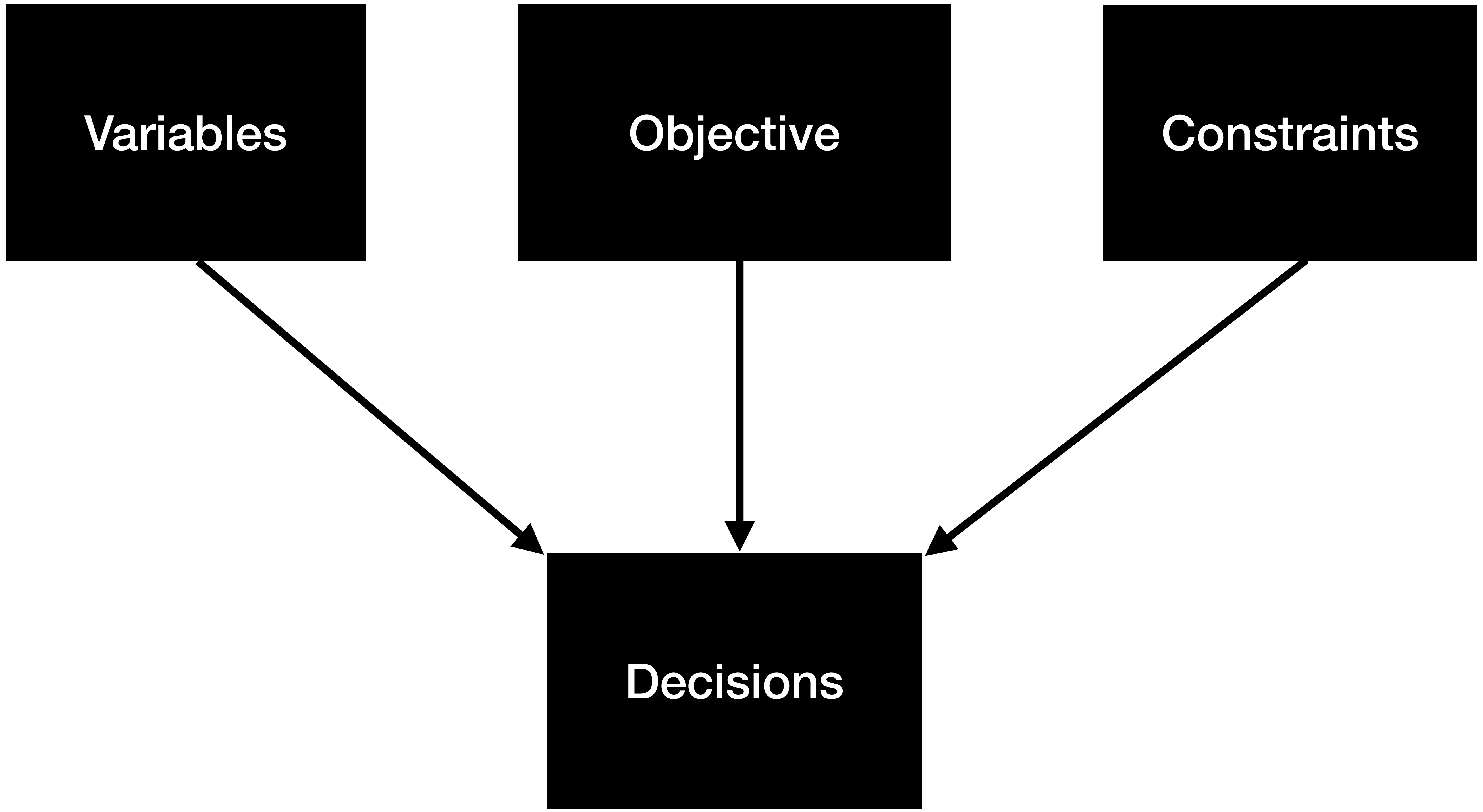
ORF522 – Linear and Nonlinear Optimization

1. Introduction

Bartolomeo Stellato - Fall 2021

What is this course about?

