Lineare Optimierung

Erweiterte Vorlesung

BSc Analysis

Inhaltsverzeichnis

Lernziele

Am Ende dieser Lektion koennen Sie:

1. Ein **lineares Programm** in Standardform formulieren und die Bestandteile (Zielfunktion, Nebenbedingungen, Entscheidungsvariablen) benennen.
2. Den **zulaessigen Bereich** geometrisch interpretieren und den **Eckpunktsatz** erklaeren.
3. Ein 2D-LP **graphisch loesen** und das Optimum an Eckpunkten bestimmen.
4. Das **Simplex-Verfahren** tabellarisch durchfuehren und die Optimalloesung ablesen.
5. Den **dualen LP** aufstellen und **Schattenpreise** oekonomisch interpretieren.
6. Die **Big-M-Methode** anwenden, wenn der Ursprung nicht zulaessig ist.
7. **Sensitivitaetsanalyse** durchfuehren und das Ergebnis interpretieren.

Lineare Optimierung ist eines der am haeufigsten eingesetzten Verfahren in Wirtschaft, Logistik und Ingenieurwesen.

Was ist ein lineares Programm?

Definition

Ein **lineares Programm (LP)** besteht aus:

- einer **linearen Zielfunktion**, die maximiert oder minimiert wird,
- **linearen Nebenbedingungen** (Gleichungen oder Ungleichungen),
- **Nichtnegativitaetsbedingungen** fuer die Entscheidungsvariablen.

Allgemeine Form:

$$\underset{\mathbf{x}}{\text{maximize}} \ \mathbf{c}^T\mathbf{x}$$

$$\text{s.t.} \quad \mathbf{A}\mathbf{x} \leq \mathbf{b}, \quad \mathbf{x} \geq \mathbf{0}$$

Dabei:

- $\mathbf{x} \in \mathbb{R}^n$: Entscheidungsvariablen
- $\mathbf{c} \in \mathbb{R}^n$: Zielfunktionskoeffizienten
- $\mathbf{A} \in \mathbb{R}^{m \times n}$: Koeffizientenmatrix
- $\mathbf{b} \in \mathbb{R}^m$: rechte Seite

Beispiel (Baeckerei):

Eine Baeckerei stellt Brot (x_1) und Kuchen (x_2) her.

- Gewinn: $3x_1 + 5x_2$ Euro
- Mehl: $x_1 + 2x_2 \leq 8$ kg
- Arbeitszeit: $3x_1 + 2x_2 \leq 12$ Std.
- $x_1, x_2 \geq 0$

$\Rightarrow$ **Wie viel von jedem Produkt maximiert den Gewinn?**

Linearitaet: Zielfunktion und Nebenbedingungen sind linear in den Entscheidungsvariablen – keine Produkte $x_1 x_2$ oder Potenzen x_1^2.

Standardform eines LP

Standardform (Maximierung)

$$\underset{\mathbf{x}\in\mathbb{R}^n}{\text{maximize}} \quad z = \mathbf{c}^\mathsf{T}\mathbf{x} \qquad \text{s.t.} \quad \mathbf{A}\mathbf{x} \leq \mathbf{b}, \quad \mathbf{x} \geq \mathbf{0}$$

Unser Baeckerei-Beispiel:

$$\underset{x_1,x_2}{\text{maximize}} \quad z = 3x_1 + 5x_2$$

$$\begin{aligned} \text{s.t.} \quad x_1 + 2x_2 &\leq 8 \\ 3x_1 + 2x_2 &\leq 12 \\ x_1, x_2 &\geq 0 \end{aligned}$$

In Matrixform:

$$\mathbf{c} = \begin{pmatrix} 3 \\ 5 \end{pmatrix}, \ \mathbf{A} = \begin{pmatrix} 1 & 2 \\ 3 & 2 \end{pmatrix}, \ \mathbf{b} = \begin{pmatrix} 8 \\ 12 \end{pmatrix}$$

Umwandlungsregeln:

- Minimierung $\to$ Maximierung: minimize $f(\mathbf{x})$ = maximize $(-f(\mathbf{x}))$
- $\geq$-Nebenbedingung: mit (-1) multiplizieren
- Gleichung $= \to$ zwei Ungleichungen ($\leq$ und $\geq$)
- Variable ohne Vorzeichen-Beschraenkung: $x_j = x_j^+ - x_j^-$ mit $x_j^+, x_j^- \geq 0$

Merke: Jedes LP laesst sich in Standardform bringen.

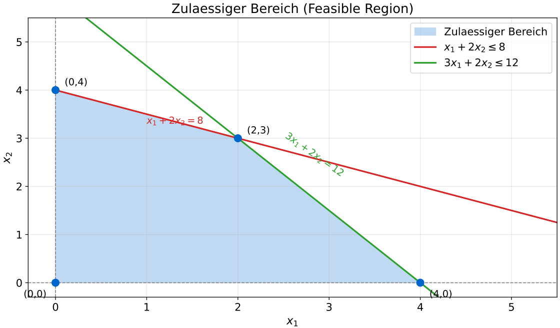
Die Standardform ist Voraussetzung fuer das Simplex-Verfahren – alle Algorithmen erwarten dieses Format.

Zulaessiger Bereich

Definition

Der **zulaessige Bereich** (engl. *feasible region*) ist die Menge aller Punkte $\mathbf{x}$, die *alle* Nebenbedingungen erfuellen:

$$\mathcal{F} = \{\mathbf{x} \in \mathbb{R}^n \mid \mathbf{A}\mathbf{x} \leq \mathbf{b}, \ \mathbf{x} \geq \mathbf{0}\}$$



Eigenschaften:

- $\mathcal{F}$ ist ein **konvexes Polyeder**
- Schnittmenge endlich vieler Halbraeume
- Jede Nebenbedingung definiert einen Halbraum
- Die Nichtnegativitaet beschraenkt auf den 1. Quadranten

Unser Beispiel:

- Halbraum 1: $x_1 + 2x_2 \leq 8$
- Halbraum 2: $3x_1 + 2x_2 \leq 12$
- Quadrant: $x_1 \geq 0, \ x_2 \geq 0$

Konvexitaet ist entscheidend: Jede Verbindungslinie zwischen zwei zulaessigen Punkten liegt komplett im zulaessigen Bereich.

Der Eckpunktsatz

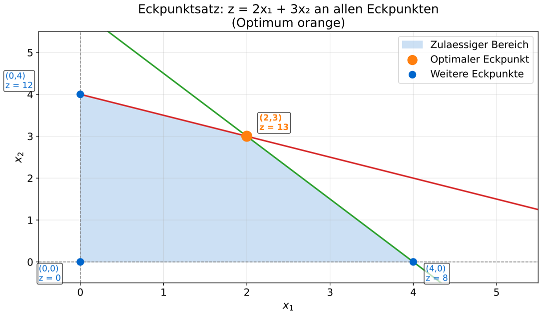
Satz (Eckpunktsatz)

Besitzt ein lineares Programm eine optimale Loesung, so gibt es eine **optimale Loesung an einem Eckpunkt** (Vertex) des zulaessigen Bereichs.

Beweisskizze:

1. Sei $\mathbf{x}^*$ optimal und *kein* Eckpunkt.
2. Dann ist $\mathbf{x}^* = \lambda\mathbf{u} + (1-\lambda)\mathbf{v}$ fuer zwei zulaessige Punkte $\mathbf{u}, \mathbf{v}$ und $0 < \lambda < 1$.
3. Da z linear: $z(\mathbf{x}^*) = \lambda z(\mathbf{u}) + (1-\lambda)z(\mathbf{v})$.
4. Mindestens einer von $z(\mathbf{u}), z(\mathbf{v})$ ist $\geq z(\mathbf{x}^*)$.
5. Durch Wiederholung gelangt man zu einem Eckpunkt.

Konsequenz:
Es genuegt, **endlich viele Eckpunkte** zu pruefen!



Eckpunkte unseres Beispiels:

- (0, 0) – Ursprung

Beispiel: Eckpunkte auswerten

Baeckerei-LP: maximize $z = 3x_1 + 5x_2$ unter den bekannten Nebenbedingungen.

Eckpunkt	x_1	x_2	$z = 3x_1 + 5x_2$
A	0	0	0
B	4	0	12
C	2	3	**21**
D	0	4	20

Berechnung von $C = (2, 3)$:

$$x_1 + 2x_2 = 8$$
$$3x_1 + 2x_2 = 12$$

Subtraktion: $2x_1 = 4 \Rightarrow x_1 = 2,\ x_2 = 3$.

Ergebnis

Optimalloesung: $x_1^* = 2,\ x_2^* = 3$
Optimaler Gewinn: $z^* = 21$ Euro
$\Rightarrow$ 2 Einheiten Brot, 3 Einheiten Kuchen.

Bei kleinen LPs mit 2 Variablen genuegt es, alle Eckpunkte zu berechnen und die Zielfunktion auszuwerten.

