1 Simplex Algorithm

1.1 Simplex Algorithm

$x$ is basic solution for $\min(c^T x)$ s.t. $Ax = b$ and $x \geq 0$ ($A \in \mathbb{R}^{n \times n}$).

- **basis** $B = (B(1), ..., B(m))$
- **columns** $A_{B(1)}, ..., A_{B(m)}$ are linearly independent.
- if $j \notin B(1), ..., B(m)$ then $x_j = 0$ ($x_j$ is non basic variable)
- $Ax = b$
- $x_B = (X_{B(1)}, ..., X_{B(n)})$ (those are the basic variables)
- $A_B = (A_{B(1)}, ..., A_{B(m)})$
- $X_B = A_B^{-1} b$

From point $x$ we now want to go to $x + \theta d$ with $\theta \geq 0$. For each $j \in [n] \setminus B$ we define $d^j$

- $d^j = 1$
- $d^i = 0$ (for each $i \in [n] \setminus (B \cup \{j\})$
- $d_B = -A_B^{-1} A_j$

It follows

- $A d^j = 0$
- $Ax = b$
- $A(x + \theta d) = Ax + \theta Ad = b$ as $\theta Ad = 0$

we now define the reduced cost $\bar{c}_j$ as

- $c^T(x + \theta d^j) - c^T x = c_j - c_B^T A_B^{-1} A_j = \bar{c}_j$

if the result vector $\bar{c}$ is strictly positive ($\bar{c} \geq 0$) then $x$ is the optimal solution.

The resulting Simplex Algorithm is as follows:
1. start with basic feasible solution $x$ and the corresponding basis $B$
2. compute $\bar{c}$
3. if $\bar{c} \geq 0$
   - TRUE then $x$ is optimal $\Rightarrow$ STOP
   - FALSE assume $\bar{c}_j < 0$
     move from $x$ in direction $d^j$ by moving to point $x + \theta^* d^j$
     where
     \[ \theta^* = \max \{ \theta \geq 0 \mid x + \theta d^j \in P \} \]
     non negtivity constraints are satisfied at $x + \theta d^j$
     for the non-negativity constraints:
     \begin{enumerate}
     \item $d^j \geq 0$ then $x + \theta d^j \geq 0$ for any $\theta \geq 0$
     \[ c^T (x + \theta d^j) = c^T x + \theta c^T d^j. \]
     It follows that optimal solution is $-\infty$. STOP
     \item if $d^j_i < 0$ for some $i$, $x_i + \theta d^j_i \geq 0$
     \[ \theta^* = \min \{ \theta \geq 0 \mid x_i + \theta d^j_i \geq 0 \} \]
     $x_i > 0$ if $i \in B(x$ is non - degenerate)$
\end{enumerate}

\[ \begin{array}{c}
\theta^* = \min_{i \mid d^j_i < 0} - \frac{-x_i}{d^j_i} \iff 0 \leq \frac{-x_i}{d^j_i} \\
\theta > 0 : d^j_i \geq 0 \text{ if } x_i \text{ is nonbasic variable} \\
x_i > 0 \text{ if } i \in B(x \text{ is non - degenerate})
\end{array} \]

let $B(l)$ be minimize for $(*)$ it follows:
\[ y = x + \theta^* d^j \]
\[ \theta^* = \frac{-x_{B(l)}}{d_{B(l)}} \]
\[ y_{B(l)} = X_{B(l)} + \theta^* d^j_{B(l)} = 0 \text{ and } d^j_{B(l)} < 0 \]
new basis $\bar{B} : \bar{B}(i) = \begin{cases} B(i) \text{ if } i \neq l \\ j \text{ if } i = l \end{cases} \theta^* > 0$
\[ \Rightarrow y \neq x \]
\[ \Rightarrow c^T y < c^T x \]

1.2 Theorem 42
1. the columns of $A_{\bar{B}}$ are linearly independent
2. $y$ is basic feasible solution corresponding to $\bar{B}$