The mathematics behind making optimal decisions

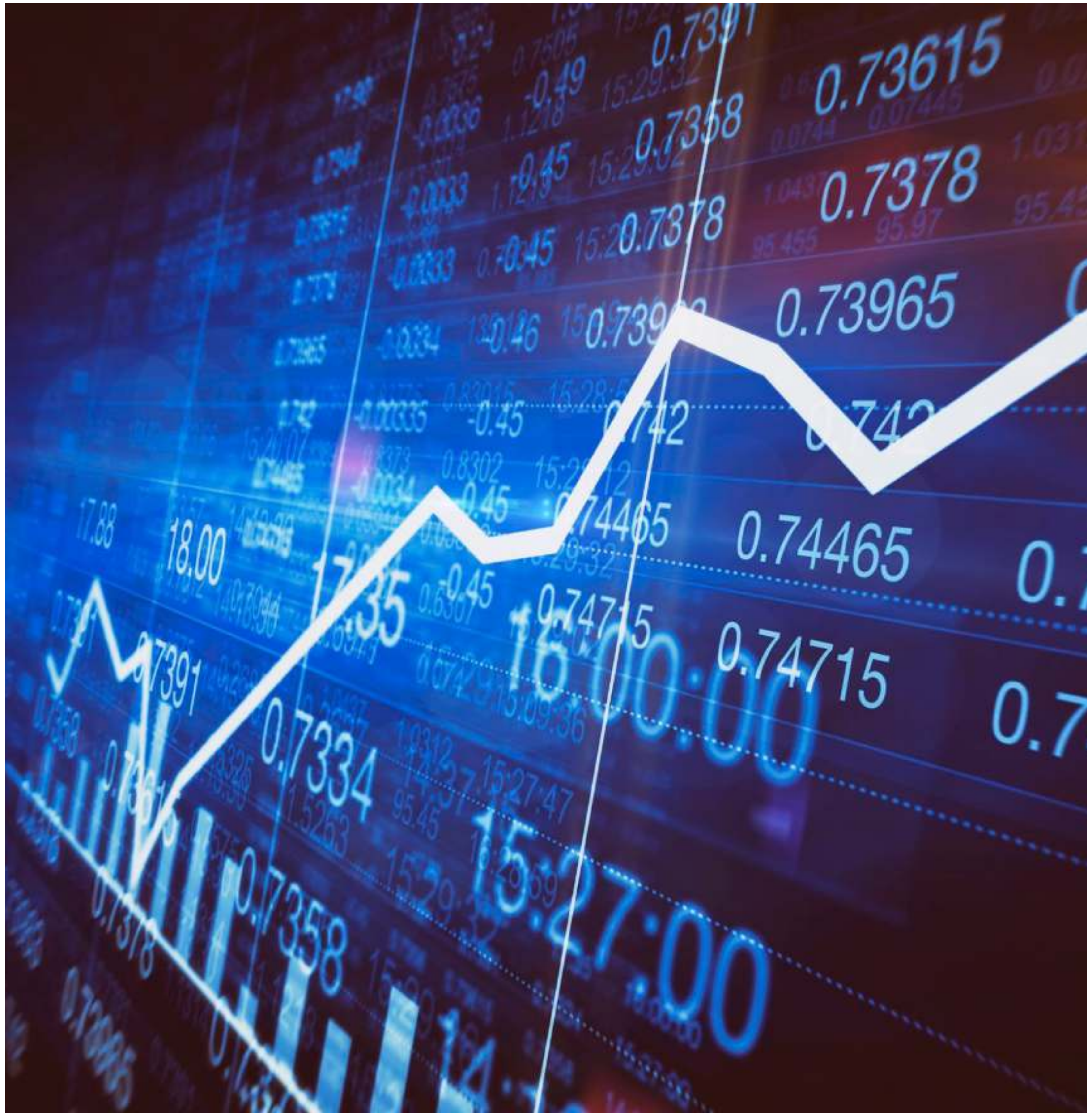


Finance

Variables
Amounts invested in each asset

Constraints
Budget, investment per asset, minimum return, etc.

Objective
Maximize profit, minus risk



Optimal control

Variables
Inputs: thrust, flaps, etc.

Constraints
System limitations, obstacles, etc.