Kanonische Form vs Standardform

Standardform (Ungleichungen)

$$\begin{aligned} \text{maximize} \quad & \mathbf{c}^\mathsf{T}\mathbf{x} \\ \text{s.t.} \quad & \mathbf{A}\mathbf{x} \leq \mathbf{b} \\ & \mathbf{x} \geq \mathbf{0} \end{aligned}$$

- Nebenbedingungen als **Ungleichungen**
- Geeignet fuer graphische Loesung
- Direkte Formulierung aus Sachproblem

Kanonische Form (Gleichungen)

$$\begin{aligned} \text{maximize} \quad & \mathbf{c}^\mathsf{T}\mathbf{x} \\ \text{s.t.} \quad & \mathbf{A}\mathbf{x} = \mathbf{b} \\ & \mathbf{x} \geq \mathbf{0} \end{aligned}$$

- Nebenbedingungen als **Gleichungen**
- Entsteht durch Einfuegen von Schlupfvariablen
- Voraussetzung fuer den Simplex-Algorithmus

Umwandlung Standardform → Kanonische Form:

$$\mathbf{A}\mathbf{x} \leq \mathbf{b} \quad \xrightarrow{+\mathbf{s}} \quad \mathbf{A}\mathbf{x} + \mathbf{I}\mathbf{s} = \mathbf{b}, \quad \mathbf{s} \geq \mathbf{0}$$

Der erweiterte Variablenvektor ist $\tilde{\mathbf{x}} = (\mathbf{x}^\mathsf{T}, \mathbf{s}^\mathsf{T})^\mathsf{T} \in \mathbb{R}^{n+m}$.

Beide Formen sind aequivalent – die Wahl haengt vom Kontext und Algorithmus ab.

Normalform-Umwandlung: Beispiele

Verschiedene NB-Typen in die Standardform bringen:

1. **$\geq$-Ungleichung:** $2x_1 + x_2 \geq 6$

$$\xrightarrow{\times(-1)} \quad -2x_1 - x_2 \leq -6$$

2. **Gleichung:** $x_1 + x_2 = 5$

$$\xrightarrow{\text{aufteilen}} \quad x_1 + x_2 \leq 5 \quad \text{und} \quad -x_1 - x_2 \leq -5$$

3. **Minimierung:** minimize $2x_1 - 3x_2$

$$\xrightarrow{\text{negieren}} \quad \text{maximize } -2x_1 + 3x_2, \quad \text{dann } z^*_{\text{min}} = -z^*_{\text{max}}$$

4. **Freie Variable** (x_j ohne Vorzeichenbedingung):

$$x_j = x_j^+ - x_j^-, \quad x_j^+, x_j^- \geq 0$$

Zusammenfassung

Jedes LP kann durch systematische Anwendung dieser Regeln in die Standardform maximize $\mathbf{c}^T\mathbf{x}$, $\mathbf{A}\mathbf{x} \leq \mathbf{b}$, $\mathbf{x} \geq \mathbf{0}$ gebracht werden.

In der Praxis uebernehmen Solver wie Gurobi oder HiGHS diese Umwandlungen automatisch (Preprocessing).

Hoehere Dimensionen: $n > 2$ Variablen

Beispiel mit 3 Produkten:

$$
\begin{aligned}
\text{maximize} \quad & z = 3x_1 + 5x_2 + 2x_3 \\
\text{s.t.} \quad & x_1 + 2x_2 + x_3 \leq 10 \\
& 3x_1 + x_2 + 2x_3 \leq 12 \\
& x_1 + x_2 + x_3 \leq 8 \\
& x_1, x_2, x_3 \geq 0
\end{aligned}
$$

Unterschiede zu 2D:

- Zulaessiger Bereich ist ein **Polyeder im** $\mathbb{R}^3$
- Graphische Loesung nicht mehr praktikabel
- Simplex-Verfahren ist **universell** anwendbar

Allgemein fuer n Variablen, m NB:

Anzahl moeglicher Eckpunkte: bis zu $\binom{n+m}{n}$.

n	m	Max. Eckpunkte
2	2	6
3	3	20
5	5	252
10	10	184 756

Fazit: Fuer $n > 2$ ist das Absuchen aller Eckpunkte **nicht effizient** – der Simplex waehlt gezielt vielversprechende Eckpunkte.

Reale LPs haben oft Tausende von Variablen und Nebenbedingungen – nur intelligente Algorithmen sind hier machbar.

Methode der Isoprofit-Linien

Idee

Eine **Isoprofit-Linie** (Niveaulinie) verbindet alle Punkte mit gleichem Zielfunktionswert:

$$\mathbf{c}^\top \mathbf{x} = k \quad \Leftrightarrow \quad 3x_1 + 5x_2 = k$$

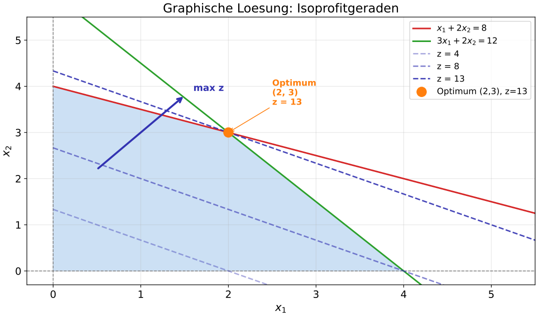
Vorgehen:

1. Zeichne den zulaessigen Bereich.
2. Zeichne eine Isoprofit-Linie fuer beliebiges k.
3. Verschiebe parallel in Richtung steigender z-Werte.
4. Der **letzte Beruehrpunkt** mit $\mathcal{F}$ ist das Optimum.

Richtung des Gradienten:

$$\nabla z = \mathbf{c} = \begin{pmatrix} 3 \\ 5 \end{pmatrix}$$

zeigt in Richtung wachsender z-Werte.



Die graphische Methode funktioniert nur fuer 2 (maximal 3) Variablen – fuer hoehere Dimensionen brauchen wir den Simplex.

Schritt 1: Nebenbedingungen als Geraden zeichnen.

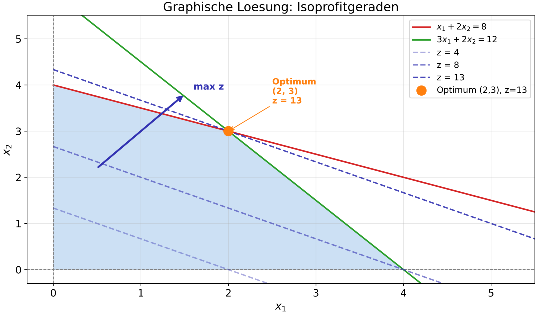
- $x_1 + 2x_2 = 8$: Achsenabschnitte $(8, 0)$ und $(0, 4)$
- $3x_1 + 2x_2 = 12$: Achsenabschnitte $(4, 0)$ und $(0, 6)$
- Zulaessig: unterhalb beider Geraden, im 1. Quadranten

Schritt 2: Isoprofit-Linien zeichnen.

- $z = 5$: $3x_1 + 5x_2 = 5$ (nah am Ursprung)
- $z = 15$: $3x_1 + 5x_2 = 15$ (mittendrin)
- $z = 21$: $3x_1 + 5x_2 = 21$ (letzter Kontakt!)

Schritt 3: Optimum ablesen.

- Letzte Isoprofit-Linie beruehrt $\mathcal{F}$ im Punkt $(2, 3)$.
- $z^* = 3 \cdot 2 + 5 \cdot 3 = 21$.



Die parallele Verschiebung der Isoprofit-Linie sichert, dass das globale Optimum gefunden wird – Konvexitaet garantiert dies.

Optimum ablesen und verifizieren

Verifikation der Loesung $(x_1^*, x_2^*) = (2, 3)$**:**

1. **Zulaessigkeit pruefen:**
   - $x_1 + 2x_2 = 2 + 6 = 8 \leq 8$ ✓
   - $3x_1 + 2x_2 = 6 + 6 = 12 \leq 12$ ✓
   - $x_1 = 2 \geq 0,\ x_2 = 3 \geq 0$ ✓
2. **Optimalitaet:** Beide Nebenbedingungen sind **aktiv** (Gleichheit) $\Rightarrow$ Schnittpunkt zweier Begrenzungsgeraden.
3. **Zielfunktionswert:** $z^* = 3 \cdot 2 + 5 \cdot 3 = 21$.

Beobachtung

Am Optimum sind beide Ressourcen **vollstaendig ausgeschoepft**: Mehl und Arbeitszeit sind knapp (binding constraints).

Oekonomische Deutung:

- 2 Einheiten Brot
- 3 Einheiten Kuchen
- Gesamtgewinn: 21 Euro
- Kein Mehl uebrig (Schlupf $= 0$)
- Keine Arbeitszeit uebrig (Schlupf $= 0$)

Frage: Was passiert, wenn wir 1 kg mehr Mehl haetten? $\rightarrow$ Schattenpreis (Dualitaet)

Aktive Nebenbedingungen (binding constraints) zeigen, welche Ressourcen den Gewinn limitieren – das fuehrt zur Dualitaet.

Sonderfaelle bei linearen Programmen

Unbeschraenktes LP

Der Zielfunktionswert kann beliebig gross werden.
$\Rightarrow$ Es gibt **keine** endliche Optimalloesung.

Beispiel: maximize x_1 mit $x_1 \geq 0, x_2 \geq 0$.
Kein oberer Halbraum begrenzt $x_1 - z \to \infty$.

Unzulaessiges LP

Die Nebenbedingungen widersprechen sich.
$\Rightarrow \mathcal{F} = \emptyset$, keine Loesung existiert.

Beispiel: $x_1 \leq 2$ und $x_1 \geq 5$ gleichzeitig.

Mehrdeutige Loesung

Unendlich viele Optima entlang einer Kante.
$\Rightarrow$ Die Isoprofit-Linie ist **parallel** zu einer aktiven Nebenbedingung.

Beispiel: maximize $x_1 + 2x_2$ mit NB $x_1 + 2x_2 \leq 8$.
Jeder Punkt auf der Kante ist optimal mit $z = 8$.

