NTNU | Norwegian University of Science and Technology

TTK4135 – Optimization and Control

Spring 2021

Lecturer: Lars Imsland
Teaching Assistant: Joakim R. Andersen
6 Student Assistants

Learning Objectives

- Optimization – important concepts and theory
- Formulating an engineering problem into a mathematical optimization problem (modeling for optimization)
- Solving an optimization problem numerically
  - choosing the right algorithm for your problem,
  - use of (the right) optimization software,
  - some implementation of algorithms

  for a few important classes of optimization problems
- Applications in control engineering – model predictive control

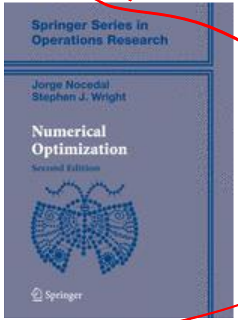
Numerical optimization is an incredibly versatile tool across most engineering domains

Course Information: General

- Description:
  - All course information is provided through Blackboard
  - Course description: http://www.itk.ntnu.no/emner/ttk4135
- Assignments and assessment (More information on Blackboard):
  - Exercises: 7 of 10 assignments must be approved
    - No extra assignments will be given, deadlines are absolute
    - Pay attention and make sure your delivered assignments are approved
    - Do not copy (kok)!
  - Helicopter lab: must be approved
    - Evaluation weighted 26% towards grade
    - In addition to deliver (group) reports, you have to give feedback on other groups' reports (new)
  - Matlab assessments (new)
    - 6 Matlab assessments, each counts 4% towards grade (pass/no pass)
  - Final exam (digital home exam)
    - Evaluation weighted 50% towards grade

Course Information: Course Material

- Lectures: Online on Blackboard
  - Will not cover the full curriculum in lectures
  - Will focus on difficult parts and build intuition
  - Will be recorded, and PDF made available afterwards
  - Online lecturing is new for lecturer
    - Constructive feedback welcome!
    - Ask questions in chat *or use* "raise *hand*"
    - If he is not recording, remind him!



- Course Material:
  - Numerical Optimization, J. Nocedal and S. J Wright, 2nd ed., Springer (ISBN-10: 0-387-30303-0 or ISBN-13: 987-0387-30303-1). Download here from campus or through VPN.
  - Errata on Blackboard
  - Note on Merging Optimization and Control, B. Foss and T. A. Heirung (Blackboard)
  - Note on Matrix Calculus, T. A. Heirung (Blackboard)

Course Information: Practical

- Grading
  - Final exam: 50%
  - Lab report (helicopter): 26% (group work)
  - Matlab assessments: 24% (6 individual tasks)
- Timetable
  - Lectures: Tuesday 08:15 - 10:00 on Blackboard
    Friday 08:15 - 10:00 on Blackboard
  - Assignment Sessions: Monday 16:15 - 17:00 on Blackboard
- Exam: May 28, 09:00 – 13:00 (?)
- Reference group!
- Video lectures from 2014: https://mediasite.ntnu.no/Mediasite/Catalog/catalogs/ttk4135-v14

Expected Background

- Linear algebra and real analysis
  - Quick recap next time (Also: Note on Blackboard + Exercise 0)
- Some numerical analysis (Newton's method)
- Basic control theory:
  - TTK4105 Control engineering
  - Advantage: TTK4115 Linear system theory

Tentative Lecture schedule

	TTK4135 Plan for Spring 2021					
Week no.	**Lectures Tuesday 08:15-10:00 Online**	**Lectures Friday 08:15-10:00 Online**	**Helicopter project**	**Exercise out (Mon 15:00)**	**Help session Monday 16:15-17:00 Online**	**Exercise in (Wed 23:59)**
2	Lecture 1 Introduction on optimization - N&W Ch.1	Lecture 2 Optimality conditions - N&W Ch. 12.1-12.2		0: Matrix Calculus, 1: KKT		
3	Lecture 3 Optimality conditons and linear algebra - N&W Ch.12.3, 12.5 (12.8, 12.9)	Lecture 4 Linear Programming - N&W Ch.13.1-13.5		2: LP	0, 1, 2	
4	Lecture 5 Linear Programming - N&W Ch.13.1-13.5	Lecture 6 Quadratic programming - N&W Ch.15.3-15.5, 16.1-2,4-5		3: LPQP	2, 3	0, 1
5	Lecture 7 Quadratic Programming - N&W Ch.15.3-15.5, 16.1-2,4-5	Lecture 8 Open loop dynamic optimization - MPC note Ch.3-3.2	Helicopter Lab week	4: QP	3, 4	2
6	Lecture 9 Model predictive control - MPC note Ch.3.3-4.2.1	Lecture 10 Model predictive control - MPC note Ch.4.2.2-4.3.1	Helicopter Lab week	5: OLMPC	4, 5	3
7	Lecture 11 Linear quadratic control - MPC note Ch.4.3.2-4.4	Lecture 12 Linear quadratic control - MPC note repetition and 4.6	Helicopter Lab week	6: MPCLQR	5	4
8	Lecture 13 Unconstrained optimization - N&W Ch.2.1-2.2	No lecture	Helicopter Lab week		5, 6	

- Updated schedule will be available on Blackboard

NTNU | Norwegian University of Science and Technology

TTK4135 – Lecture 1
Optimization: What and Why?