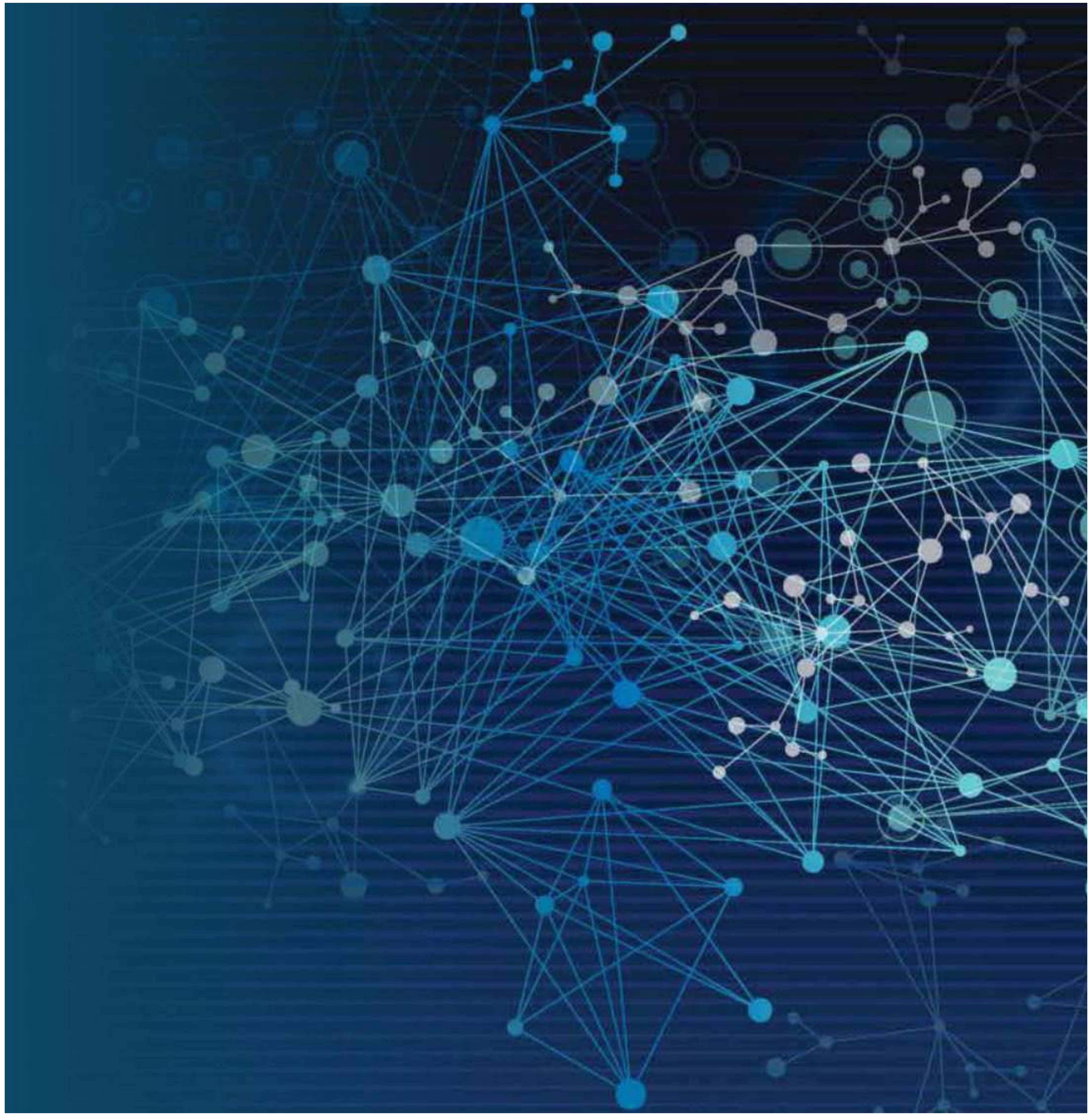
Objective
Minimize distance to target and
fuel consumption

Machine learning

Variables
Model parameters

Constraints
Prior information, parameter limits

Objective
Minimize prediction error, plus regularization



Mathematical optimization

$$
\begin{array}{ll}
\text{minimize} & f(x) \\
\text{subject to} & g_i(x) \leq 0, \quad i = 1, \ldots, m
\end{array}
$$

