Chapter 4

Linear, Quadratic, and Geometric Models

A linear program (LP) is an optimization problem that all the functions involved are *affine*. The feasible set is thus a *polyhedron*, that is, an intersection of half-spaces. *Quadratic programs (QPs)* is an extension of linear programs, in which all constraint functions involved are *affine*, and the objective is the sum of a linear function and a *positive semi-definite quadratic* form. QPs generalize both LPs and oridinary least-squares. The objective is the same as in ordinary-least-squares, and the problem includes polyhedral constraints, just as in LP.

4.1 Unconstrained Minimization of Quadratic Functions

- The linear function $f_0(x) = c^T x + d$ with no constraints $x \in \mathbb{R}^n$ has the optimal solution as follow:

$$p^* = \min_{x \in \mathbb{R}^n} c^T x + d$$

$$p^* = \begin{cases} d & \text{if } c = 0 \\ -\infty & \text{otherwise} \end{cases}$$

- For the quadratic case

$$p^* = \min_{x \in \mathbb{R}^n} \frac{1}{2} x^T H x + c^T x + d$$

the minimum value p^* of the quadratic function depends on the sign of the eigenvalues of H (H is symmetric).

$$p^* = \begin{cases} -\frac{1}{2} c^T H^+ c + d & \text{if } H \succeq 0 \text{ and } c \in \mathcal{R}(H) \\ -\infty & \text{otherwise} \end{cases}$$

4.2 Geometry of Linear and Convex Quadratic Inequalities

4.2.1 Linear Inequalities and Polyhedra

- **Closed Half-space**. The set of points $x \in \mathbb{R}^n$ satisfying a linear inequality $a_i^T x \leq b_i$ is a closed half-space; the vector a_i is *normal* to the boundary of the half-space and points outwards.

- **Polyhedron**. A collection of m linear inequalities $a_i^T \leq b_i$ defines a region in $\mathbb{R}^m$ which is the ***intersection of*** m ***half-spaces***, and is called a *polyhedron*. It is equivalent to the matrix form $Ax \leq b$.

- **Polytope**. Depending on the actual inequalities, the region can be bounded or unbounded. If it is bounded, it is called *polytope*.

- **Face**. The intersection of a polytope P with a supporting hyperplane H is called a *face* of P, which is a convex polytope.

- **Vertices**. *Vertices* are the faces of dimension 0.

- **Edges**. The faces of dimension 1 are the *edges* of P.

- **Facets**. The faces of dimension dim $P - 1$ are called the *facets*.

- A polyhedron is a ***convex set***, with boundary made up of **flat** boundaries (facet). Each facet corresponds to one of the hyperplanes defined by $a_i^T x = b_i$. The vectors a_i are *orthogonals* to the facets, and point outside the polyhedra.

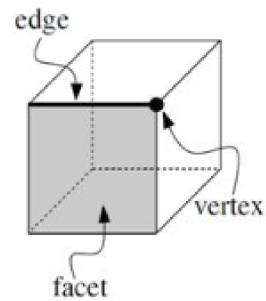

Figure 4.1: Polytope in $\mathbb{R}^3$

- **Equality Constraints**. *Equality constraints* are allowed. Sets defined by affine inequalities and equalities are also polyhedra. The set $p = \{Ax \leq b, Cx = d\}$ can be expressed as an **inequalities-only** polyhedron:

$$p = \{Ax \leq b, Cx \leq d, -Cx \leq -d\}$$

4.2.2 Quadratic Inequalities and Ellipsoids

- **Quadratic inequality**. The zero-level set $x \in \mathbb{R}^n$ of a quadratic inequality

$$f_0(x) = \frac{1}{2}x^T Hx + c^T x + d \leq 0 \tag{4.1}$$

is convex if $H \succeq 0$. (4.1) can be written as

$$f_0(x) = \frac{1}{2}(x - \hat{x})H(x - \hat{x}) - \frac{1}{4}c^T H^{-1}c + d \leq 0 \tag{4.2}$$

which is a (possibly unbounded) **ellipsoid** with center in $\hat{x} = -\frac{1}{2}H^{-1}c$.

