New uses for old tools

An introduction to mathematical programming

Dr Gianluca Campanella

29th April 2017

Contents

Mathematical programming

What is mathematical programming?

- Also known as (mathematical) *optimisation*
- Goal is to select the 'best' element from some set of available alternatives

Typically we have an *objective* function, e.g. $f : \mathbb{R}^p \to \mathbb{R}$, that:

- Takes p inputs as a vector, e.g. $\mathbf{x} \in \mathbb{R}^p$
- Maps the input to some output value $f(\mathbf{x}) \in \mathbb{R}$

We want to find the *optimal* $\mathbf{x}^\star$ that minimises (or maximises) f

What is mathematical programming?

- Many ML methods rely on minimisation of *cost functions*

Linear regression

$$\mathrm{MSE}(\hat{\beta}) = \frac{1}{n}\sum_i (\hat{y}_i - y_i)^2$$

where $\hat{y}_i = \mathbf{x}_i^\intercal \hat{\beta}$

Logistic regression

$$\mathrm{LogLoss}(\hat{\beta}) = -\sum_i [y_i \log \hat{p}_i + (1 - y_i)\log(1 - \hat{p}_i)]$$

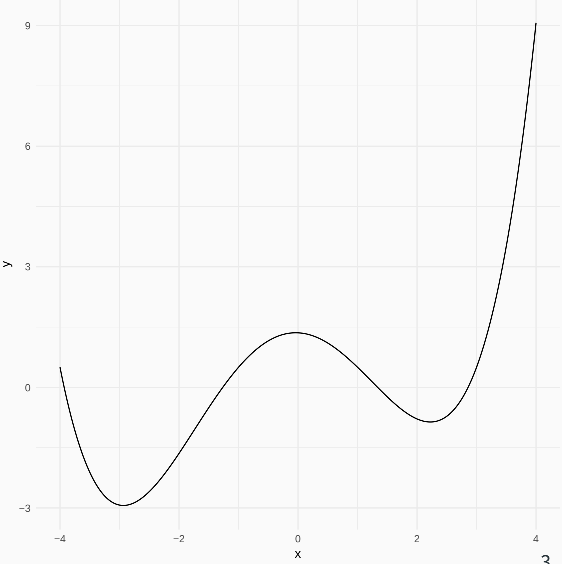
where $\hat{p}_i = \mathrm{logit}^{-1}(\mathbf{x}_i^\intercal \hat{\beta})$

Local and global optima

A function may have *multiple* optima

↓

Some will be *local*, some will be *global*



Hard optimisation problems

Consider these three functions:

$$f : \mathbb{R}^{100} \rightarrow \mathbb{R}$$
$$g : [0, 1]^{100} \rightarrow \mathbb{R}$$
$$h : \{0, 1\}^{100} \rightarrow \mathbb{R}$$

Which one is ‘harder’ to optimise, and why?

Combinatorial optimisation

Combinatorial problems like optimising $h : \{0,1\}^{100} \to \mathbb{R}$ are intrinsically hard

- Need to try all $2^{100} \approx 1.27 \times 10^{30}$ combinations
- *Variable selection* is a notable example

Side note

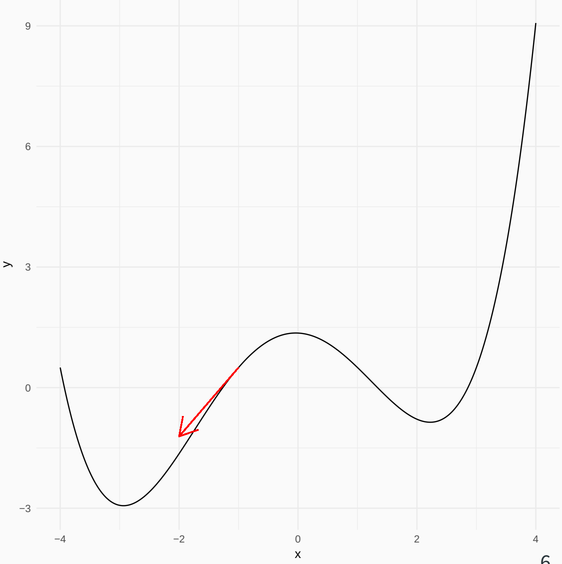
If h is continuous and we're actually constraining $\mathbf{x} \in \{0,1\}^{100}$, approximate solutions (*relaxations*) are normally easier to obtain

