \cdot x_4 - 8, \]
Motivating Examples
Problem Formulation
Constrained Optimization

The generalized reduced gradient method
Non-smooth Optimization: Subgradient methods

Linear equality constraints

Solving the linear equations

\[ x_2 = 3 \cdot x_1 + 8 \cdot x_4 - 8, \]
\[ x_3 = -x_1 - 3 \cdot x_4 + 3, \]
Motivating Examples

Problem Formulation

Constrained Optimization

The generalized reduced gradient method

Non-smooth Optimization: Subgradient methods

Linear equality constraints
Solving an unconstrained non-linear program
Motivating Examples

Problem Formulation

Constrained Optimization

The generalized reduced gradient method

Non-smooth Optimization: Subgradient methods

Linear equality constraints

Solving an unconstrained non-linear program

The above equations lead to the following unconstrained non-linear program:
Solving an unconstrained non-linear program

The above equations lead to the following unconstrained non-linear program:

$$\min f(x_1, x_4) := x_1^3 + (3 \cdot x_1 + 8 \cdot x_4 - 8) + (-x_1 - 3 \cdot x_4 + 3)^3 + x_4.$$
Problem Formulation

Constrained Optimization

The generalized reduced gradient method

Non-smooth Optimization: Subgradient methods

Non-linear equality constraints

Consider the following non-linear program similar to the previous one:

\[
\begin{align*}
\text{min} \quad & f(x) = x_1^2 + x_2^2 + x_3^3 + x_4^4 \\
\text{subject to} \quad & g_1(x) = x_1^2 + x_2^2 + 4 \cdot x_3 + 4 \cdot x_4 - 4 = 0 \\
& g_2(x) = -x_1 + x_2 + 2 \cdot x_3 - 2 \cdot x_4^2 + 2 = 0 
\end{align*}
\]

In this example, the constraints are non-linear.

Taylor series

Approximate the constraint functions by their Taylor series:

\[
g(x) \approx g(\bar{x}) + \nabla g(\bar{x}) \cdot (x - \bar{x})^T,
\]

where \( \bar{x} \) is the current point.

Mkrtchyan

Optimization Methods in Finance
Consider the following non-linear program similar to the previous one:

\[
\begin{align*}
\min & \quad f(x) := x_1^2 + x_2^2 + x_3^2 + x_4^2 \\
\text{s.t.} & \quad g_1(x) := x_1^2 + x_2^2 + 4 \cdot x_3 + 4 \cdot x_4 - 4 = 0, \\
& \quad g_2(x) := -x_1 + x_2 + 2 \cdot x_3 - 2 \cdot x_2^4 + 2 = 0.
\end{align*}
\]

In this example, the constraints are non-linear. Taylor series can approximate the constraint functions by their Taylor series:

\[
g(x) \approx g(\bar{x}) + \nabla g(\bar{x}) \cdot (x - \bar{x})^T,
\]

where \(\bar{x}\) is the current point.
Solving an unconstrained non-linear program

Consider the following non-linear program similar to the previous one:

\[
\begin{align*}
\min & \quad f(x) := x_1^2 + x_2^2 + x_3^2 + x_4^4 \\
\text{s.t.} & \quad g_1(x) := x_1^2 + x_2^2 + 4x_3 + 4x_4 - 4 = 0 \\
& \quad g_2(x) := -x_1 + x_2 + 2x_3 - 2x_4 = 0
\end{align*}
\]
Non-linear equality constraints

Solving an unconstrained non-linear program

Consider the following non-linear program similar to the previous one:

$$\min f(x) := x_1^2 + x_2 + x_3^2 + x_4$$
Non-linear equality constraints

Solving an unconstrained non-linear program

Consider the following non-linear program similar to the previous one:

\[
\min f(x) := x_1^2 + x_2 + x_3^2 + x_4
\]
\[
g_1(x) := x_1^2 + x_2 + 4 \cdot x_3 + 4 \cdot x_4 - 4 = 0,
\]
Motivating Examples
Problem Formulation
Constrained Optimization
The generalized reduced gradient method
Non-smooth Optimization: Subgradient methods

Non-linear equality constraints

Solving an unconstrained non-linear program

Consider the following non-linear program similar to the previous one:

\[
\begin{align*}
\min f(x) &:= x_1^2 + x_2 + x_3^2 + x_4 \\
g_1(x) &:= x_1^2 + x_2 + 4 \cdot x_3 + 4 \cdot x_4 - 4 = 0, \\
g_2(x) &:= -x_1 + x_2 + 2 \cdot x_3 - 2 \cdot x_4^2 + 2 = 0,
\end{align*}
\]
Non-linear equality constraints

Solving an unconstrained non-linear program

Consider the following non-linear program similar to the previous one:

$$\min f(x) := x_1^2 + x_2 + x_3^2 + x_4$$

$$g_1(x) := x_1^2 + x_2 + 4 \cdot x_3 + 4 \cdot x_4 - 4 = 0,$$

$$g_2(x) := -x_1 + x_2 + 2 \cdot x_3 - 2 \cdot x_4^2 + 2 = 0,$$

In this example, the constraints are non-linear.
Solving an unconstrained non-linear program

Consider the following non-linear program similar to the previous one:

\[
\begin{align*}
\min f(x) := x_1^2 + x_2 + x_3^2 + x_4 \\
g_1(x) := x_1^2 + x_2 + 4 \cdot x_3 + 4 \cdot x_4 - 4 = 0, \\
g_2(x) := -x_1 + x_2 + 2 \cdot x_3 - 2 \cdot x_4^2 + 2 = 0,
\end{align*}
\]

In this example, the constraints are non-linear.

Taylor series
Solving an unconstrained non-linear program

Consider the following non-linear program similar to the previous one:

\[
\min f(x) := x_1^2 + x_2 + x_3^2 + x_4 \\
g_1(x) := x_1^2 + x_2 + 4 \cdot x_3 + 4 \cdot x_4 - 4 = 0, \\
g_2(x) := -x_1 + x_2 + 2 \cdot x_3 - 2 \cdot x_4^2 + 2 = 0,
\]

In this example, the constraints are non-linear.

Taylor series

Approximate the constraint functions by their Taylor series:
Solving an unconstrained non-linear program

Consider the following non-linear program similar to the previous one:

\[ \min f(x) := x_1^2 + x_2 + x_3^2 + x_4 \]

\[ g_1(x) := x_1^2 + x_2 + 4 \cdot x_3 + 4 \cdot x_4 - 4 = 0, \]

\[ g_2(x) := -x_1 + x_2 + 2 \cdot x_3 - 2 \cdot x_4^2 + 2 = 0, \]

In this example, the constraints are non-linear.

Taylor series

Approximate the constraint functions by their Taylor series:

\[ g(x) \approx g(\bar{x}) + \nabla g(\bar{x}) \cdot (x - \bar{x})^T, \]
Solving an unconstrained non-linear program

Consider the following non-linear program similar to the previous one:

\[
\min f(x) := x_1^2 + x_2 + x_3^2 + x_4
\]

\[
g_1(x) := x_1^2 + x_2 + 4 \cdot x_3 + 4 \cdot x_4 - 4 = 0,
\]

\[
g_2(x) := -x_1 + x_2 + 2 \cdot x_3 - 2 \cdot x_4^2 + 2 = 0,
\]

In this example, the constraints are non-linear.

