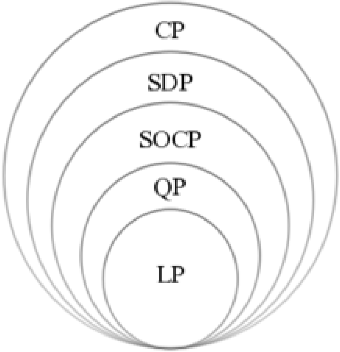
Optimization in Machine Learning

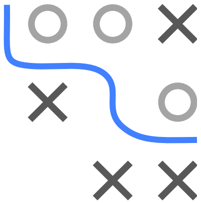
Linear Programming



Learning goals

- Definition and different forms of an LP
- Geometric intuition of LPs
- Chracteristics of vertices
- Simplex algorithms

LINEAR PROGRAMMING



Linear program (LP):

optimization problem with **linear** objective function + **linear** constraints

General form

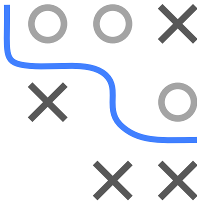
$$
\begin{aligned}
\min_{\mathbf{x}} \quad & \mathbf{c}^\top \mathbf{x} \\
\text{s.t.} \quad & \mathbf{A}_1 \mathbf{x} \leq \mathbf{b}_1 \\
& \mathbf{A}_2 \mathbf{x} \geq \mathbf{b}_2 \\
& \mathbf{A}_3 \mathbf{x} = \mathbf{b}_3
\end{aligned}
$$

Examples:

$$
\begin{aligned}
\min_{\mathbf{x}} \quad & \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_1 \Leftrightarrow \\
\min_{\mathbf{x},\mathbf{s}} \quad & \mathbf{1}^\top \mathbf{s} \\
\text{s.t.} \quad & \mathbf{A}\mathbf{x} - \mathbf{b} \leq \mathbf{s} \\
& \mathbf{A}\mathbf{x} - \mathbf{b} \geq -\mathbf{s}
\end{aligned}
$$

$$
\begin{aligned}
\min_{\mathbf{x}} \quad & \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_\infty \Leftrightarrow \\
\min_{\mathbf{x},t} \quad & t \\
\text{s.t.} \quad & \mathbf{A}\mathbf{x} - \mathbf{b} \leq t\mathbf{1} \\
& \mathbf{A}\mathbf{x} - \mathbf{b} \geq -t\mathbf{1}
\end{aligned}
$$

GEOMETRIC INTERPRETATION



Feasible set:

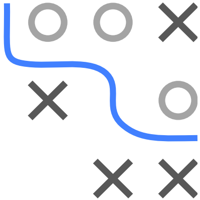
- Points $\{\mathbf{x} : \mathbf{A}_i^\top \mathbf{x} = b_i\}$ form a hyperplane in $\mathbb{R}^n$.
  $\mathbf{A}_i$ is perpendicular to the hyperplane and called **normal vector**.

- Points $\{\mathbf{x} : \mathbf{A}_i^\top \mathbf{x} \leq b_i\}$ lie on one side of the hyperplane, which form a convex half-space.

- Points satisfying **all** inequalities form a **convex polytope**.
  The intersection of convex sets is still convex, also easy to prove using definition of convex set.

Polytope $\{\mathbf{x} : \mathbf{A}\mathbf{x} \leq \mathbf{b},\ \mathbf{A} \in \mathbb{R}^{m \times n}\}$ is an n-**simplex**, i.e., convex hull of $n + 1$ **affinely independent** points, which we call vertices.

VERTICES



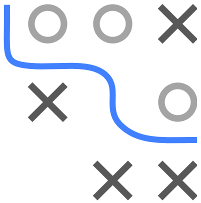
We assume that the rows of $\mathbf{A} \in \mathbb{R}^{m \times n}$ are linearly independent and $m \leq n$ to form a bounded unempty feasible set.

$\mathbf{A}\mathbf{x} = \mathbf{b}$ imposes m equality constraints:

- Each equality constraint reduces the dimension of the feasible set by 1.
- Starting with n-dimensional space, applying m independent equality constraints leaves a solution space of dimension $n - m$.

$\mathbf{x} \geq \mathbf{0}$ imposes n non-negativity.

SIMPLEX ALGORITHMS



The **simplex algorithm** solves linear programs by moving from vertex to vertex of the feasible set, and produces an optimal vertex.

It operates on equality-form linear programs $\mathbf{Ax} = \mathbf{b}, \mathbf{x} \geq \mathbf{0}$. We still assume that the rows of $\mathbf{A} \in \mathbb{R}^{m \times n}$ are linearly independent and $m \leq n$.

The method is guaranteed to arrive at an optimal solution so long as the linear program is feasible and bounded.

The simplex algorithm operates in two phases:

- **Initialization** phase: identifies a vertex partition.
- **Optimization** phase: transitions between vertex partitions toward a partition corresponding to an optimal vertex.

EXAMLPES

$$\mathbf{A}_{\mathcal{V}} = \begin{pmatrix} 2 & 1 & 1 & 0 \\ -4 & 2 & 0 & 1 \end{pmatrix}, \mathbf{b} = (9, 2)^{\top}, \mathbf{c} = (3, -1, 0, 0)^{\top}$$

Solution: $\mathcal{V} = \{1, 2\}, \mathcal{B} = \{3, 4\}$

$\mathbf{x}_{\mathcal{B}} = \mathbf{A}_{\mathcal{B}}^{-1}\mathbf{b} = (9, 2)^{\top}$

$\boldsymbol{\lambda} = \mathbf{A}_{\mathcal{B}}^{-1}\mathbf{c}_{\mathcal{B}} = \mathbf{0}$

$\boldsymbol{\mu}_{\mathcal{V}} = \mathbf{c}_{\mathcal{V}} - \left(\mathbf{A}_{\mathcal{B}}^{-1}\mathbf{A}_{\mathcal{V}}\right)^{\top}\mathbf{c}_{\mathcal{B}} = (3, -1)^{\top}$

$\boldsymbol{\mu}_{\mathcal{V}}$ contains negative elements, so our current $\mathcal{B}$ is suboptimal.

We will pivot on the negative one with $q = 2, -\mathbf{A}_{\mathcal{B}}^{-1}\mathbf{A}_{\{q\}} = (1, 2)^{\top}$.

This causes $x_4 = 0$, so updated $\mathcal{B} = \{2, 3\}$.

In the second iteration, we find

$\mathbf{x}_{\mathcal{B}} = (1, 8)^{\top}, \boldsymbol{\lambda} = (0, -\frac{1}{2})^{\top}, \boldsymbol{\mu}_{\mathcal{V}} = (1, \frac{1}{2})^{\top}$.

This is optimal with no negative entry, thus we have $x^* = (0, 1, 8, 0)^{\top}$