GEMS OF TCS

LINEAR PROGRAMMING

Sasha Golovnev
September 25, 2023
Linear Programming

- Optimization problems: among all solutions satisfying certain constraints find optimal one.
Linear Programming

- Optimization problems: among all solutions satisfying certain constraints find optimal one
- Find shortest cycle through all vertices
LINEAR PROGRAMMING

- Optimization problems: among all solutions satisfying certain constraints find optimal one
- Find shortest cycle through all vertices
- Find optimal coloring
LINEAR PROGRAMMING

- Optimization problems: among all solutions satisfying certain constraints find optimal one
- Find shortest cycle through all vertices
- Find optimal coloring
- Find minimum vertex cover
LINEAR PROGRAMMING

- Optimization problems: among all solutions satisfying certain constraints find optimal one
- Find shortest cycle through all vertices
- Find optimal coloring
- Find minimum vertex cover
- Linear programming: class of optimization problems where constraints and optimization criterion are linear functions
Avoiding Scurvy
• Orange costs $1, grapefruit costs $1; we have budget of $2/day
• Orange costs $1, grapefruit costs $1; we have budget of $2/day

• Orange weighs 100gm, grapefruit weighs 200gm, we can carry 300gm
• Orange costs $1, grapefruit costs $1; we have budget of $2/day

• Orange weighs 100gm, grapefruit weighs 200gm, we can carry 300gm

• Orange has 50gm of vitamin C, grapefruit has 75gm of vitamin C, maximize daily vitamin C intake.
AVOIDING SCURVY. PLOT

\[ \text{max} \ 2x + 3y \]

\[ x + y \leq 2 \]
\[ x + 2y \leq 3 \]
\[ x \geq 0 \]
\[ y \geq 0 \]
Avoiding Scurvy. Plot

\[
\begin{align*}
\text{max } & \quad 2x + 3y \\
\text{subject to } & \quad x + y \leq 2 \\
& \quad x + 2y \leq 3 \\
& \quad x \geq 0 \\
& \quad y \geq 0
\end{align*}
\]
AVOIDING SCURVY. PLOT

\[
\max 2x + 3y
\]

\[
x + y \leq 2
\]
\[
x + 2y \leq 3
\]
\[
x \geq 0
\]
\[
y \geq 0
\]
AVOIDING SCURVY. PLOT

\[
\begin{align*}
\text{max } & 2x + 3y \\
x + y & \leq 2 \\
x + 2y & \leq 3 \\
x & \geq 0 \\
y & \geq 0
\end{align*}
\]
max $2x + 3y$

$x + y \leq 2$
$x + 2y \leq 3$
$x \geq 0$
$y \geq 0$
AVOIDING SCURVY. PLOT

\[
\begin{align*}
\text{max } & \quad 2x + 3y \\
\text{s.t. } & \quad x + y \leq 2 \\
& \quad x + 2y \leq 3 \\
& \quad x \geq 0 \\
& \quad y \geq 0
\end{align*}
\]
AVOIDING SCURVY. PLOT

\[
\begin{align*}
\text{max} & \quad 2x + 3y \\
x + y & \leq 2 \\
x + 2y & \leq 3 \\
x & \geq 0 \\
y & \geq 0
\end{align*}
\]
AVOIDING SCURVY. PLOT

\[ \begin{align*}
\text{max} & \quad 2x + 3y \\
x + y & \leq 2 \\
x + 2y & \leq 3 \\
x & \geq 0 \\
y & \geq 0
\end{align*} \]
AVOIDING SCURVY. PLOT

\[
\begin{align*}
\max & \quad 2x + 3y \\
\text{s.t.} & \quad x + y \leq 2 \\
& \quad x + 2y \leq 3 \\
& \quad x \geq 0 \\
& \quad y \geq 0
\end{align*}
\]
\[ \text{AVOIDING SCURVY. PLOT} \]

\[ \begin{align*}
\text{max } & 2x + 3y \\
& x + y \leq 2 \\
& x + 2y \leq 3 \\
& x \geq 0 \\
& y \geq 0
\end{align*} \]
AVOIDING SCURVY. PLOT

\[
\begin{align*}
\max & \quad 2x + 3y \\
\text{subject to} & \quad x + y \leq 2 \\
& \quad x + 2y \leq 3 \\
& \quad x \geq 0 \\
& \quad y \geq 0
\end{align*}
\]
Profit Maximization
PROFIT MAXIMIZATION

• We have 6 machines and 20 workers
Profit Maximization

- We have 6 machines and 20 workers
- A machine takes two workers to operate
Profit Maximization

- We have 6 machines and 20 workers
- A machine takes two workers to operate
- Each machine produces 20 chocolates/hour, each worker produces 5 chocolates/hour