- **Representation of ellipsoid**. A bounded, full-dimensional ellipsoid is usually represented in the form

$$\epsilon = \{x : (x - \hat{x})^T p^{-1}(x - \hat{x}) \leq 1\}, \quad P \succ 0$$

where P is the *shape matrix* of the ellipsoid. This representation is analogous to (4.1) and (4.2), with

$$H = 2P^{-1}, \quad \frac{c^T H^{-1}c}{4} - d = 1$$

- **Directions and lengths** The *eigenvectors* v_i of P define the *directions* of the semi-axes of the ellipsoid; the *lengths* of the semi-axes are given by the *eigenvalues* of P, $\sqrt{\lambda_i}$.

- The previous discussion suggests that the family of m convex quadratic inequalities

$$\frac{1}{2}x^T H_i x + c_i^T x + d_i \leq 0,\ H_i \succeq 0, i = 1, \cdots, m$$

includes the family of *polyhedra* and *polytopes*, but it is much richer.

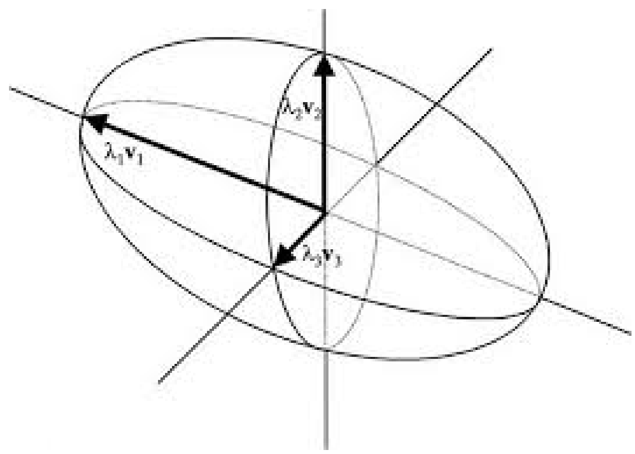

Figure 4.2: A 3-dimensional ellipsoid

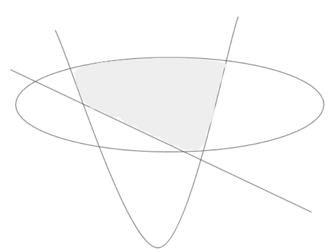

Figure 4.3: Intersection of the feasible sets of three quadratic inequalities in $\mathbb{R}^2$

4.3 Linear Programs

4.3.1 Definition

- A *linear program* is an optimization problem with **linear objective** and **affine inequality constraints**.

$$\begin{aligned} p^* = \min_x \quad & c^T x + d \\ \text{s.t.}: \quad & A_{eq} x = b_{eq}, \\ & Ax \leq b \end{aligned}$$

  where the constant term d in the objective does not matter.

- **Geometric Interpretation of LP**. The set of points that satisfy the constraints of an LP is a ***polyhedron*** (or a ***polytope*** when it is bounded):

$$\mathcal{X} = \{x \in \mathbb{R}^n : A_{eq} x = b_{eq},\ Ax \leq b\}$$

  - **Empty feasible set**. If the feasible set is empty (i.e. the linear equalities and inequalities have no intersection), then there is no feasible and hence no optimal solution. By convention, the optimal objective is $p^* = +\infty$.
  - **Non-empty feasible set and bounded**. If the feasible set is nonemtpy and bounded, then the LP attains an optimal solution and the objective value p^* is finite. In this case, any optimal solution x^* is on a **vertex**, **edge** or **facet** of the feasible polytope. In particular, the optimal solution is unique if the optimal cost hyperplane $\{x : c^T x = p^*\}$ intersects the feasible polytope only at a vertex.
  - **Non-empty feasible set and unbounded**. If the feasible set is nonempty and unbounded, then the LP may or may not attain an optimal solution, depending on the cost direction c, and there exist direction c such that the LP is unbounded below. (i.e. $p^* = -\infty$ and the solution x^* "drifts" to **infinity**)