$x = (x_1, \ldots, x_n)$ Variables

$f : \mathbf{R}^n \to \mathbf{R}$ Objective function

$g_i : \mathbf{R}^n \to \mathbf{R}$ Constraint functions

$x^\star$ Solution/Optimal point

$f(x^\star)$ Optimal value

Most optimization problems cannot be solved

Solving optimization problems

General case → **Very hard!**

Compromises

- Long computation times
- Not finding the solution
  (in practice it may not matter)

Exceptions

- Linear optimization
- Convex optimization

→ **Can be solved very efficiently and reliably**

Meet your teaching staff

Instructor

Bartolomeo Stellato

I am a Professor at ORFE. I obtained my PhD from Oxford and I was a postdoc at MIT.

email: bstellato@princeton.edu

office hours: Wed 2pm—4pm EST, Sherred 323

website: stellato.io

Assistants in instruction

Rajiv Sambharya

PhD student at ORFE.

email: rajivs@princeton.edu

office hours: Fri 11am—1pm, at Sherrerd 003

Scander Mustapha

PhD student at PACM.

email: mustapha@g.princeton.edu

office hours: Mon 3pm — 5pm, at Sheered 003

Meet your classmates!

Name?

Year?

What is your department?

https://www.menti.com/5jp334nxuj

What do you want to use optimization for?

Today's agenda

- Optimization problems
- History of optimization
- Course contents and information
- A glance into modern optimization

Linear optimization

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & a_i^T x \leq b_i, \quad i = 1. \dots, m \end{array}$$

No analytical formula (99% of the time there will be none in this course!)

Efficient algorithms and software we can solve problems with several thousands of variables and constraints

Extensive theory (duality, degeneracy, sensitivity)

Linear optimization

Example: resource allocation

$$
\begin{array}{ll}
\text{maximize} & \sum_{i=1}^n c_i x_i \\
\text{subject to} & \sum_{i=1}^n a_{ji} x_i \leq b_j, \quad j = 1, \ldots, m \\
& x_i \geq 0, \quad i = 1, \ldots, n
\end{array}
$$

- c_i: profit per unit of product i shipped
- b_j: units of raw material j on hand
- a_{ji}: units of raw material j required to produce on unit of product i

Nonlinear optimization

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & g_i(x) \leq 0, \quad i = 1, \ldots, m \end{array}$$

Hard to solve in general

- multiple local minima
- discrete variables $x \in \mathbf{Z}^n$
- hard to certify optimality

Convex optimization

Convex functions

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & g_i(x) \leq 0, \quad i = 1, \dots, m \end{array}$$

All local minima are global!

Efficient algorithms and software

Extensive theory (convex analysis and conic optimization) [ORF523]

Used to solve non convex problems

Prehistory of optimization

Calculus of variations

Fermat/Newton

minimize $f(x),\ x \in \mathbf{R}$

$$\frac{\mathrm{d}f(x)}{\mathrm{d}x} = 0$$

1670

Euler

minimize $f(x),\ x \in \mathbf{R}^n$

$$\nabla f(x) = 0$$

1755

Lagrange

minimize $f(x)$

subject to $g(x) = 0$

1797

Time

History of optimization

Age of computers

Algorithms

- **1930s**: Origin of linear optimization (Kantorovich, Koopmans, von Neumann)
- **1947**: Simplex algorithm (Dantzig)
- **1984**: Interior-point methods (Karmarkar)
- **2000s**: Large-scale optimization

Applications

- Operations Research, Economics
- **1990s**: Engineering, Statistics
- **2000s**: Machine learning, Image processing, Communication systems, Embedded intelligent systems