Degeneriertes LP

Ein Eckpunkt liegt im Schnitt von **mehr als** n aktiven Nebenbedingungen.
$\Rightarrow$ Kann zu Zyklen im Simplex fuehren (selten in der Praxis).

In der Praxis sind unbeschraenkte und unzulaessige LPs meist Zeichen eines Modellierungsfehlers – die Nebenbedingungen pruefen!

Erkennung

Wenn in der Pivotspalte **kein positiver Eintrag** existiert, ist das LP **unbeschraenkt**:

$$a_{ij} \leq 0 \quad \text{fuer alle } i \quad \Rightarrow \quad z \to \infty$$

Beispiel:

$$\begin{aligned} \text{maximize} \quad & z = 2x_1 + x_2 \\ \text{s.t.} \quad & -x_1 + x_2 \leq 4 \\ & x_1, x_2 \geq 0 \end{aligned}$$

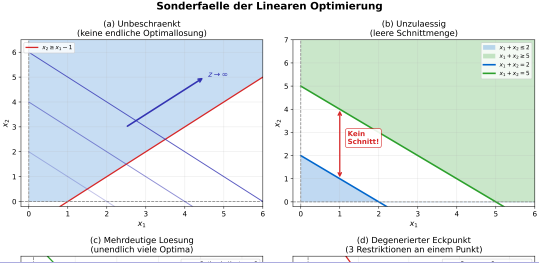
Tableau:

Basis	x_1	x_2	s_1	RHS
s_1	-1	1	1	4
z	-2	-1	0	0

Pivotspalte x_1: Eintrag $= -1 < 0$.
$\Rightarrow$ Kein positiver Quotient moeglich!

Geometrische Deutung:

Der zulaessige Bereich ist nach rechts **offen** – man kann x_1 beliebig erhoehen, ohne eine Nebenbedingung zu verletzen.



Unzulaessiges LP: Leerer zulaessiger Bereich

Erkennung

Ein LP ist **unzulaessig**, wenn die Nebenbedingungen sich widersprechen und $\mathcal{F} = \emptyset$.

Beispiel:

$$\begin{aligned} \text{maximize} \quad & z = x_1 + x_2 \\ \text{s.t.} \quad & x_1 + x_2 \leq 4 \\ & x_1 + x_2 \geq 6 \\ & x_1, x_2 \geq 0 \end{aligned}$$

Die Menge $\{x_1 + x_2 \leq 4\} \cap \{x_1 + x_2 \geq 6\} = \emptyset$.

Im Simplex:
Bei der Big-M-Methode oder Zweiphasen-Methode erkennt man Unzulaessigkeit, wenn kuenstliche Variablen **nicht** auf 0 gedrueckt werden koennen.

Haeufige Ursachen:

- Widerspruechliche Kapazitaetsgrenzen
- Mindestmengen uebersteigen Kapazitaet
- Falsche Vorzeichen in der Modellierung
- Typische Formulierungsfehler

Diagnose-Strategie:

1. Nebenbedingungen paarweise auf Widersprueche pruefen.
2. Relaxierung: Nebenbedingungen einzeln entfernen, bis zulaessig.
3. Die entfernte NB ist die problematische.

Software-Hilfe:
Solver wie Gurobi liefern ein *Irreducible Infeasible Subsystem* (IIS) – die minimale Menge widerspruechlicher NB.

Unzulaessigkeit ist kein algorithmisches, sondern ein Modellierungsproblem – ueberpruefen Sie Ihre Annahmen sorgfaeltig.

Degeneriertes LP

Definition

Eine Basisloesung ist **degeneriert**, wenn mindestens eine Basisvariable den Wert 0 hat (d. h. der zugehoerige RHS-Eintrag ist 0).

Beispiel:

$$\begin{aligned} \text{maximize} \quad & z = x_1 + x_2 \\ \text{s.t.} \quad & x_1 \leq 2 \\ & x_2 \leq 3 \\ & x_1 + x_2 \leq 5 \\ & x_1, x_2 \geq 0 \end{aligned}$$

Am Eckpunkt $(2, 3)$: Drei aktive NB bei $n = 2$ Variablen.

Tableau-Erkennung:
Mindestens ein RHS-Eintrag $= 0$ in einer Basiszeile.

Konsequenzen fuer den Simplex:

- Ein Pivotschritt kann z **nicht** verbessern (Seitensprung: Basiswechsel ohne z-Aenderung).
- Theoretisch moeglich: **Zykeln** (Simplex besucht Basen wiederholt).
- In der Praxis extrem selten.

Gegenstrategien:

- **Bland-Regel:** Waehle stets die Variable mit kleinstem Index $\Rightarrow$ garantiert keine Zyklen.
- **Perturbation:** Kleine Stoerung der rechten Seite.
- **Lexikographische Regel.**

Degenerierung ist geometrisch: Ein Eckpunkt hat "zu viele" aktive NB – algebraisch: eine Basisvariable ist Null.

Mehrdeutige Loesung: Alternative Optima

Erkennung im Optimaltableau

Alternative Optima existieren, wenn im Optimaltableau eine **Nichtbasisvariable** reduzierte Kosten $\bar{c}_j = 0$ hat.

Beispiel:

$$\begin{aligned} \text{maximize} \quad & z = 2x_1 + 4x_2 \\ \text{s.t.} \quad & x_1 + 2x_2 \leq 8 \\ & 3x_1 + 2x_2 \leq 12 \\ & x_1, x_2 \geq 0 \end{aligned}$$

Die Isoprofit-Linie $2x_1 + 4x_2 = k$ ist **parallel** zur NB $x_1 + 2x_2 = 8$.

Optimale Kante:
Alle Punkte auf der Kante zwischen $(0, 4)$ und $(2, 3)$ sind optimal mit $z = 16$.

Allgemein: Jede konvexe Kombination

$$\mathbf{x}^* = \lambda \mathbf{x}_1^* + (1 - \lambda)\mathbf{x}_2^*, \quad 0 \leq \lambda \leq 1$$

zweier optimaler Eckpunkte ist ebenfalls optimal.

Praxis-Vorteil:
Bei alternativen Optima hat der Entscheider **Spielraum** – er kann sekundaere Kriterien (z. B. Robustheit, Praeferenzen) beruecksichtigen.

Alternative Optima sind kein Problem, sondern eine Chance: Sie bieten Flexibilitaet in der Entscheidung.

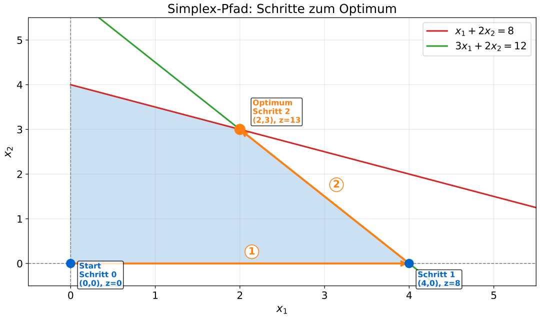
Simplex-Verfahren: Grundidee

Idee

Der Simplex-Algorithmus startet an einem Eckpunkt und wandert entlang der Kanten des Polyeders zu benachbarten Eckpunkten mit **hoeheren** Zielfunktionswerten, bis das Optimum erreicht ist.

Algorithmus (Ueberblick):

1. Waehle einen zulaessigen Eckpunkt (Startloesung).
2. Teste: Ist der aktuelle Eckpunkt optimal?
3. Falls nein: Waehle eine **verbessernde Richtung** (Pivotspalte).
4. Bestimme, wie weit man gehen kann (Pivotzeile).
5. Fuehre den Basiswechsel durch (Pivotschritt).
6. Wiederhole ab Schritt 2.



Warum funktioniert es?

- Endlich viele Eckpunkte ⇒ Terminierung

Schlupfvariablen einfuehren

Umwandlung: Ungleichung $\to$ Gleichung

Fuer jede Nebenbedingung $\leq$ fuehren wir eine **Schlupfvariable** $s_i \geq 0$ ein:

$$\mathbf{Ax} \leq \mathbf{b} \quad \Longrightarrow \quad \mathbf{Ax} + \mathbf{s} = \mathbf{b}, \quad \mathbf{s} \geq \mathbf{0}$$

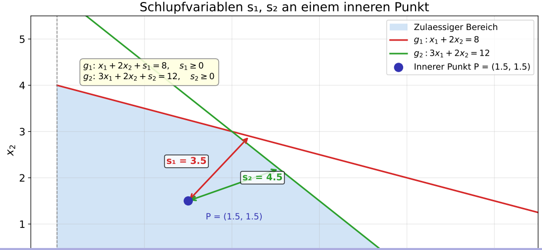
Unser Beispiel:

$$x_1 + 2x_2 + s_1 = 8 \qquad (s_1 \geq 0 : \text{ueberschuessiges Mehl})$$

$$3x_1 + 2x_2 + s_2 = 12 \qquad (s_2 \geq 0 : \text{ueberschuessige Arbeitszeit})$$

Startloesung (Basis = Schlupfvariablen):

- Setze $x_1 = 0, x_2 = 0$ (Nichtbasisvariablen)
- Dann: $s_1 = 8, s_2 = 12$ (Basisvariablen)
- Zielfunktion: $z = 0$
- Das ist der Ursprung $(0, 0)$.