Taylor series

Approximate the constraint functions by their Taylor series:

\[
g(x) \approx g(\bar{x}) + \nabla g(\bar{x}) \cdot (x - \bar{x})^T, \text{ where } \bar{x} \text{ is the current point.}
\]
Non-linear equality constraints
Non-linear equality constraints

Our constraints become

\[ g_1(x) \approx 2 \cdot \bar{x}_1 \cdot x_1 + x_2 + 4 \cdot x_4 - (\bar{x}_2 + 4) = 0, \]

\[ g_2(x) \approx -x_1 + x_2 + 2 \cdot x_3 - 4 \cdot \bar{x}_4 \cdot x_4 + (\bar{x}_2 + 2) = 0, \]
Non-linear equality constraints

Our constraints become

\[ g_1(x) \approx 2 \cdot \bar{x}_1 \cdot x_1 + x_2 + 4 \cdot x_3 + 4 \cdot x_4 - (\bar{x}_1^2 + 4) = 0, \]
Our constraints become

\[ g_1(x) \approx 2 \cdot \bar{x}_1 \cdot x_1 + x_2 + 4 \cdot x_3 + 4 \cdot x_4 - (\bar{x}_1^2 + 4) = 0, \]
\[ g_2(x) \approx -x_1 + x_2 + 2 \cdot x_3 - 4 \cdot \bar{x}_4 \cdot x_4 + (\bar{x}_4^2 + 2) = 0, \]
The idea of Generalized Reduced Gradient Method (GRG) is to solve a sequence of sub-problems, each of which uses a linear approximation of the constraints. In each iteration of the algorithm, the constraint linearization is recalculated at the point found from the previous iteration. Though the constraints are only approximated, the subproblems yield points that are progressively closer to the optimal point.
The idea of Generalized Reduced Gradient Method (GRG) is to solve a sequence of sub-problems, each of which uses a linear approximation of the constraints. In each iteration of the algorithm, the constraint linearization is recalculated at the point found from the previous iteration. Though the constraints are only approximated, the subproblems yield points that are progressively closer to the optimal point.
The idea of Generalized Reduced Gradient Method (GRG) is to solve a sequence of sub-problems, each of which uses a linear approximation of the constraints.
The general idea

The idea of Generalized Reduced Gradient Method (GRG) is to solve a sequence of sub-problems, each of which uses a linear approximation of the constraints.

In each iteration of the algorithm, the constraint linearization is recalculated at the point found from the previous iteration.
The general idea

The idea of Generalized Reduced Gradient Method (GRG) is to solve a sequence of sub-problems, each of which uses a linear approximation of the constraints.

In each iteration of the algorithm, the constraint linearization is recalculated at the point found from the previous iteration.

Though the constraints are only approximated, the subproblems yield points that are progressively closer to the optimal point.
Motivating Examples
Non-smooth Optimization: Subgradient methods

Non-linear equality constraints

Our example: Starting point
Let us start with
\[ x_0 = (0, -8, 3, 0) \]

This point satisfies our constraints.

It is quite possible to start with an infeasible point.

Our example: the resulting program
Using the approximation formulas derived earlier, we get:

\[
\begin{align*}
\min & \quad f(x) := x_1^2 + x_2^2 + x_3^2 + x_4^2 \\
\text{s.t.} & \quad g_1(x) := x_1^2 + 4 \cdot x_3 + 4 \cdot x_4 - 4 = 0 \\
& \quad g_2(x) := -x_1 + x_2 + 2 \cdot x_3 + 2 = 0
\end{align*}
\]
Our example: Starting point

Let us start with \( x_0 = (0, -8, 3, 0) \).
This point satisfies our constraints.
It is quite possible to start with an infeasible point.

Our example: the resulting program
Using the approximation formulas derived earlier, we get:

\[
\begin{align*}
\text{min} & \quad f(x) := x_1^2 + x_2^2 + x_3^2 + x_4^4 \\
\text{subject to} & \quad g_1(x) := x_2^2 + 4 \cdot x_3 + 4 \cdot x_4 = 0, \\
& \quad g_2(x) := -x_1 + x_2 + 2 \cdot x_3 + 2 = 0.
\end{align*}
\]
Non-linear equality constraints

Our example: Starting point

Let us start with $x^0 = (0, -8, 3, 0)$. 
Motivating Examples
Problem Formulation
Constrained Optimization
The generalized reduced gradient method
Non-smooth Optimization: Subgradient methods

Non-linear equality constraints

Our example: Starting point

Let us start with $\mathbf{x}^0 = (0, -8, 3, 0)$.

This point satisfies our constraints.
Our example: Starting point

Let us start with $x^0 = (0, -8, 3, 0)$.

This point satisfies our constraints.

It is quite possible to start with an infeasible point.
Motivating Examples
Problem Formulation
Constrained Optimization
The generalized reduced gradient method
Non-smooth Optimization: Subgradient methods

Non-linear equality constraints

Our example: Starting point

Let us start with $x^0 = (0, -8, 3, 0)$.

This point satisfies our constraints.

It is quite possible to start with an infeasible point.

Our example: the resulting program
Non-linear equality constraints

Our example: Starting point

Let us start with \( \mathbf{x}^0 = (0, -8, 3, 0) \).

This point satisfies our constraints.

It is quite possible to start with an infeasible point.

Our example: the resulting program

Using the approximation formulas derived earlier, we get:
Motivating Examples
Problem Formulation
Constrained Optimization
The generalized reduced gradient method
Non-smooth Optimization: Subgradient methods

Non-linear equality constraints

Our example: Starting point
Let us start with $x^0 = (0, -8, 3, 0)$.

This point satisfies our constraints.

It is quite possible to start with an infeasible point.

Our example: the resulting program
Using the approximation formulas derived earlier, we get:

$$\min f(x) := x_1^2 + x_2 + x_3^2 + x_4$$
Our example: Starting point

Let us start with $x^0 = (0, -8, 3, 0)$.  

This point satisfies our constraints.  

It is quite possible to start with an infeasible point.

Our example: the resulting program

Using the approximation formulas derived earlier, we get:

$$
\min f(x) := x_1^2 + x_2 + x_3^2 + x_4 \\
g_1(x) := x_2 + 4 \cdot x_3 + 4 \cdot x_4 - 4 = 0,
$$
Motivating Examples
Problem Formulation
Constrained Optimization
The generalized reduced gradient method
Non-smooth Optimization: Subgradient methods

Non-linear equality constraints

Our example: Starting point
Let us start with $x^0 = (0, -8, 3, 0)$.

This point satisfies our constraints.

It is quite possible to start with an infeasible point.

Our example: the resulting program
Using the approximation formulas derived earlier, we get:

$$
\min f(x) := x_1^2 + x_2 + x_3^2 + x_4
$$

$$
\begin{align*}
g_1(x) &:= x_2 + 4 \cdot x_3 + 4 \cdot x_4 - 4 = 0, \\
g_2(x) &:= -x_1 + x_2 + 2 \cdot x_3 + 2 = 0,
\end{align*}
$$
Non-linear equality constraints

Solving the equality constraints with respect to $x_2$ and $x_3$, we get:

$$x_2 = 2x_1 + 4x_4 - 8,$$

$$x_3 = -1.5x_1 - 2x_4 + 3.$$
Solving the equality constraints

\[ x_2 = 2 \cdot x_1 + 4 \cdot x_4 - 8, \]
\[ x_3 = -\frac{1}{2} \cdot x_1 - 2 \cdot x_4 + 3. \]
Solving the equality constraints

Solving with respect to $x_2$ and $x_3$, we get:

\[ x_2 = 2 \cdot x_1 + 4 \cdot x_4 - 8, \]
\[ x_3 = -\frac{1}{2} \cdot x_1 - 2 \cdot x_4 + 3. \]
Motivating Examples

Problem Formulation

Constrained Optimization

The generalized reduced gradient method

Non-smooth Optimization: Subgradient methods

Non-linear equality constraints

Solving the equality constraints

Solving with respect to $x_2$ and $x_3$, we get:

$$x_2 = 2 \cdot x_1 + 4 \cdot x_4 - 8,$$
Non-linear equality constraints

Solving the equality constraints

Solving with respect to $x_2$ and $x_3$, we get:

\[
\begin{align*}
  x_2 &= 2 \cdot x_1 + 4 \cdot x_4 - 8, \\
  x_3 &= -\frac{1}{2} \cdot x_1 - 2 \cdot x_4 + 3,
\end{align*}
\]
Non-linear equality constraints

Minimizing the objective function $f(x) = x_2^1 + (2 \cdot x_1 + 4 \cdot x_4 - 8) + (-0.5 \cdot x_1 - 2 \cdot x_4 + 3)^2 + x_4^4$.