Each chocolate costs $10, each worker gets $40 per hour.
Profit Maximization

- We have 6 machines and 20 workers
- A machine takes two workers to operate
- Each machine produces 20 chocolates/hour, each worker produces 5 chocolates/hour
- We need to produce at most 100 chocolates/hour
Profit Maximization

- We have 6 machines and 20 workers
- A machine takes two workers to operate
- Each machine produces 20 chocolates/hour, each worker produces 5 chocolates/hour
- We need to produce at most 100 chocolates/hour
- Each chocolate costs $10, each worker gets $40 per hour
WORKERS AND MACHINES
TWO WORKERS OPERATE A MACHINE
Linear Classifier
Linear Classifier

- Given $n_1$ spam emails, and $n_2$ ham emails as points in $\mathbb{R}^d$
Linear Classifier

• Given $n_1$ spam emails, and $n_2$ ham emails as points in $\mathbb{R}^d$

• Find a linear function $h(a_1, \ldots, a_d)$ s.t.
Linear Classifier

- Given $n_1$ spam emails, and $n_2$ ham emails as points in $\mathbb{R}^d$

- Find a linear function $h(a_1, \ldots, a_d)$ s.t.
  - $h(a_1, \ldots, a_d) < 0$ for all spam emails
  - $h(a_1, \ldots, a_d) > 0$ for all ham emails
Linear Programming
Linear Programming

• Find real numbers $x_1, \ldots, x_n$ that satisfy linear constraints

$$a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n \geq b_1$$
$$a_{21}x_1 + a_{22}x_2 + \ldots + a_{2n}x_n \geq b_2$$

$$\ldots$$

$$a_{m1}x_1 + a_{m2}x_2 + \ldots + a_{mn}x_n \geq b_m$$
LINEAR PROGRAMMING

• Find real numbers $x_1, \ldots, x_n$ that satisfy linear constraints

\[
\begin{align*}
    a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n & \geq b_1 \\
    a_{21}x_1 + a_{22}x_2 + \ldots + a_{2n}x_n & \geq b_2 \\
    \vdots \\
    a_{m1}x_1 + a_{m2}x_2 + \ldots + a_{mn}x_n & \geq b_m
\end{align*}
\]

• So that linear objective is maximized

\[
    c_1x_1 + c_2x_2 + \ldots + c_nx_n
\]
EQUIVALENT FORMULATIONS

• Turn minimization problem into maximization problem:

\[
\min \quad c_1 x_1 + c_2 x_2 + \ldots - c_n x_n
\]
EQUIVALENT FORMULATIONS

• Turn minimization problem into maximization problem:

\[
\begin{align*}
\text{min} & \quad c_1 x_1 + c_2 x_2 + \ldots - c_n x_n \\
\text{max} & \quad -c_1 x_1 - c_2 x_2 - \ldots - c_n x_n
\end{align*}
\]
**Equivalent Formulations**

- Turn **minimization** problem into **maximization** problem:

  \[
  \begin{align*}
  \min & \quad c_1 x_1 + c_2 x_2 + \ldots - c_n x_n \\
  \max & \quad -c_1 x_1 - c_2 x_2 - \ldots - c_n x_n
  \end{align*}
  \]

- Turn \( \leq \) into \( \geq \):

  \[
  a_{11} x_1 + a_{12} x_2 + \ldots + a_{1n} x_n \leq b_1
  \]
EQUIVALENT FORMULATIONS

• Turn minimization problem into maximization problem:

\[
\begin{align*}
\min & \quad c_1x_1 + c_2x_2 + \ldots - c_nx_n \\
\max & \quad -c_1x_1 - c_2x_2 - \ldots - c_nx_n
\end{align*}
\]

• Turn \( \leq \) into \( \geq \):

\[
\begin{align*}
a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n & \leq b_1 \\
- a_{11}x_1 - a_{12}x_2 - \ldots - a_{1n}x_n & \geq -b_1
\end{align*}
\]
• Turn $=$ into $\geq$:

$$a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n = b_1$$
EQUIVALENT FORMULATIONS

• Turn $=$ into $\geq$:

$$a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n = b_1$$

$$a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n \geq b_1$$

$$-a_{11}x_1 - a_{12}x_2 - \ldots - a_{1n}x_n \geq -b_1$$
Input is a matrix $A \in \mathbb{R}^{m \times n}$, and vectors $b \in \mathbb{R}^m$ and $c \in \mathbb{R}^n$
Input is a matrix $A \in \mathbb{R}^{m \times n}$, and vectors $b \in \mathbb{R}^m$ and $c \in \mathbb{R}^n$. 