Spring 2021

Lecturer: Lars Imsland

Purpose of Lecture

- Brief Timeline & Motivation

- Formulation of optimization problems, classes of optimization problems

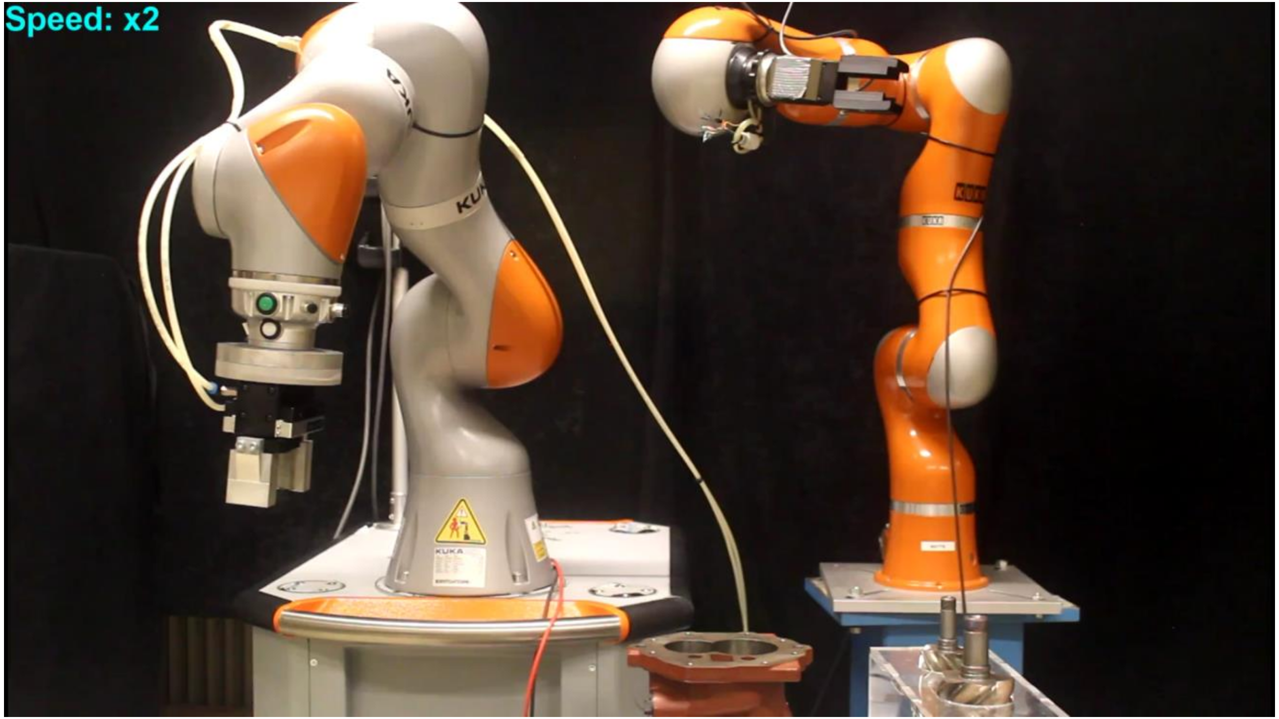
- Definition of important terms
  – Convexity and non-convexity
  – Global vs. local solution
  – Constrained vs. unconstrained problems
  – Feasible set

