Optimisation and Decision Models

MSc Business Analytics 2016/17 — Wolfram Wiesemann

Imperial College London
BUSINESS SCHOOL

Content

Imperial College London
BUSINESS SCHOOL

Content

Imperial College London
BUSINESS SCHOOL

Course Objectives

This course introduces the *theory and application of modern optimisation* from a *business analytics perspective*.

The course should enable you to:

1. formulate *decision problems* in management and analytics as *mathematical optimisation problems*
2. solve the resulting optimisation problems using state-of-the-art *optimisation software packages*
3. appreciate the *computational complexity* of various classes of optimisation problems
4. get a *strong foundation in optimisation* in order to tackle later MSc courses (e.g. Machine Learning)

Imperial College London
BUSINESS SCHOOL

Course Philosophy

This course introduces the *theory and application of modern optimisation* from a *business analytics perspective*.

In order to achieve those goals within 20 hours, we prefer:

1. *modelling techniques* over *optimisation algorithms*
2. *intuitive (often visual) explanations* over *lengthy proofs*
3. *transferrable concepts* over *isolated facts*

This course will be tough, but we are here to help!

Imperial College London
BUSINESS SCHOOL

Course Structure

PART ONE: Linear Optimisation

- formulating linear programs
- solving linear programs
- linear programming duality

1.5 hours tutorial

8 hours lecture

PART TWO: Discrete Optimisation

- formulating discrete programs
- logical constraints
- solving discrete programs

1.5 hours tutorial

8 hours lecture

PART THREE: Nonlinear Optimisation

- recognising convexity
- special classes of convex programs
- solving convex programs

1 hour tutorial

4 hours lecture

4 individual homeworks: count for 40% of the final grade

4 quizzes: count for 10% of the final grade

Imperial College London
BUSINESS SCHOOL

Course Structure

PART ONE: Linear Optimisation

- formulating linear programs
- solving linear programs
- linear programming duality

PART TWO: Discrete Optimisation

- formulating discrete programs
- logical constraints
- solving discrete programs

PART THREE: Nonlinear Optimisation

- recognising convexity
- special classes of convex programs
- solving convex programs

1.5 hours tutorial
8 hours lecture
4 individual homeworks: count for 40% of the final grade
4 quizzes: count for 10% of the final grade

1.5 hours tutorial
8 hours lecture

1 hour tutorial
4 hours lecture

+2h final exam

Imperial College London
BUSINESS SCHOOL

Timetabling Nightmare!

Mon 10	Tue 11	Wed 12	Thu 13	Fri 14
		11:30-13:30 ACEX 554		9:30-11:30 LT 3
				13:00-14:00 LT 2

Mon 17/24	Tue 18/25	Wed 19/26	Thu 20/27	Fri 21/28
	16:00-17:00 ACEX 554	11:30-13:30 ACEX 554		11:30-12:30 LT 2
				13:30-15:30 LT 2

Mon 31	Tue 1	Wed 2	Thu 3	Fri 4
	16:00-17:00 ACEX 554	11:30-13:30 **LT 3**		11:30-12:30 LT 2
				13:30-15:30 LT 2

