ORF 522

Linear Programming and Convex Analysis

The Simplex Method

Marco Cuturi

Reminder: Basic Feasible Solutions, Extreme points, Optima

- Some important theorems last time for **standard forms**:

(i) Existence of one feasible solution $\Rightarrow$ Existence of a **basic feasible** solution;
(ii) **basic feasible** solutions $\Leftrightarrow$ **extreme** points of the feasible region;
(iii) **Optimum** of an LP occurs at an **extreme** point of the feasible region;

- Extreme points in canonical and corresponding standard form are equivalent.

Today

- The simplex algorithm with an initial feasible solution,
- How to check for optimality,
- How to check for unboundedness of the feasible set and/or the objective in that feasible region.

Golden slide. Always remember

- A Linear Program is a program with linear constraints and objectives.
- Equivalent formulations for LP's: **canonical** (inequalities) and **standard** (equalities) form.
- Both have feasible **convex** sets that are **bounded from below**.
- **Simplex Algorithm** to solve LP's works in standard form.

- In **standard form**, the optimum occurs on an extreme point of this polyhedron.
- All **extreme points** are **basic feasible solutions**.
- That is, all extreme points are of the type $\mathbf{x_I} = B_{\mathbf{I}}^{-1}\mathbf{b}$ for a subset $\mathbf{I}$ of coordinates, zero elsewhere.
- Looking for an optimum? **only need to check extreme points/BFS**
- Looking for an optimum? **there exists a basis $\mathbf{I}$ which realizes that optimum**.

Improving a Basic Feasible Solution

Improving a BFS

- Remember that a **standard form** LP is

$$\begin{array}{rr} \text{maximize} & \mathbf{c}^T\mathbf{x} \\ \text{subject to} & A\mathbf{x} = \mathbf{b}, \\ & \mathbf{x} \geq \mathbf{0}. \end{array}$$

- Given $\mathbf{I} = (i_1, \cdots, i_m)$, the base $B_\mathbf{I} = [\mathbf{a}_{i_1}, \mathbf{a}_{i_2}, \cdots, \mathbf{a}_{i_m}]$, suppose we have a **basic feasible solution** where $\mathbf{x}_\mathbf{I} = B^{-1}\mathbf{b}$, that is an **extreme** point of the feasible polyhedron.
- **We know that the optimum is reached on an optimal $\mathbf{I}^\star$.**
- There is finite number of families $\{\mathbf{I} | B_\mathbf{I}$ is invertible, $x_\mathbf{I}$ is feasible$\}$.
- How can we find a family $\mathbf{I}'$ such that $\mathbf{x}_{\mathbf{I}'}$ is still feasible and $\mathbf{c}_{\mathbf{I}'}^T\mathbf{x}_{\mathbf{I}'} > \mathbf{c}_\mathbf{I}^T\mathbf{x}_\mathbf{I}$?.
- The **simplex algorithm** provides an answer, where an index of $\mathbf{I}$ is replaced by a new integer in $\mathbf{O} = [1, \cdots, n] \setminus \mathbf{I}$.
- Note that we only have methods that change **one index at a time**.

The simplex does three things

Given a BFS $\mathbf{I}$

- shows how to select a **base** $\mathbf{I}'$ by changing one index in $\mathbf{I}$ (an index goes out, an index goes in)
- check how to select an **improved basic** solution by telling which index to include.
- check how we can select a **improved basic feasible** solution linked to $\mathbf{I}'$ by telling which index to remove.

In practice, given a BFS $\mathbf{I}$, the 3 steps of the simplex

1. Look for an index that would **improve** the objective.

2. check we can **improve** and obtain a valid **base** $\mathbf{I}'$ by incorporating that index and checking there is at least one we can remove.

3. **basic** & **improve** objective accomplished, ensure now $\mathbf{x}_{\mathbf{I}'}$ is **feasible** by choosing the index we remove.