\[
Ax = \begin{bmatrix}
a_{11} & \cdots & a_{1n} \\
\vdots & \ddots & \vdots \\
a_{m1} & \cdots & a_{mn}
\end{bmatrix}
\begin{bmatrix}
x_1 \\
x_n
\end{bmatrix}
= \begin{bmatrix}
a_{11}x_1 + \cdots + a_{1n}x_n \\
\vdots \\
a_{m1}x_1 + \cdots + a_{mn}x_n
\end{bmatrix} \geq \begin{bmatrix}
b_1 \\
\vdots \\
b_m
\end{bmatrix}
\]
Input is a matrix $A \in \mathbb{R}^{m \times n}$, and vectors $b \in \mathbb{R}^m$ and $c \in \mathbb{R}^n$

$$Ax = \begin{bmatrix} a_{11} & \ldots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \ldots & a_{mn} \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} a_{11}x_1 + \ldots + a_{1n}x_n \\ \vdots \\ a_{m1}x_1 + \ldots + a_{mn}x_n \end{bmatrix} \geq \begin{bmatrix} b_1 \\ \vdots \\ b_m \end{bmatrix}$$

$Ax \geq b$
**Matrix Formulation**

Input is a matrix \( A \in \mathbb{R}^{m \times n} \), and vectors \( b \in \mathbb{R}^m \) and \( c \in \mathbb{R}^n \)

\[
Ax = \begin{bmatrix}
  a_{11} & \cdots & a_{1n} \\
  \vdots & \ddots & \vdots \\
  a_{m1} & \cdots & a_{mn}
\end{bmatrix}
\begin{bmatrix}
  x_1 \\
  \vdots \\
  x_n
\end{bmatrix}
= \begin{bmatrix}
  a_{11}x_1 + \cdots + a_{1n}x_n \\
  \vdots \\
  a_{m1}x_1 + \cdots + a_{mn}x_n
\end{bmatrix} \geq \begin{bmatrix}
  b_1 \\
  \vdots \\
  b_m
\end{bmatrix}
\]

\[Ax \geq b\]

maximize \( cx = \begin{bmatrix}
  c_1 & \cdots & c_n
\end{bmatrix}
\begin{bmatrix}
  x_1 \\
  \vdots \\
  x_n
\end{bmatrix} = c_1x_1 + \cdots + c_nx_n\]
HISTORY OF LINEAR PROGRAMMING

• Kantorovich, 1939, started studying Linear Programming

• Dantzig, 1947, developed Simplex Method for US Air force planning problems

• Koopmans, 1947, showed how to use LP for analysis of economic theories

• Kantorovich and Koopmans won Nobel Prize in Economics in 1971

• Dantzig's algorithm is "One of top 10 algorithms of the 20th century"
HISTORY OF LINEAR PROGRAMMING

• Kantorovich, 1939, started studying Linear Programming
• Dantzig, 1947, developed Simplex Method for US Air force planning problems
HISTORY OF LINEAR PROGRAMMING

• Kantorovich, 1939, started studying Linear Programming
• Dantzig, 1947, developed Simplex Method for US Air force planning problems
• Koopmans, 1947, showed how to use LP for analysis of economic theories

Kantorovich and Koopmans won Nobel Prize in Economics in 1971

Dantzig's algorithm is "One of top 10 algorithms of the 20th century"
HISTORY OF LINEAR PROGRAMMING

- Kantorovich, 1939, started studying Linear Programming
- Dantzig, 1947, developed Simplex Method for US Air force planning problems
- Koopmans, 1947, showed how to use LP for analysis of economic theories
- Kantorovich and Koopmans won Nobel Prize in Economics in 1971
HISTORY OF LINEAR PROGRAMMING

• Kantorovich, 1939, started studying Linear Programming
• Dantzig, 1947, developed Simplex Method for US Air force planning problems
• Koopmans, 1947, showed how to use LP for analysis of economic theories
• Kantorovich and Koopmans won Nobel Prize in Economics in 1971
• Dantzig’s algorithm is “One of top 10 algorithms of the 20th century”
### Theorem

A linear function takes its maximum and minimum values on vertices
Theorem

A linear function takes its maximum and minimum values on vertices

- Start at any vertex
Theorem
A linear function takes its maximum and minimum values on vertices

- Start at any vertex
- While there is an adjacent vertex with higher profit
  - Move to that vertex
CORNER CASES

• No solutions
CORNER CASES

• No solutions

• Unbounded profit
ALGORITHMS FOR LINEAR PROGRAMMING

- Simplex method
ALGORITHMS FOR LINEAR PROGRAMMING

• Simplex method

• Many professional packages that implement efficient algorithms for LP
ALGORITHMS FOR LINEAR PROGRAMMING

• Simplex method

• Many professional packages that implement efficient algorithms for LP

• Ellipsoid method
Algorithms for Linear Programming

- Simplex method

- Many professional packages that implement efficient algorithms for LP

- Ellipsoidal method

- Projective algorithm
ALGORITHMS FOR LINEAR PROGRAMMING

• Simplex method
• Many professional packages that implement efficient algorithms for LP
• Ellipsoid method
• Projective algorithm
• Recent results!
ELLIPSOID METHOD