linear programs

\[ \begin{align*}
\text{max } & \quad c^T x \\
\text{st } & \quad Ax = b \\
& \quad Mx \leq \nu
\end{align*} \]

Notice: cases:

- Domain is always convex
  (i.e. if \( x_1, x_2 \) are in, then \( tx_1 + (1-t)x_2 \) is in) for \( 0 \leq t \leq 1 \)
  - called the Feasible Set
  - could be
    - empty (then it's a polytope)
    - compact
    - a cone
    - non-existent (empty feasible set)
    - a k-face of the feasible set \( (k \text{ usually zero}) \)
    - infinite
Linear programs
We can always convert to (equational form).

\[ \text{max } c^T x \]
\[ \text{st. } A x = b \]
\[ x \geq 0 \]

By:
- introduce a slack for each inequality,
- mult by -1 if necessary.

If we have \( x_i \) which is unbounded, then replace with:

\[ x_u - x_v \quad x_u \geq 0 \quad x_v \geq 0 \]
The Simplex Method:

- Assume there is at least one \( x \) s.t. \( Ax = b \).

Notice geoem of standard form
- Take the orthant, slice w/ linear space

2D:

3D:
All this means that there must be basic solutions, where "enough" of the vars are zero.

\[ \begin{pmatrix} m \to \mathbf{A} \downarrow \smash{\mathbf{x}} = \mathbf{b} \end{pmatrix} \]

Basic solution is an \( m \)-element set of \( \overline{\text{indices}} \) \( \mathbf{B} \) st. \( \mathbf{A}_\mathbf{B} = \{ \text{cols of } \mathbf{A} \text{ indexed by } \mathbf{B} \} \) has full rank.

\[ x_j = 0, \quad j \in \mathbf{B} \]

Notice if we know \( \mathbf{B} \), we know \( \mathbf{x} \) is a feasible point (non-singular \( \mathbf{A}_\mathbf{B} \)) if there is a feasible point, and if there is a feasible point with an objective function bounded above, there is an optimal solution. If there is an optimal solution, then there is a basic optimal solution.
Simplex example (after Matousek + Gartner)

\[
\begin{align*}
\text{Max} & \quad x_1 + x_2 \\
- x_1 + x_2 & \leq 1 \\
x_1 & \leq 3 \\
x_2 & \leq 2 \\
x_1, x_2 & > 0
\end{align*}
\]

to get standard form, introduce slacks \(x_3, x_4, x_5\)

\[
\begin{align*}
\text{Max} & \quad x_1 + x_2 \\
- x_1 + x_2 + x_3 & = 1 \\
x_1 + x_4 & = 3 \\
x_2 + x_6 & = 2 \\
\text{all} & \quad x \geq 0
\end{align*}
\]
Represent with Tableaux:

\[ x_3 = 1 + x_1 - x_2 \]
\[ x_4 = 3 - x_1 \]
\[ x_5 = 2 - x_2 \]

Interpret as \((0, 0, 1, 3, 2)\).

Notice: \(\overline{0}\) - feasible

Q: Can we improve?

2 = x_1 + x_2

Currently, 0

Notice \(x_1\), strategy: choose a variable that is 0 and one that is non-zero and swap:

In this case, notice that \(x_2\) can get bigger and this would make 2 bigger

\(\Rightarrow\) swap \(x_3, x_2\)
New Tableau:

\[ x_2 = 1 + x_1 - x_3 \]
\[ x_4 = 3 - x_1 \]
\[ x_5 = 2 - x_2 = 2 - (1 + x_1 - x_3) = 1 - x_1 + x_3 \]

Interpret as \((0, 1, 0, 3, 1)\)

and \[ z = x_1 + x_2 = x_1 + (1 + x_1 - x_3) = 2 + 2x_1 - x_3 \]

Value is 1 (so it gets better)

 Notice: to increase \(z\), we need to increase \(x_1\), but \(x_1\) can't get bigger than 1 (cause \(z = 1 - x_1 + x_3 \geq 0\).

\(x_1\) goes in, \(x_5\) goes out.

Subs \(x_1 = 1 + x_3 - x_5\) to get

\[ x_1 = 1 + x_3 - x_5 \]
\[ x_2 = 2 - x_5 \]
\[ x_4 = 2 - x_3 + x_5 \]

\[ z = 3 + x_3 - 2x_5 \]
Notice

- Increasing $x_3$ will inc. $z$
- It can go to 2

$\Rightarrow x_3$ goes in, $x_4$ goes out.

$x_1 = 3 - x_4$
$x_2 = 2 - x_5$
$x_3 = 2 - x_4 + x_5$
$z = 5 - x_4 - x_5$

No more makes $z$ better
So stop

$(3, 2, 2, 0, 0)$

$(x: \pi_0)$
Notice we are moving along 1-faces of feasible polytope.

i.e. \[ \text{tableau} = \underbrace{0 \text{-face}}_{\text{vertex}} \]

1 eqn goes out, another goes in = move to another 0-face connected to last 0-face by a 1-face.

and we do this along edges such that \[ c \cdot (\text{edge vector}) > 0 \]

**Killer algorithmic question:**

- Who goes in, who goes out?

This turns out to be complicated as methods can cycle.
cycling

- all available o-verts might have
  same value of objective.

- if this happens, we could
  come back to where we came
  from.

Various rules for pivoting:

- largest coefficient in objective
- largest increase in objective
- steepest edge

\[
\frac{c^T (x_{new} - x_{old})}{\| x_{new} - x_{old} \|}
\]

is best

V. good: approx this test.

random

(best provable bounds)

Bland

No cycling | incoming var has smallest
           | ditto outgoing, if choice
Very painful [practical] problem

→ Simplex is super-good on large problems

→ no pivot rule known can be proved polynomial

→ existence of a polynomial pivot rule is a major open qu. in geometry

One can prove various average case results (Spielman + Teng, etc).