Initial Setting

- Let $\mathbf{I} = (i_1, \cdots, i_m)$, the base $B_{\mathbf{I}} = [\mathbf{a}_{i_1}, \mathbf{a}_{i_2}, \cdots, \mathbf{a}_{i_m}]$ and suppose we have a **basic feasible solution** $\mathbf{x}_{\mathbf{I}} = B_{\mathbf{I}}^{-1}\mathbf{b}$.
- The **column vectors** of B are **l.i.**, and can thus be used as a basis of $\mathbf{R}^m$. Thus $\exists Y \in \mathbf{R}^{m\times n} \mid A = BY$, namely $Y = B^{-1}A$, the coordinates of all vectors of A in base B.

$$m\left\{\overbrace{\begin{bmatrix} \vdots & \vdots & \cdots\cdots & \vdots \\ \mathbf{a}_1 & \mathbf{a}_2 & \cdots\cdots & \mathbf{a}_n \\ \vdots & \vdots & \cdots\cdots & \vdots \end{bmatrix}}^{n}\right. = \overbrace{\begin{bmatrix} \vdots & \vdots & \cdot\cdot & \vdots \\ \mathbf{a}_{i_1} & \mathbf{a}_{i_2} & \cdot\cdot & \mathbf{a}_{i_m} \\ \vdots & \vdots & \cdot\cdot & \vdots \end{bmatrix}}^{m} \overbrace{\begin{bmatrix} \vdots & \vdots & \cdots\cdots & \vdots \\ \mathbf{y}_1 & \mathbf{y}_2 & \cdots\cdots & \mathbf{y}_n \\ \vdots & \vdots & \cdots\cdots & \vdots \end{bmatrix}}^{n}$$

or individually $\mathbf{a}_j = \sum_{k=1}^{m} y_{k,j}\, \mathbf{a}_{i_k}$. We write $\mathbf{y}_j = \begin{bmatrix} y_{1,j} \\ \vdots \\ y_{m,j} \end{bmatrix}$ and $\mathbf{a}_j = B\mathbf{y}_j$.

- Hence $\mathbf{y}_j = B^{-1}\mathbf{a}_j$ and B^{-1} is a change of coordinate matrix from the canonical base to the base in B.

Change an element in the basis and still have a *basic* solution

- Change an index in $\mathbf{I}$? everything depends on

$$Y = \begin{bmatrix} \vdots & \vdots & \cdots\cdots & \vdots \\ \mathbf{y}_1 & \mathbf{y}_2 & \cdots\cdots & \mathbf{y}_n \\ \vdots & \vdots & \cdots\cdots & \vdots \end{bmatrix} \in \mathbf{R}^{m \times n}$$

- Claim: if $y_{r,e} \neq 0$ for two indices, $r \leq m$, $e \leq n$ and not in $\mathbf{I}$,
  - r **for remove,** e **for enter**,
  - one can substitute the $\boldsymbol{r}^{\text{th}}$ column of $B, \mathbf{a}_{i_r}$, for the $\boldsymbol{e}^{\text{th}}$ column of $A, \mathbf{a}_e$.
  - That is we can select the basis $\hat{\mathbf{I}} = (\mathbf{I} \setminus i_r) \cup e$ and we are sure that
    - ▷ $B_{\hat{\mathbf{I}}}$ is invertible,
    - ▷ $\mathbf{x}_{\hat{\mathbf{I}}}$ is a basic solution.

basic solution

- **Proof** if $y_{r,e} \neq 0$,$\mathbf{a}_e = y_{r,e}\,\mathbf{a}_{i_r} + \sum_{k\neq r} y_{k,j}\,\mathbf{a}_{i_k} \Rightarrow \mathbf{a}_{i_r} = \frac{1}{y_{r,e}}\mathbf{a}_e - \sum_{k\neq r}\frac{y_{k,j}}{y_{r,e}}\mathbf{a}_{i_k}$.