Proof of (a)
\[ \sum_{i=1}^m \lambda_i A_{\bar{B}(i)} = 0 \]
\[ \Rightarrow \sum_{i=1}^m \lambda_i A_B^{-1} A_{\bar{B}(i)} = 0 \]
\[ \text{vectors } A_B^{-1} A_{\bar{B}(i)}, ..., A_B^{-1} A_{\bar{B}(m)} \text{ are linearly dependent} \]
\[
A_B^{-1} * A_B = I = A_B^{-1} * A_B(i) = l_i
\]
\[
A_B(i) = A_B(i) \text{ for all } i \neq l
\]
\[
\Rightarrow A_B^{-1} * A_B(i) = A_B^{-1} * A_B(i) = e_i \text{ if } i \neq l
\]
\[
B(l) = j
\]
\[
A_B^{-1} A_B(l) = A_B^{-1} A_j = -d_j^B \text{(the } l^{th} \text{ entry of } d_j^B = d_j^B(l))
\]
\[
A_B^{-1} A_B = \begin{pmatrix}
1 & \ldots & ? & \ldots & \ldots & 1 \\
\ldots & \ldots & ? & \ldots & \ldots & \ldots \\
\ldots & \ldots & \ldots & \ldots & ? & 1 \\
\end{pmatrix}
\]
\[
det(A_B^{-1} A_B) \neq 0
\]
\[
\Rightarrow \text{vectors } A_B^{-1} A_B(i), \ldots, A_B^{-1} A_B(m) \text{ are linearly independent. Contradiction!}
\]

Proof of (b)

\[
y \geq 0, Ay = b, y_i = 0 \text{ if } i \notin B
\]
\[
columns \ of \ A_B \ are \ linearly \ independent
\]
\[
\Rightarrow y \text{ is basic feasible solution for basis } B
\]

1.3 One Iteration of Simplex

One iteration of Simplex algorithm is called a pivot.

1. start with basis matrix \( A_B \), defining basic feasible solution \( x \)
2. compute reduced costs \( c_j = c_j - c_B^T A_B^{-1} A_j \)
   (a) if \( c_j \geq 0 \) ⇒ \( x \) optimal solution. STOP
   (b) choose some \( j \) with \( c_j < 0 \)
3. compute \( u = A_B^{-1} A_j = -d_j^B \) if \( u \leq 0 \) the optimum is \( -\infty \). STOP
4. choosse index \( l \) such that \( \frac{X_B(i)}{u_i} = \theta^* = min \{ \frac{X_B(i)}{u_i} | i \in [n] and u_i > 0 \} \)
5. Form new basis \( B \) by replacing \( B(l) \) with \( j \)
6. new basic feasible solution \( y \) with

\[
y_i = \theta^*
\]
\[ y_i = 0 \text{ if } i \notin \mathcal{B} = B \cup \{ j \} \setminus B(l) \]
\[ y_{B(i)} = X_{B(i)} - \theta^* u_i \]

1.4 Theorem 43

Assume \( P \neq \emptyset \), every basic feasible solution is non-degenerate, and that the algorithm is initialized with a basic feasible solution. Then it terminates after a finite amount of iterations. At termination, there are the following two options:

- We have a optimal Basis \( B \) and a associated basic feasible solution that is optimal
- We have a vector \( d \) satisfying \( Ad = 0 \), \( d \geq 0 \), \( c^T d < 0 \) and thus the optimal cost is \( -\infty \).

Proof

- If algorithm terminates in step 2 the solution is optimal because \( \bar{c} \geq 0 \)
- If algorithm terminates in step 3
  \[ \Rightarrow \exists \text{ basic feasible solution } x \text{ and direction } d^j \text{ with } Ad^j = 0, x + \theta d^j \in P, \forall \theta \geq 0 \]
  \[ c^T d^j = \bar{c}_j < 0 \]
  cost of \( x + \theta d^j \) is \( c^T (x + \theta d^j) = c^T x + \theta c^T d^j \Rightarrow \text{The optimum is } -\infty. \]

in each pivot the objective value strictly decreases. Thus, no vertex is visited twice. As there are only a limited amount of vertices the algorithm has to terminate after a finite amount of iterations.