Tue 8
12:00-13:00 LT 1

Lecture

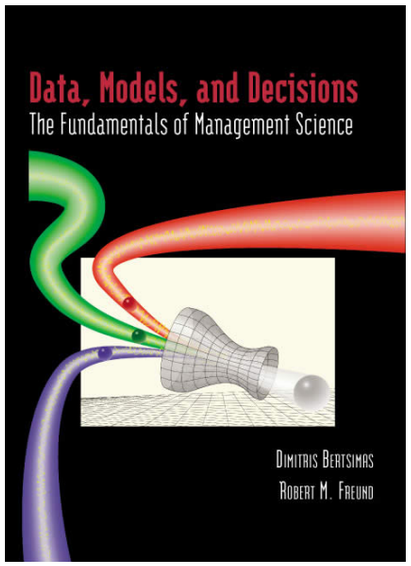
Tutorial

Quiz

Assignment

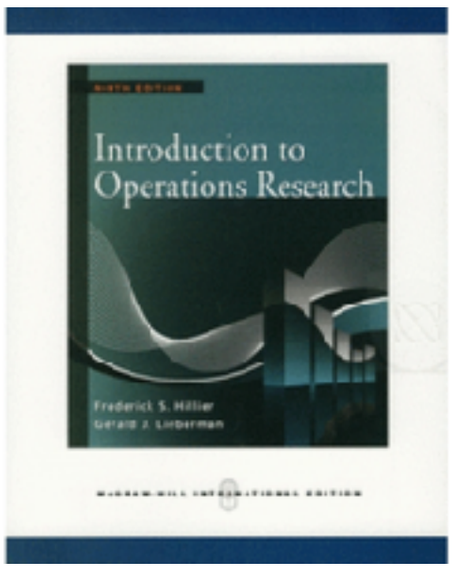
Imperial College London
BUSINESS SCHOOL

Recommended Textbooks



Bertsimas/Freund: ***Data, Models, and Decisions***

- standard MBA textbook at MIT Sloan, HBS, Wharton, Columbia, Northwestern, …
- good for basic concepts, exercises, MS Excel
- copies available at Central Library

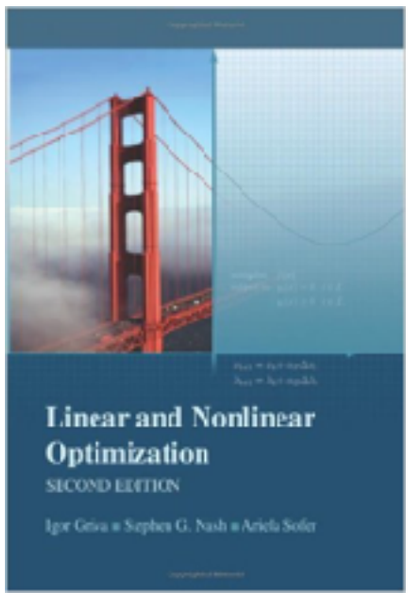


Hillier/Lieberman: ***Introduction to OR***

- prize-winning Industrial Engineering classic (first version 1967), used throughout the world
- good coverage of duality + logical constraints
- copies available at Central Library

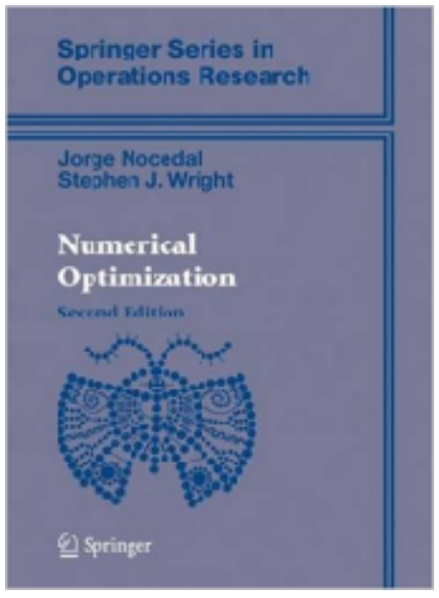
Imperial College London
BUSINESS SCHOOL

Recommended Textbooks



Griva/Nash/Sofer: ***Linear and Nonlinear Opt.***

- good coverage of nonlinear optimisation
- goes beyond what is discussed in class
- 1 copy available at Central Library



Nocedal/Wright: ***Numerical Optimization***

- classic textbook for nonlinear optimisation
  covers many details of real-life implementations
- goes beyond what is discussed in class
- copies available at Central Library

Imperial College London
BUSINESS SCHOOL

Content

Mathematical Optimisation

Leonhard Euler:*

"...nothing at all takes place in the universe in which some rule of maximum or minimum does not appear..."