Numerical optimisation using directional information

Function is differentiable
(analytically or numerically)

↓

Gradient gives a search *direction*
and
Hessian can be used to confirm optimality

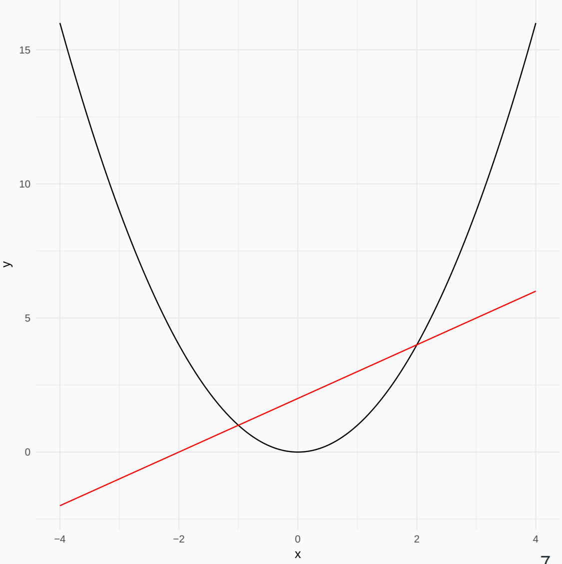


Convex functions

Function is *convex*

$\downarrow$

Any local minimum is also a global minimum



Constrained optimisation

What about $g : [0,1]^{100} \to \mathbb{R}$?

- Harder than $f : \mathbb{R}^{100} \to \mathbb{R}$… but not much
- Directional information still useful
- Need to ensure search strategy doesn't escape the *feasible region*

Linear and quadratic programs

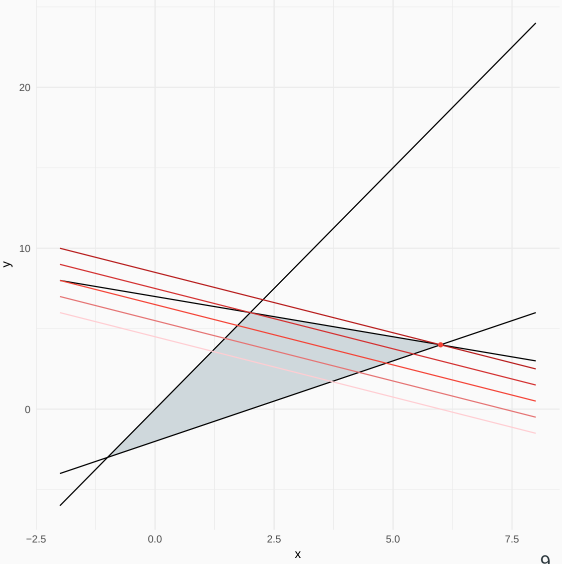
Linear programs

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

- Linear objective, linear constraints
- Linear objective is convex $\rightsquigarrow$ global maximum
- An optimal solution need not exist:
  - Inconsistent constraints $\rightsquigarrow$ infeasible
  - Feasible region unbounded in the direction of the gradient of the objective

Linear programs

$$\max_{x,y} \; 3x + 4y$$
$$\text{s.t.} \; x + 2y \leq 14$$
$$3x - y \geq 0$$
$$x - y \leq 2$$



Linear programs

Linear programs can be solved efficiently using:

- Simplex algorithm
- Interior-point (barrier) methods

Performance is *generally* similar, but might differ drastically for specific problems

Convex quadratic programs

$$\begin{aligned} \min_{\mathbf{x}} \quad & \frac{1}{2}\mathbf{x}^\top \mathbf{Q}\mathbf{x} + \mathbf{c}^\top \mathbf{x} \\ \text{s.t.} \quad & \mathbf{A}\mathbf{x} \preceq \mathbf{b} \\ & \mathbf{x} \succeq \mathbf{0} \end{aligned}$$

- Quadratic objective, quadratic constraints
- Are quadratic objectives always convex?
- **Q** must be (semi)definite