Simplex-Tableau: Aufbau

Tableau-Form

Das Simplex-Tableau organisiert die Koeffizienten des Gleichungssystems:

Ausgangstableau unseres Beispiels:

Basis	x_1	x_2	s_1	s_2	**RHS**
s_1	1	2	1	0	8
s_2	3	2	0	1	12
z	-3	-5	0	0	0

Leseweise:

- Zeilen: Nebenbedingungen (Gleichungen)
- Letzte Zeile: Zielfunktion (z-Zeile)
- RHS: rechte Seite
- Basis: aktuell in der Basisloesung

Aktuelle Basisloesung:

- $s_1 = 8,\ s_2 = 12$ (Basisvariablen)
- $x_1 = 0,\ x_2 = 0$ (Nichtbasis)
- $z = 0$

Die z-Zeile enthaelt die negativen Zielfunktionskoeffizienten: $z - 3x_1 - 5x_2 = 0$.

Pivotelement waehlen

Schritt 1: Pivotspalte (eintretende Variable)

- Waehle die Spalte mit dem **negativsten** Eintrag in der z-Zeile.
- Hier: x_2 hat -5 (negativster Wert).
- Interpretation: x_2 verbessert z am staerksten pro Einheit.

Schritt 2: Pivotzeile (austretende Variable)

- Berechne die **Quotienten** $\frac{\text{RHS}}{\text{Pivotspalte}}$ fuer positive Eintraege:
  - Zeile s_1: $\frac{8}{2} = 4$
  - Zeile s_2: $\frac{12}{2} = 6$
- Waehle die Zeile mit dem **kleinsten positiven** Quotienten: s_1 $(= 4)$.
- s_1 verlaesst die Basis, x_2 tritt ein.

	x_1	x_2	s_1	s_2	RHS	Quotient
s_1	1	**2**	1	0	8	$8/2 = 4$
s_2	3	2	0	1	12	$12/2 = 6$
z	-3	-5	0	0	0	

Pivotelement

Schnittpunkt von Pivotspalte und Pivotzeile: **2** (orange markiert).

Die Quotientenregel (Minimum-Ratio-Test) stellt sicher, dass die neue Basisloesung zulaessig bleibt (RHS ≥ 0).

Pivotschritt durchfuehren

Ziel: Pivotelement auf 1 bringen, Rest der Pivotspalte auf 0 (Gauss-Elimination).

Rechenregeln:

1. **Pivotzeile** durch Pivotelement dividieren: $\text{Zeile}_1' = \frac{1}{2}\,\text{Zeile}_1$.
2. Alle **anderen Zeilen**: Vielfaches der neuen Pivotzeile subtrahieren.

Berechnung:

- Neue Zeile 1: $\frac{1}{2}(1, 2, 1, 0 \mid 8)$
  $= (\frac{1}{2}, 1, \frac{1}{2}, 0 \mid 4)$
- Neue Zeile 2: $\text{Zeile}_2 - 2 \cdot \text{Zeile}_1'$
  $= (3, 2, 0, 1 \mid 12) - 2 \cdot (\frac{1}{2}, 1, \frac{1}{2}, 0 \mid 4)$
  $= (2, 0, -1, 1 \mid 4)$
- Neue z-Zeile: $\text{Zeile}_z + 5 \cdot \text{Zeile}_1'$
  $= (-3, -5, 0, 0 \mid 0) + 5 \cdot (\frac{1}{2}, 1, \frac{1}{2}, 0 \mid 4)$
  $= (-\frac{1}{2}, 0, \frac{5}{2}, 0 \mid 20)$

Neues Tableau nach Pivot:

Basis	x_1	x_2	s_1	s_2	RHS
x_2	$\frac{1}{2}$	1	$\frac{1}{2}$	0	4
s_2	2	0	-1	1	4
z	$-\frac{1}{2}$	0	$\frac{5}{2}$	0	20

Neue Basisloesung:

$x_2 = 4,\ s_2 = 4,\ x_1 = 0,\ s_1 = 0$

$z = 20$

Der Pivotschritt ist nichts anderes als ein Schritt der Gauss-Elimination – die Variable x_2 ersetzt s_1 in der Basis.

Ist die aktuelle Loesung optimal?

- In der z-Zeile steht noch $-\frac{1}{2}$ bei x_1 $\Rightarrow$ **nicht optimal** (negative Eintraege vorhanden).
- x_1 tritt in die Basis ein (einziger negativer Eintrag).

Quotientenregel:

- Zeile x_2: $\frac{4}{1/2} = 8$
- Zeile s_2: $\frac{4}{2} = 2 \leftarrow$ Minimum

$\Rightarrow$ Pivotzeile: s_2, Pivotelement: 2.

Pivotschritt:

- Neue Zeile 2: $\frac{1}{2}(2, 0, -1, 1 \mid 4) = (1, 0, -\frac{1}{2}, \frac{1}{2} \mid 2)$
- Neue Zeile 1: $(0, 1, \frac{3}{4}, -\frac{1}{4} \mid 3)$
- Neue z-Zeile: $(0, 0, \frac{9}{4}, \frac{1}{4} \mid 21)$

Endergebnis-Tableau:

Basis	x_1	x_2	s_1	s_2	**RHS**
x_2	0	1	$\frac{3}{4}$	$-\frac{1}{4}$	3
x_1	1	0	$-\frac{1}{2}$	$\frac{1}{2}$	2
z	0	0	$\frac{9}{4}$	$\frac{1}{4}$	21

Basisloesung:
$x_1^* = 2,\ x_2^* = 3$
$z^* = \mathbf{21}$

Optimalitaetstest: Alle Eintraege in der z-Zeile ≥ 0 ✓
$\Rightarrow$ **Optimum erreicht!**

Zwei Pivotschritte genuegen: $(0, 0) \rightarrow (0, 4) \rightarrow (2, 3)$ – **der Simplex hat zwei Ecken besucht, bevor das Optimum gefunden wurde.**

Optimalitaetstest

Kriterium

Die aktuelle Basisloesung ist **optimal**, wenn alle Eintraege in der z-Zeile (ausser RHS) ≥ 0 sind:

$$\bar{c}_j \geq 0 \quad \text{fuer alle } j$$

Dabei sind $\bar{c}_j$ die **reduzierten Kosten** der Nichtbasisvariablen.

Interpretation der reduzierten Kosten:

- $\bar{c}_j < 0$: Erhoehung von x_j **verbessert** z.
  $\Rightarrow$ Noch nicht optimal.
- $\bar{c}_j > 0$: Erhoehung von x_j **verschlechtert** z.
  $\Rightarrow$ Variable bleibt bei 0.
- $\bar{c}_j = 0$: Erhoehung von x_j aendert z **nicht**.
  $\Rightarrow$ Alternative Optimalloesung moeglich.

Unser Endtableau:

	x_1	x_2	s_1	s_2	RHS
x_2	0	1	$\frac{3}{4}$	$-\frac{1}{4}$	3
x_1	1	0	$-\frac{1}{2}$	$\frac{1}{2}$	2
z	0	0	$\frac{9}{4}$	$\frac{1}{4}$	21

Alle $\bar{c}_j \geq 0$ ✓ $\Rightarrow$ **optimal!**

Die reduzierten Kosten der Schlupfvariablen im Optimaltableau sind die Schattenpreise – dazu mehr bei der Dualitaet.

Vollstaendiges Beispiel: Ausgangstableau

Neues Beispiel: maximize $z = 4x_1 + 3x_2$ unter:

$$2x_1 + x_2 \leq 10, \qquad x_1 + x_2 \leq 8, \qquad x_1, x_2 \geq 0$$

Mit Schlupfvariablen:

$$2x_1 + x_2 + s_1 = 10$$
$$x_1 + x_2 + s_2 = 8$$

Basis	x_1	x_2	s_1	s_2	RHS	Quotient
s_1	**2**	1	1	0	10	$10/2 = 5 \leftarrow$
s_2	1	1	0	1	8	$8/1 = 8$
z	-4	-3	0	0	0	

Pivotspalte: x_1 (negativster Eintrag: -4). **Pivotzeile:** s_1 (Quotient $5 < 8$).

Startwerte: $s_1 = 10$, $s_2 = 8$, $x_1 = x_2 = 0$, $z = 0$ – **wir starten am Ursprung.**

Vollstaendiges Beispiel: Nach erstem Pivot

Pivotschritt: Pivotelement $= 2$.

- Neue Zeile 1: $\frac{1}{2}(2, 1, 1, 0 \mid 10) = (1, \frac{1}{2}, \frac{1}{2}, 0 \mid 5)$
- Neue Zeile 2: $(1, 1, 0, 1 \mid 8) - 1 \cdot (1, \frac{1}{2}, \frac{1}{2}, 0 \mid 5) = (0, \frac{1}{2}, -\frac{1}{2}, 1 \mid 3)$
- Neue z-Zeile: $(-4, -3, 0, 0 \mid 0) + 4 \cdot (1, \frac{1}{2}, \frac{1}{2}, 0 \mid 5) = (0, -1, 2, 0 \mid 20)$

Basis	x_1	x_2	s_1	s_2	**RHS**	**Quotient**
x_1	1	$\frac{1}{2}$	$\frac{1}{2}$	0	5	$5/(1/2) = 10$
s_2	0	$\mathbf{\frac{1}{2}}$	$-\frac{1}{2}$	1	3	$3/(1/2) = 6 \leftarrow$
z	0	-1	2	0	20	

Basisloesung: $x_1 = 5, s_2 = 3, x_2 = 0, s_1 = 0, z = 20$. Noch -1 in z-Zeile $\Rightarrow$ nicht optimal.
Naechster Pivot: x_2 tritt ein (Spalte mit -1), s_2 tritt aus (Quotient $6 < 10$).