  Thus
$$B_{\mathbf{I}}\mathbf{x}_{\mathbf{I}} = \sum_{k=1}^{m} x_{i_k}\mathbf{a}_{i_k} = x_{i_r}\mathbf{a}_{i_r} + \sum_{k=1,k\neq r}^{m} x_{i_k}\mathbf{a}_{i_k} = \mathbf{b}$$
  is replaced by
$$\frac{x_{i_r}}{y_{r,e}}\mathbf{a}_e + \sum_{k=1}^{m}\left(x_{i_k} - x_{i_r}\frac{y_{k,e}}{y_{r,e}}\right)\mathbf{a}_{i_k} = \mathbf{b}$$
  and we have a new solution $\hat{\mathbf{x}}$ with $\hat{I} = (i_1, \cdots, i_{r-1}, e, i_{r+1}, \cdots, i_m)$ and
$$\begin{aligned}\hat{x}_{i_k} &= x_{i_k} - x_{i_r}\frac{y_{k,e}}{y_{r,e}} \quad \text{for } 1 \leq k \leq m,\ (k \neq r)\\ \hat{x}_e &= \frac{x_{i_r}}{y_{r,e}}\end{aligned}$$
  note that $\hat{x}_{i_r} = 0$ and we still have a **basic** solution.

basic & *better*: restriction on e

- The objective value, $\mathbf{c}_{\mathbf{I}}^T\mathbf{x}_{\mathbf{I}}$ becomes $\mathbf{c}_{\hat{I}}^T\hat{\mathbf{x}}_{\hat{I}}$ with $\hat{c}_{i_k} = c_{i_k}$ for $k \neq r$ and $\hat{c}_e = \boldsymbol{c_e}$. Thus

$$\begin{aligned}\hat{z} &= \mathbf{c}_{\hat{I}}^T\hat{\mathbf{x}}_{\hat{I}} = \textstyle\sum_{k\neq r} c_{i_k}\hat{x}_{i_k} + \boldsymbol{c_e}\hat{x}_e \\ &= \textstyle\sum_{k\neq r} c_{i_k}\left(x_{i_k} - \boldsymbol{x_{i_r}}\frac{y_{k,e}}{\boldsymbol{y_{r,e}}}\right) + \boldsymbol{c_e}\frac{\boldsymbol{x_{i_r}}}{\boldsymbol{y_{r,e}}} \\ &= \textstyle\sum_k c_{i_k}x_{i_k} - \frac{\boldsymbol{x_{i_r}}}{\boldsymbol{y_{r,e}}}\sum_k c_{i_k}y_{k,e} + \boldsymbol{c_e}\frac{\boldsymbol{x_{i_r}}}{\boldsymbol{y_{r,e}}} \\ &= z - \frac{\boldsymbol{x_{i_r}}}{\boldsymbol{y_{r,e}}}\mathbf{c}_{\mathbf{I}}^T\mathbf{y}_e + \boldsymbol{c_e}\frac{\boldsymbol{x_{i_r}}}{\boldsymbol{y_{r,e}}} \\ &= z + \frac{\boldsymbol{x_{i_r}}}{\boldsymbol{y_{r,e}}}(\boldsymbol{c_e} - z_e),\end{aligned}$$

  where $z_e = \mathbf{c}_{\mathbf{I}}^T\mathbf{y}_e = \mathbf{c}_{\mathbf{I}}^T B^{-1}\mathbf{a}_e$.

- $\hat{z} > z$ if $y_{r,e} > 0$ and $\boldsymbol{c_e} - \boldsymbol{z_e} > 0$, hence we choose a column e such that
  - $\boldsymbol{c_e} - \boldsymbol{z_e} > 0$
  - there exists $y_{i,e} > 0$

- **Important Remark** if $\mathbf{x}_{\mathbf{I}}$ is **non-degenerate**, $x_{i_r} > 0$ and hence $\hat{z} > z$.