Technological innovations

Lots of data

easy storage
and
transmission

Massive computations

computers
are
super fast

High-level programming languages

easy to
do complex
stuff

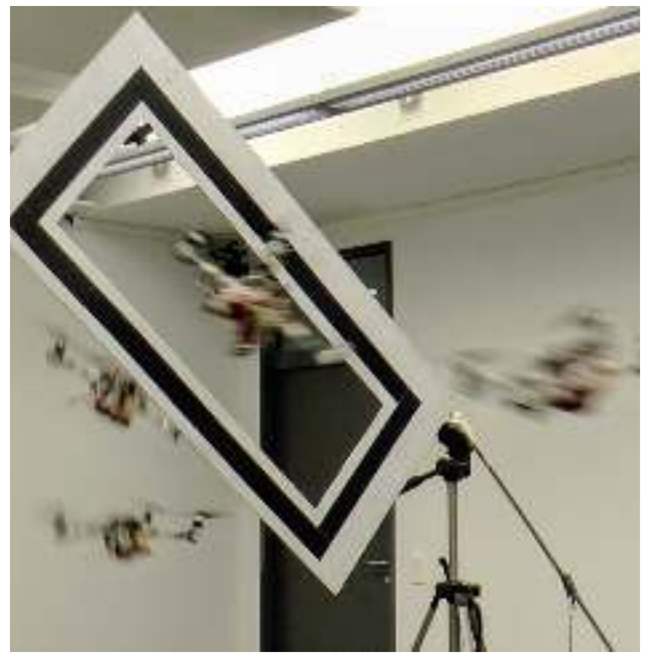
What is happening today?

Huge scale optimization

Massive data + Massive computations

Real-time optimization

Fast real-time requirements + Low-cost computing platforms



Renewed interest in old methods (70s)

- Subgradient methods
- Proximal algorithms

$\longrightarrow$

- Cheap iterations
- Simple implementation

Contents of this course

Linear optimization

- Modelling and applications
- Geometry
- Duality
- Degeneracy
- The simplex method
- Sensitivity analysis
- Interior point methods

Nonlinear optimization

- Modelling and applications
- Optimality conditions
- First-order methods
- Operator-splitting algorithms
- Acceleration schemes

Extensions

- Sequential convex programming
- Branch and bound algorithms
- Real-time optimization

Course information

Grading

- **25% Homeworks**
  5 bi-weekly homeworks with coding component. Collaborations are encouraged!

- **25% Midterm**
  90 minutes written exam at home. No collaborations.

- **40% Final**
  Take-home assignment with coding component. No collaborations.

- **10% Participation**
  One question or note on Ed after each lecture.

Course information

10% Participation notes/questions

What?

- Briefly summarize what you learned in the last lecture
- Highlight the concepts that were most confusing/you would like to review.
- Can be anonymous (to your classmates, not to the instructor) or public, as you choose.

Why?

- We will use your ideas to clarify previous lectures, and to improve the course in future iterations.
- You can ask questions you don’t feel comfortable asking in class.
- You can use these to gather your thoughts on the previous lecture and solidify your understanding.

Course information

Course website

`https://stellato.io/teaching/orf522`

Prerequisites

- Good knowledge of linear algebra and calculus.

  For a refresher, read Appendices A & C of [CO] Boyd, Vandenberghe: *Convex Optimization* (available **online**).

- Familiarity with Python.

PRINCETON UNIVERSITY **Bartolomeo Stellato** ≡

ORF522: Linear and Nonlinear Optimization

Previous years: **2020**

Description

This course introduces analytical and computational tools for linear and nonlinear optimization. Topics include linear optimization modeling, duality, the simplex method, degeneracy, sensitivity analysis and interior point methods. Nonlinear optimality conditions, KKT conditions, first order and operator splitting methods for nonlinear optimization, real-time optimization and data-driven algorithms. A broad spectrum of applications in engineering, finance and statistics is presented.

Learning objectives

This course introduces analytical and computational tools for linear and nonlinear optimization. Upon successful completion of this course you should be able to:

- Model decision-making problems across different disciplines as mathematical optimization problems.
- Apply the most appropriate optimization tools when faced with a concrete problem.
- Implement optimization algorithms and prove their convergence.

Course information

Materials

Linear optimization

- [LP] R. J. Vanderbei: *Linear Programming: Foundations & Extensions* (available on **SpringerLink**)
- [LO] D. Bertsimas, J. Tsitsiklis: *Introduction to Linear Optimization* (available **Princeton Controlled Digital Lending**)