Reference: Chapter 1 Nocedal & Wright

Brief Timeline

~1600 BC	Ancient Geometry: Babylonian method for solving $x^2 + bx = c$
~300 BC	Ancient Geometry: Euclid's minimal distance between point and line
~200s	Iterative approaches: Han Dynasty methods for solving $\sum_{i=0}^{3} a_i x^i = 0$
~900s	Modern algebra and arithmetics: Muhammad Al-Khwarizmi ("Algorismi") gives various root solving methods
1600s	Basis of Calculus of Variations: Newton's Body of minimal resistance, Bernoulli's Brachistochrone problem
1700s	Calculus of Variations and combinatorial optimization: Maupertius' Principle of Least Action, Samuel König's optimal honeycomb
1800s	First "Optimization algorithms": Hamilton-Jacobi Equation, Extreme Value Theorem, Rolle's Theorem, Cauchy's Gradient Descent
1900-1957	Rigorous theory and applications: Minkowski's Convex Sets, Hancock's Theory of Minima and Maxima, Kantorevich's Linear Optimization Problems, Dantzig's Simplex method, Neumann and Morgenstern's Dynamic Programming, Karush-Kuhn-Tucker's Optimality Conditions, Bellman's Optimality principle, Pontryagin's Maximum Principle
1950+	Optimization is applied to economics, agriculture, space travel, social media, robots, manufacturing, art and everything in between

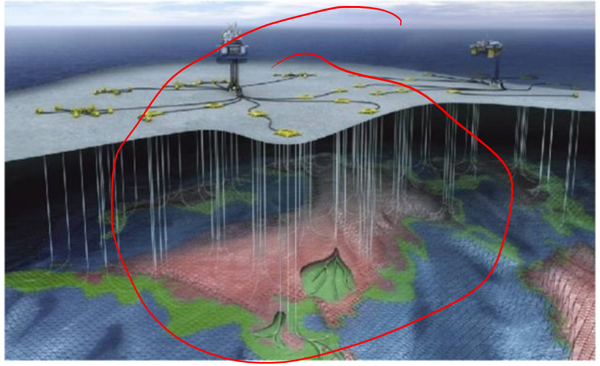
Modern Applications: Motion Control

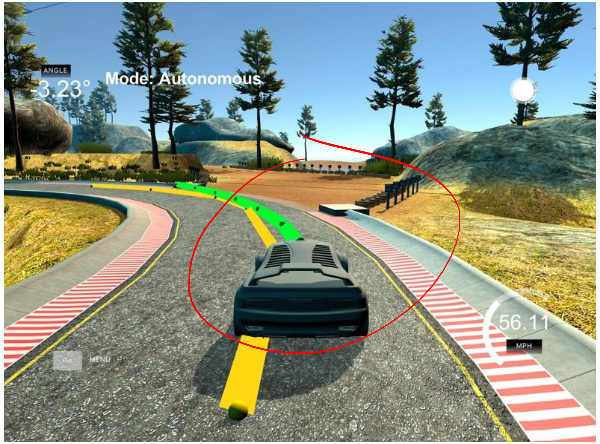


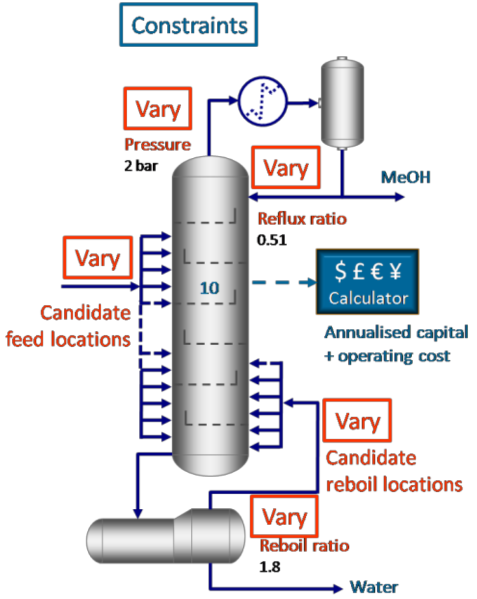
LBR IIWA avoiding collision with static LBR 4+ while performing assembly

Source: Y. Pane, M. H. Arbo, E. Aertbeliën, W. Decré, “A System Architecture for CAD-Based Robotic Assembly with Sensor-Based Skills”, T-ASE 2019

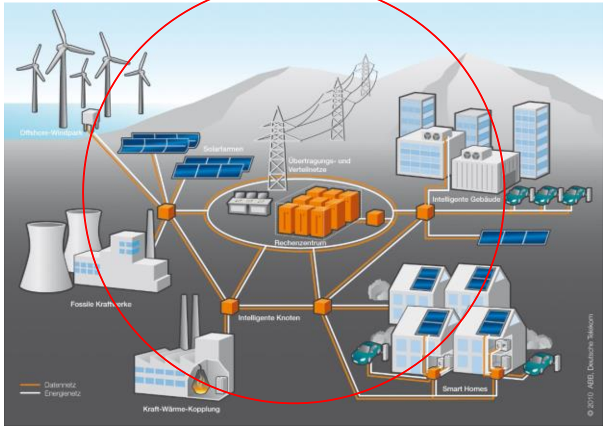
Control Applications: Model Predictive Control for all domains