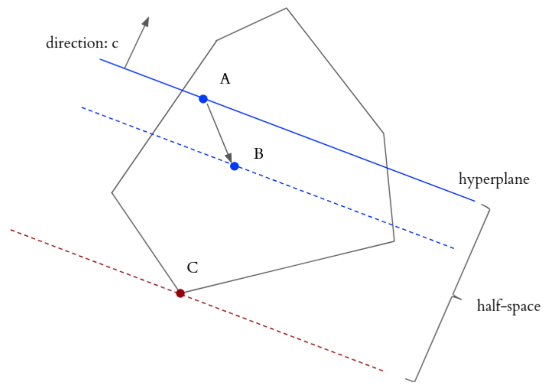

Figure 4.4: The bounded region is the feasible set, point A is x_f, point B is x, and the direction from A to B is $x - x_f$. The hyperplane is $\{x : c^T x = c^T x_f\}$ with direction c, and the half-space is $\{x : c^T(x - x_f) < 0\}$. Point c is the optimal value x^*.

4.3.2 Polyhedral Functions

- **Polyhedral function**. We say that a function $f : \mathbb{R}^n \to \mathbb{R}$ is polyhedral if its *epigraph*

  $$\text{epi}\, f = \left\{(x, t) \in \mathbb{R}^{n+1} : f(x) \le t\right\}$$

  can be expressed as a *polyhedron*

  $$\text{epi}\, f = \left\{(x, t) \in \mathbb{R}^{n+1} : C \begin{bmatrix} x \\ t \end{bmatrix} \le d\right\} \tag{4.3}$$

  for some matrix $C \in \mathbb{R}^{m,n+1}$, and vector $d \in \mathbb{R}^m$.

Examples of Polyhedral Function

- **Maxima of affine functions**. Polyhedra functions include functions that can be expressed as a maximum of a finite number of affine functions:

  $$f(x) = \max_{i=i,\cdots,m} a_i^T x + b_i$$

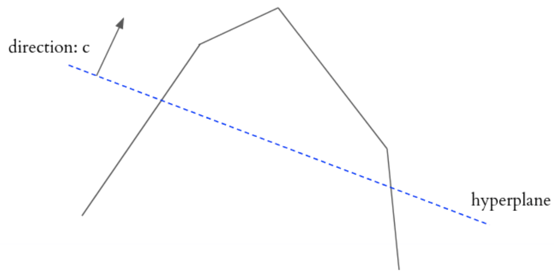

Figure 4.5: An LP with unbounded optimal objective

where $a_i \in \mathbb{R}^n, b_i \in \mathbb{R}$. For any family of functions $f_\alpha(x)$ parameterized by $\alpha \in \mathcal{A}$, it holds that

$$\max_{\alpha \in \mathcal{A}} f_\alpha(x) \leq t \Leftrightarrow f_\alpha(x) \leq t,\ \forall \alpha \in \mathcal{A}$$

The epigraph of f

$$\text{epi } f = \left\{(x,t) \in \mathbb{R}^{n+1} : \max_{i=1,\cdots,m} a_i^T x + b_i \leq t\right\}$$

can be expressed as the polyhedron

$$\text{epi } f = \left\{(x,t) \in \mathbb{R}^{n+1} : \alpha_i^T x + b_i \leq t,\ i = 1,\cdots,m\right\}$$

- L_1**-norm function**. The L_1-norm function $f(x) = \|x\|_1, x \in \mathbb{R}^m$, is polyhedra since it can be written as the maximum of $2n$ affine functions:

$$f(x) = \max_{i=1,\cdots,m} \max(x_i, -x_i)$$

- **Sum of maxima of affine functions**. Polyhedra functions also include functions that can be expressed as a **sum** of functions which are themselves maxima of affine functions:

$$f(x) = \sum_{j=1}^{q} \max_{i=1,\cdots,m} a_{ij}^T x + b_{ij}$$

- The condition $(x, t) \in \text{epi} f$ is equivalent to the existence of a vector $u \in \mathbb{R}^q$ such that

$$\sum_{j=1}^{q} u_j \leq t, \quad a_{ij}^T x + b_{ij} \leq u_{ij}, i = 1, \cdots, m; j = 1, \cdots, q \tag{4.4}$$

hence, epi f is the projection (on the space of (x, t)-variables) of a polyhedron, which is itself a polyhedron.