Jeder Pivotschritt bewegt die Loesung zu einem benachbarten Eckpunkt: $(0, 0) \rightarrow (5, 0)$.

Vollstaendiges Beispiel: Optimalloesung

Zweiter Pivotschritt: Pivotelement $= \frac{1}{2}$.

- Neue Zeile 2: $2 \cdot (0, \frac{1}{2}, -\frac{1}{2}, 1 \mid 3) = (0, 1, -1, 2 \mid 6)$
- Neue Zeile 1: $(1, \frac{1}{2}, \frac{1}{2}, 0 \mid 5) - \frac{1}{2} \cdot (0, 1, -1, 2 \mid 6) = (1, 0, 1, -1 \mid 2)$
- Neue z-Zeile: $(0, -1, 2, 0 \mid 20) + 1 \cdot (0, 1, -1, 2 \mid 6) = (0, 0, 1, 2 \mid 26)$

Basis	x_1	x_2	s_1	s_2	**RHS**
x_1	1	0	1	-1	2
x_2	0	1	-1	2	6
z	0	0	1	2	**26**

Optimalloesung

$x_1^* = 2,\ x_2^* = 6,\ z^* = 4 \cdot 2 + 3 \cdot 6 = 26$.
Alle $\bar{c}_j \geq 0$ ✓ $\Rightarrow$ Optimum erreicht!

Pfad des Simplex: $(0,0) \xrightarrow{z=0} (5,0) \xrightarrow{z=20} (2,6) \xrightarrow{z=26}$ **optimal**.

Verifikation: $2 \cdot 2 + 6 = 10 \leq 10$ ✓ **und** $2 + 6 = 8 \leq 8$ ✓ **– beide Nebenbedingungen aktiv.**

Big-M-Methode: Motivation

Problem

Wenn die Nebenbedingungen $\geq$-Ungleichungen oder Gleichungen enthalten, liefern die Schlupfvariablen **keine** zulaessige Startbasis. Der Ursprung ist nicht zulaessig!

Beispiel:

$$\begin{aligned} \text{minimize} \quad & z = 2x_1 + 3x_2 \\ \text{s.t.} \quad & x_1 + x_2 \geq 4 \\ & 2x_1 + x_2 \leq 10 \\ & x_1, x_2 \geq 0 \end{aligned}$$

Erste NB: $x_1 + x_2 - s_1 = 4$ (Ueberschussvariable).

Setze $x_1 = x_2 = 0 \Rightarrow s_1 = -4 < 0$ ✗
Nicht zulaessig!

Loesungsidee:

1. Fuehre eine **kuenstliche Variable** $a_1 \geq 0$ ein:
$$x_1 + x_2 - s_1 + a_1 = 4$$
2. Bestrafe a_1 in der Zielfunktion mit einem sehr grossen $M > 0$:
$$\text{minimize } z = 2x_1 + 3x_2 + Ma_1$$
3. Im Optimum wird $a_1 = 0$ (sonst waere die Strafe zu hoch).

Die kuenstliche Variable macht den Ursprung "kuenstlich" zulaessig.

M **muss "gross genug" sein – typisch** $M = 10^4$ **bis** 10^6. **Zu gross: numerische Probleme; zu klein: falsche Loesung.**

Kuenstliche Variablen einfuehren

Regeln fuer kuenstliche Variablen:

NB-Typ	Schlupf/Ueberschuss	Kuenstliche Var.
$\leq$	$+s_i$	nicht noetig (Schlupf als Basis)
$\geq$	$-s_i$	$+a_i$ noetig
$=$	–	$+a_i$ noetig

Unser Beispiel (Minimierung $\rightarrow$ Maximierung):

$$\begin{aligned} \text{maximize} \quad & z' = -2x_1 - 3x_2 - Ma_1 \\ \text{s.t.} \quad & x_1 + x_2 - s_1 + a_1 = 4 \\ & 2x_1 + x_2 + s_2 = 10 \\ & x_1, x_2, s_1, s_2, a_1 \geq 0 \end{aligned}$$

Startbasis: $a_1 = 4$ (Zeile 1), $s_2 = 10$ (Zeile 2), $z' = -4M$.

Wichtig: Vor dem Start muss die z-Zeile korrigiert werden, damit a_1-Spalte den Eintrag 0 hat (Gauss-Schritt).

Die kuenstliche Variable ist ein Hilfsgeruest – am Ende muss $a_i = 0$ gelten, sonst ist das urspruengliche LP unzulaessig.

Korrigiertes Ausgangstableau (nach Elimination von a_1 aus z-Zeile):

Basis	x_1	x_2	s_1	s_2	a_1	**RHS**
a_1	1	1	-1	0	1	4
s_2	2	1	0	1	0	10
z'	$-2+M$	$-3+M$	M	0	0	$-4M$

Pivotierung:

- z-Zeile: $-3+M$ bei x_2 (fuer grosses M am "negativsten")
- $\Rightarrow$ x_2 tritt ein, a_1 tritt aus (Quotient $4/1 = 4$).

Nach Pivot:

Basis	x_1	x_2	s_1	s_2	a_1	RHS
x_2	1	1	-1	0	1	4
s_2	1	0	1	1	-1	6
z'	1	0	-3	0	$3-M$	-12

Weiter pivotieren (s_1 hat $\bar{c} = -3 < 0$):

- s_1 tritt ein, s_2 tritt aus (Quotient $6/1$, Zeile x_2 hat negativen Eintrag).

Finales Tableau:

Basis	x_1	x_2	s_1	s_2	RHS
x_2	2	1	0	1	10
s_1	1	0	1	1	6
z'	4	0	0	3	-6

$x_1^* = 0, x_2^* = 10, z_{\min} = 6.$

$a_1 = 0$ ✓ – kuenstliche Variable eliminiert.

Die Big-M-Methode fuegt genau so viele kuenstliche Variablen hinzu, wie $\geq$- und $=$-Nebenbedingungen vorhanden sind.

Big-M: Bewertung und Alternativen

Vorteile der Big-M-Methode:

- Einfach zu verstehen und zu implementieren
- Nur **eine** Phase noetig
- Direkt auf beliebige LP-Formen anwendbar

Nachteile:

- Wahl von M kritisch:
  - M zu gross $\rightarrow$ numerische Instabilitaet
  - M zu klein $\rightarrow$ kuenstliche Variable bleibt > 0
- Grosse M-Werte verschlechtern die Kondition der Tableau-Matrix
- In der Praxis selten besser als Zweiphasen-Methode

Alternative: Zweiphasen-Methode

Idee

- **Phase I:** Loese ein Hilfsproblem, das die kuenstlichen Variablen minimiert: minimize $\sum a_i$.
- Wenn Minimum $= 0$: zulaessige Basis gefunden.
- **Phase II:** Loese das urspruengliche LP mit dieser Startbasis.

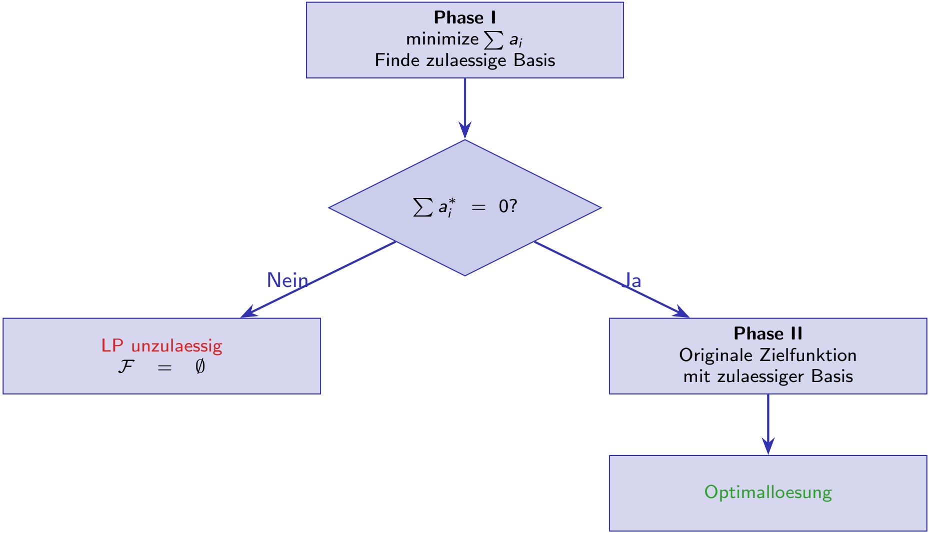
Vorteil: Kein grosses M noetig $\Rightarrow$ numerisch stabiler.

Vergleich:

	Big-M	2-Phasen
Einfachheit	✓	
Num. Stabilitaet		✓
Praxis-Einsatz	selten	oft

In modernen LP-Solvern werden weder Big-M noch Zweiphasen explizit verwendet – stattdessen nutzen sie Crash-Heuristiken und Preprocessing.

Vorschau: Zweiphasen-Methode



Merke: Phase I beantwortet die Frage "Ist das LP ueberhaupt zulaessig?", Phase II findet dann die Optimalloesung.