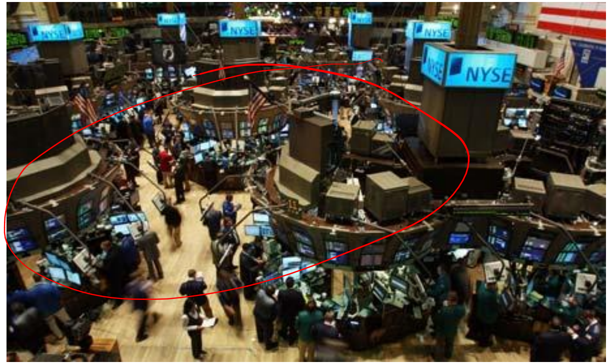


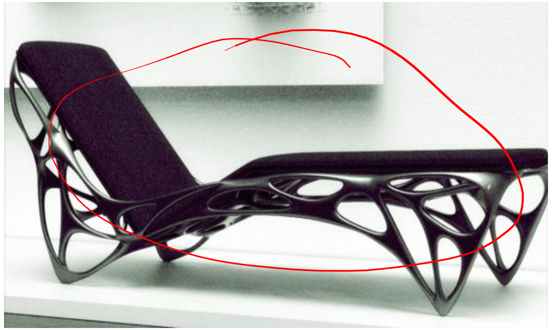


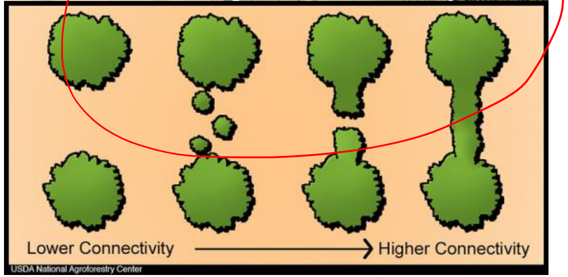


Lots of other applications





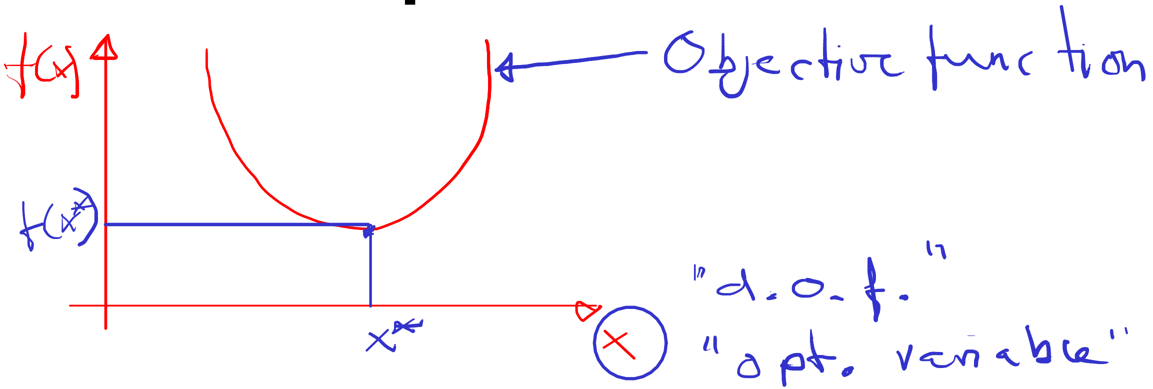




What is optimization?

unconstrained opt.

$$\boxed{\min_{x \in \mathbb{R}^n} f(x)}$$



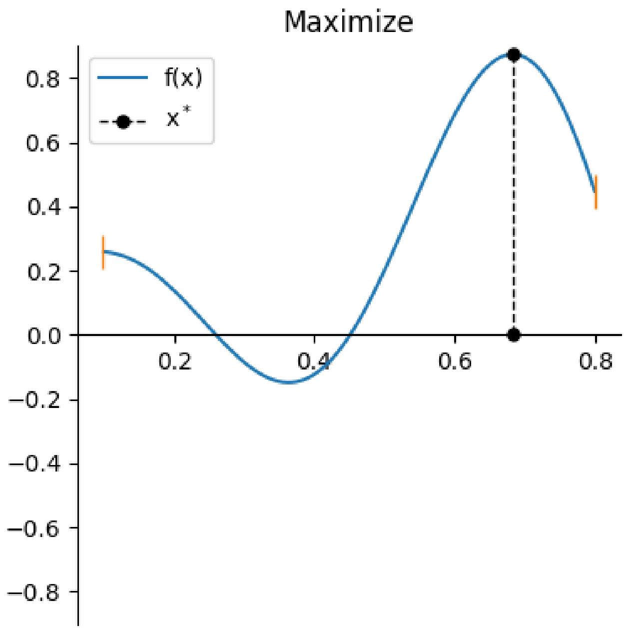
"d.o.f."
"opt. variable"

