Introduction to LP modelling

Sapienza Università di Roma

Outline

What we shall do

Linear models and acronyms used

Notation

A deliberate choice: continuous variables only

Map of the models

Structure of each chapter

Course organization

Learning objectives (1/2)

At the end of this chapter you will be able to:

- describe the path of every chapter: managerial problem, model, Python and solver, interpretation;
- use the notation of the course for data, variables, duals, reduced costs and slacks;

Learning objectives (2/2)

At the end of this chapter you will be able to:

- explain why the course uses continuous variables only and what this buys;
- find your way in the map of the models and in the structure of each chapter.

Prerequisites

- From the module *Linear Programming*, Chapter 4 «Linear Programming problems»: the form of a Linear Programming problem: objective function, linear constraints, domains of the variables
- From the module *Linear Programming*, Chapter 16 «Linear Programming modeling techniques»: the basic techniques to turn a statement into a linear programming model

What we shall do

Introduction: What we shall do

- In this section we see the **purpose** of the course
- We shall use linear programming as a tool for *decision making*, not merely for computation
- Understanding the goal guides us through all the sessions that follow

From the managerial problem to the model

- Every chapter starts from a **concrete managerial problem**:
  - how much to produce, how to charge a fleet
  - how efficient a branch is, how much risk to accept
- The problem becomes an **linear programming model**
- The model is implemented in **Python** and solved with a **solver**

Interrogating the model

- Above all, the model is *interrogated*:
  - how much is one extra hour of capacity worth?
  - does the solution hold up if the data change by 5%?
  - which trade-off would we recommend to a decision maker?

The right question

- At the end of every laboratory session the question is not only

  "what is the optimum?"

  but

  "which decision do we recommend and how robust is it?"

- The value of a model lies in the **quality of the recommendation** it allows us to formulate

Summary: What we shall do

- Managerial problem $\rightarrow$ model $\rightarrow$ Python and solver
- The model does not just return a number: it must be **interrogated**
- The final goal is a **robust recommendation** for the decision maker

Linear models and acronyms used

Linear models

- All the models of the course are **LP** (*Linear Programming*): the objective and the constraints are linear functions of the variables
- An LP is always **convex**: every local optimum is global and duality holds without exceptions
- The other recurring acronyms are defined at first use in the respective chapter

Prerequisites and neighbouring modules

- Prerequisite: the module **Linear Programming**, in particular duality and sensitivity analysis
- Models with a quadratic or nonlinear objective are in the module **NLP Modelling**

Notation

Introduction: Notation

- Throughout the course the **same notational convention** applies
- A uniform notation lets us read every model without ambiguity
- We see how scalars, sets, data, vectors, matrices, graphs, dual variables and solutions are written

Scalars, indices and sets

- **Scalars and indices**: lowercase italic letters (x_{it}, b_t, λ, i, t)
- **Sets of objects indexed by numbers**: the objects of a model (products, channels, securities, vehicles, districts, scenarios) are numbered

$$i \in \{1, 2, \ldots, n\}$$

  so that the notation of the text is already that of the model being built

- *Counts* are positive integers ($n \in \mathbb{Z}_{\geq 1}$)

Data, vectors, matrices and graphs

- **Numerical data**: rational values, $\mathbb{Q}_{>0}$ or $\mathbb{Q}_{\geq 0}$
  (the data of a real instance are always machine numbers)
- **Vectors**: lowercase boldface letters ($\boldsymbol{x}$, $\boldsymbol{w}$, $\boldsymbol{\mu}$); all vectors are column vectors and $\boldsymbol{a}'\boldsymbol{b}$ denotes the scalar product
- **Matrices**: uppercase boldface letters ($\boldsymbol{Q}$, $\boldsymbol{A}$); $\boldsymbol{Q} \succeq 0$ means positive semidefinite
- **Graphs**: $G = (N, A)$ with nodes N and arcs A (the only sets that are not enumerated, by their very nature)

Dual variables and solutions

- **Dual variables**: π_i, one per constraint
- **Reduced costs**: $\bar{c}_j$; **slacks**: $\bar{s}_i$
- **Solutions**:
  - unknown variables are written plain (x_j, π_i)
  - the *bar* denotes the values of a *feasible* solution $(\bar{x}_j, \bar{\pi}_i, \bar{s}_i, \bar{c}_j)$
  - the *tilde* those of an *optimal* one $(\tilde{x}_j, \tilde{\pi}_i, \tilde{z})$