Nonlinear optimization

- [NO] J. Nocedal, S. J. Wright: *Numerical Optimization* (available on **SpringerLink**)
- [CO] S. Boyd, L. Vandenberghe: *Convex Optimization* (available for **free**)
- [FMO] A. Beck: *First-order methods in optimization* (available on **SIAM**)
- [FCA] J. B. Hiriart-Hrruty, C. Lemarechal: *Fundamentals of Convex Analysis* (available on **SpringerLink**)
- [ILCO] Y. Nesterov: *Introductory Lectures to Convex Optimization* (available on **SpringerLink**)
- [e364b] S. Boyd: *Convex Optimization II Lecture Notes* (available **online**)
- [COAC] S. Bubeck: *Convex Optimization: Algorithms and Complexity* (available for **free**)
- [MINLO] P. Belotti, C. Kirches, S. Leyffer, J. Linderoth, J. Luedtke, A. Mahajan: *Mixed-integer nonlinear optimization* (available **online**)

Operator splitting algorithms

- [PA] N. Parikh, S. Boyd: *Proximal Algorithms* (available for **free**)
- [PMO] E. K. Ryu, S. Boyd: *A primer on monotone operators* (available for **free**)
- [LSMO] E. K. Ryu and W. Yin: *Large-Scale Convex Optimization via Monotone Operators (Draft)* (available for **free**)
- [ADMM] S. Boyd, N. Parikh, B. Peleato, J. Eckstein: *Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers* (available for **free**)

Software (open-source)

python™

jupyter

Numerical computations

Numerical computations on *numpy* and *scipy*.

CVXPY

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax \leq b \end{array}$$

```
x = cp.Variable(n)
prob = cp.Problem(
    cp.Minimize(c.T@x),
    [A @ x <= b]
)
prob.solve()
print("The optimal value is", prob.value)
print("The solution x is", x.value)
```

Learning goals

- **Model** your favorite decision-making problems as mathematical optimization problems.
- **Apply** the most appropriate optimization tools when faced with a concrete problem.
- **Implement** optimization algorithms and prove their convergence.

Glance into modern optimization

Huge scale optimization

Dataset with
billions of datapoints (x^i, y^i) $\longrightarrow$ **Goal:** Design predictor $\hat{y}^i = g_\theta(x^i)$

Optimization problem

minimize $\underbrace{\mathcal{L}(\theta)}_{\text{Loss}} + \lambda \underbrace{r(\theta)}_{\text{Regularizer}} = \sum_{i=1}^n \ell(\hat{y}^i, y^i) + \lambda r(\theta)$

Many examples

- Support vector machines
- Regularized regression
- Neural networks

Large-scale computing

- Parallel
- Distributed

How large are the largest problems we can solve? (how many variables?)

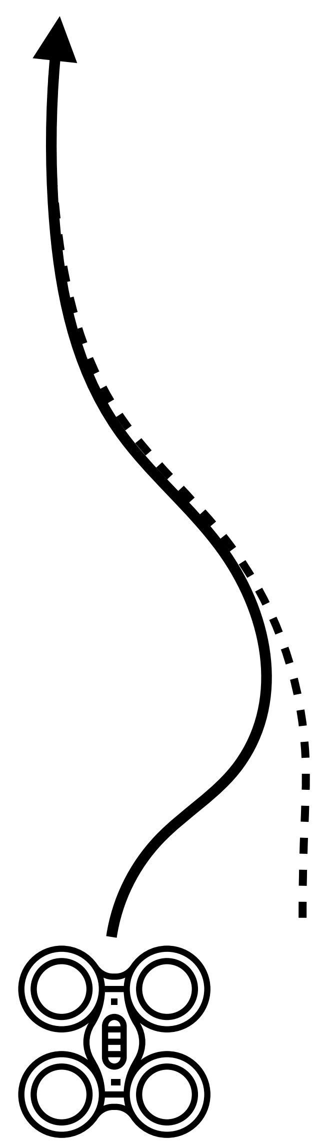
Glance into modern optimization

Real-time optimization

$x_t \in \mathbf{R}^n$: state

$u_t \in \mathbf{R}^m$: input

Dynamical system: $x_{t+1} = Ax_t + Bu_t$



Goal: track trajectory minimize $\sum_{t=0}^{T} \|x_t - x_t^{\mathrm{des}}\|$

Constraints: inputs $\|u\| \leq U$, states $a \leq x_t \leq b$

Solve and repeat…..

How fast can we solve these problems?

1-norm $\longrightarrow$???

∞-norm $\longrightarrow$???

Next lecture

Linear optimization

- Definitions
- Modelling
- Formulations
- Examples