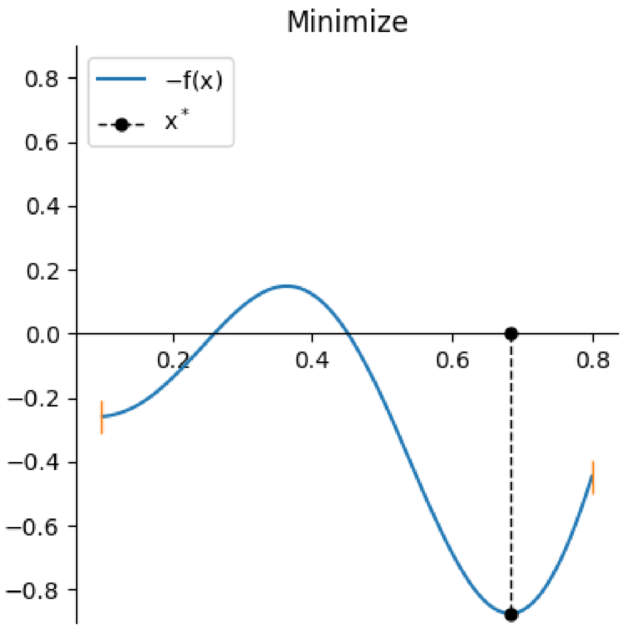
What characterizes an optimum?

Necessary conditions for optimality of x^*:

1. order: $f'(x^*) = 0$
2. order: $f''(x^*) \geq 0$

Minimization or Maximization?

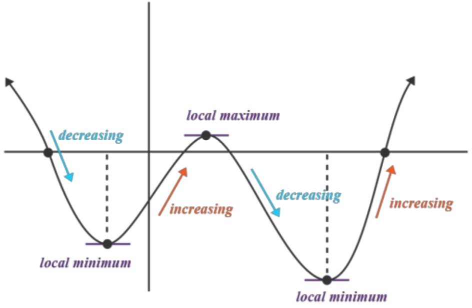


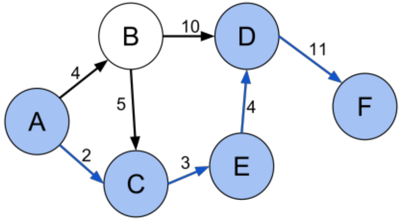


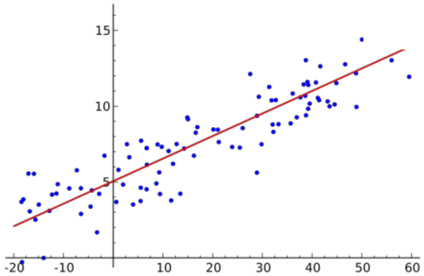
Convention this course: Minimization!

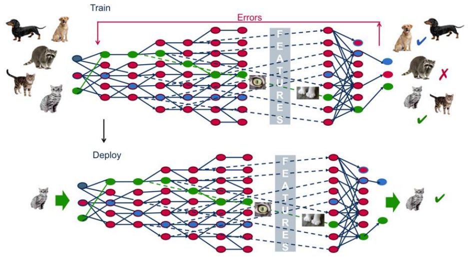
Optimization – A recurring friend

- Finding max and min of a function (Calc 1)
- The Lagrange Method (Calc 2)
- Algorithms course (Shortest path, dynamic programming, max flow, travelling salesman, etc)
- Statistics (Least-squares, data fitting)
- Machine Learning (Gradient descent)
- (And many applications in control…)