Solving this unconstrained minimization problem, we get $x_1 = -0.375$ and $x_4 = 0.96875$.

Substituting in equations for $x_2$ and $x_3$ gives $x_2 = -4.875$ and $x_3 = 1.25$.

Thus, the iteration of GRG method is $x_1 = (-0.375, -4.875, 1.25, 0.96875)$. 
Non-linear equality constraints

The resulting unconstrained program

\[
\begin{align*}
\min & \quad f(x_1, x_4) := x_2 + (2 \cdot x_1 + 4 \cdot x_4 - 8) + (-12 \cdot x_1 - 2 \cdot x_4 + 3)^2 + x_4. \\
\end{align*}
\]

Solving this unconstrained minimization problem, we get \( x_1 = -0.375 \) and \( x_4 = 0.96875 \).

Substituting in equations for \( x_2 \) and \( x_3 \) gives \( x_2 = -4.875 \) and \( x_3 = 1.25 \).

Thus, the iteration of GRG method is \( x_1 = (-0.375, -4.875, 1.25, 0.96875) \).
The resulting unconstrained program

\[
\min f(x_1, x_4) := x_1^2 + (2 \cdot x_1 + 4 \cdot x_4 - 8) + \left(-\frac{1}{2} \cdot x_1 - 2 \cdot x_4 + 3\right)^2 + x_4.
\]
The resulting unconstrained program

$$\min f(x_1, x_4) := x_1^2 + (2 \cdot x_1 + 4 \cdot x_4 - 8) + (-\frac{1}{2} \cdot x_1 - 2 \cdot x_4 + 3)^2 + x_4.$$
The resulting unconstrained program

\[ \min f(x_1, x_4) := x_1^2 + (2 \cdot x_1 + 4 \cdot x_4 - 8) + (-\frac{1}{2} \cdot x_1 - 2 \cdot x_4 + 3)^2 + x_4. \]

The next point

Solving this unconstrained minimization problem, we get \( x_1 = -0.375 \) and \( x_4 = 0.96875 \).
The resulting unconstrained program

\[ \min f(x_1, x_4) := x_1^2 + (2 \cdot x_1 + 4 \cdot x_4 - 8) + (-\frac{1}{2} \cdot x_1 - 2 \cdot x_4 + 3)^2 + x_4. \]

The next point

Solving this unconstrained minimization problem, we get \( x_1 = -0.375 \) and \( x_4 = 0.96875 \).

Substituting in equations for \( x_2 \) and \( x_3 \) gives \( x_2 = -4.875 \) and \( x_3 = 1.25 \).
The resulting unconstrained program

\[ \min f(x_1, x_4) := x_1^2 + (2 \cdot x_1 + 4 \cdot x_4 - 8) + (-\frac{1}{2} \cdot x_1 - 2 \cdot x_4 + 3)^2 + x_4. \]

The next point

Solving this unconstrained minimization problem, we get \( x_1 = -0.375 \) and \( x_4 = 0.96875 \).

Substituting in equations for \( x_2 \) and \( x_3 \) gives \( x_2 = -4.875 \) and \( x_3 = 1.25 \).

Thus, the iteration of GRG method is \( x^1 = (-0.375, -4.875, 1.25, 0.96875) \).
Non-linear equality constraints

Continuing the process

To continue the solution process, we would re-linearize the constraint functions at the new point,

Use the resulting system of linear equations to express two of the variables in terms of the others,

Substitute into the objective to get the new reduced problem,

Solve the reduced problem for $x_2$, and so forth.

Stopping criterion

The stopping criterion is $\|x_{k+1} - x_k\| < T$, where $T$ is a small constant.
Continuing the process

To continue the solution process, we would re-linearize the constraint functions at the new point, use the resulting system of linear equations to express two of the variables in terms of the others, substitute into the objective to get the new reduced problem, solve the reduced problem for $x_2$, and so forth.

Stopping criterion

The stopping criterion is $\|x_{k+1} - x_k\| < T$, where $T$ is a small constant.
Continuing the process

To continue the solution process, we would re-linearize the constraint functions at the new point,

Stopping criterion

The stopping criterion is \( \| x^{k+1} - x^k \| < T \), where \( T \) is a small constant.
Non-linear equality constraints

Continuing the process

To continue the solution process, we would re-linearize the constraint functions at the new point,

Use the resulting system of linear equations to express two of the variables in terms of the others,
Non-linear equality constraints

Continuing the process

To continue the solution process, we would re-linearize the constraint functions at the new point,

Use the resulting system of linear equations to express two of the variables in terms of the others,

Substitute into the objective to get the new reduced problem,
Continuing the process

To continue the solution process, we would re-linearize the constraint functions at the new point,

Use the resulting system of linear equations to express two of the variables in terms of the others,

Substitute into the objective to get the new reduced problem,

Solve the reduced problem for $x^2$, and so forth.
Non-linear equality constraints

Continuing the process

To continue the solution process, we would re-linearize the constraint functions at the new point,

Use the resulting system of linear equations to express two of the variables in terms of the others,

Substitute into the objective to get the new reduced problem,

Solve the reduced problem for $x^2$, and so forth.

Stopping criterion
Non-linear equality constraints

Continuing the process
To continue the solution process, we would re-linearize the constraint functions at the new point,

Use the resulting system of linear equations to express two of the variables in terms of the others,

Substitute into the objective to get the new reduced problem,

Solve the reduced problem for $x^2$, and so forth.

Stopping criterion
The stopping criterion is $||x^{k+1} - x^k|| < T$, where $T$ is a small constant.
Non-linear equality constraints

Our example

For example, if we take $T = 0.0025$ in the above example, we get $x_k = (-0.498, -4.823, 1.534, 0.610)$ and $f(x_k) = -1.612$.

The optimum solution is $x^* = (-0.500, -4.825, 1.534, 0.610)$ and has an objective value of $-1.612$. 

Motivating Examples

Problem Formulation

Constrained Optimization

The generalized reduced gradient method

Non-smooth Optimization: Subgradient methods
Motivating Examples
Problem Formulation
Constrained Optimization
The generalized reduced gradient method
Non-smooth Optimization: Subgradient methods

Non-linear equality constraints

Our example

For example if we take, $T = 0.0025$ in the above example, we get $x_k = (-0.498, -4, 8.23, 1.534, 0.610)$ and $f(x_k) = -1.612$.

The optimum solution is $x^* = (-0.500, -4.825, 1.534, 0.610)$ and has an objective value of $-1.612$. 
For example if we take, $T = 0.0025$ in the above example, we get $x^k = (-0.498, -4.823, 1.534, 0.610)$ and $f(x^k) = -1.612$. 
For example if we take, \( T = 0.0025 \) in the above example, we get 
\[ \mathbf{x}^k = (-0.498, -4, 823, 1.534, 0.610) \]  
and 
\[ f(\mathbf{x}^k) = -1.612. \]

The optimum solution is 
\[ \mathbf{x}^* = (-0.500, -4.825, 1.534, 0.610) \]  
and has an objective value of \(-1.612\).
Non-linear equality constraints
Non-linear equality constraints

Remark

During the iteration, the values of $f(x_k)$ can sometimes be smaller than the minimum value. How this is possible?

The reason is that the points $x_k$ computed by GRG are usually not feasible. They are only feasible for linear approximations of these constraints.
Remark

During the iteration, the values of $f(x^k)$ can sometimes be smaller than the minimum value.
Remark

During the iteration, the values of $f(x^k)$ can sometimes be smaller than the minimum value.