- Much better than $\hat{z} \geq z$ as it implies convergence.

basic & better & *feasible*: restriction on r

- We require $\hat{x}_i \geq 0$ for all i. In particular, for basic variables we need that

$$\begin{cases} \hat{x}_{i_k} = x_{i_k} - \boldsymbol{x_{i_r}} \frac{y_{k,e}}{\boldsymbol{y_{r,e}}} \geq 0 \text{ for } 1 \leq k \leq m \ (k \neq r) \\ \hat{x}_e = \frac{\boldsymbol{x_{i_r}}}{\boldsymbol{y_{r,e}}} \geq 0 \end{cases}$$

- Let r be chosen such that

$$\frac{x_{i_r}}{y_{r,e}} = \min_{k=1,..,m} \left\{ \frac{x_{i_k}}{y_{k,e}} \middle| \ y_{k,e} > 0 \right\}$$

From one basic feasible solution to a better one

Theorem 1. *Let* $\mathbf{x}$ *be a* ***basic feasible solution (BFS)*** *to a LP with index set* $\mathbf{I}$ *and objective value* z*. If there exists* $e \notin \mathbf{I}, 1 \leq e \leq n$ *such that*

(i) *a* ***reduced cost coefficient*** $\boldsymbol{c_e} - \boldsymbol{z_e} > 0$,

(ii) *at least* ***one coordinate of*** $\mathbf{y}_e$ ***is positive****,* $\exists i$ *such that* $\boldsymbol{y_{i,e} > 0}$,

then it is possible to obtain ***a new BFS*** *by replacing an index in* $\mathbf{I}$ *by* e*, and the new value of the objective value* $\hat{z}$ *is such that* $\hat{z} \geq z$*,* ***strictly*** *if* $x_{\mathbf{I}}$ *is non-degenerate.*

From one basic feasible solution to a better one

- **Remark**: coefficients $\boldsymbol{c_e} - \boldsymbol{z_e}$ are called reduced cost coefficients.
- **Remark** "$e \notin \mathbf{I}$" is redundant: if $e \in \mathbf{I}$, that is $\exists k, i_k = e$ then $\boldsymbol{c_e} - \boldsymbol{z_e} = 0$. Indeed, $\boldsymbol{c_e} - \boldsymbol{z_e} = c_e - \mathbf{c}_{\mathbf{I}}^T B^{-1}\mathbf{a}_e = c_e - \mathbf{c}_{\mathbf{I}}^T \mathbf{e}_{i_k} = c_e - c_e = 0$ where $\mathbf{e}_i$ is the i^{th} canonical vector of $\mathbf{R}^m$. Indeed, if $B\mathbf{x} = \mathbf{a}$ and $\mathbf{a}$ is the k^{th} vector of B then necessarily $\mathbf{x} = \mathbf{e}_k$.
- **Remember**: if $k \in \mathbf{I}$ then **necessarily the reduced cost** $(c_k - z_k)$ **is 0**.

Testing for Optimality

Optimality: $c_i - z_i \leq 0$ for all i

Theorem 2. *Let* $\mathbf{x}^\star$ *be a* ***basic feasible solution (BFS)*** *to a LP with index set* $\mathbf{I}^\star$ *and objective value* $z^\star$*. If* $c_i - z_i^\star \leq 0$ *for all* $1 \leq i \leq n$ *then* $\mathbf{x}^\star$ *is optimal.*

- **Proof idea**: the conditions $c_i - z_i^\star \leq 0$ allow us to write that $\sum c_i x_i$ is smaller than $\sum z_i^\star x_i$ for all $\mathbf{x}$ in $\mathbf{R}_+^m$. Moreover, $z_i^\star$ integrates information about the base $\mathbf{I}^\star$ and we show that the point that realizes $\sum z_i^\star x_i = \mathbf{c}^T\mathbf{x}$ is necessarily $\mathbf{x}^\star$ and thus every $\mathbf{c}^T\mathbf{x}$ is smaller than $\mathbf{c}^T\mathbf{x}^\star$.