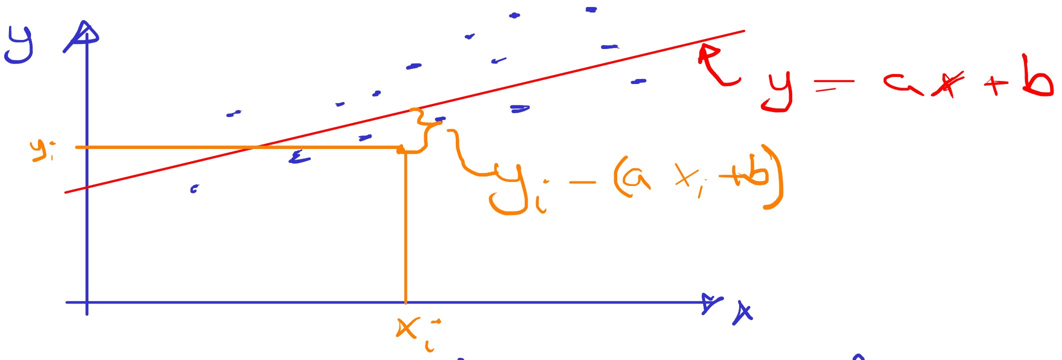


Example optimization problem: Least Squares

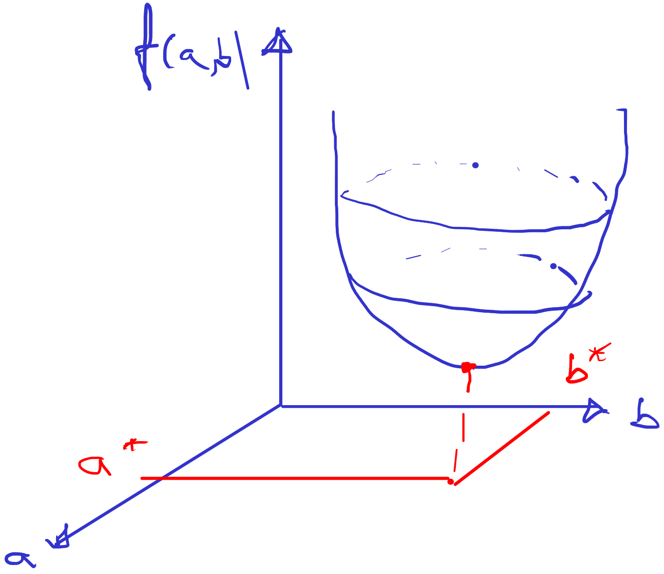
x_i : Temperature day i

y_i : # ice creams sold day i

Previous data



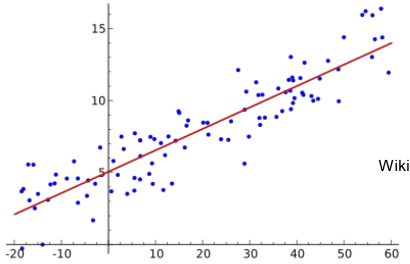
Find a, b: $\min_{a,b} \underbrace{\sum_{i=1}^{n} (y_i - a x_i - b)^2}_{f(a,b)}$



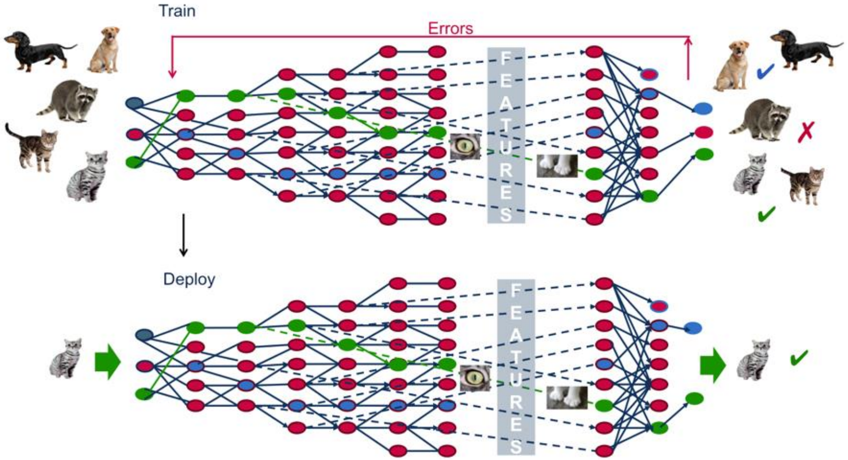
Predict for $y = a^* x + b^*$

Example: Machine Learning

- Learn, and make predictions, from data
- Linear regression is the most basic ML algorithm, solved using optimization
  - "Least squares", Ch. 10, N&W
- In a similar fashion: ML, neural networks, deep learning etc. are "trained" using "gradient descent" algorithms
  - Topic of Ch. 2-10, N&W



Wikipedia.org



Constrained optimization problems

Unconstr. opt.

$$\min_{x \in \mathbb{R}^n} f(x)$$

Constrained opt.

$$\boxed{\begin{aligned} &\min_{x \in \mathbb{R}^n} f(x) \\ &\text{s.t.} \quad c_i(x) = 0, \quad i \in \mathcal{E} \\ &\qquad\quad c_i(x) \geq 0, \quad i \in \mathcal{I} \end{aligned}}$$

Feasible set:

$$\Omega = \{ x \in \mathbb{R}^n \mid c_i(x) = 0, \ i \in \mathcal{E} \text{ and } c_i(x) \geq 0, \ i \in \mathcal{I} \}$$