4.3.3 Minimization of Polyhedra Functions

- The problem of minimizing a polyhedral function f, under linear equality or inequality (polyhedra) constraints, such as

$$\min_x f(x) : \; Ax \leq b$$

can be cast as an LP

$$\min_{x,t} t : Ax \leq b, \; (x, t) \in \text{epi} f$$

- Since epi f is a polyhedron, it can be expressed as in (4.3), hence the problem is an LP of the form

$$\min_{x,t} t : C \begin{bmatrix} x \\ t \end{bmatrix} \leq d$$

- Note that explicit representation of the LP in a standard form may require the introduction of additional **slack variables**, as was done in (4.4).

- **Minimization of Maxima of Affine Functions**. Assume that f is defined as the maximum of linear functions. Then the problem

$$\min_x \max_{1 \leq i \leq m} (a_i^T x + b_i) : Cx \leq d$$

can be expressed as the LP

$$\min_{x,t} t : Cx \leq d, \; a_i^T x + b_i \leq t$$

The objective function is linear in the variables (x, t) and the constraints are ordinary inequalities involving affine functions.

- **Minimization of a Sum of Maxima of Affine Functions**. We can formulate the problem of minimizing the function f with values

$$f(x) = \sum_{j=1}^{p} \max_{1 \leq i \leq m} (a_{ij}^T x + b_{ij})$$

under polytopic constraints as an LP by **introducing a new variable** for each max-linear function that appears in the function f. We obtain the LP representation

$$\min_{x,t} \sum_{j=1}^{p} t_j : t_j \geq a_{ij}^T x + b_{ij},\ Cx \leq d$$
$$i = 1, \cdots, m,\ j = 1, \cdots, p$$

4.4 Quadratic Programs

4.4.1 Definition

- A *quadratic program* (QP) is an optimization problem of the standard form where the **objective function** f_0 is a *quadratic* function and the **constraint functions**, $f_0, \cdots, f_m$ are *affine* functions.

$$p^* = \min_x \quad \frac{1}{2} x^T H x + c^T x \tag{4.5}$$
$$\text{s.t.} \quad A_{eq} x = b_{eq} \tag{4.6}$$
$$Ax \leq b \tag{4.7}$$

- The feasible set of QP is polyhedron (as in LP), but the objective is *quadratic*, rather than linear.

- If the H matrix is ***positive-semidefinite***, then the QP is *convex*.

- LPs are special cases of QPs, in which the matrix H is zero.

4.4.2 Constrained Least Squares

Quadratic programs arise naturally from least-squares problems when linear equality or inequality constraints need to be enforced on the decision variables. A linearly-constrained LS problem takes the form

$$\begin{aligned} p^* = \min_x \quad & \|Rx - y\|_2^2 \\ \text{s.t.}: \quad & A_{eq}x = b_{eq} \\ & Ax \leq b \end{aligned}$$

This is a convex QP, having objective (neglecting a constant term $d = \|y\|^2$)

$$f_0(x) = \frac{1}{2}x^T H x + c^T x$$

with $H = 2R^T R \succeq 0, c^T = -2y^T R$.

4.4.3 Quadratic Constrained Quadratic Programs

A generalization of the QP model is obtained by allowing quadratic equality and inequalities constraints. A *quadratic constrained quadratic program* (QCQP) takes the form

$$\begin{aligned} P^* = \min_x \quad & x^T H_0 x + 2c_0^T x + d_0 \\ \text{s.t.}: \quad & x^T H_i x + 2c_i^T x + d_i \leq 0, \quad i \in \mathcal{I} \\ & x^T H_j x + 2c_j^T x + d_j = 0, \quad j \in \mathcal{J} \end{aligned}$$

where $\mathcal{I}, \mathcal{J}$ denote the index sets relative to constraints.

A QCQP is convex if and only if the objective and the inequality constraints are convex quadratic, and all the equality constraints are actually affine, $H_0 \succeq 0, H_i \succeq 0, H_j = 0$.

4.5 Modeling with LP and QP

4.5.1 Problems Involving Cardinality and Their L_1 Relaxations

Many engineering applications require the determination of solutions that are *sparse*, that possess only few *non-zero* entries (*low-cardinality solutions*). However, finding for low-cardinality solutions (i.e., solutions with small L_0 norm) is hard in general, from a computational point of view. For this reason, several *heuristics* are often used. For example, replacing the L_0 norm with the L_1 norm.