Proof

- For any *feasible solution* $\mathbf{x}$ we have $\sum_{k=1}^{n} c_k x_k \leq \sum_{k=1}^{n} \boldsymbol{z_k^\star} x_k$. Yet,

$$\sum_{k=1}^{n} \boldsymbol{z_k^\star} x_k = \sum_{k=1}^{n} \mathbf{c}_{\mathbf{I}^\star}^T \mathbf{y}_k \, x_k = \sum_{k=1}^{n} \left(\sum_{j=1}^{m} \boldsymbol{c_{i_j} y_{j,k}} \right) x_k = \sum_{j=1}^{m} \boldsymbol{c_{i_j}} \left(\sum_{k=1}^{n} \boldsymbol{y_{j,k}} x_k \right)$$

- We have found a maxima of $\mathbf{c}^T \mathbf{x}$ with base $\mathbf{I}^\star$...
- The terms $u_j \stackrel{\text{def}}{=} \sum_{k=1}^{n} \boldsymbol{y_{j,k}} \, x_k$ **are actually equal to** $x_{i_j}^\star$. Indeed, remember $\sum_{j=1}^{m} \boldsymbol{x_{i_j}^\star} \mathbf{a}_{i_j} = \mathbf{b}$ and that since $\mathbf{x}$ is feasible, $\sum_{k=1}^{n} x_k \mathbf{a}_k = \mathbf{b}$. Yet,

$$\sum_{k=1}^{n} x_k (\boldsymbol{B_{\mathbf{I}^\star} \mathbf{y}_k}) = \sum_{k=1}^{n} \left(\sum_{j=1}^{m} \boldsymbol{y_{k,j}} \mathbf{a}_{i_j} \right) x_k = \sum_{j=1}^{m} \left(\sum_{k=1}^{n} \boldsymbol{y_{k,j}} x_k \right) \mathbf{a}_{i_j} = \sum_{j=1}^{m} u_j \mathbf{a}_{i_j} = \mathbf{b}.$$

  Hence

$$z \leq \sum_{j=1}^{m} \boldsymbol{c_{i_j}} \, \boldsymbol{x_{i_j}^\star} = z^\star.$$

Testing for Boundedness

(un)boundedness

- Sometimes programs are trivially unbounded

$$\begin{array}{ll} \text{maximize} & \mathbf{1}^T\mathbf{x} \\ \text{subject to} & \mathbf{x} \geq \mathbf{0}. \end{array}$$

- Here **both** the feasible set and the objective on that feasible set are **unbounded**.
- Feasible set is **bounded** $\Rightarrow$ objective is bounded.
- Feasible set is **unbounded**, optimum might be bounded **or** unbounded, no implication.
- Two different issues.
- Can we check quickly?

(un)boundedness of the feasible set *and/or* of the objective.

Theorem 3. *Consider an LP in* **standard** *form and a basic feasible index set* $\mathbf{I}$. *If there exists an index* $e \notin \mathbf{I}$ *such that* $\mathbf{y}_e \leq 0$ *then the* ***feasible region*** *is* ***unbounded***. *If moreover for* e *the reduced cost* $c_e - z_e > 0$ *then there exists a feasible solution with at most* $\mathbf{m+1}$ *nonzero variables and an* ***arbitrary large objective function***.

Proof sketch:

- Take advantage of $\mathbf{y}_e \leq 0$ to modify a BFS $\mathbf{b} = \sum x_{i_j} \mathbf{a}_{i_j}$ to get a new **nonbasic** feasible solution using $\mathbf{a}_e$, $\mathbf{b} = \sum x_{i_j} \mathbf{a}_{i_j} - \theta \mathbf{a}_e + \theta \mathbf{a}_e$. This solution is arbitrarily large.
- If for that e, $c_e > z_e$ then it is easy to prove that we can have an arbitrarily high objective.

(un)boundedness of the feasible set *and/or* of the objective.

Proof. • Let $\mathbf{I}$ be an index set and $\mathbf{x_I}$ the corresponding BFS.