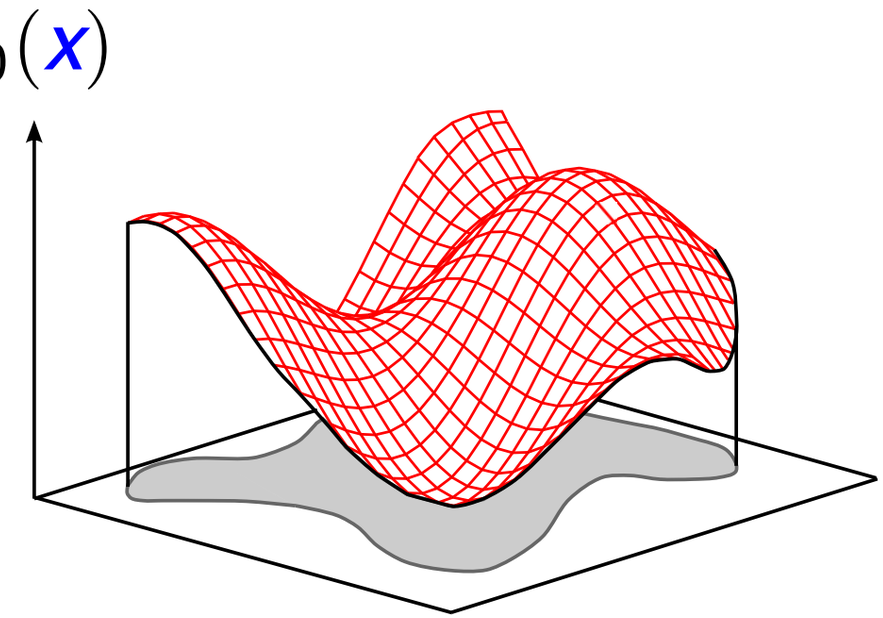
* Leonhard Euler. *Methodus inviendi lineas curvas maximi minimive proprietate gaudentes, sive, solutio problematis isoperimetrici latissimo sensu accepti*. Lausannæ; Genevæ: M.-M. Bousquet, 1744, p. 245.

Imperial College London
BUSINESS SCHOOL

The Generic Optimisation Problem

The generic optimisation problem:

$$\begin{aligned} \underset{x}{\text{minimise}} \quad & f_0(x) \\ \text{subject to} \quad & x \in \mathcal{X} \end{aligned}$$



where:

- $x \in \mathbb{R}^n$ is the vector of *decision variables*
- $f_0 : \mathbb{R}^n \mapsto \mathbb{R}$ is the *objective function* (e.g. costs of x)
- $\mathcal{X} \subseteq \mathbb{R}^n$ is the *feasible region* (admissible values of x)

$\mathcal{X}$

The feasible region is typically expressed through *constraints*:

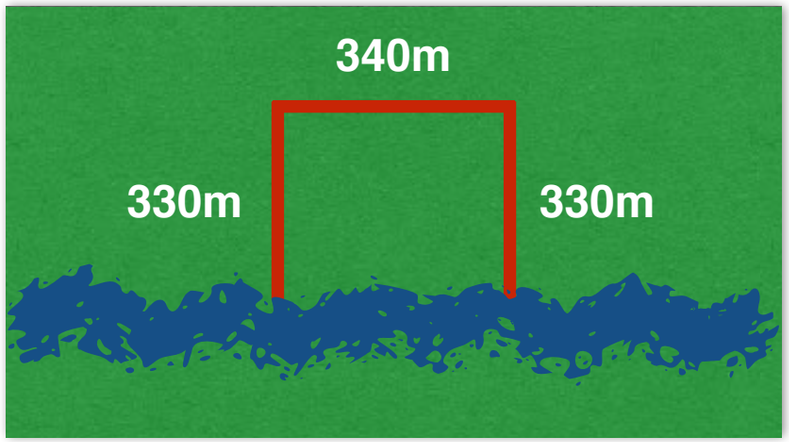
$$\begin{aligned} f_i(x) &\leq 0 \qquad \forall i = 1, \ldots, m \\ h_i(x) &= 0 \qquad \forall i = 1, \ldots, p \end{aligned}$$

where $f_i, h_i : \mathbb{R}^n \mapsto \mathbb{R}$.

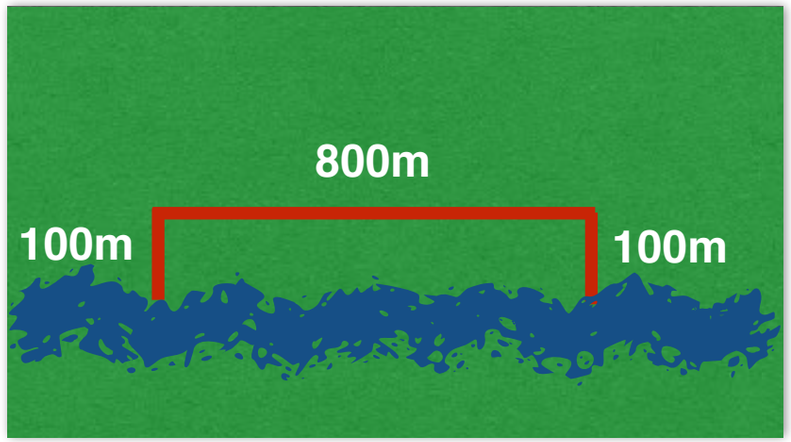
Imperial College London
BUSINESS SCHOOL

Example*

A farmer has 1000m of fencing with which he wants to use to fence off a rectangular field that borders a straight river. Assuming that he does not need any fence along the river, what are the dimensions of the field that maximises the overall area?