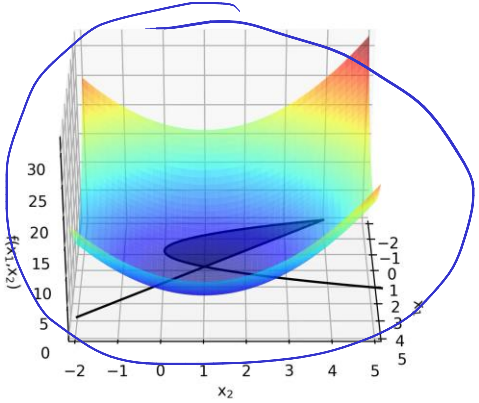
General Optimization Problem

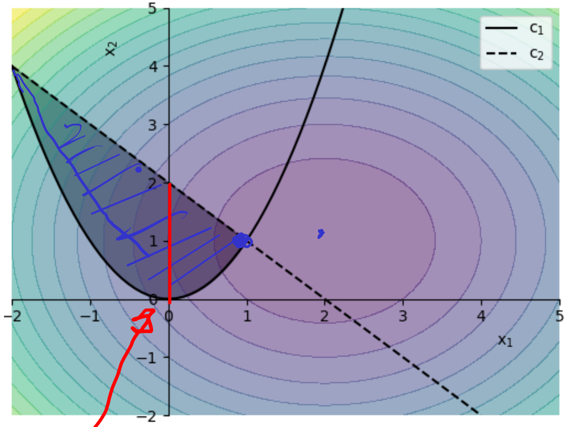
$$\min_{x \in \mathbb{R}^n} f(x) \quad \text{subject to} \quad \begin{cases} c_i(x) = 0, & i \in \mathcal{E}, \\ c_i(x) \geq 0, & i \in \mathcal{I}. \end{cases}$$

$\{1, 2\}$

- Example:

$$\min\ (x_1 - 2)^2 + (x_2 - 1)^2 \quad \text{subject to} \quad \begin{array}{l} x_1^2 - x_2 \leq 0, \quad \leftarrow c_1 \\ x_1 + x_2 \leq 2. \quad \leftarrow c_2 \end{array}$$





- What if we add equality-constraint $x_1 = 0$?