Cardinality Minimization

- **Cardinality (L_0 norm)**. The *cardinality* of a vector $x \in \mathbb{R}^n$ is the number of **non-zero** elements in it. It is sometimes called the L_0 norm of x, although the cardinality function is ***not a norm***. The cardinality is denoted $\text{card}(x)$ or $\|x\|_0$. The cardinality function is difficult to optimize; thus, in cardinality minimization problems, the L_1 norm is often used as a surrogate.

- **Convex envelope**. The *convex envelope* env f of a function $f : C \to \mathbb{R}$ is the **largest** convex function that is an **under estimator** of f on C, i.e. env $f \leq f(x)\, \forall x \in C$ and no other convex function is uniformly larger then env f on C

$$\text{env}\, f = \sup \{\phi : C \to \mathbb{R} : \phi \text{ is convex and } \phi \leq f\}$$

- Intuitively, the epigraph of the convex envelope of f corresponds to *convex hull* of the epigraph of f (see Figure 4.6).

- **Cardinality minimization**. Many problems in engineering and scientific computing can be cast as

$$\min_x \text{Card}(x) : x \in P$$

  where P is a *polyhedron* (a convex set). A related problem is a penalized version of the above, where we seek to trade-off an objective function against cardinality:

$$\min_x f(x) + \lambda \text{Card}(x) : x \in P$$

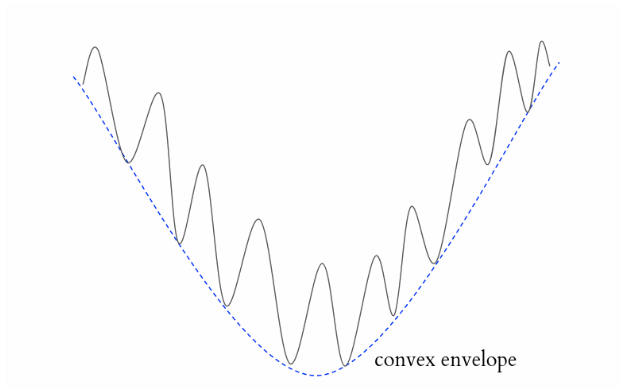

Figure 4.6: A non-convex function f and its convex envelope (dashed) env f

, where $f(x)$ is some (usually convex) **cost** function, and $\lambda > 0$ is a **penalty** parameter.

- **The L_1 norm heuristic**. The L_1 *norm heuristic* consists in replacing the above (non-convex) cardinality function $\text{Card}(x)$ with a polyhedral (convex) one, involving the L_1 norm. This heuristic leads to replace the above prolem with

$$\min_x \|x\|_1 : x \in P$$

where P is a polyhedron.

- The reason why this works is that L_1 norm provides a *lower bound* for the original L_0 problem. The L_1 norm heuristic is convex and can be written as the QP by adding slack variables.

4.5.2 LP Relaxations of Boolean Problems

Definition

- **Boolean problems**. A Boolean optimization problem is one where the variables are constrained to be *Boolean* (i.e. to take on values in $\{0, 1\}$).

$$p^* = \min_x c^T x : Ax \leq b,\ x \in \{0, 1\}^n$$

Such problems are usually very hard to solve exactly, since they potentially require combinatorial enumeration of all the 2^n possible points in $\{0, 1\}^n$.

- **LP relaxation**. A tractable *relaxation* of a Boolean problem is typically obtained by replacing the discrete set $\{0,1\}^n$ with the hypercube $[0,1]^n$, which is a convex set.

$$\bar{p}^* = \min_x c^T x \quad \text{s.t. : } \; Ax \le b, \; x \in [0,1]^n$$

- **Lower bound**. The feasible set of the relaxed problem is larger than (includes) the feasible set of the original problem, the relaxation provides a *lower bound* on the original problem: $\bar{p}^* \le p^*$.

Total Unimodularity and Exact Solutions

Boolean problems are not always hard to solve. If the solution of the LP relaxation is Boolean, then this solution provides an exact solution (optimal) for the original Boolean problem. Such a solution arises when b is an integer and the A matrix has a property called *total unimodularity*.

- **Totally unimodular (TUM)**. A matrix A is *totally unimodular (TUM)* if every square submatrix of A has determinant -1, 1, or 0. Polytopes defines via TUM matrices have *integer* vertices.