Das duale lineare Programm

Konstruktionsregeln

Zu jedem **primalen LP** (Maximierung) gehoert ein **duales LP** (Minimierung):

Primales LP:

$$\begin{aligned} \underset{\mathbf{x}}{\text{maximize}} \quad & \mathbf{c}^\top \mathbf{x} \\ \text{s.t.} \quad & \mathbf{A}\mathbf{x} \leq \mathbf{b} \\ & \mathbf{x} \geq \mathbf{0} \end{aligned}$$

n **Variablen,** m **Nebenbedingungen**

$$\Longleftrightarrow$$

Duales LP:

$$\begin{aligned} \underset{\mathbf{y}}{\text{minimize}} \quad & \mathbf{b}^\top \mathbf{y} \\ \text{s.t.} \quad & \mathbf{A}^\top \mathbf{y} \geq \mathbf{c} \\ & \mathbf{y} \geq \mathbf{0} \end{aligned}$$

m **Variablen,** n **Nebenbedingungen**

Merkregel:

- Primale NB-Koeffizienten $\mathbf{A} \rightarrow$ duale NB-Koeffizienten $\mathbf{A}^\top$
- Primale Zielfunktion $\mathbf{c} \rightarrow$ duale rechte Seite $\mathbf{c}$
- Primale rechte Seite $\mathbf{b} \rightarrow$ duale Zielfunktion $\mathbf{b}$
- maximize $\leftrightarrow$ minimize, $\quad \leq \leftrightarrow \geq$

Das Dual des Duals ist wieder das Primal – die Beziehung ist symmetrisch.

Schwacher Dualitaetssatz

Satz (Schwache Dualitaet)

Sei $\mathbf{x}$ zulaessig fuer das primale LP und $\mathbf{y}$ zulaessig fuer das duale LP. Dann gilt:

$$\mathbf{c}^\mathsf{T}\mathbf{x} \;\leq\; \mathbf{b}^\mathsf{T}\mathbf{y}$$

Der primale Zielfunktionswert ist stets $\leq$ dem dualen Zielfunktionswert.

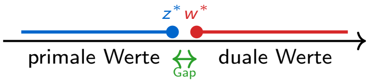
Beweisskizze:

$$\mathbf{c}^\mathsf{T}\mathbf{x} \leq (\mathbf{A}^\mathsf{T}\mathbf{y})^\mathsf{T}\mathbf{x} = \mathbf{y}^\mathsf{T}\mathbf{A}\mathbf{x} \leq \mathbf{y}^\mathsf{T}\mathbf{b} = \mathbf{b}^\mathsf{T}\mathbf{y}$$

1. Erste $\leq$: duale NB $\mathbf{A}^\mathsf{T}\mathbf{y} \geq \mathbf{c}$, $\mathbf{x} \geq 0$
2. Zweite $\leq$: primale NB $\mathbf{A}\mathbf{x} \leq \mathbf{b}$, $\mathbf{y} \geq 0$

Konsequenzen:

- Jede duale zulaessige Loesung liefert eine **obere Schranke** fuer das primale Optimum.
- Jede primale zulaessige Loesung liefert eine **untere Schranke** fuer das duale Optimum.
- Falls $\mathbf{c}^\mathsf{T}\mathbf{x}^* = \mathbf{b}^\mathsf{T}\mathbf{y}^*$, dann sind **beide** optimal.



Der starke Dualitaetssatz zeigt: Im Optimum ist der Gap $= 0$, d. h. $z^* = w^*$ (bei Zulaessigkeit beider Probleme).

Schattenpreise: Oekonomische Interpretation

Definition

Der **Schattenpreis** (engl. *shadow price*) der i-ten Nebenbedingung ist die optimale duale Variable y_i^*. Er gibt an, um wie viel sich z^* aendert, wenn die rechte Seite b_i um eine Einheit erhoeht wird.

Baeckerei-Beispiel:
Optimaltableau liefert z-Zeile:
$(0, 0, \frac{9}{4}, \frac{1}{4} \mid 21)$

- $y_1^* = \frac{9}{4}$ (Schattenpreis Mehl):
  1 kg mehr Mehl $\Rightarrow$ Gewinn steigt um 2.25 Euro.
- $y_2^* = \frac{1}{4}$ (Schattenpreis Arbeitszeit):
  1 Std. mehr $\Rightarrow$ Gewinn steigt um 0.25 Euro.

Folgerung:
Mehl ist die **wertvollere** Ressource!

Allgemein:

$$\frac{\partial z^*}{\partial b_i} = y_i^*$$

Ablesen aus dem Optimaltableau

Die Schattenpreise stehen in der z-Zeile unter den **Schlupfvariablen**:

$$\bar{c}_{s_i} = y_i^*$$

Nichtaktive NB:
Schlupf $s_i > 0 \Rightarrow y_i^* = 0$.
Mehr Ressource bringt nichts, da sie ohnehin nicht knapp ist.

Schattenpreise sind nur in einem gewissen Bereich gueltig (RHS-Ranging) – groessere Aenderungen erfordern eine Neuberechnung.

Beispiel: Primal und Dual

Primales LP (Baeckerei):

$$\begin{aligned} \text{maximize} \quad & z = 3x_1 + 5x_2 \\ \text{s.t.} \quad & x_1 + 2x_2 \leq 8 \\ & 3x_1 + 2x_2 \leq 12 \\ & x_1, x_2 \geq 0 \end{aligned}$$

Optimalloesung:

$x_1^* = 2,\ x_2^* = 3,\ z^* = 21.$

Duales LP:

$$\begin{aligned} \text{minimize} \quad & w = 8y_1 + 12y_2 \\ \text{s.t.} \quad & y_1 + 3y_2 \geq 3 \\ & 2y_1 + 2y_2 \geq 5 \\ & y_1, y_2 \geq 0 \end{aligned}$$

Optimalloesung:

$y_1^* = \frac{9}{4},\ y_2^* = \frac{1}{4},\ w^* = 21.$

Verifikation: Starke Dualitaet

$z^* = 21 = w^*$ ✓ Die Optimalwerte stimmen ueberein!

Duale Deutung: y_1, y_2 sind die Preise fuer Mehl bzw. Arbeitszeit. Das duale LP fragt: Was ist der **minimale Gesamtwert** der Ressourcen, sodass kein Produkt profitabler waere als sein Ressourcenverbrauch?

Primal = Produzent (maximiert Gewinn). Dual = Ressourcenbewerter (minimiert Ressourcenkosten bei fairem Preis).

Starker Dualitaetssatz

Satz (Starke Dualitaet)

Wenn das primale LP eine optimale Loesung $\mathbf{x}^*$ hat, dann hat auch das duale LP eine optimale Loesung $\mathbf{y}^*$, und es gilt:

$$\mathbf{c}^T\mathbf{x}^* = \mathbf{b}^T\mathbf{y}^*$$

Der **Dualitaets-Gap ist Null**.

Alle Faelle:

	Dual opt.	**Dual unzul.**
Primal opt.	$z^* = w^*$	unmoeglich
Primal unzul.	unmoeglich	moeglich
Primal unbeschr.	unmoeglich	ja

Merke:

- Beide optimal $\Rightarrow$ gleicher Wert
- Eines unbeschraenkt $\Rightarrow$ anderes unzulaessig
- Beide unzulaessig: moeglich!

Praktische Bedeutung:

- Der Simplex-Algorithmus liefert die optimale duale Loesung **gratis mit** (aus dem Optimaltableau).
- Optimalitaetszertifikat: Wenn $\mathbf{c}^T\mathbf{x} = \mathbf{b}^T\mathbf{y}$ fuer zulaessige $\mathbf{x}, \mathbf{y}$, dann sind beide optimal.
- Kein separates duales Loesen noetig!

Verifikation (Baeckerei):
$\mathbf{c}^T\mathbf{x}^* = 3 \cdot 2 + 5 \cdot 3 = 21$
$\mathbf{b}^T\mathbf{y}^* = 8 \cdot \frac{9}{4} + 12 \cdot \frac{1}{4} = 18 + 3 = 21$ ✓

Der starke Dualitaetssatz ist eines der zentralen Resultate der linearen Optimierung – er verbindet Primal und Dual vollstaendig.

Komplementaerer Schlupf

Satz (Complementary Slackness)

Seien $\mathbf{x}^*$ und $\mathbf{y}^*$ optimale Loesungen des primalen bzw. dualen LP. Dann gilt:

$$y_i^* \cdot (b_i - \mathbf{a}_i^T \mathbf{x}^*) = 0 \quad \forall\, i = 1, \ldots, m$$
$$x_j^* \cdot (\mathbf{a}_j^T \mathbf{y}^* - c_j) = 0 \quad \forall\, j = 1, \ldots, n$$

Interpretation:

- Wenn $y_i^* > 0$ (positive duale Variable), dann ist die i-te primale NB **aktiv** (Schlupf $= 0$).
- Wenn primale NB i **nicht aktiv** (Schlupf > 0), dann ist $y_i^* = 0$.

In Kurzform:
"Duale Variable positiv $\Leftrightarrow$ Primale NB aktiv"

Baeckerei-Verifikation:

- $y_1^* = \frac{9}{4} > 0$ und $s_1 = 8 - (2 + 6) = 0$ ✓
- $y_2^* = \frac{1}{4} > 0$ und $s_2 = 12 - (6 + 6) = 0$ ✓

Beide dualen Variablen sind positiv, beide primalen NB sind aktiv.

Anwendung:
Komplementaerer Schlupf kann genutzt werden, um die duale Loesung aus der primalen zu **berechnen**, ohne das Dual separat zu loesen.

Komplementaerer Schlupf ist das Brueckenprinzip zwischen primalem und dualem Problem – zentral fuer Sensitivitaetsanalyse.