- Remember that for any index, e in particular, $\mathbf{a}_e = B_\mathbf{I}\mathbf{y}_e = \sum_{j=1}^m y_{j,e}\mathbf{a}_{i_j}$.
- Let's play with $\mathbf{a}_e$: $\mathbf{b} = \sum_{j=1}^m x_{i_j}\mathbf{a}_{i_j} - \theta\mathbf{a}_e + \theta\mathbf{a}_e$.
- $\mathbf{b} = \sum_{j=1}^m \left(x_{i_j} - \theta y_{j,e}\right)\mathbf{a}_{i_j} + \theta\mathbf{a}_e$
- Since $y_{j,e} \leq 0$ is negative we have a **nonbasic** & **feasible** solution with $m+1$ nonzero variables.
- θ can be set arbitrarily large: $\mathbf{x_I} + \theta\mathbf{a}_e$ is feasible $\Rightarrow$ **unboundedness**.
- If moreover $c_e > z_e$ then writing $\hat{z}$ for the objective of the point above,

$$\begin{aligned}\hat{z} &= \textstyle\sum_{j=1}^m (x_{i_j} - \theta y_{j,e})c_{i_j} + \theta c_e,\\ &= \textstyle\sum_{j=1}^m x_{i_j}c_{i_j} - \theta\sum_{j=1}^m y_{j,e}c_{i_j} + \theta c_e,\\ &= c_\mathbf{I}^T\mathbf{x_I} - \theta c_\mathbf{I}^T\mathbf{y}_e + \theta c_e = z - \theta z_e + \theta c_e,\\ &= z + \theta(c_e - z_e).\end{aligned}$$

■

A simple example

An example

- Let's consider the following example:

$$A = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 1 & 0 & 0 & 1 \end{bmatrix}, \mathbf{c} = \left[\begin{smallmatrix} 2 \\ 5 \\ 6 \\ 8 \end{smallmatrix} \right], \mathbf{b} = \left[\begin{smallmatrix} 5 \\ 2 \end{smallmatrix} \right].$$

- Let us choose the starting $\mathbf{I}$ as $(1, 4)$. $B_{\mathbf{I}} = \left[\begin{smallmatrix} 1 & 4 \\ 1 & 1 \end{smallmatrix} \right]$, and we check easily that $\mathbf{x}_{\mathbf{I}} = \left[\begin{smallmatrix} 1 \\ 1 \end{smallmatrix} \right]$ which is feasible (lucky here) with objective

$$z = c_{\mathbf{I}}^T x_{\mathbf{I}} = \left[\begin{smallmatrix} 2 & 8 \end{smallmatrix} \right] \left[\begin{smallmatrix} 1 \\ 1 \end{smallmatrix} \right] = 10.$$

An example: 4 out, 2 in

- Here $B_{\mathbf{I}}^{-1} = \frac{1}{3}\left[\begin{smallmatrix} -1 & 4 \\ 1 & -1 \end{smallmatrix}\right]$ the y_{ij} are given by $B_{\mathbf{I}}^{-1}A = \begin{bmatrix} 1 & -\frac{2}{3} & -1 & 0 \\ 0 & \frac{2}{3} & 1 & 1 \end{bmatrix}$, namely

$$\mathbf{y}_1 = \left[\begin{smallmatrix} 1 \\ 0 \end{smallmatrix}\right], \mathbf{y}_2 = \begin{bmatrix} -\frac{2}{3} \\ \frac{2}{3} \end{bmatrix}, \mathbf{y}_3 = \left[\begin{smallmatrix} -1 \\ 1 \end{smallmatrix}\right], \mathbf{y}_4 = \left[\begin{smallmatrix} 0 \\ 1 \end{smallmatrix}\right]$$

- Hence, $z_2 = \left[\begin{smallmatrix} 2 & 8 \end{smallmatrix}\right] \begin{bmatrix} -\frac{2}{3} \\ \frac{2}{3} \end{bmatrix} = 4, z_3 = \left[\begin{smallmatrix} 2 & 8 \end{smallmatrix}\right]\left[\begin{smallmatrix} -1 \\ 1 \end{smallmatrix}\right] = 6.$
- Because $\mathbf{I} = [1, 4]$, we know $z_1 - c_1 = z_4 - c_4 = 0.$
- We have $c_2 - z_2 = \mathbf{1}$; $\; c_3 - z_3 = 0$ so only one choice for e, that is 2.
- We check $\mathbf{y}_2$ and see that y_{22} is the only positive entry. Hence we remove the second index of $\mathbf{I}$, $i_2 = 4$. $\mathbf{I}' = (1, 2)$ and $B_{\mathbf{I}'} = \left[\begin{smallmatrix} 1 & 2 \\ 1 & 0 \end{smallmatrix}\right]$
- The corresponding basic solution is $\mathbf{x}_{\mathbf{I}'} = \begin{bmatrix} 2 \\ \frac{3}{2} \end{bmatrix}$, **feasible as expected**.
- The objective is now $z' = \left[\begin{smallmatrix} 2 & 5 \end{smallmatrix}\right] \begin{bmatrix} 2 \\ \frac{3}{2} \end{bmatrix} = 11.5 > z$, **better, as expected**.

