

Quadratic Programming

Aayush Bajaj

July 10, 2026

Contents

1	standard form	1
2	convexity	2
3	kkt conditions	2
4	equality-constrained QP: the closed form	2
4.1	worked example	3
5	inequality constraints: two algorithm families	3
5.1	active-set methods	3
5.2	interior-point methods	4
6	applications	4
6.1	the svm dual	4
6.2	markowitz portfolio selection	4
6.3	ridge regression is an unconstrained QP	5
7	code	5
8	see also	5

promote the objective of a **linear program** from a plane to a bowl and you get quadratic programming: minimise a quadratic function over a polyhedron. it is the smallest step beyond LP, yet it captures a startling share of applied mathematics — support vector machines, portfolio selection, ridge regression, model-predictive control — because “squared penalty subject to linear rules” is how half the world states its preferences.

the other half states them as expected log-wealth and gets convex programming anyway

1 standard form

a quadratic program (QP):

$$\begin{aligned} & \text{minimise} && \frac{1}{2}x^\top Qx + c^\top x \\ & \text{subject to} && Ax \leq b \\ & && Ex = d, \end{aligned}$$

with $Q \in \mathbb{R}^{n \times n}$ symmetric (any quadratic can be symmetrised: replace Q by $\frac{1}{2}(Q + Q^\top)$ without changing the objective). the constraints are still linear — the feasible

region is the same polyhedron as in LP — only the objective has curved.

2 convexity

everything hinges on the spectrum of Q :

- $Q \succeq 0$ (positive semidefinite): the objective is **convex**, every local minimum is global, and the problem is solvable in polynomial time.
- $Q \succ 0$ (positive definite): **strictly** convex — the minimiser is unique.
- Q indefinite: the objective is a saddle, the problem is **NP-hard** in general (it can encode binary choices in the concave directions), and one honest local minimum is all you should expect.

why psd means convex: for any x, y and $\theta \in [0, 1]$, expanding $f(\theta x + (1 - \theta)y)$ leaves the linear parts interpolating exactly and the quadratic part short by $\frac{1}{2}\theta(1 - \theta)(x - y)^\top Q(x - y)$, which is ≥ 0 precisely when $Q \succeq 0$. one bad eigenvalue and the bowl folds into a pringle.

unlike LP, a convex QP's optimum need not sit at a vertex — the bowl's bottom can rest anywhere in the polyhedron: interior (constraints irrelevant), on a face, or at a vertex. the diagram below makes this concrete.

3 kkt conditions

lagrangian $L(x, \mu, \lambda) = \frac{1}{2}x^\top Qx + c^\top x + \mu^\top(Ax - b) + \lambda^\top(Ex - d)$. the karush-kuhn-tucker conditions for the QP: x^* is optimal (for convex $Q \succeq 0$) iff there exist multipliers μ^*, λ^* with

$$\begin{aligned} Qx^* + c + A^\top \mu^* + E^\top \lambda^* &= 0 && \text{(stationarity)} \\ Ax^* &\leq b, \quad Ex^* = d && \text{(primal feasibility)} \\ \mu^* &\geq 0 && \text{(dual feasibility)} \\ \mu_i^* (Ax^* - b)_i &= 0 \quad \forall i && \text{(complementary slackness).} \end{aligned}$$

since the constraints are affine, no constraint qualification is needed — KKT is necessary *and* (with convexity) sufficient. this is LP duality's complementary slackness with one new term: the gradient of the objective is no longer the constant c but $Qx + c$, so the stationarity condition genuinely involves x . that single change replaces the simplex tableau with linear algebra.

4 equality-constrained QP: the closed form

with only equality constraints the KKT conditions are a *linear system*. minimise $\frac{1}{2}x^\top Qx + c^\top x$ subject to $Ex = d$, $E \in \mathbb{R}^{m \times n}$ full row rank:

stationarity $Qx + c + E^\top \lambda = 0$ and **feasibility** $Ex = d$, stacked:

$$\underbrace{\begin{pmatrix} Q & E^\top \\ E & 0 \end{pmatrix}}_{\text{KKT matrix}} \begin{pmatrix} x \\ \lambda \end{pmatrix} = \begin{pmatrix} -c \\ d \end{pmatrix}. \quad (1)$$

if Q is positive definite on the null space of E (i.e. $z^\top Q z > 0$ for all $z \neq 0$ with $Ez = 0$), the KKT matrix is nonsingular and the unique solution is

$$\begin{aligned}\lambda^* &= -(EQ^{-1}E^\top)^{-1}(EQ^{-1}c + d) \\ x^* &= -Q^{-1}(c + E^\top \lambda^*),\end{aligned}$$

obtained by block elimination (solve the first row for x , substitute into the second). one factorisation, done. no iteration, no tableau: equality-constrained QP is “just” linear algebra, which is why it is the inner engine of everything else on this page.