Dualitaet: Oekonomische Interpretation vertieft

Primale Sicht (Produzent):

- Entscheidet ueber Produktionsmengen $\mathbf{x}$
- Maximiert Gewinn $\mathbf{c}^T\mathbf{x}$
- Beschraenkt durch Ressourcen $\mathbf{Ax} \leq \mathbf{b}$

Duale Sicht (Ressourcenkaeufer):

- Bietet Preise $\mathbf{y}$ fuer Ressourcen
- Minimiert Gesamtkaufpreis $\mathbf{b}^T\mathbf{y}$
- Muss faire Preise bieten: $\mathbf{A}^T\mathbf{y} \geq \mathbf{c}$ (fuer jedes Produkt muss der Ressourcenwert mindestens so gross sein wie der Gewinn)

Gleichgewicht (Starke Dualitaet):
Im Optimum sind Produzent und Kaeufer gleich gut gestellt:

$$\text{Max-Gewinn} = \text{Min-Ressourcenwert}$$

Baeckerei:

- 1 Brot verbraucht 1 kg Mehl + 3 Std. Arbeit
- Faire Preise: $y_1 + 3y_2 \geq 3$ (Gewinn von Brot)
- 1 Kuchen verbraucht 2 kg Mehl + 2 Std. Arbeit
- Faire Preise: $2y_1 + 2y_2 \geq 5$ (Gewinn von Kuchen)

Ergebnis: Mehlpreis $y_1^* = 2.25$ Euro/kg, Arbeitspreis $y_2^* = 0.25$ Euro/Std.

Die duale Interpretation gibt Managern direkt Aufschluss darueber, welche Ressourcen wertvoll sind und wo Investitionen lohnen.

Anwendung: Produktionsplanung

Szenario

Eine Fabrik produziert zwei Produkte P_1 und P_2 mit drei Maschinen.

	Maschine A (Std/Stueck)	**Maschine B** (Std/Stueck)	**Maschine C** (Std/Stueck)
Produkt P_1	1	3	1
Produkt P_2	2	2	0
Kapazitaet	8 Std	18 Std	5 Std
Gewinn/Stueck	P_1: 4 Euro, P_2: 5 Euro		

LP-Formulierung:

$$
\begin{aligned}
\text{maximize} \quad & z = 4x_1 + 5x_2 \\
\text{s.t.} \quad & x_1 + 2x_2 \leq 8 \\
& 3x_1 + 2x_2 \leq 18 \\
& x_1 \leq 5 \\
& x_1, x_2 \geq 0
\end{aligned}
$$

Erweiterungspotenzial:

- Mehr Produkte: $n > 2$ Variablen
- Nachfragegrenzen: $x_j \leq d_j$
- Mindestmengen: $x_j \geq l_j$
- Mehrere Perioden (dynamisch)

Produktionsplanung ist die klassischste LP-Anwendung – das gleiche Grundmodell skaliert auf Hunderte von Produkten und Maschinen.

Anwendung: Transportproblem

Problem

m Lagerhaeuser mit Angebot a_i beliefern n Kunden mit Nachfrage d_j. Transportkosten: c_{ij} pro Einheit. Minimiere die Gesamtkosten.

LP-Formulierung:

$$\begin{aligned} \text{minimize} \quad & \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} x_{ij} \\ \text{s.t.} \quad & \sum_{j=1}^{n} x_{ij} \leq a_i \quad \forall\, i \\ & \sum_{i=1}^{m} x_{ij} \geq d_j \quad \forall\, j \\ & x_{ij} \geq 0 \quad \forall\, i, j \end{aligned}$$

x_{ij}: transportierte Menge von Lager i zu Kunde j.

Beispiel (2 Lager, 3 Kunden):

	K_1	K_2	K_3	Angebot
L_1	2	3	1	20
L_2	5	4	8	30
Nachfr.	10	15	15	

Besonderheiten:

- Spezialstruktur: Transportalgorithmus
- Ganzzahlige Loesung automatisch, wenn $a_i, d_j \in \mathbb{Z}$

Das Transportproblem ist ein Spezialfall des LP – die Netzwerkstruktur erlaubt besonders effiziente Loesungsverfahren.

Anwendung: Portfolio-Optimierung (LP-Ansatz)

Vereinfachtes Modell

Ein Investor verteilt Kapital auf n Anlagen. Ziel: Maximiere die erwartete Rendite unter Risikobeschraenkungen.

LP-Formulierung:

$$\begin{aligned} \text{maximize} \quad & \sum_{j=1}^{n} r_j x_j \\ \text{s.t.} \quad & \sum_{j=1}^{n} x_j = 1 \quad \text{(Budgetrest.)} \\ & x_j \leq u_j \quad \forall j \quad \text{(Diversifikation)} \\ & x_j \geq 0 \quad \forall j \end{aligned}$$

Beispiel:

Asset	r_j	u_j
Aktien	8%	0.40
Anleihen	3%	0.50
Immobilien	5%	0.30

Hinweis: Das klassische Markowitz-Modell ist ein *QP*, kein LP. Hier die lineare Vereinfachung.

LP-Portfoliomodelle eignen sich fuer Szenarien mit linearen Risikomassen (z. B. CVaR, MAD) statt Varianz.

Python: LP loesen mit scipy

Baeckerei-Beispiel in Python:

Code

```
from scipy.optimize import linprog

# Zielfunktion (linprog minimiert!)
c = [-3, -5]  # negativ fuer max

# Nebenbedingungen: A_ub @ x <= b_ub
A_ub = [[1, 2],
        [3, 2]]
b_ub = [8, 12]

# Variablengrenzen
x1_bounds = (0, None)
x2_bounds = (0, None)

res = linprog(c, A_ub=A_ub, b_ub=b_ub,
              bounds=[x1_bounds, x2_bounds],
              method='highs')

print(f"x* = {res.x}")
```

Ausgabe

```
x* = [2. 3.]
z* = 21.0
```

Wichtige Punkte:

- `linprog` **minimiert** $\Rightarrow$ fuer Maximierung **c** negieren.
- `method='highs'`: moderner Solver.
- `res.fun`: Optimalwert (negiert).
- `res.x`: Optimalvektor.
- `res.slack`: Schlupfwerte.

Frage

Wie aendert sich der **optimale Zielfunktionswert** z^*, wenn die rechte Seite b_i einer Nebenbedingung leicht variiert?

Schattenpreis = Marginalwert:

$$y_i^* = \frac{\Delta z^*}{\Delta b_i} \quad \text{(lokal)}$$

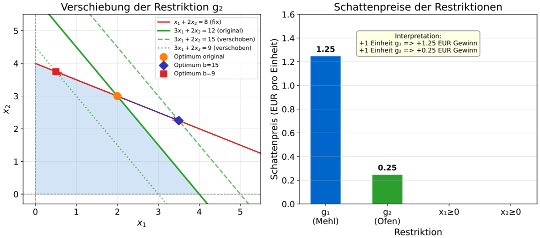
Baeckerei:

- Mehl ($b_1 = 8$): $y_1^* = \frac{9}{4} = 2.25$
  $\Rightarrow$ 1 kg mehr $\rightarrow$ +2.25 Euro Gewinn
- Arbeitszeit ($b_2 = 12$): $y_2^* = \frac{1}{4} = 0.25$
  $\Rightarrow$ 1 Std. mehr $\rightarrow$ +0.25 Euro Gewinn

Management-Entscheidung:
Mehl ist 9-mal wertvoller als Arbeitszeit!
$\Rightarrow$ In zusaetzliches Mehl investieren.

Gueltigkeitsbereich:

Schattenpreise gelten nur, solange die **aktuelle Basis optimal bleibt**. Bei zu grossen Aenderungen von b_i wechselt die Basis.



Im Gueltigkeitsbereich ist z^* eine **lineare Funktion** von b_i

RHS-Ranging: Gueltigkeitsbereich der Schattenpreise

Frage

In welchem Bereich darf b_i variieren, sodass die **aktuelle Basis** (und damit die Schattenpreise) erhalten bleibt?

Methode:
Im Optimaltableau hat die Basisloesung die Form $\mathbf{x}_B = \mathbf{B}^{-1}\mathbf{b}$. Die Basis bleibt zulaessig, solange:

$$\mathbf{B}^{-1}\mathbf{b} \geq \mathbf{0}$$

Setze $b_i \to b_i + \Delta$ und bestimme den Bereich von Δ, fuer den alle Basisvariablen ≥ 0 bleiben.

Baeckerei-Mehl ($b_1 = 8$):

- $x_2 = \frac{3}{4}(8 + \Delta) - \frac{1}{4} \cdot 12 = 3 + \frac{3}{4}\Delta \geq 0$
  $\Rightarrow \Delta \geq -4$
- $x_1 = -\frac{1}{2}(8 + \Delta) + \frac{1}{2} \cdot 12 = 2 - \frac{1}{2}\Delta \geq 0$
  $\Rightarrow \Delta \leq 4$

Ergebnis:

$$b_1 \in [8 - 4,\ 8 + 4] = [4,\ 12]$$

Im Bereich $4 \leq b_1 \leq 12$ gilt der Schattenpreis $y_1^* = \frac{9}{4}$ uneingeschraenkt.

Interpretation:

- Erhoehung des Mehls von 8 auf 12 (+4 kg):
  $\Delta z^* = 4 \cdot 2.25 = +9$ Euro
  $z^*_{\text{neu}} = 21 + 9 = 30$ Euro.
- Reduktion auf 4 kg:
  $\Delta z^* = (-4) \cdot 2.25 = -9$ Euro
  $z^*_{\text{neu}} = 21 - 9 = 12$ Euro.

Ausserhalb dieses Bereichs: Basiswechsel, neue Schattenpreise.

RHS-Ranging ist ein Standardoutput professioneller Solver – es zeigt, wie "robust" die aktuelle Loesung ist.