An example: that's it

- Since $B_{\mathbf{I}'}^{-1} = \frac{1}{2}\left[\begin{smallmatrix} 0 & 2 \\ 1 & -1 \end{smallmatrix}\right]$ the new coefficients y'_{ij} in

$$B_{\mathbf{I}'}^{-1}A == \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & \frac{3}{2} & \frac{3}{2} \end{bmatrix}$$

  are given by

$$\mathbf{y}'_1 = \left[\begin{smallmatrix} 1 \\ 0 \end{smallmatrix}\right],\ \mathbf{y}'_2 = \left[\begin{smallmatrix} 0 \\ 1 \end{smallmatrix}\right],\ \mathbf{y}'_3 = \left[\begin{smallmatrix} 0 \\ 3/2 \end{smallmatrix}\right],\ \mathbf{y}'_4 = \left[\begin{smallmatrix} 1 \\ 3/2 \end{smallmatrix}\right],$$

- Now $c_3 - z_3 = 6 - \left[\begin{smallmatrix} 2 & 5 \end{smallmatrix}\right] \begin{bmatrix} 0 \\ \frac{3}{2} \end{bmatrix} = \mathbf{-1.5}$ and $c_4 - z_4 = 8 - \left[\begin{smallmatrix} 2 & 5 \end{smallmatrix}\right] \begin{bmatrix} 1 \\ \frac{3}{2} \end{bmatrix} = \mathbf{-1.5}$.
- since all $c_j - z_j \geq 0$, the set of indices $1, 2$ is optimal.
- The solution is $\mathbf{x}^\star = \begin{bmatrix} 2 \\ \frac{3}{2} \\ 0 \\ 0 \end{bmatrix}$.

Nice algorithm but...

Issues with the previous example

- Clean mathematically, but **very heavy notation-wise**.
- **Worse**: lots of **redundant** computations: we only change one column from $B_{\mathbf{I}}$ to $B_{\mathbf{I}'}$ but always recompute at each iteration:
  - the inverse $B_{\mathbf{I}}^{-1}$,
  - the $\mathbf{y}_i$'s, that is the matrix $Y = B_{\mathbf{I}}^{-1}A$,
  - the z_i's which can be found through $c_{\mathbf{I}}^T Y = c_{\mathbf{I}}^T B_{\mathbf{I}}^{-1} A$ and the reduced costs.
- **Plus** we *assumed* we had an initial feasible solution immediately... what if?
- Imagine someone solves the problem $(\mathbf{c}, A, \mathbf{b})$ before us and finds $\mathbf{x}^\star$ as the optimal solution such that $\mathbf{c}^T\mathbf{x}^\star = z^\star$.
- He gives it back to us adding the constraint $\mathbf{c}^T\mathbf{x} \geq z^\star$. Finding an initial feasible solution is as hard as finding **the optimal solution** itself!

Next time

- For all these reasons, we look for a
  - compact (less redundant variables and notations),
  - fast computationally (rank one updates),

  methodology: the tableaux and dictionaries methods to go through the simplex step by step.
- We also study how to find an **initial** BFS and address additional issues.

- **YET** The simplex is **not** just a dictionary or a tableau method.
- The latter are **tools**. The **simplex algorithm is 100% algebraic and combinatorial**.
- The truth is that it is just an "optimization tool in disguise".