the kkt matrix is symmetric but indefinite — it has m negative eigenvalues — so factorise it with ldl-transpose, not cholesky

4.1 worked example

minimise $x_1^2 + x_2^2 - 2x_1 - 5x_2$ subject to $x_1 + x_2 = 1$. here $Q = 2I$, $c = (-2, -5)^\top$, $E = (1 \ 1)$, $d = 1$. the unconstrained minimiser is $\hat{x} = -Q^{-1}c = (1, 2.5)$, which violates the constraint ($1 + 2.5 = 3.5 \neq 1$); the constrained answer is its orthogonal projection onto the line (orthogonal because Q is a multiple of I — in general the projection is skewed by the metric Q):

$$\begin{pmatrix} 2 & 0 & 1 \\ 0 & 2 & 1 \\ 1 & 1 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \lambda \end{pmatrix} = \begin{pmatrix} 2 \\ 5 \\ 1 \end{pmatrix} \implies x^* = (-\frac{1}{4}, \frac{5}{4}), \quad \lambda^* = \frac{5}{2}. \quad (2)$$

check: rows give $2x_1 + \lambda = 2$ and $2x_2 + \lambda = 5$, so $x_1 - x_2 = -\frac{3}{2}$; with $x_1 + x_2 = 1$ that pins x^* . the code below solves exactly this system.

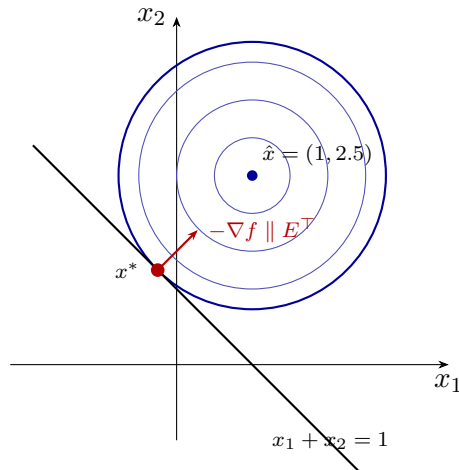


Figure 1: a convex QP: circular contours of the bowl centred at the unconstrained minimiser $\hat{x} = (1, 2.5)$, and the constraint line $x_1 + x_2 = 1$. the constrained optimum $x^* = (-\frac{1}{4}, \frac{5}{4})$ is the point where a contour first kisses the line — gradient perpendicular to the constraint.

5 inequality constraints: two algorithm families

5.1 active-set methods

the fix for inequalities: guess which of them bind at the optimum (the *active set*), treat those as equalities, ignore the rest, and solve the equality-constrained QP above.

- **solve** the equality QP on the current working set; get x and multipliers μ .
- **if** the step is blocked by an inactive constraint, walk only as far as feasibility allows and **add** the blocking constraint to the working set.
- **if** the subproblem solution has some $\mu_i < 0$, that constraint is being held against its will — **drop** it.
- **stop** when the subproblem is solved and all multipliers are non-negative: KKT holds.

this is simplex’s spiritual successor — combinatorial in the worst case (there are 2^m candidate active sets), excellent with warm starts, which is why model-predictive controllers re-solving almost-identical QPs at 100 hz love it.

5.2 interior-point methods

exactly the LP barrier story: replace complementarity $\mu_i s_i = 0$ (with slacks $s = b - Ax$) by $\mu_i s_i = \tau$, solve the smoothed KKT system by newton’s method, drive $\tau \rightarrow 0$ along the central path. each newton step is one KKT-like linear solve; a few dozen steps suffice essentially regardless of m . polynomial time for convex QP, no combinatorial search, but every solve starts cold.