Koeffizientenanalyse: Aenderung der Zielfunktion

Frage

In welchem Bereich darf ein Zielfunktionskoeffizient c_j variieren, sodass die **aktuelle Basisloesung optimal bleibt** (auch wenn sich z^* aendert)?

Methode:
Die Basisloesung bleibt optimal, solange alle reduzierten Kosten $\bar{c}_j \geq 0$ bleiben.

Baeckerei – Koeffizient $c_1 = 3$ (Brot):
Im Optimaltableau: $\bar{c}_1 = 0$. Aenderung von c_1 um Δ:

$$\bar{c}_1' = 0 + \Delta \cdot (\text{Empfindlichkeit})$$

Analyse: c_1 darf zwischen $\frac{5}{2}$ und ∞ variieren, ohne dass sich die Basis aendert (nur z^* aendert sich).

Interpretation:

- Solange der Brotgewinn ≥ 2.50 Euro bleibt, lohnt es sich weiterhin, $x_1 = 2$ Brot zu produzieren.
- Sinkt $c_1 < 2.50$: Es lohnt sich, auf reinen Kuchen umzusteigen (Basiswechsel).

Geometrische Deutung:
Die Aenderung von c_j dreht die Isoprofit-Linie. Solange dieselbe Ecke "zuletzt beruehrt" wird, bleibt die Basisloesung gleich.

Praxis: Koeffizientenanalyse zeigt die Robustheit gegenueber Gewinn- oder Kostenschwankungen.

RHS-Ranging und Koeffizientenanalyse zusammen bilden die vollstaendige Sensitivitaetsanalyse eines LP.

Baeckerei-LP – Zusammenfassung aller Sensitivitaetsinformationen:

Variable	Wert	Reduz. Kosten	Untergrenze c_j	Obergrenze c_j
x_1 (Brot)	2	0	2.50	∞
x_2 (Kuchen)	3	0	0	6.00

Nebenbedingung	Schlupf	Schattenpreis	Untergrenze b_i	Obergrenze b_i
Mehl ($b_1 = 8$)	0	2.25	4	12
Arbeitszeit ($b_2 = 12$)	0	0.25	8	24

Entscheidungshilfe:

- Zusaetzliches Mehl bis 12 kg lohnt sich mit 2.25 Euro/kg Grenzwert.
- Zusaetzliche Arbeitszeit bis 24 Std. lohnt sich mit 0.25 Euro/Std. Grenzwert.
- Brotgewinn kann bis 2.50 Euro sinken, ohne die Produktionsentscheidung zu aendern.

Sensitivitaetsanalyse liefert dem Management direkt umsetzbare Handlungsempfehlungen – ohne das LP neu loesen zu muessen.

scipy.optimize.linprog: Ausfuehrliches Beispiel

Vollstaendiger Code

```
from scipy.optimize import linprog
import numpy as np

c = [-3, -5]         # Maximierung
A_ub = [[1, 2],      # Mehl
        [3, 2]]      # Arbeitszeit
b_ub = [8, 12]

bounds = [(0, None), (0, None)]

res = linprog(c, A_ub=A_ub, b_ub=b_ub,
              bounds=bounds, method='highs')

print("Status:", res.message)
print("x* =", res.x)
print("z* =", -res.fun)
print("Schlupf:", res.slack)
print("Iterationen:", res.nit)
```

Ausgabe

```
Status: Optimization terminated
  successfully.
x* = [2. 3.]
z* = 21.0
Schlupf: [0. 0.]
Iterationen: 2
```

Rueckgabe-Attribute:

- `res.x`: Optimalloesung $\mathbf{x}^*$
- `res.fun`: Optimalwert (Minimierung!)
- `res.slack`: Schlupfwerte $\mathbf{s}$
- `res.nit`: Anzahl Iterationen
- `res.status`: $0 =$ Erfolg
- `res.message`: Beschreibung

Der HiGHS-Solver in scipy kombiniert Simplex und Interior-Point-Methoden – ab scipy 1.9 ist er der Standard.

Schlupfwerte interpretieren:

```
res.slack = [0. 0.]
```

- $s_1 = 0$: Mehl-NB ist **aktiv** (binding)
- $s_2 = 0$: Arbeitszeit-NB ist **aktiv**
- Beide Ressourcen vollstaendig ausgeschoepft.

Status-Codes:

Code	Bedeutung
0	Optimum gefunden
1	Iterationslimit erreicht
2	LP unzulaessig
3	LP unbeschraenkt
4	Numerische Probleme

Gleichungs-NB und Schranken:

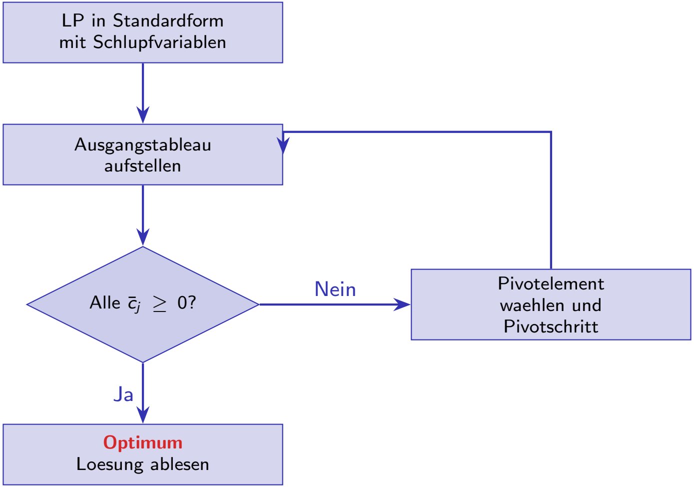
```
# Gleichung: A_eq @ x = b_eq
A_eq = [[1, 1]]
b_eq = [5]

# Obere Schranken
bounds = [(0, 4), (0, 3)]

res = linprog(c,
    A_ub=A_ub, b_ub=b_ub,
    A_eq=A_eq, b_eq=b_eq,
    bounds=bounds,
    method='highs')
```

Fuer groessere Probleme:

- **PuLP:** Modellierungssprache, bindet verschiedene Solver an
- **Pyomo:** Umfangreiches Framework
- **gurobipy:** Gurobi Python API

Pruefen Sie immer den Status-Code – "Optimum gefunden" ist keine Selbstverstaendlichkeit bei realen Problemen.

Simplex-Algorithmus: Flussdiagramm



Komplexitaet: Im Worst Case exponentiell (Klee-Minty), in der Praxis typisch $O(m)$ bis $O(m^2)$ Pivotschritte bei m Nebenbedingungen.

Fuer polynomiale Worst-Case-Garantie: Innere-Punkte-Methoden (Karmarkar, 1984) – in der Praxis oft aehnlich schnell wie Simplex.

Wichtige Formeln

Standardform:

$$\underset{\mathbf{x}}{\text{maximize}}\ \mathbf{c}^\top \mathbf{x} \quad \text{s.t. } \mathbf{A}\mathbf{x} \leq \mathbf{b},\ \mathbf{x} \geq \mathbf{0}$$

Schlupfvariablen:

$$\mathbf{A}\mathbf{x} + \mathbf{s} = \mathbf{b}, \quad \mathbf{s} \geq \mathbf{0}$$

Pivotspalte: Spalte j mit kleinstem $\bar{c}_j < 0$.

Pivotzeile: Zeile i mit min $\frac{b_i}{a_{ij}}$ $(a_{ij} > 0)$.

Optimalitaet: $\bar{c}_j \geq 0$ fuer alle j.

Dualitaet:

$$\text{Primal: maximize } \mathbf{c}^\top \mathbf{x},\ \mathbf{A}\mathbf{x} \leq \mathbf{b}$$

$$\text{Dual: minimize } \mathbf{b}^\top \mathbf{y},\ \mathbf{A}^\top \mathbf{y} \geq \mathbf{c}$$

Schwache Dualitaet: $\mathbf{c}^\top \mathbf{x} \leq \mathbf{b}^\top \mathbf{y}$
Starke Dualitaet: $z^* = w^*$
Komplementaerer Schlupf: $y_i^* s_i^* = 0$, $x_j^* \bar{c}_j = 0$

Schattenpreis: $y_i^* = \frac{\partial z^*}{\partial b_i}$

Diese Formeln bilden das Grundgeruest der linearen Optimierung – beherrschen Sie sie fuer Klausur und Praxis.

Ausblick und Literatur

Weiterfuehrende Themen:

- **Zweiphasen-Methode:** Numerisch stabile Alternative zu Big-M.
- **Ganzzahlige Optimierung:** Branch-and-Bound, Schnittebenen.
- **Innere-Punkte-Methoden:** Polynomiale Alternative zum Simplex.
- **Nichtlineare Optimierung:** QP, konvexe Optimierung.
- **Stochastische Optimierung:** Unsichere Parameter.

Literatur:

- **Bertsimas & Tsitsiklis:** *Introduction to Linear Optimization*, Athena Scientific, 1997.
- **Chvatal:** *Linear Programming*, W. H. Freeman, 1983.
- **Dantzig & Thapa:** *Linear Programming*, Springer, 1997/2003.
- **Vanderbei:** *Linear Programming*, 5th ed., Springer, 2020.

Software:

- Python: `scipy`, PuLP, Pyomo
- Kommerziell: Gurobi, CPLEX, MOSEK
- Open Source: GLPK, HiGHS, CBC

Lineare Optimierung ist das Fundament fuer viele weitergehende Methoden – von Machine Learning bis Supply-Chain-Management.