Convex quadratic programs

Quadratic programs can be solved efficiently using:

- Active set method
- Augmented Lagrangian method
- Conjugate gradient method
- Interior-point (barrier) methods

LPs and QPs in Python

Many Python libraries exist:

Linear programming

- PuLP
- Google Optimization Tools
- `clpy`

Convex quadratic programming

- CVXOPT
- CVXPY

Regression problems as LPs and QPs

We can rewrite the least-squares problem

$$\min_{\mathbf{x}} \ ||\,\mathbf{Ax} - \mathbf{b}\,||_2^2 = \sum_i \varepsilon_i^2$$

as the convex quadratic objective

$$f(\mathbf{x}) = \mathbf{x}^\top \mathbf{A}^\top \mathbf{A}\mathbf{x} - 2\mathbf{b}^\top \mathbf{A}\mathbf{x} + \mathbf{b}^\top \mathbf{b}$$

Side note

Setting the gradient to 0 and solving for **x** recovers the normal equations:

$$\nabla f = 2\mathbf{A}^\top \mathbf{A}\mathbf{x} - 2\mathbf{A}^\top \mathbf{b} = 0 \quad \rightsquigarrow \quad \mathbf{A}^\top \mathbf{A}\mathbf{x} = \mathbf{A}^\top \mathbf{b} \quad \rightsquigarrow \quad \mathbf{x}^\star = \left(\mathbf{A}^\top \mathbf{A}\right)^{-1} \mathbf{A}^\top \mathbf{b}$$

Regularised linear regression

Let's add a penalisation term:

$$\min_{\mathbf{x}} \; ||\,\mathbf{Ax} - \mathbf{b}\,||_2^2 + \lambda\,||\,\mathbf{x}\,||_2^2$$

Our quadratic objective becomes:

$$f(\mathbf{x}) = \mathbf{x}^\top \left(\mathbf{A}^\top\mathbf{A} + \lambda\mathbf{I}_p\right)\mathbf{x} - 2\mathbf{b}^\top\mathbf{Ax} + \mathbf{b}^\top\mathbf{b}$$

Side note

This is a good trick to use when the columns of **A** are not perfectly independent

Constraints on $\mathbf{x}$

Nonnegativity

- $\mathbf{x} \geq \mathbf{0}$
- Parameters known to be nonnegative, e.g. intensities or rates

Bounds

- $\mathbf{l} \leq \mathbf{x} \leq \mathbf{u}$
- Prior knowledge of permissible values

Unit sum

- $\mathbf{x} \geq \mathbf{0}$ and $\mathbf{1}_p^\top \mathbf{x} = 1$
- Useful for proportions and probability distributions

Least squares *vs* least absolute deviations

Why do we minimise *squared* residuals?

- Stable, unique, analytical solution
- Not very robust!

Least squares *vs* least absolute deviations

Least *absolute* deviations

- Predates least squares by around 50 years (Bošković)
- Adopted by Laplace, but shadowed by Legendre and Gauss
- *Robust*
- Possibly multiple solutions

Robust regression

We can rewrite the LAD problem

$$\min_{\mathbf{x}} \ ||\, \mathbf{Ax} - \mathbf{b}\, ||_1 = \sum_i |\varepsilon_i|$$

as the linear program

$$\begin{aligned} \min_{\mathbf{x},\mathbf{t}} \quad & \mathbf{1}_n^\top \mathbf{t} \\ \text{s.t.} \quad & -\mathbf{t} \leq \mathbf{Ax} - \mathbf{b} \leq \mathbf{t} \\ & \mathbf{t} \in \mathbb{R}^n \end{aligned}$$