How this is possible?
Non-linear equality constraints

Remark

During the iteration, the values of $f(x^k)$ can sometimes be smaller than the minimum value.

How this is possible?

The reason is that the points $x^k$ computed by GRG are usually not feasible.
Remark

During the iteration, the values of \( f(x^k) \) can sometimes be smaller than the minimum value.

How this is possible?

The reason is that the points \( x^k \) computed by GRG are usually not feasible.

They are only feasible for linear approximations of these constraints.
Non-linear equality constraints
Starting from an infeasible solution

Assume that we have chosen the point $x_0$ so that it is infeasible. We consider a phase 1 problem, which is the construction of a feasible solution. The objective function for the phase 1 problem is the sum of the absolute values of the violated constraints. The constraints for the phase 1 problem are the non-violated ones.
Starting from an infeasible solution

Assume that we have chosen the point $x^0$ so that it is infeasible.
Starting from an infeasible solution

Assume that we have chosen the point \( \mathbf{x}^0 \) so that it is infeasible.

We consider a phase 1 problem, which is the construction of a feasible solution.
Starting from an infeasible solution

Assume that we have chosen the point $x^0$ so that it is infeasible.

We consider a phase 1 problem, which is the construction of a feasible solution.

The objective function for the phase 1 problem is the sum of the absolute values of the violated constraints.
Non-linear equality constraints

Starting from an infeasible solution

Assume that we have chosen the point $x^0$ so that it is infeasible.

We consider a phase 1 problem, which is the construction of a feasible solution.

The objective function for the phase 1 problem is the sum of the absolute values of the violated constraints.

The constraints for the phase 1 problem are the non-violated ones.
Motivating Examples
Problem Formulation
Constrained Optimization
The generalized reduced gradient method
Non-smooth Optimization: Subgradient methods

Non-linear equality constraints
Motivating Examples
Problem Formulation
Constrained Optimization
The generalized reduced gradient method
Non-smooth Optimization: Subgradient methods

Non-linear equality constraints

Starting from an infeasible solution: Our example

If we had started from the point $x_0 = (1, 1, 0, 1)$, which happens to be infeasible, then the phase 1 problem would be the following:

$$
\begin{align*}
\min & \quad |x_1^2 + x_2^2 + 4x_3 + 4x_4| - x_1 + x_2 + 2x_3 - 2x_4 + 4 = 0,
\end{align*}
$$

This is because $x_0$ violates the first constraint and satisfies the second one. Observe that the value of the objective function is 0, if and only if the corresponding point is a feasible solution.
Starting from an infeasible solution: Our example

If we had started from the point $\mathbf{x}^0 = (1, 1, 0, 1)$, which happens to be infeasible, then the phase 1 problem would be the following:

$$
\begin{align*}
\min & \quad |x_2^2 + x_3^4 + 4x_4| - x_1 + x_2 + 2x_3 - 2x_2^4 + 2x_1^2 \\
\text{s.t.} & \quad x_1 - x_2 + 2x_3 - 2x_2^4 + 2x_1^2 = 0
\end{align*}
$$

This is because $\mathbf{x}^0$ violates the first constraint and satisfies the second one.

Observe that the value of the objective function is 0, if and only if the corresponding point is a feasible solution.
Non-linear equality constraints

Starting from an infeasible solution: Our example

If we had started from the point $\mathbf{x}^0 = (1, 1, 0, 1)$, which happens to be infeasible, then the phase 1 problem would be the following:

$$\min |x_1^2 + x_2 + 4 \cdot x_3 + 4 \cdot x_4 - 4|$$
Starting from an infeasible solution: Our example

If we had started from the point $x^0 = (1, 1, 0, 1)$, which happens to be infeasible, then the phase 1 problem would be the following:

$$\begin{align*}
\min & \quad |x_1^2 + x_2 + 4 \cdot x_3 + 4 \cdot x_4 - 4| \\
- & \quad x_1 + x_2 + 2 \cdot x_3 - 2 \cdot x_4^2 + 2 = 0,
\end{align*}$$

This is because $x^0$ violates the first constraint and satisfies the second one. Observe that the value of the objective function is 0, if and only if the corresponding point is a feasible solution.
Starting from an infeasible solution: Our example

If we had started from the point $\mathbf{x}^0 = (1, 1, 0, 1)$, which happens to be infeasible, then the phase 1 problem would be the following:

$$\min |x_1^2 + x_2 + 4 \cdot x_3 + 4 \cdot x_4 - 4|$$
$$- x_1 + x_2 + 2 \cdot x_3 - 2 \cdot x_4^2 + 2 = 0,$$

This is because $\mathbf{x}^0$ violates the first constraint and satisfies the second one.
Starting from an infeasible solution: Our example

If we had started from the point $x^0 = (1, 1, 0, 1)$, which happens to be infeasible, then the phase 1 problem would be the following:

$$
\begin{align*}
\min & \quad |x_1^2 + x_2 + 4 \cdot x_3 + 4 \cdot x_4 - 4| \\
& - x_1 + x_2 + 2 \cdot x_3 - 2 \cdot x_4^2 + 2 = 0,
\end{align*}
$$

This is because $x^0$ violates the first constraint and satisfies the second one.

Observe that the value of the objective function is 0, if and only if the corresponding point is a feasible solution.
The general strategy

We will discuss how GRG solves problems when there are inequality as well as equality constraints.

At each iteration, only the tight inequality constraints enter into the system of linear equations used for eliminating variables (active inequality constraints).

The process is complicated by the fact that active inequality constraints at the current point may need to be released in order to move to a better solution.
We will discuss how GRG solves problems when there are inequality as well as equality constraints. At each iteration, only the tight inequality constraints enter into the system of linear equations used for eliminating variables (active inequality constraints). The process is complicated by the fact that active inequality constraints at the current point may need to be released in order to move to a better solution.
Linear inequality constraints

The general strategy

We will discuss how GRG solves problems when there are inequality as well as equality constraints.
Linear inequality constraints

The general strategy

We will discuss how GRG solves problems when there are inequality as well as equality constraints.

At each iteration, only the tight inequality constraints enter into the system of linear linear equations used for eliminating variables (active inequality constraints).
Linear inequality constraints

The general strategy

We will discuss how GRG solves problems when there are inequality as well as equality constraints.

At each iteration, only the tight inequality constraints enter into the system of linear linear equations used for eliminating variables (active inequality constraints).

The process is complicated by the fact that active inequality constraints at the current point may need to be released in order to move to a better solution.
Motivating Examples
Problem Formulation
Constrained Optimization
Non-smooth Optimization: Subgradient methods

Linear inequality constraints

The generalized reduced gradient method

An example

\[
\begin{align*}
\min & \quad f(x_1, x_2) = (x_1 - 1)^2 + (x_2 - 5)^2 \\
\text{s.t.} & \quad x_1 - x_2 \geq 0, \\
& \quad x_1 \geq 0, \\
& \quad x_2 \leq 2.
\end{align*}
\]

The process

Assume that the initial feasible solution is \(x_0 = (1, 0)\).

It can be checked directly that the constraints \(x_1 - x_2 \geq 0, x_1 \geq 0, \) and \(x_2 \leq 2\) are inactive, whereas the constraint \(x_2 \geq 0\) is active.
Linear inequality constraints

An example

\[ \begin{align*}
\min & \quad f(x_1, x_2) = (x_1 - 1)^2 + (x_2 - 5)^2 \\
\text{subject to} & \quad x_1 - x_2 \geq 0, \\
& \quad x_1 \geq 0, \\
& \quad x_2 \leq 2.
\end{align*} \]

The process

Assume that the initial feasible solution is \( x_0 = (1, 0) \).