6 applications

6.1 the svm dual

the maximum-margin classifier is a QP twice over. the primal — minimise $\frac{1}{2}\|w\|^2 + C \sum_i \xi_i$ subject to $y_i(w^\top x_i + b) \geq 1 - \xi_i$, $\xi_i \geq 0$ — is already a QP in (w, b, ξ) . its lagrangian dual is the QP everyone actually solves:

$$\begin{aligned} & \text{maximise}_{\alpha} \quad \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j x_i^\top x_j \\ & \text{subject to} \quad 0 \leq \alpha_i \leq C, \quad \sum_i \alpha_i y_i = 0, \end{aligned}$$

with hessian $Q_{ij} = y_i y_j x_i^\top x_j$ — a gram matrix, hence psd, hence convex (Hastie, Trevor and Tibshirani, Robert and Friedman, Jerome, 2009). two gifts fall out: complementary slackness makes most $\alpha_i = 0$ (only support vectors survive), and the data enter purely through inner products, so swapping $x_i^\top x_j$ for a kernel $k(x_i, x_j)$ costs nothing — the kernel trick is a property of the QP dual, not of the classifier. details on the [svm page](#).

6.2 markowitz portfolio selection

hold weights w in n assets with mean returns μ and covariance $\Sigma \succeq 0$. the 1952 formulation: minimise risk at a target return,

$$\begin{aligned} & \text{minimise} \quad w^\top \Sigma w \\ & \text{subject to} \quad \mu^\top w \geq r, \quad \mathbf{1}^\top w = 1, \quad w \geq 0, \end{aligned}$$

a convex QP whose optimal objective traced over r is the *efficient frontier*. drop the no-short-selling constraint $w \geq 0$ and pin the return target to equality $\mu^\top w = r$, and it becomes equality-constrained — solvable in closed form by the KKT system above.

markowitz shared the 1990 nobel for this; the qp is the easy part — estimating sigma is the hard part

6.3 ridge regression is an unconstrained QP

$\min_w \|y - Xw\|^2 + \lambda \|w\|^2$ expands to $\frac{1}{2}w^\top Qw + c^\top w + \text{const}$ with $Q = 2(X^\top X + \lambda I) \succ 0$ and $c = -2X^\top y$. no constraints, so KKT collapses to stationarity $Qw + c = 0$, i.e. the familiar $w^* = (X^\top X + \lambda I)^{-1}X^\top y$. the lasso swaps the squared penalty for ℓ_1 and stops being a QP — but its constrained form (minimise squared loss subject to $\|w\|_1 \leq t$) is a QP over a polyhedron again, since the ℓ_1 ball is one. more on the [regularised regression](#) page.

7 code

the worked equality-constrained QP, solved two ways: assemble and solve the KKT system by hand, then verify with scipy.

```

1  import numpy as np
2  from scipy.optimize import minimize
3
4  # equality-constrained QP:
5  #   min (1/2) x^T Q x + c^T x   s.t.   E x = d
6  Q = np.array([[2.0, 0.0], [0.0, 2.0]])
7  c = np.array([-2.0, -5.0])
8  E = np.array([[1.0, 1.0]])
9  d = np.array([1.0])
10
11 # KKT system: [[Q, E^T], [E, 0]] [x; lam] = [-c; d]
12 n, m = 2, 1
13 K = np.block([[Q, E.T], [E, np.zeros((m, m))]])
14 rhs = np.concatenate([-c, d])
15 sol = np.linalg.solve(K, rhs)
16 x_kkt, lam = sol[:n], sol[n:]
17 obj = lambda x: 0.5 * x @ Q @ x + c @ x
18 print("KKT solve:   x* =", x_kkt, " lambda* =", lam, " f* =", obj(x_kkt))
19
20 # verify against scipy (SLSQP)
21 res = minimize(obj, x0=np.zeros(n), jac=lambda x: Q @ x + c,
22               constraints={"type": "eq", "fun": lambda x: E @ x - d},
23               method="SLSQP")
24 print("scipy SLSQP: x* =", res.x, " f* =", res.fun)
25
26 # gradient of lagrangian vanishes?
27 print("stationarity residual:", np.linalg.norm(Q @ x_kkt + c + E.T @ lam))

```

```

KKT solve:   x* = [-0.25  1.25]  lambda* = [2.5]  f* = -4.125
scipy SLSQP: x* = [-0.25  1.25]  f* = -4.125
stationarity residual: 0.0

```

both routes land on $x^* = (-\frac{1}{4}, \frac{5}{4})$, $f^* = -4.125$, and the lagrangian gradient is zero to machine precision — the KKT system *is* the answer, the iterative solver merely rediscovers it.

8 see also

- [linear programming](#) — flatten the bowl back to a plane

- **non-linear programming** — arbitrary smooth objectives, same KKT skeleton
- **integer programming** — miqp: bowls over lattice points
- **support vector machines** — the QP dual worn as a classifier
- **regularised regression** — ridge as the unconstrained special case

references

Hastie, Trevor and Tibshirani, Robert and Friedman, Jerome (2009). *The Elements of Statistical Learning*, Springer.