area: 330m * 340m = 112,200m^2



area: 100m * 800m = 80,000m^2

$$\begin{aligned} \underset{w,h}{\text{maximise}} \quad & w \cdot h \\ \text{subject to} \quad & w + 2h \leq 1000 \\ & w, h \geq 0 \end{aligned}$$

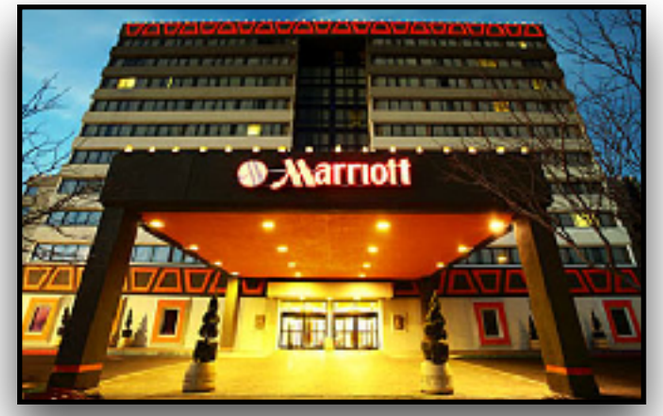
where:

- w is the field's width
- h is the field's height

* Adapted from https://cims.nyu.edu/~kiryl/Calculus/Section_4.5--Optimization%20Problems/Optimization_Problems.pdf.

Imperial College London
BUSINESS SCHOOL

Other Applications



Marriott International developed an optimisation-based real-time pricing strategy that improved hotel revenues by $47m p.a.

Sloan Kettering Cancer Center improved prostrate cancer therapeutics using optimisation, offering safer treatment and $459m savings p.a.

TNT Express developed optimisation-based supply chain design and routing solutions that saved €207m over 4 years

dunnhumby uses optimisation to place, promote and price products for Tesco, P&G, Unilever, …

Wien Energie uses optimisation to run a cascade of hydropower plants at Ottenstein Stausee (Austria)

Imperial College London
BUSINESS SCHOOL

Content

Imperial College London
BUSINESS SCHOOL

Feasible Solutions, Minimisers

$$\begin{aligned} \underset{x}{\text{minimise}} \quad & f_0(x) \\ \text{subject to} \quad & x \in \mathcal{X} \end{aligned}$$

- x is a *feasible solution* if $x \in \mathcal{X}$, that is, if

$$f_i(x) \leq 0 \qquad \forall i = 1, \ldots, m$$
$$h_i(x) = 0 \qquad \forall i = 1, \ldots, p$$

- x^* is a *(global) minimiser* if x^* is feasible and $f_0(x^\star) \leq f_0(x)$ for any feasible solution x; $f_0(x^\star)$ is the *(global) minimum*
- x^* is a *local minimiser* if x^* is feasible and $f_0(x^\star) \leq f_0(x)$ for for any feasible solution x "close to" x^*; $f_0(x^\star)$ is the *local minimum*. Formally put:

  $\exists \delta > 0$ such that $f_0(x^\star) \leq f_0(x)$ for all $x \in \mathcal{X}$ with $\|x - x^\star\|_2 \leq \delta$