It can be checked directly that the constraints \( x_1 - x_2 \geq 0, x_1 \geq 0, \) and \( x_2 \leq 2 \) are inactive, whereas the constraint \( x_2 \geq 0 \) is active.
Linear inequality constraints

An example

\[ \min f(x_1, x_2) = (x_1 - \frac{1}{2})^2 + (x_2 - \frac{5}{2})^2 \]
An example

\[
\min f(x_1, x_2) = (x_1 - \frac{1}{2})^2 + (x_2 - \frac{5}{2})^2 \\
x_1 - x_2 \geq 0,
\]
Linear inequality constraints

An example

\[ \min f(x_1, x_2) = (x_1 - \frac{1}{2})^2 + (x_2 - \frac{5}{2})^2 \]
\[ x_1 - x_2 \geq 0, \]
\[ x_1 \geq 0, \]
\[ x_2 \geq 0, \]
Motivating Examples
Problem Formulation
Constrained Optimization
The generalized reduced gradient method
Non-smooth Optimization: Subgradient methods

Linear inequality constraints

An example

\[ \min f(x_1, x_2) = (x_1 - \frac{1}{2})^2 + (x_2 - \frac{5}{2})^2 \]

\[ x_1 - x_2 \geq 0, \]

\[ x_1 \geq 0, \]

\[ x_2 \geq 0, \]
Linear inequality constraints

An example

\[
\min f(x_1, x_2) = (x_1 - \frac{1}{2})^2 + (x_2 - \frac{5}{2})^2
\]

\[
x_1 - x_2 \geq 0,
\]

\[
x_1 \geq 0,
\]

\[
x_2 \geq 0,
\]

\[
x_2 \leq 2,
\]
Linear inequality constraints

An example

\[ \min f(x_1, x_2) = (x_1 - \frac{1}{2})^2 + (x_2 - \frac{5}{2})^2 \]

\[ x_1 - x_2 \geq 0, \]

\[ x_1 \geq 0, \]

\[ x_2 \geq 0, \]

\[ x_2 \leq 2, \]

The process
An example

\[
\begin{align*}
\min f(x_1, x_2) &= (x_1 - \frac{1}{2})^2 + (x_2 - \frac{5}{2})^2 \\
x_1 - x_2 &\geq 0, \\
x_1 &\geq 0, \\
x_2 &\geq 0, \\
x_2 &\leq 2,
\end{align*}
\]

The process

Assume that the initial feasible solution is \(x^0 = (1, 0)\).
Motivating Examples
Problem Formulation
Constrained Optimization
The generalized reduced gradient method
Non-smooth Optimization: Subgradient methods

Linear inequality constraints

An example

\[ \min f(x_1, x_2) = (x_1 - \frac{1}{2})^2 + (x_2 - \frac{5}{2})^2 \]

\[ x_1 - x_2 \geq 0, \]
\[ x_1 \geq 0, \]
\[ x_2 \geq 0, \]
\[ x_2 \leq 2, \]

The process

Assume that the initial feasible solution is \( x^0 = (1, 0) \).

It can be checked directly that the constraints \( x_1 - x_2 \geq 0, x_1 \geq 0, \) and \( x_2 \leq 2 \) are inactive, whereas the constraint \( x_2 \geq 0 \) is active.
The feasible region

\[ x_1 - x_2 = 0 \]
The feasible region

\[ x_1 - x_2 = 0 \]

\[ x_2 = 2 \]
Motivating Examples
Problem Formulation
Constrained Optimization
The generalized reduced gradient method
Non-smooth Optimization: Subgradient methods

The feasible region

\[ x_1 - x_2 = 0 \]

\[ x_2 = 2 \]
Motivating Examples
Problem Formulation
Constrained Optimization
The generalized reduced gradient method
Non-smooth Optimization: Subgradient methods

Linear inequality constraints

First let us evaluate the gradient of the objective function at $x_0 = (1, 0)$. 
\[ \nabla f(x_0) = (2 \cdot x_0^1 - 1, 2 \cdot x_0^2 - 5) = (1, -5) \].

This means that we will get the largest decrease in $f$ if we move in the direction $d_0 = -\nabla f(x_0) = (-1, 5)$, that is, if we decrease $x_1$ and increase $x_2$.

The new point will be $x_1 = x_0 + \alpha_0 \cdot d_0$ for some $\alpha_0 > 0$.

The constraints imply that $\alpha_0 \leq 0.8333$.

Now we perform a line search to determine the best value of $\alpha_0$ in this range.

It can be shown that $\alpha_0 = 0.8333$, so $x_1 = (0.8333, 0)$.
An example

First let us evaluate the gradient of the objective function at \( x_0 = (1, 0) \).

\[
\nabla f(x_0) = (2 \cdot x_0^1 - 1, 2 \cdot x_0^2 - 5) = (1, -5).
\]

This means that we will get the largest decrease in \( f \) if we move in the direction \( d_0 = -\nabla f(x_0) = (-1, 5) \), that is, if we decrease \( x_1 \) and increase \( x_2 \).

The new point will be \( x_1 = x_0 + \alpha_0 \cdot d_0 \) for some \( \alpha_0 > 0 \).

The constraints imply that \( \alpha_0 \leq 0.8333 \).

Now we perform a line search to determine the best value of \( \alpha_0 \) in this range. It can be shown that \( \alpha_0 = 0.8333 \), so \( x_1 = (0.8333, 0.8333) \).
First let us evaluate the gradient of the objective function at $x^0 = (1, 0)$.

\[ \nabla f(x_0) = (2 \cdot x_0^1 - 1, 2 \cdot x_0^2 - 5) = (1, -5). \]

This means that we will get the largest decrease in $f$ if we move in the direction $d_0 = -\nabla f(x_0) = (-1, 5)$, that is, if we decrease $x_1$ and increase $x_2$.

The new point will be $x_1 = x_0 + \alpha_0 \cdot d_0$ for some $\alpha_0 > 0$.

The constraints imply that $\alpha_0 \leq 0.8333$.

Now we perform a line search to determine the best value of $\alpha_0$ in this range. It can be shown that $\alpha_0 = 0.8333$, so $x_1 = (0.8333, 0.8333)$. 
An example

First let us evaluate the gradient of the objective function at $\mathbf{x}^0 = (1, 0)$.

$$\nabla f(\mathbf{x}^0) = (2 \cdot x_1^0 - 1, 2 \cdot x_2^0 - 5) = (1, -5).$$
Linear inequality constraints

An example

First let us evaluate the gradient of the objective function at $\mathbf{x}^0 = (1, 0)$.

$$\nabla f(\mathbf{x}^0) = (2 \cdot x_1^0 - 1, 2 \cdot x_2^0 - 5) = (1, -5).$$

This means that we will get the largest decrease in $f$ if we move in the direction $\mathbf{d}^0 = -\nabla f(\mathbf{x}^0) = (-1, 5)$, that is, if we decrease $x_1$ and increase $x_2$. 
An example

First let us evaluate the gradient of the objective function at $x^0 = (1, 0)$.

$$\nabla f(x^0) = (2 \cdot x_1^0 - 1, 2 \cdot x_2^0 - 5) = (1, -5).$$

This means that we will get the largest decrease in $f$ if we move in the direction $d^0 = -\nabla f(x^0) = (-1, 5)$, that is, if we decrease $x_1$ and increase $x_2$.

The new point will be $x^1 = x^0 + \alpha^0 \cdot d^0$ for some $\alpha^0 > 0$. 
An example

First let us evaluate the gradient of the objective function at $x^0 = (1, 0)$.

$$\nabla f(x^0) = (2 \cdot x_1^0 - 1, 2 \cdot x_2^0 - 5) = (1, -5).$$

This means that we will get the largest decrease in $f$ if we move in the direction $d^0 = -\nabla f(x^0) = (-1, 5)$, that is, if we decrease $x_1$ and increase $x_2$.

