Large-Scale Optimization

L21. LINEAR PROGRAM 1
## Summary of Methods

<table>
<thead>
<tr>
<th></th>
<th>Gradient Descent</th>
<th>Stochastic Gradient Descent</th>
<th>ISTA / FISTA</th>
<th>BCGD / RCDM / Acc. BCGD</th>
<th>ADMM</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Convex</strong></td>
<td>Sublinear (O(1/\epsilon))</td>
<td>Sublinear (O(1/\epsilon^2))</td>
<td>Sublinear (O(1/\epsilon)) (/O(1/\sqrt{\epsilon}))</td>
<td>(O(1/\epsilon)) (/O(1/\epsilon)) (/O(1/\sqrt{\epsilon}))</td>
<td>Sublinear (O(1/\epsilon))</td>
</tr>
<tr>
<td><strong>Strongly Convex</strong></td>
<td>Linear (O(\log(1/\epsilon)))</td>
<td>Sublinear (O(1/\epsilon))</td>
<td>Linear (O(\log(1/\epsilon)))</td>
<td>Linear (O(\log(1/\epsilon)))</td>
<td>Linear (O(\log(1/\epsilon)))</td>
</tr>
<tr>
<td><strong>Smoothness</strong></td>
<td>Required</td>
<td>Any nonsmooth</td>
<td>Simple Nonsmooth</td>
<td>Required</td>
<td>(Separable) Nonsmooth</td>
</tr>
<tr>
<td><strong>Constraints</strong></td>
<td>Projection</td>
<td>Projection</td>
<td>(Thresholding)</td>
<td>Projection</td>
<td>Primal-dual</td>
</tr>
</tbody>
</table>
Today

Linear Programs

- Standard form
- Simplex method
max \ z = 5x + 4y \\
\text{s.t. } x \leq 6, \\
.25x + y \leq 6, \\
3x + 2y \leq 22, \\
x, y \geq 0.

Solution (4,5)
Min. Cost Network Flow

Flow of commodity along the arcs in a network

Each node \( i \) is associated with a divergence \( b_i \)
- \( b_i > 0 \) : supply node
- \( b_i < 0 \) : demand node

\( x_{ij} \) : the amount of the commodity to be moved along the arc \((i,j)\)

\( c_{ij} \) : the cost of moving one unit of flow along the arc \((i,j)\)
Min. Cost Network Flow LP

\[
\min_x \quad z = \sum_{(i,j) \in A} c_{ij} x_{ij}
\]

s.t. \[
\sum_{j : (i,j) \in A} x_{ij} - \sum_{j : (j,i) \in A} x_{ji} = b_i \quad \text{for all nodes } i \in \mathcal{N}
\]

\[
l_{ij} \leq x_{ij} \leq u_{ij} \quad \text{for all arcs } (i,j) \in \mathcal{A}
\]
Complexity of LP

When a solution exists, it must be a vertex of the convex polyhedron defined by constraints.

However, in the worst case the number of vertices in $\mathbb{R}^n$ defined with $\ell$ inequality constraints and bounds can be

$$\binom{\ell}{n}$$

$n = 10, \ell = 20, \binom{\ell}{n} \approx 185,000$

Therefore a brute force approach to find a solution can very slow.
Algorithms to Solve an LP

Simplex Method

- Starts from a feasible vertex
- Moves to an adjacent vertex which improves the objective function value
- Stops and returns a vertex solution
- Can be slow (# of iterations can be exponential of n)

Interior-Point Methods

- A break-through by Khachiyan (1979), the ellipsoid method
  - LP can be solved in poly(n, L), L: bits to store the data
  - Hard to implement and slow in practice
- Karmarkar (1984): a new algorithm with a similar poly bound, which has ideas of interior point methods
Standard Form

\[
\begin{align*}
    \min_{x \in \mathbb{R}^n} & \quad z = c^T x \\
    \text{subject to} & \quad Ax \geq b, \\
    & \quad x \geq 0.
\end{align*}
\]

Data:
\[c \in \mathbb{R}^n, \quad A \in \mathbb{R}^{m \times n}, \quad b \in \mathbb{R}^m\]

Inequalities can be converted to equalities by introducing slack variables:

\[
\begin{align*}
    \min_{x_N \in \mathbb{R}^n, x_B \in \mathbb{R}^m} & \quad z = c^T x_N + 0^T x_B \\
    \text{subject to} & \quad x_B = Ax - b, \\
    & \quad x_B, x_N \geq 0.
\end{align*}
\]