Linear Programming

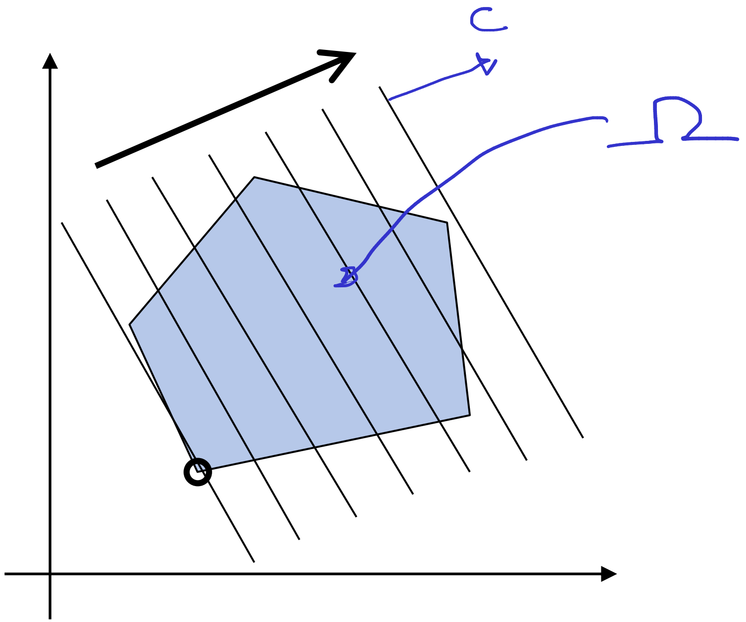
$$\min_x \; c^T x$$

$$\text{s.t.} \quad a_i^T x = b_i, \; i \in \mathcal{E}$$

$$a_i^T x \geq b_i, \; i \in \mathcal{I}$$

$\cdot(-1)$

$$\boxed{\min \; c^T x \quad \text{s.t.} \quad Ax \leq b, \quad x \geq 0}$$

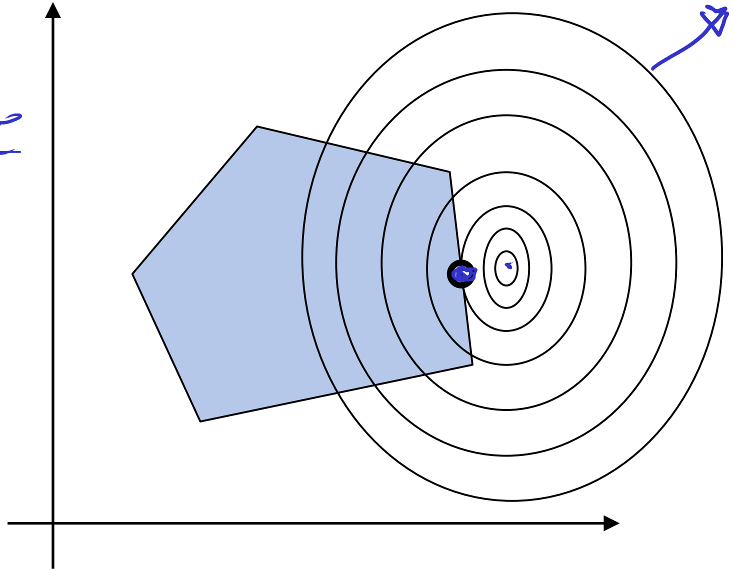


Quadratic Programming

$$\min_x \quad q(x) = \frac{1}{2} x^T G x + d^T x$$

$$\text{s.t} \quad a_i^T x = b_i \,, \quad i \in \mathcal{E}$$

$$a_i^T x \geq b_i \,, \quad i \in \mathcal{I}$$



LP Example: Farming

- A farmer wants to grow apples (A) and bananas (B)
- He has a field of size 100 000 m^2
- Growing 1 tonne of A requires an area of 4 000 m^2, growing 1 tonne of B requires an area of 3 000 m^2
- A requires 60 kg fertilizer per tonne grown, B requires 80 kg fertilizer per tonne grown
- The profit for A is 7000 per tonne (including fertilizer cost), the profit for B is 6000 per tonne (including fertilizer cost)
- The farmer can legally use up to 2000 kg of fertilizer

- The farmer wants to maximize his profits

Formulating a LP optimization problem

x_1: # tonnes A

x_2: # tonnes B

Objective $7000 \cdot x_1 + 6000 \cdot x_2$

Constraints

$$4000\, x_1 + 3000\, x_2 \leq 100\,000$$
$$60\, x_1 + 80\, x_2 \leq 2000$$

$$\min_{x_1, x_2} \; -7000\, x_1 - 6000\, x_2$$

s.t.

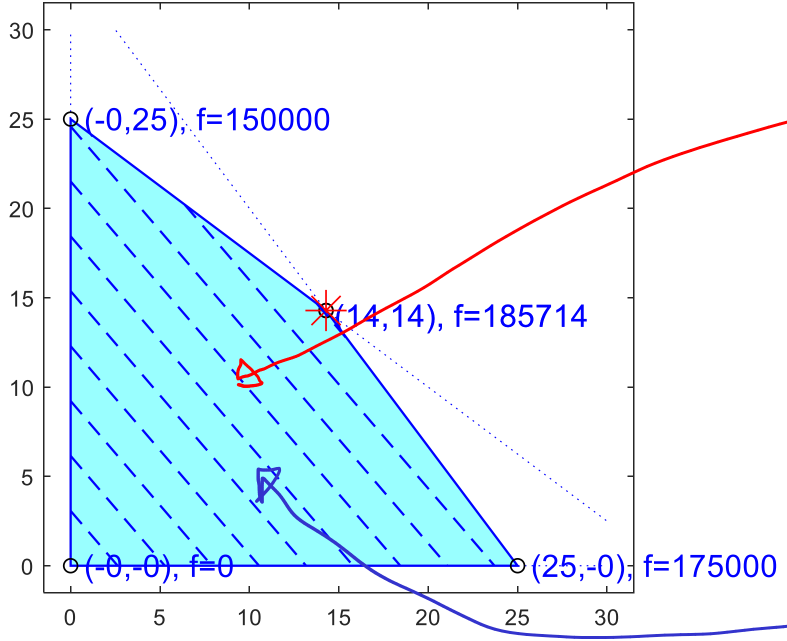
$$x_1 \geq 0$$
$$x_2 \geq 0$$

$\min c^T x$ s.t. $Ax \leq b,\ x \geq 0$

$$c = [-7000, -6000]$$

$$A = \begin{bmatrix} 4000 & 3000 \\ 60 & 80 \end{bmatrix}, \quad b = \begin{bmatrix} 100000 \\ 2000 \end{bmatrix}$$

Farming Example: Geometric Interpretation and Solution



$$
\begin{aligned}
\max_{x_1,\, x_2} \quad & 7000x_1 + 6000x_2 \\
\text{subject to:} \quad & 4000x_1 + 3000x_2 \leq 100000 \\
& 60x_1 + 80x_2 \leq 2000 \\
& x_1 \geq 0 \\
& x_2 \geq 0
\end{aligned}
$$