or

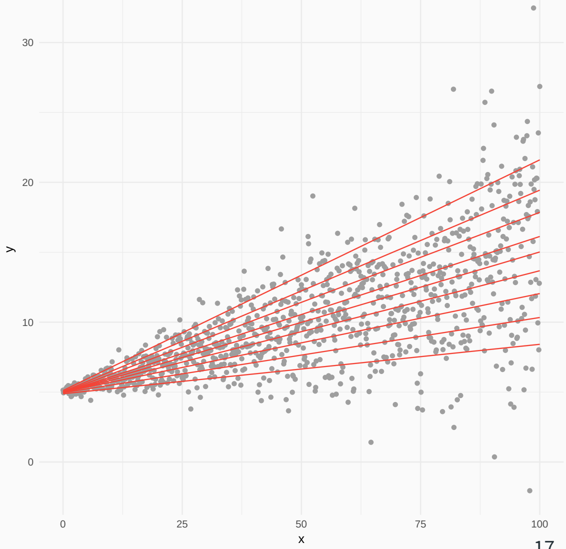
$$\begin{aligned} \min_{\mathbf{x},\mathbf{u},\mathbf{v}} \quad & \mathbf{1}_n^\top \mathbf{u} + \mathbf{1}_n^\top \mathbf{v} \\ \text{s.t.} \quad & \mathbf{Ax} + \mathbf{u} - \mathbf{v} = \mathbf{b} \\ & \mathbf{u}, \mathbf{v} \geq \mathbf{0} \end{aligned}$$

Quantile regression

Let's now introduce a weight $\tau \in [0, 1]$

$$\begin{aligned} \min_{\mathbf{x},\mathbf{u},\mathbf{v}} \quad & \tau\, \mathbf{1}_n^\top \mathbf{u} + (1-\tau)\mathbf{1}_n^\top \mathbf{v} \\ \text{s.t.} \quad & \mathbf{A}\mathbf{x} + \mathbf{u} - \mathbf{v} = \mathbf{b} \\ & \mathbf{u}, \mathbf{v} \geq \mathbf{0} \end{aligned}$$

This is the τ^{th} **quantile regression** problem



An application to portfolio theory

Example

- Consider these two assets:

  A Equally likely to go up 20% or down 10% in a year

  A Equally likely to go up 20% or down 10% in a year

- Assume they’re perfectly inversely correlated
- How would you allocate your money?

Example

- Consider these two assets:

  A Equally likely to go up 20% or down 10% in a year

  A Equally likely to go up 20% or down 10% in a year
- Assume they're perfectly inversely correlated
- How would you allocate your money?

The portfolio 50% **A** + 50% **B** goes up 5% every year!

Mean-variance approach of Markowitz

Given historical ROIs, denoted $r_i(t)$ for asset i at time $t \leq T$, we can compute:

- The *reward* of asset i:
$$\text{reward}_i = \frac{1}{T} \sum_t r_i(t)$$
- The *risk* of asset i:
$$\text{risk}_i = \frac{1}{T} \sum_t \left[r_i(t) - \text{reward}_i \right]^2$$

We can compute the same quantities for a *portfolio* $\mathbf{x} \geq \mathbf{0}, \mathbf{1}_p^\top \mathbf{x} = \mathbf{1}$

Mean-variance approach of Markowitz

Our objective is to *maximise reward* and *minimise risk*

Instead, we solve

$$\max_{\mathbf{x}} \ \mathrm{reward}(\mathbf{x}) - \mu \, \mathrm{risk}(\mathbf{x})$$

for multiple values of the *risk aversion parameter* $\mu \geq 0$

- Linear constraints: $\mathbf{x} \geq \mathbf{0}$, $\mathbf{1}_p^\top \mathbf{x} = \mathbf{1}$
- What about the objective function?

Why is variance a reasonable measure of risk?

- Variance-based measures are not monotonic
- Quantile-based measures (e.g. VaR) are not subadditive
- The loss beyond the VaR is ignored

Other risk measures

Artzner et al. provided a foundation for ‘coherent’ risk measures:

- Expected shortfall
- Conditional VaR (CVaR)
- α-risk

Other risk measures

Artzner et al. provided a foundation for ‘coherent’ risk measures:

- Expected shortfall
- Conditional VaR (CVaR)
- α-risk

Linear programming solutions

- Portfolios with CVaR constraints are linear programs
- α-risk models are α^{th} quantile regression problems

Recap

- Optimisation is at the core of what we do!
- Some problems are much harder than others $\rightsquigarrow$ convexity
- LPs and QPs are ‘easy’, with plenty of tools available
- Different commonly used regression models are actually LPs or QPs
- So are some portfolio allocation models!