How every model is written

- Every coefficient and every set is *defined before being used*
- In every model:
  - (i) the variables are introduced explicitly before the formulation
  - (ii) the objective and each family of constraints carry an equation number
  - (iii) the non-negativity constraints, which define the variables, close the model
  - (iv) after the model, a list describes the objective and the constraints family by family, with the number of constraints in each

Summary: Notation

- Lowercase italic for scalars and indices, boldface for vectors (lowercase) and matrices (uppercase)
- Enumerated sets $\{1, \ldots, n\}$; rational data
- π_i duals, $\bar{c}_j$ reduced costs, $\bar{s}_i$ slacks
- Bar = feasible solution, tilde = optimal solution
- Every model: variables, numbered objective and constraints, non-negativity, description

A deliberate choice: continuous variables only

Introduction: Continuous variables only

- All the models of the course use **exclusively continuous variables**
- No binary or integer variables
- It is a deliberate choice: let us see its **three motivations**

Three motivations

1. **Teaching**: with continuous variables we can rely on duality, shadow prices and reduced costs, the tools that turn a numerical solution into an economic explanation; with binary variables these tools are lost or weakened
2. **Coverage**: a surprising number of real decisions is naturally continuous: quantities, flows, portfolio weights, power levels, weights
3. **Progression**: the same tools (an LP, its dual, the shadow prices) serve production, finance and statistics, without changing solver

Summary: Continuous variables only

- Continuous variables only, by choice
- So **duality**, **shadow prices** and **reduced costs** apply
- Many real decisions are already continuous
- The same tools serve every chapter

Map of the models

Introduction: Map of the models

- We see the **applications** we shall tackle in the course
- For each one: the chapter, the class of the model and the central concept
- The map shows the path from deterministic applications to stochastic ones

Map of the models

Ch.	Application	Class	Central concept
4	Multi-period production and inventory	LP	duality, capacity
5	Electric vehicle charging	LP	energy, peaks
6	Data Envelopment Analysis	LP	relative efficiency
7	Newsvendor and variants	stoch. LP	deciding before knowing
8	VaR and CVaR	LP	tail risk
9	Arbitrage and pricing	LP	the duality that prices
10	Robust and quantile regression	LP	estimating with an LP

Summary: Map of the models

- Seven managerial, financial and estimation applications
- All LPs: deterministic and scenario-based stochastic ones
- Each application has a **central concept** to take home

Structure of each chapter

Introduction: Structure of each chapter

- Every application chapter follows the **same eight-step outline**
- The outline is designed to be replicated in the **laboratory reports**
- Knowing it helps you find your way in every chapter

The eight-step outline (1/2)

1. **Managerial motivation**: the problem told as a business decision
2. **Guided construction of the model**: sets, parameters, variables, constraints and objective, explained line by line
3. **Numerical example worked out by hand**: a minimal instance solved with all the steps, to understand the mechanism before switching on the solver
4. **Case study**: a realistic instance with reproducible data (CSV)

The eight-step outline (2/2)

5. **Python/Gurobi implementation**: the complete code, with comments
6. **Results**: the solver output translated into managerial language
7. **Sensitivity analysis**: shadow prices, scenarios, trade-offs
8. **Exercises**: from verification to a small research extension

Reproducible numbers

Observation 1: Warning

The numbers reported in each chapter are generated by the scripts in the `python/` folder: running `python3 run_all.py` regenerates data, tables and figures.

If you modify the data, the numbers in the lecture notes must be regenerated.

Summary: Structure of each chapter

- Motivation $\rightarrow$ model $\rightarrow$ example by hand $\rightarrow$ case study
- Code $\rightarrow$ results $\rightarrow$ sensitivity $\rightarrow$ exercises
- Use the same outline in your reports
- All the numbers can be regenerated by the Python scripts

Course organization

Introduction: Course organization

- We see the **essential path** of the course
- The details are in chapter 11

The essential path: four laboratory sessions

1. Production and inventory with duality and shadow prices
2. Data Envelopment Analysis and the envelopment form
3. Scenario-based decisions (newsvendor or CVaR)
4. A project of your own choice with a managerial presentation

Summary: Course organization

- Four sessions: production, DEA, scenarios, project

Final summary (1/2)

The purpose:

- Mathematical programming as a decision-making tool
- A model is solved and then **interrogated**

The tools:

- Linear models only: an LP is always convex
- Continuous variables only: duality, shadow prices, reduced costs

Final summary (2/2)

The method:

- A uniform notation in all models
- Seven applications, each with a central concept
- The same eight-step outline in every chapter

The course:

- Four lab sessions: production, DEA, scenarios, project