The new point will be $x^1 = x^0 + \alpha^0 \cdot d^0$ for some $\alpha^0 > 0$.

The constraints imply that $\alpha^0 \leq 0.8333$. 
An example

First let us evaluate the gradient of the objective function at $x^0 = (1, 0)$.

$$\nabla f(x^0) = (2 \cdot x_1^0 - 1, 2 \cdot x_2^0 - 5) = (1, -5).$$

This means that we will get the largest decrease in $f$ if we move in the direction $d^0 = -\nabla f(x^0) = (-1, 5)$, that is, if we decrease $x_1$ and increase $x_2$.

The new point will be $x^1 = x^0 + \alpha^0 \cdot d^0$ for some $\alpha^0 > 0$.

The constraints imply that $\alpha^0 \leq 0.8333$.

Now we perform a line search to determine the best value of $\alpha^0$ in this range.
An example

First let us evaluate the gradient of the objective function at $x^0 = (1, 0)$.

$$\nabla f(x^0) = (2 \cdot x_1^0 - 1, 2 \cdot x_2^0 - 5) = (1, -5).$$

This means that we will get the largest decrease in $f$ if we move in the direction $d^0 = -\nabla f(x^0) = (-1, 5)$, that is, if we decrease $x_1$ and increase $x_2$.

The new point will be $x^1 = x^0 + \alpha^0 \cdot d^0$ for some $\alpha^0 > 0$.

The constraints imply that $\alpha^0 \leq 0.8333$.

Now we perform a line search to determine the best value of $\alpha^0$ in this range.

It can be shown that $\alpha^0 = 0.8333$, so $x^1 = (0.8333, 0.8333)$. 
Motivating Examples
Problem Formulation
Constrained Optimization
The generalized reduced gradient method
Non-smooth Optimization: Subgradient methods

The feasible region and the progress of the algorithm

\[ x_1 - x_2 = 0 \]
\[ x_2 = 2 \]
The feasible region and the progress of the algorithm

\[ x_1 - x_2 = 0 \]

\[ x_2 = 2 \]

\[ x^0 = (1, 0) \]
Motivating Examples
Problem Formulation
Constrained Optimization
The generalized reduced gradient method
Non-smooth Optimization: Subgradient methods

The feasible region and the progress of the algorithm

\[ x_1 - x_2 = 0 \]

\[ x_2 = 2 \]

\[ x^1 = (0.8333, 0.8333) \]

\[ x^0 = (1, 0) \]
The feasible region and the progress of the algorithm

\[ x_1 - x_2 = 0 \]

\[ x_2 = 2 \]

\[ x^2 = (1.5, 1.5) \]

\[ x^1 = (0.8333, 0.8333) \]

\[ x^0 = (1, 0) \]
Problem Formulation

Consider a general non-linear optimization problem:

\[ \min_{x} f(x) \]

\[ g_i(x) = 0, \quad i \in E \]

\[ g_i(x) \geq 0, \quad i \in I \]

Quadratic Programs

In order to solve this problem, we can use methods available for quadratic programs. This is the idea behind sequential quadratic programming.
Consider a general non-linear optimization problem:

$$\min f(x)$$

$$g_i(x) = 0, \quad i \in E$$

$$g_i(x) \geq 0, \quad i \in I$$

Quadratic Programs

In order to solve this problem, we can use methods available for quadratic programs. This is the idea behind sequential quadratic programming.
Non-linear programming

**Problem Formulation**

Consider a general non-linear optimization problem:

$$\min f(x)$$
Consider a general non-linear optimization problem:

\[
\min f(x) \\
g_i(x) = 0, i \in \mathcal{E}
\]
Problem Formulation

Consider a general non-linear optimization problem:

$$\min f(x)$$

$$g_i(x) = 0, i \in \mathcal{E}$$

$$g_i(x) \geq 0, i \in \mathcal{I}$$
Motivating Examples

Problem Formulation

Consider a general non-linear optimization problem:

\[
\begin{align*}
\min f(x) \\
g_i(x) &= 0, i \in \mathcal{E} \\
g_i(x) &\geq 0, i \in \mathcal{I}
\end{align*}
\]

Quadratic Programs
Consider a general non-linear optimization problem:

\[
\min f(x)
\]

\[
g_i(x) = 0, i \in \mathcal{E}
\]

\[
g_i(x) \geq 0, i \in \mathcal{I}
\]

In order to solve this problem, we can use methods available for quadratic programs.
Problem Formulation

Consider a general non-linear optimization problem:

\[
\min f(x) \\
g_i(x) = 0, i \in \mathcal{E} \\
g_i(x) \geq 0, i \in \mathcal{I}
\]

Quadratic Programs

In order to solve this problem, we can use methods available for quadratic programs.

This is the idea behind sequential quadratic programming.
The idea
At the current feasible point $x_k$, the problem is approximated by a quadratic program:

A quadratic approximation of the Lagrangian function is computed as well as linear approximations of the constraints.

The resulting quadratic program is of the form:

$$\min \left( \nabla f(x_k) \cdot (x - x_k) + \frac{1}{2} \cdot (x - x_k)^T \cdot B_k \cdot (x - x_k) \right) \nabla g_i(x_k) \cdot (x - x_k) + g_i(x_k) = 0, \quad i \in E$$

$$\nabla g_i(x_k) \cdot (x - x_k) + g_i(x_k) \geq 0, \quad i \in I$$

Where $B_k = \nabla^2_{xx} \mathcal{L}(x_k, \lambda_k)$ is the Hessian of the Lagrangian function with respect to the variables $x$ and $\lambda_k$ is the current estimate of the Lagrangian multipliers.
The idea

At the current feasible point $x_k$, the problem is approximated by a quadratic program:

A quadratic approximation of the Lagrangian function is computed as well as linear approximations of the constraints. The resulting quadratic program is of the form:

$$\min_{x} \left( \nabla f(x_k) \cdot (x - x_k) + \frac{1}{2} \cdot (x - x_k)^T \cdot B_k \cdot (x - x_k) \right)$$

$$\nabla g_i(x_k) \cdot (x - x_k) + g_i(x_k) = 0, \quad i \in E$$

$$\nabla g_i(x_k) \cdot (x - x_k) + g_i(x_k) \geq 0, \quad i \in I$$

Where $B_k = \nabla^2 xL(x_k, \lambda_k)$ is the Hessian of the Lagrangian function with respect to the variables $x$ and $\lambda_k$ is the current estimate of the Lagrangian multipliers.
The idea

At the current feasible point $\mathbf{x}^k$, the problem is approximated by a quadratic program:

$$\begin{align*}
\min & \quad \nabla f(\mathbf{x}^k)^T (\mathbf{x} - \mathbf{x}^k) + \frac{1}{2} (\mathbf{x} - \mathbf{x}^k)^T \nabla^2 f(\mathbf{x}^k, \lambda^k) (\mathbf{x} - \mathbf{x}^k)
\end{align*}$$

$$\begin{align*}
\nabla g_i(\mathbf{x}^k)^T (\mathbf{x} - \mathbf{x}^k) + g_i(\mathbf{x}^k) &= 0, \quad i \in E \\
\nabla g_i(\mathbf{x}^k)^T (\mathbf{x} - \mathbf{x}^k) + g_i(\mathbf{x}^k) &\geq 0, \quad i \in I
\end{align*}$$

Where $\nabla^2 f(\mathbf{x}^k, \lambda^k)$ is the Hessian of the Lagrangian function with respect to the variables $\mathbf{x}$ and $\lambda^k$ is the current estimate of the Lagrangian multipliers.
At the current feasible point $x^k$, the problem is approximated by a quadratic program:

A quadratic approximation of the Lagrangian function is computed as well as linear approximations of the constraints.
At the current feasible point $x^k$, the problem is approximated by a quadratic program:

A quadratic approximation of the Lagrangian function is computed as well as linear approximations of the constraints.