Imperial College London
BUSINESS SCHOOL

Near-Optimal Solutions

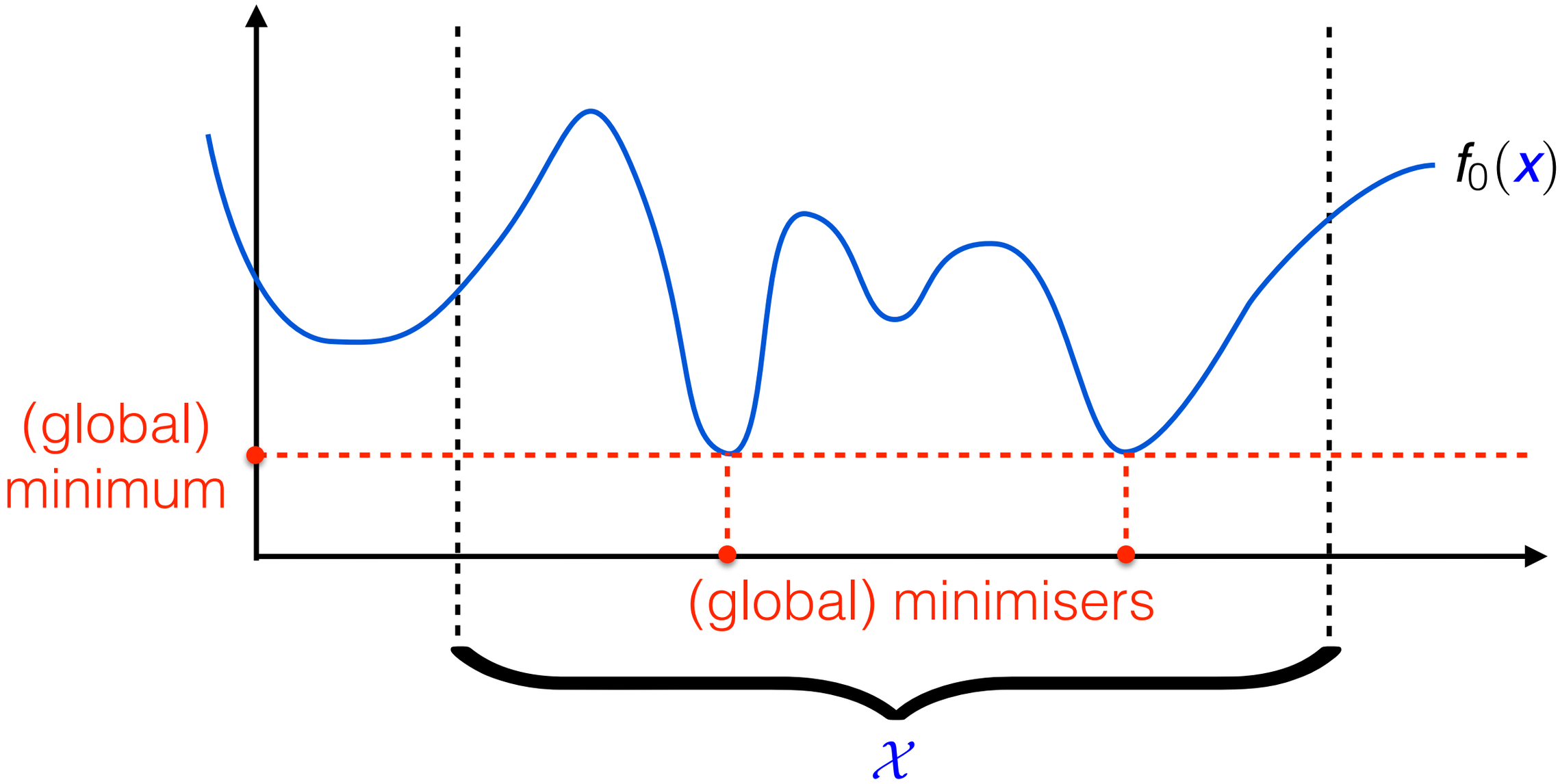
$$\begin{aligned} \underset{x}{\text{minimise}} \quad & f_0(x) \\ \text{subject to} \quad & x \in \mathcal{X} \end{aligned}$$

- for $\varepsilon > 0$, x^* is called an *ε-(global) minimiser* if $x^\star \in \mathcal{X}$ and $f_0(x^\star) \leq f_0(x) + \varepsilon$ for every feasible solution $x \in \mathcal{X}$
- one can similarly define *ε-local minimisers*
- an optimisation problem is either:
  - *infeasible* if the set $\mathcal{X}$ is empty
  - *feasible and solvable* if it has at least one minimiser
  - *feasible but unbounded* if for every value θ there is a solution x that satisfies $f(x) \leq \theta$
  - *feasible and bounded but not solvable* if for every feas. sol. x there is a feas. sol. x' such that $f_0(x') < f_0(x)$