This is called the standard form (or the canonical form).
### Tableau Form

\[
\min_{x_N \in \mathbb{R}^n, x_B \in \mathbb{R}^m} \quad z = c^T x_N + 0^T x_B
\]

\[
x_B = A x - b,
\]

\[
x_B, x_N \geq 0.
\]

This can be represented as a tableau:

<table>
<thead>
<tr>
<th>Basic variables (slack variables, dependent variables)</th>
<th>Nonbasic variables (independent variables)</th>
</tr>
</thead>
<tbody>
<tr>
<td>( x_N )</td>
<td>( x_B )</td>
</tr>
</tbody>
</table>
| \( z = \)                                              | \( \begin{array}{c|c}
A & -b \\
\hline
c^T & 0
\end{array} \) |

TU Dortmund, Dr. Sangkyun Lee
\begin{align*}
\text{min}_{x,y} & \quad z = -5x - 4y \\
\text{s.t.} & \quad x \leq 6, \\
& \quad .25x + y \leq 6, \\
& \quad 3x + 2y \leq 22, \\
& \quad x, y \geq 0.
\end{align*}

\begin{align*}
\begin{array}{ccc|c}
 & x & y & 1 \\
\hline
u & -1 & 0 & 6 \\
v & -.25 & -1 & 6 \\
w & -3 & -2 & 22 \\
z & -5 & -4 & 0 \\
\end{array}
\end{align*}
Simplex Algorithm

This tableau represents the current state:
- Current pt: \((x,y) = (0,0)\)
- Current obj. value: \(z = 0\)

We seek a pivot exchanging a basic and a nonbasic variable which leads to a smaller \(z\):
- First choose a column (nonbasic var.) to be a basic var (thus can be nonzero)
  - Choose \(x\) (\(y=0\) fixed): objective will decrease by -5
  - Choose \(y\) (\(x=0\) fixed): objective will decrease by -4
  - \(\rightarrow\) Choose \(x\)
- This procedure is called “pricing”
Simplex Algorithm

\[
\begin{array}{cc|c}
  x & y & 1 \\
  -1 & 0 & 6 \\
  v & -1 & 6 \\
 -0.25 & -2 & 22 \\
  w & -3 & 22 \\
  z & -5 & 0 \\
\end{array}
\]

After exchange, \( x \) will set to some nonneg value, \( x = r \geq 0 \)

Now we choose a row which *blocks* the increment of \( r \):

\[
\begin{align*}
  u &= -r + 6 \geq 0 & r \leq 6 \\
  v &= -0.25r + 6 \geq 0 & r \leq 24 \\
  w &= -3r + 22 \geq 0 & r \leq \frac{22}{3} \\
\end{align*}
\]

This procedure is called “ratio test”
## Simplex Algorithm

### Initial tableau

<table>
<thead>
<tr>
<th></th>
<th>x</th>
<th>y</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>u</td>
<td>-1</td>
<td>0</td>
<td>6</td>
</tr>
<tr>
<td>v</td>
<td>-0.25</td>
<td>-1</td>
<td>6</td>
</tr>
<tr>
<td>w</td>
<td>-3</td>
<td>-2</td>
<td>22</td>
</tr>
<tr>
<td>z</td>
<td>-5</td>
<td>-4</td>
<td>0</td>
</tr>
</tbody>
</table>

### After eliminating x

<table>
<thead>
<tr>
<th></th>
<th>x</th>
<th>y</th>
<th>u</th>
<th>v</th>
<th>w</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>u</td>
<td>-1</td>
<td>0</td>
<td>-1</td>
<td>0</td>
<td>0</td>
<td>6</td>
</tr>
<tr>
<td>v</td>
<td>-0.25</td>
<td>-1</td>
<td>0</td>
<td>-1</td>
<td>0</td>
<td>6</td>
</tr>
<tr>
<td>w</td>
<td>-3</td>
<td>-2</td>
<td>0</td>
<td>0</td>
<td>-1</td>
<td>22</td>
</tr>
<tr>
<td>z</td>
<td>-5</td>
<td>-4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Eliminate x from rows except for the 1st row.