The resulting quadratic program is of the form:

$$\min (\nabla f(x^k)^T \cdot (x - x^k) + \frac{1}{2} \cdot (x - x^k)^T \cdot B_k \cdot (x - x^k))$$
The idea

At the current feasible point \( x^k \), the problem is approximated by a quadratic program:

A quadratic approximation of the Lagrangian function is computed as well as linear approximations of the constraints.

The resulting quadratic program is of the form:

\[
\min (\nabla f(x^k)^T \cdot (x - x^k) + \frac{1}{2} \cdot (x - x^k)^T \cdot B_k \cdot (x - x^k))
\]

\[
\nabla g_i(x^k)^T \cdot (x - x^k) + g_i(x^k) = 0, \ i \in \mathcal{E}
\]
Non-linear programming

The idea

At the current feasible point $x^k$, the problem is approximated by a quadratic program:

A quadratic approximation of the Lagrangian function is computed as well as linear approximations of the constraints.

The resulting quadratic program is of the form:

$$\min(\nabla f(x^k)^T \cdot (x - x^k) + \frac{1}{2} \cdot (x - x^k)^T \cdot B_k \cdot (x - x^k))$$

$$\nabla g_i(x^k)^T \cdot (x - x^k) + g_i(x^k) = 0, \ i \in \mathcal{E}$$

$$\nabla g_i(x^k)^T \cdot (x - x^k) + g_i(x^k) \geq 0, \ i \in \mathcal{I}$$
The idea

At the current feasible point $x^k$, the problem is approximated by a quadratic program:

A quadratic approximation of the Lagrangian function is computed as well as linear approximations of the constraints.

The resulting quadratic program is of the form:

$$
\min (\nabla f(x^k)^T \cdot (x - x^k) + \frac{1}{2} (x - x^k)^T \cdot B_k \cdot (x - x^k))
$$

$$
\nabla g_i(x^k)^T \cdot (x - x^k) + g_i(x^k) = 0, \quad i \in \mathcal{E}
$$

$$
\nabla g_i(x^k)^T \cdot (x - x^k) + g_i(x^k) \geq 0, \quad i \in \mathcal{I}
$$

Where $B_k = \nabla^2_{xx} L(x^k, \lambda^k)$ is the Hessian of the Langrangian function with respect to the variables $x$ and $\lambda^k$ is the current estimate of the Lagrangian multipliers.
Motivating Examples

Problem Formulation

Constrained Optimization

The generalized reduced gradient method

Non-smooth Optimization: Subgradient methods

Non-linear programming

Next iteration

This problem can be solved with one of the methods developed for quadratic programs. The optimal solution of the quadratic program is used to determine a search direction. Then a line search is performed to determine the next iterate.
This problem can be solved with one of the methods developed for quadratic programs. The optimal solution of the quadratic program is used to determine a search direction. Then a line search is performed to determine the next iterate.
Next iteration

This problem can be solved with one of the methods developed for quadratic programs.
This problem can be solved with one of the methods developed for quadratic programs. The optimal solution of the quadratic program is used to determine a search direction.
Non-linear programming

Next iteration

This problem can be solved with one of the methods developed for quadratic programs.

The optimal solution of the quadratic program is used to determine a search direction.

Then a line search is performed to determine the next iterate.
Non-smooth Optimization

Problem Formulation

We will consider unconstrained nonlinear programs of the form

\[ \min f(x) \]

where \( x = (x_1, \ldots, x_n) \) and \( f \) is a non-differentiable convex function.

Optimality conditions based on the gradient are not available since the gradient is not always defined.

However, the notion of gradient can be generalized as follows.

Definition

A subgradient of \( f \) at point \( x^* \) is a vector \( s^* = (s^*_1, \ldots, s^*_n) \) such that

\[ s^* \cdot (x - x^*) \leq f(x) - f(x^*), \]

for every \( x \).

Mkrtchyan
Optimization Methods in Finance
Non-smooth Optimization

Problem Formulation

We will consider unconstrained nonlinear programs of the form

\[
\min f(x)
\]

where \(x = (x_1, \ldots, x_n)\) and \(f\) is a non-differentiable convex function.

Optimality conditions based on the gradient are not available since the gradient is not always defined. However, the notion of gradient can be generalized as follows.

**Definition**

A subgradient of \(f\) at point \(x^*\) is a vector \(s^* = (s^*_1, \ldots, s^*_n)\) such that

\[
s^* \cdot (x - x^*) \leq f(x) - f(x^*),
\]

for every \(x\).
Problem Formulation

We will consider unconstrained nonlinear programs of the form

\[ \min f(x) \]
Problem Formulation

We will consider unconstrained nonlinear programs of the form

$$\min f(x)$$

where $x = (x_1, \ldots, x_n)$ and $f$ is a non-differentiable convex function.
Problem Formulation

We will consider unconstrained nonlinear programs of the form

$$\min f(x)$$

where $x = (x_1, \ldots, x_n)$ and $f$ is a non-differentiable convex function.

Optimality conditions based on the gradient are not available since the gradient is not always defined.
Non-smooth Optimization

Problem Formulation

We will consider unconstrained nonlinear programs of the form

$$\min f(x)$$

where $x = (x_1, \ldots, x_n)$ and $f$ is a non-differentiable convex function.

Optimality conditions based on the gradient are not available since the gradient is not always defined.

However, the notion of gradient can be generalized as follows.
Non-smooth Optimization

Problem Formulation

We will consider unconstrained nonlinear programs of the form

$$\min f(x)$$

where $x = (x_1, \ldots, x_n)$ and $f$ is a non-differentiable convex function.

Optimality conditions based on the gradient are not available since the gradient is not always defined.

However, the notion of gradient can be generalized as follows.

Definition
Problem Formulation

We will consider unconstrained nonlinear programs of the form

\[ \min f(x) \]

where \( x = (x_1, \ldots, x_n) \) and \( f \) is a non-differentiable convex function.

Optimality conditions based on the gradient are not available since the gradient is not always defined.

However, the notion of gradient can be generalized as follows.

Definition

A subgradient of \( f \) at point \( x^* \) is a vector \( s^* = (s_1^*, \ldots, s_n^*) \) such that
Non-smooth Optimization

Problem Formulation

We will consider unconstrained nonlinear programs of the form

\[
\min f(x)
\]

where \( x = (x_1, \ldots, x_n) \) and \( f \) is a non-differentiable convex function.

Optimality conditions based on the gradient are not available since the gradient is not always defined.

However, the notion of gradient can be generalized as follows.

Definition

A subgradient of \( f \) at point \( x^* \) is a vector \( s^* = (s_1^*, \ldots, s_n^*) \) such that

\[
 s^* \cdot (x - x^*) \leq f(x) - f(x^*), \text{ for every } x.
\]
Non-smooth Optimization

Subgradients

When the function is differentiable, the subgradient is identical to the gradient.

When $f$ is not differentiable at point $x$, there are typically many subgradients at $x$.

For example consider the function $f(x) = |x - 1|$.
Non-smooth Optimization

Subgradients

When the function is differentiable, the subgradient is identical to the gradient. When \( f \) is not differentiable at point \( x \), there are typically many subgradients at \( x \). For example, consider the function \( f(x) = |x - 1| \).
Non-smooth Optimization

Subgradients

When the function is differentiable, the subgradient is identical to the gradient.
Subgradients

When the function is differentiable, the subgradient is identical to the gradient.

When $f$ is not differentiable at point $\mathbf{x}$, there are typically many subgradients at $\mathbf{x}$. 
Non-smooth Optimization

Subgradients

When the function is differentiable, the subgradient is identical to the gradient.

When \( f \) is not differentiable at point \( \mathbf{x} \), there are typically many subgradients at \( \mathbf{x} \).