Imperial College London
BUSINESS SCHOOL

Summary of Terminology

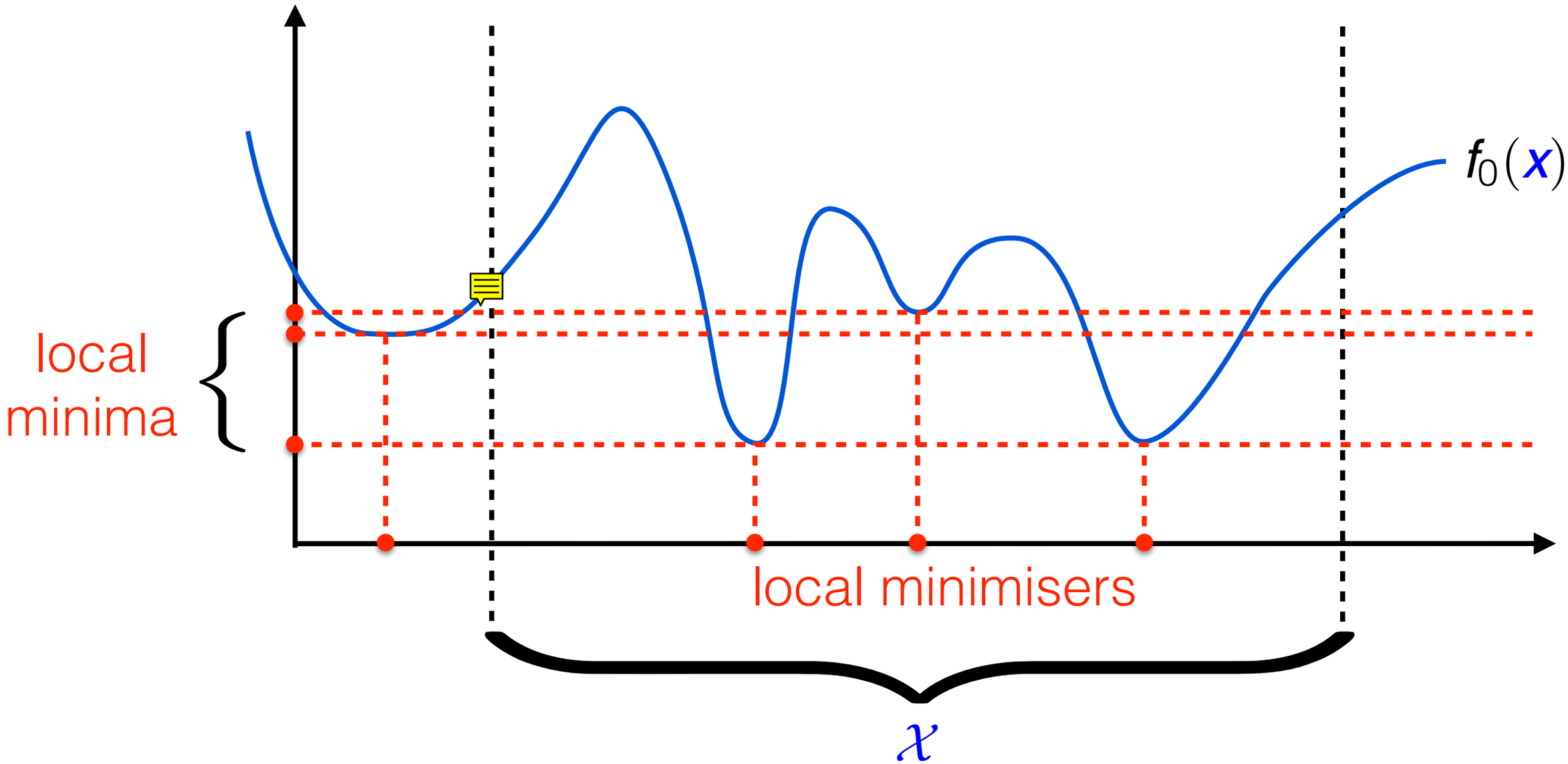
$$\begin{aligned} &\underset{x}{\text{minimise}} && f_0(x) \\ &\text{subject to} && x \in \mathcal{X} \end{aligned}$$