<table>
<thead>
<tr>
<th></th>
<th>x</th>
<th>y</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>u</td>
<td>-1</td>
<td>0</td>
<td>6</td>
</tr>
<tr>
<td>v</td>
<td>-0.25</td>
<td>1</td>
<td>-18/4</td>
</tr>
<tr>
<td>w</td>
<td>3</td>
<td>-2</td>
<td>4</td>
</tr>
<tr>
<td>z</td>
<td>5</td>
<td>-4</td>
<td>-30</td>
</tr>
</tbody>
</table>
\[
\begin{array}{ccc|c}
\hline
x & y & 1 \\
\hline
u = & -1 & 0 & 6 \\
v = & -0.25 & -1 & 6 \\
w = & -3 & -2 & 22 \\
z = & -5 & -4 & 0 \\
\end{array}
\]

\[
\begin{array}{ccc|c}
\hline
x & y & 1 \\
\hline
u = & -1 & 0 & 6 \\
v = & -0.25 & 1 & -18/4 \\
w = & 3 & -2 & 4 \\
z = & 5 & -4 & -30 \\
\end{array}
\]

\[
\begin{array}{cc|c}
\hline
x & y \\
\hline
u = & -1 & 0 \\
v = & -1.25 & 0.5 \\
y = & 1.5 & -0.5 \\
z = & -1 & 2 \\
\end{array}
\]

\[
\begin{array}{cc|c}
\hline
x & y \\
\hline
v = & 0.8 & -0.4 \\
u = & -0.8 & 0.4 \\
y = & -1.2 & 0.1 \\
z = & 0.8 & 1.6 \\
\end{array}
\]

This is optimal: no further improvement possible!
Remarks: Linear Independence

Consider the feasible set:

\[ S := \{ x \in \mathbb{R}^n : Ax \geq b, \ x \geq 0 \} \]

Let slack (basic) variables be:

\[ x_{n+i} = A_i \cdot x - b_i, \ i = 1, 2, \ldots, m. \]

Then any point \((x_1, x_2, \ldots, x_n) \in S\) satisfying the equations

\[ x_N = 0 \]

is a vertex of \(S\) when the \(n\) linear functions defined with variables in \(N\) are linearly independent.

Otherwise, an LP can have no solution or infinitely many solutions.
Remarks: N = Active Set

The set N defines the set of constraints active at a vertex, that is,

\[ x_N = 0 \]

Let \( x_j \in N \):

- \( x_j \) is the original variable \( \Rightarrow x_j \geq 0 \) is active
- \( x_j \) is a slack variable such that \( x_j = A_j \cdot x - b_j \Rightarrow A_j \cdot x \geq b_j \) active
A point \((x_1, x_2, \ldots, x_n) \in \mathbb{R}^n\) with \(x_N = 0\) may not be a vertex even with the linear independency.

Pivoting in the simplex algorithm make sure of feasibility via *ratio testing*. 

Not a vertex! (infeasible)
Remarks: Degenerate Vertex

For a vertex, the set $N$ may not be unique:

$$R^2$$

This vertex can be defined by any of

$$N = \{u, v\}, \{v, w\}, \{u, w\}$$
**What Happens on the Tableau?**

$$x_{B_0} = \begin{bmatrix} x_{N_0} & 1 \\ A & -b \\ c^T & 0 \end{bmatrix}$$

$$A = \begin{bmatrix} N_0 & B_0 \end{bmatrix}$$

A vertex solution \( \bar{x} = [x_B \ x_N] \) satisfies

$$A_B x_B + A_N x_N = b, \quad x_B \geq 0, \quad x_N = 0$$

where \( A_B \in \mathbb{R}^{m \times m} \) is invertible. Moreover,

$$x_B = \begin{bmatrix} x_N & 1 \end{bmatrix} \begin{bmatrix} -A_B^{-1} A_N \\ c_N^T - c_B^T A_B^{-1} A_N \end{bmatrix}$$

$$z = \begin{bmatrix} \frac{A_B^{-1}}{p_B} b \\ p_B A_B^{-1} b \end{bmatrix}$$
Remarks: Feasible Point

Simplex algorithm requires to start from a feasible point

- A feasible point can be found by another simplex algorithm with an auxiliary variable
- As hard as running the simplex algorithm from a feasible point
Reference

Ferris, Mangasarian, & Wright, Linear Programming with Matlab, MPS-SIAM Series on Optimization, 2007