For example consider the function

\[
    f(x) = |x - 1|.
\]
The subgradients of the function $y = |x - 1|$
The subgradients of the function \( y = |x - 1| \)
The subgradients of the function $y = |x - 1|$
The subgradients of the function $y = |x - 1|$
The subgradients of the function $y = |x - 1|$
A property of convex functions

Theorem
Let $f$ be a non-differentiable convex function. The point $x^*$ is a minimum of $f$, if and only if $f$ has a zero subgradient at $x^*$.

Example
In the case of $f(x) = |x - 1|$, 0 is a subgradient at point $x^* = 1$, therefore this is the point where the minimum of $f$ is achieved.
A property of convex functions

**Theorem**

Let $f$ be a non-differentiable convex function. The point $x^*$ is a minimum of $f$, if and only if $f$ has a zero subgradient at $x^*$. 

**Example**

In the case of $f(x) = |x - 1|$, 0 is a subgradient at point $x^* = 1$, therefore this is the point where the minimum of $f$ is achieved.
A property of convex functions

Theorem

*Let $f$ is a non-differentiable convex function.*
A property of convex functions

**Theorem**

Let $f$ be a non-differentiable convex function.

The point $x^*$ is a minimum of $f$, if and only if $f$ has a zero subgradient at $x^*$. 

A property of convex functions

**Theorem**

Let $f$ be a non-differentiable convex function.

The point $\mathbf{x}^*$ is a minimum of $f$, if and only if $f$ has a zero subgradient at $\mathbf{x}^*$.

**Example**

In the case of $f(\mathbf{x}) = |\mathbf{x} - 1|$, 0 is a subgradient at point $\mathbf{x}^* = 1$, therefore this is the point where the minimum of $f$ is achieved.
A property of convex functions

**Theorem**

Let $f$ be a non-differentiable convex function.

The point $x^*$ is a minimum of $f$, if and only if $f$ has a zero subgradient at $x^*$.

**Example**

In the case of $f(x) = |x - 1|$, 0 is a subgradient at point $x^* = 1$, therefore this is the point where the minimum of $f$ is achieved.
The method of steepest decent for convex functions

Idea
The method of steepest decent can be extended to non-differentiable functions. First we compute any subgradient direction at the current point, and use its opposite direction to make the next step.

Though subgradient directions are not always directions of ascent, one can still guarantee convergence to the optimum point by choosing the step size appropriately.
The method of steepest decent for convex functions

Idea

The method of steepest decent can be extended to non-differentiable functions. First we compute any subgradient direction at the current point, and use its opposite direction to make the next step. Though subgradient directions are not always directions of ascent, one can still guarantee convergence to the optimum point by choosing the step size appropriately.
The method of steepest decent for convex functions

Idea

The method of steepest decent can be extended to non-differentiable functions.
The method of steepest decent for convex functions

Idea

The method of steepest decent can be extended to non-differentiable functions.

First we compute any subgradient direction at the current point, and use its opposite direction to make the next step.
The method of steepest decent can be extended to non-differentiable functions.

First we compute any subgradient direction at the current point, and use its opposite direction to make the next step.

Though subgradient directions are not always directions of ascent, one can still guarantee convergence to the optimum point by choosing the step size appropriately.
The method of steepest decent for convex functions
The method of steepest decent for convex functions

Formal description of the method

1: Input: Any point $x^0$. 
The method of steepest decent for convex functions

Formal description of the method

1: Input: Any point \( x^0 \).
2: Output: A point \( x^k \).
The method of steepest decent for convex functions

Formal description of the method

1: Input: Any point $x^0$.
2: Output: A point $x^k$.
3: Set $i = 0$. 
The method of steepest decent for convex functions

Formal description of the method

1: Input: Any point $\mathbf{x}^0$.
2: Output: A point $\mathbf{x}^k$.
3: Set $i = 0$.
4: Compute a subgradient $\mathbf{s}^i$ of $f$ at point $\mathbf{x}^i$. 

5: if ($\mathbf{s}^i$ is 0 or close to it)
6: stop.
7: else
8: Let $\mathbf{x}^{i+1} = \mathbf{x}^i - \alpha_i \cdot \mathbf{s}^i$, where $\alpha_i > 0$ denotes the step size.
9: Perform the next iteration.
10: end if
The method of steepest decent for convex functions

Formal description of the method

1: Input: Any point $x^0$.
2: Output: A point $x^k$.
3: Set $i = 0$.
4: Compute a subgradient $s^i$ of $f$ at point $x^i$.
5: if ($s^i$ is 0 or close to it) then
The method of steepest decent for convex functions

Formal description of the method

1: Input: Any point $x^0$.
2: Output: A point $x^k$.
3: Set $i = 0$.
4: Compute a subgradient $s^i$ of $f$ at point $x^i$.
5: if ($s^i$ is 0 or close to it) then
6: stop.

The method of steepest decent for convex functions

Formal description of the method

1: Input: Any point $x^0$.
2: Output: A point $x^k$.
3: Set $i = 0$.
4: Compute a subgradient $s^i$ of $f$ at point $x^i$.
5: \textbf{if} ($s^i$ is 0 or close to it) \textbf{then}
6: \hspace{1em} stop.
7: \textbf{else}
Formal description of the method

1: Input: Any point $x^0$.
2: Output: A point $x^k$.
3: Set $i = 0$.
4: Compute a subgradient $s^i$ of $f$ at point $x^i$.
5: if ($s^i$ is 0 or close to it) then
6: stop.
7: else
8: Let $x^{i+1} = x^i - \alpha_i \cdot s^i$, where $\alpha_i > 0$ denotes the step size.
Formal description of the method

1: Input: Any point $x^0$.
2: Output: A point $x^k$.
3: Set $i = 0$.
4: Compute a subgradient $s^i$ of $f$ at point $x^i$.
5: if ($s^i$ is 0 or close to it) then
6: stop.
7: else
8: Let $x^{i+1} = x^i - \alpha_i \cdot s^i$, where $\alpha_i > 0$ denotes the step size.
9: Perform the next iteration.
The method of steepest decent for convex functions

Formal description of the method

1: Input: Any point $x^0$.
2: Output: A point $x^k$.
3: Set $i = 0$.
4: Compute a subgradient $s^i$ of $f$ at point $x^i$.
5: if ($s^i$ is 0 or close to it) then
6: stop.
7: else
8: Let $x^{i+1} = x^i - \alpha_i \cdot s^i$, where $\alpha_i > 0$ denotes the step size.
9: Perform the next iteration.
10: end if
The method of steepest decent for convex functions

Several choices of step size $\alpha_i$ have been proposed in the literature. To guarantee convergence to the optimum, the step size $\alpha_i$ needs to be decreased very slowly. For example, for the choice of $\alpha_i \to 0$ such that $\sum \alpha_i = +\infty$, will do.
Several choices of step size $\alpha_i$ have been proposed in the literature. To guarantee convergence to the optimum, the step size $\alpha_i$ needs to be decreased very slowly. For example, for the choice of $\alpha_i \to 0$ such that $\sum \alpha_i = +\infty$, will do.
Choice of $\alpha_i$

Several choices of step size $\alpha_i$ have been proposed in the literature.
Choice of $\alpha_i$

Several choices of step size $\alpha_i$ have been proposed in the literature.

To guarantee convergence to the optimum, the step size $\alpha_i$ needs to be decreased very very slowly.
The method of steepest decent for convex functions

Choice of $\alpha_i$

Several choices of step size $\alpha_i$ have been proposed in the literature.

To guarantee convergence to the optimum, the step size $\alpha_i$ needs to be decreased very slowly.

For example, for the choice of $\alpha_i \to 0$ such that $\sum_i \alpha_i = +\infty$, will do.
The generalized reduced gradient method

Non-smooth Optimization: Subgradient methods

Literature


The list of references

The list of references

The list of references