Imperial College London
BUSINESS SCHOOL

Summary of Terminology

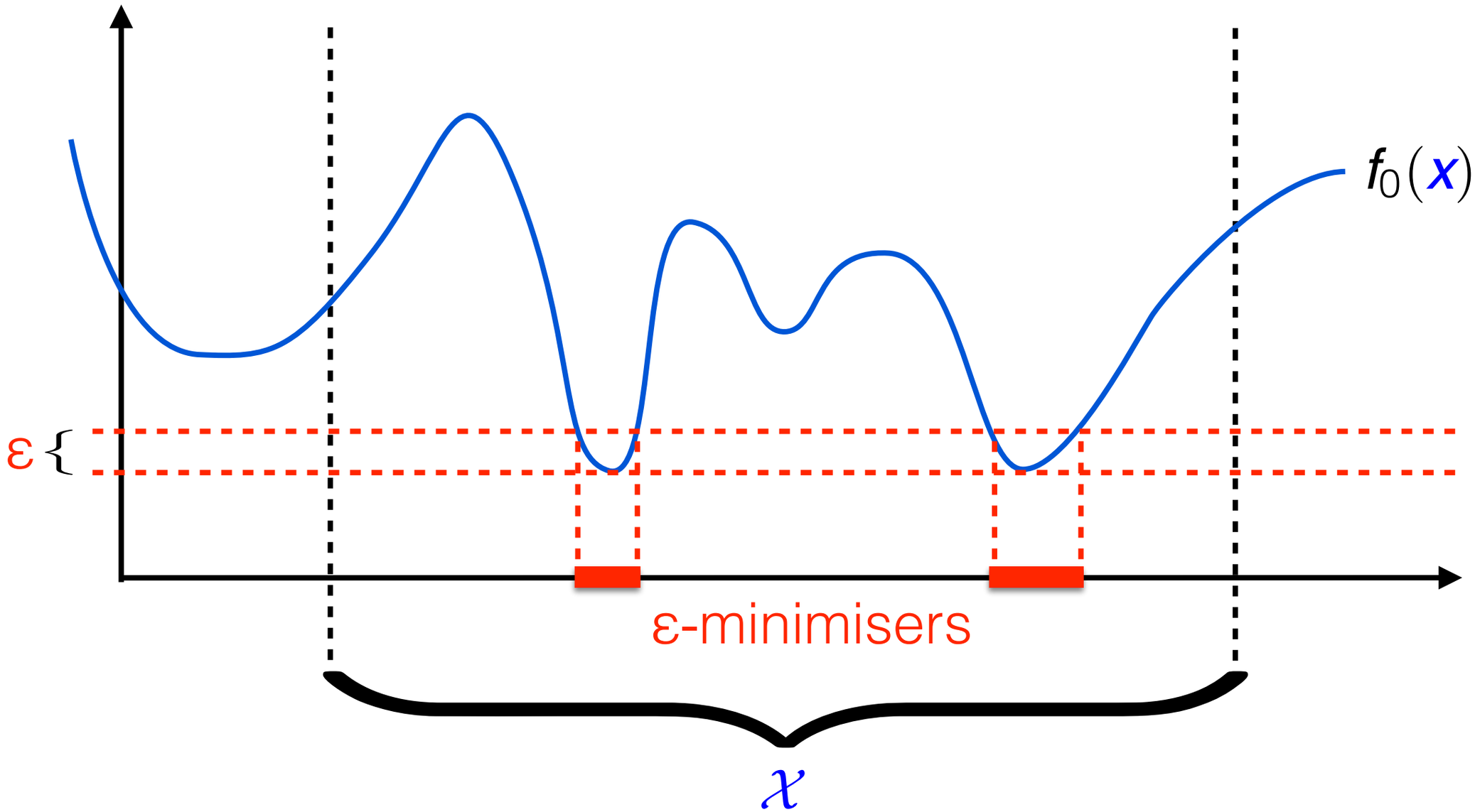
$$\begin{aligned} &\underset{x}{\text{minimise}} && f_0(x) \\ &\text{subject to} && x \in \mathcal{X} \end{aligned}$$



Imperial College London
BUSINESS SCHOOL

Summary of Terminology

$$\begin{aligned} &\underset{x}{\text{minimise}} && f_0(x) \\ &\text{subject to} && x \in \mathcal{X} \end{aligned}$$



Imperial College London
BUSINESS SCHOOL

Content

Imperial College London
BUSINESS SCHOOL

Solving "Simple" Problems

How long (in terms of # of operations) does the fastest algorithm take to solve a "simple" optimisation problem of the form

$$\begin{aligned} \underset{x}{\text{minimise}} \quad & f_0(x) \\ \text{subject to} \quad & x \in [0,1]^n \end{aligned}$$

where f_0 is Lipschitz-continuous with Lipschitz constant L?[1]

Answer:[2] To find an ε-minimiser for the problem, any algorithm requires at least $\left(\left\lfloor \frac{L}{2\varepsilon} \right\rfloor\right)^n$ evaluations of f_0.

[1] A *Lipschitz-continuous function f* is a continuous function whose Lipschitz constant L describes how "rapidly" f changes ($L = 0$: f is flat; $L \longrightarrow \infty$: f can change very rapidly).

[2] Proof: Y. Nesterov. *Introductory Lectures on Convex Optimization*. Kluwer, 2004. (Not required!)

Imperial College London
BUSINESS SCHOOL

Solving "Simple" Problems

How long (in terms of # of operations) does the fastest algorithm take to solve a "simple" optimisation problem of the form

$$\begin{aligned} \underset{x}{\text{minimise}} \quad & f_0(x) \\ \text{subject to} \quad & x \in [0,1]^n \end{aligned}$$

where f_0 is Lipschitz-continuous with Lipschitz constant L?

Example: $L = 2$, $n = 10$, $\varepsilon = 0.01$ (tiny problem, low accuracy)

We need $\left(\left\lfloor \frac{2}{2 \cdot 0.01} \right\rfloor\right)^{10} = 10^{20}$ evaluations of f_0.

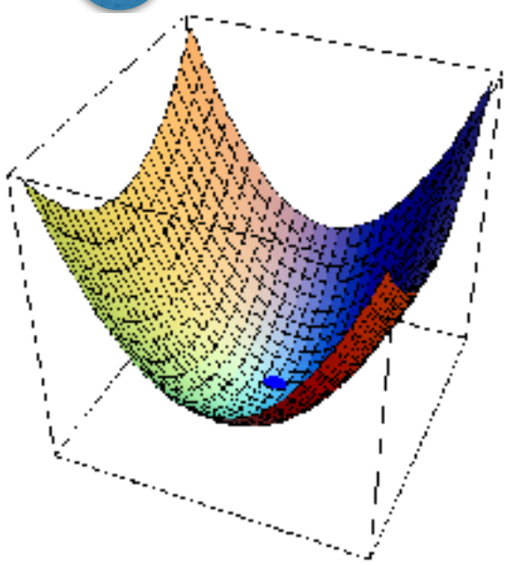
If each evaluation requires 10 arithmetic operations and our computer can execute 10^6 arithmetic operations per second, then it takes **31,250,000 years**!

Imperial College London
BUSINESS SCHOOL

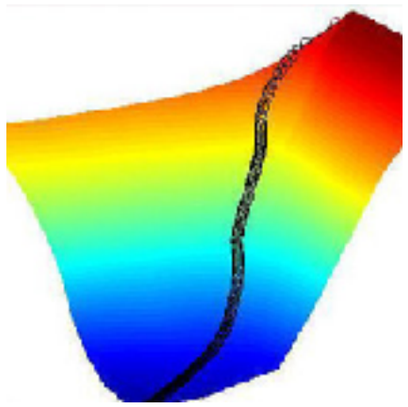
Structure is Key!

We can do much better, but for that we need:

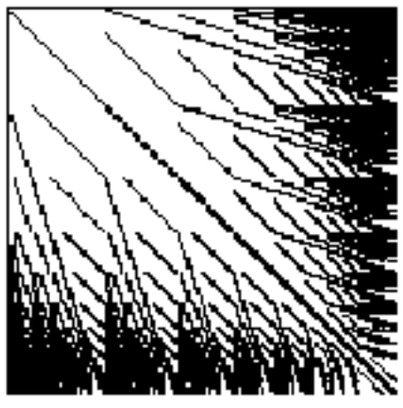
1. **Convexity:** implies that local minimum = global minimum



2. **Availability of derivatives:** helps us find a local minimum



3. **Sparsity:** each constraint involves only a few variables; makes underlying algebraic operations fast (e.g. matrix inverses)



*We can then reliable solve problems with **10,000,000 variables**!*