- **Weighted bipartite matching**. A weighted bipartite matching problem arises when n agents need to be assigned to n tasks, in a one-to-one fashion, and the cost of matching agent i to task j is w_{ij}.

  We define variables x_{ij} such that $x_{ij} = 1$ if agent i is assigned to task j and $x_{ij} = 0$ otherwise, the problem can be written as

$$\begin{aligned} p^* = \min_x \quad & \sum_{i,j=1}^{n} w_{ij} x_{ij} \\ \text{s.t. : } \quad & x_{ij} \in \{0,1\} \quad \forall i,j = 1,\cdots,n \\ & \sum_{i=1}^{n} x_{ij} = 1 \quad \forall j = 1,\cdots,n \text{ (one agent for each task)} \\ & \sum_{j=1}^{n} x_{ij} = 1 \quad \forall i = 1,\cdots,n \text{ (one task for each agent)} \end{aligned}$$

An LP relaxation is obtained by dropping the integer constraint on the x_{ij} variables, obtaining $x_{ij} \geq 0$.

- **Shortest path** The shortest path problem is the problem of finding a path between two vertices (or nodes) in a directed graph such that the sum of the weights along the edges in the path is minimized. The shortest path problem can be solved very efficiently with specialized algorithms based on the LP relaxation.

4.5.3 Other LP and QP Problems

- Linear binary classification
- Network flows
- Portfolio optimization
- Nash equilibria in zero-sum games
- Filter design

4.6 LS-related Quadratic Programs

A major source of quadratic problems comes from LS problems. The standard LS objective

$$f_0(x) = \|Ax - y\|_2^2$$

is a convex quadratic function, which can be written in the standard form

$$f_0(x) = \frac{1}{2}x^T Hx + c^T x + d$$

with $H = 2(A^T A), c = -2A^T y, d = y^T y$.

Finding the unconstrained minimum of f_0 is a linear algebra problem. This amounts to finding the solution for the system of linear equations from the optimality condition $\nabla f_0(x) = 0$ (*normal equation*):

$$A^T Ax = A^T y$$

We next illustrate some variants of the basic LS problem.

4.6.1 Equality Constrained LS

Minimizing a convex quadratic function under linear equality constraints is equivalent to solving an **augmented system of linear equations**. Solving the linear equality constrained LS problem

$$\begin{aligned} \min_{x} \quad & \|Ax - y\|_2^2 \\ \text{s.t.}: \quad & Cx = d \end{aligned}$$

is equivalent to solving the following linear equations in x, λ:

$$\begin{bmatrix} C & 0 \\ A^T A & C^T \end{bmatrix} \begin{bmatrix} x \\ \lambda \end{bmatrix} = \begin{bmatrix} d \\ A^T y \end{bmatrix}$$

4.6.2 L_1 Regularization and The LASSO Problem

- **Basis pursuit denoising problem (BPDN).** The regularized LS problems with L_1 norm is known as the *basis pursuit denoising problem* (BPDN):

$$\min_{x \in \mathbb{R}^n} \|Ax - y\|_2^2 + \lambda \|x\|_1, \; \lambda \geq 0 \tag{4.8}$$

  where $\|x\|_1 = |x_1| + \cdots + |x_n|$. The basic idea is that the L_1 norm of x us used as a proxy for the cardinality of x (the number of nonzero entries in x).

- **Trade-off**. The interpretation is that it formalizes a trade-off between the ***accuracy*** with which Ax approximates y and the ***complexity*** of the solution, intended as the number of nonzero entries in x.

- Larger λ means the problem is biased towards finding low-complexity (more zeros) solutions.

- A problem similar to (4.8) is in the context of *piece-wise constant fitting*. Problem (4.8) can be cast in the form of a standard QP by introducing slack variables $u \in \mathbb{R}^n$:

$$\begin{aligned} \min_{x, u \in \mathbb{R}^n} \quad & \|Ax - y\|_2^2 + \lambda \sum_{i=1}^{n} u_i \\ \text{s.t.}: \quad & |x_i| \leq u_i, \; i = 1, \cdots, n \end{aligned}$$

- **Least absolute shrinkage and selection operator (LASSO)**. An analogous version of problem (4.8) is obtained by imposing a constraint on the L_1 norm of x, instead of inserting this term in the objective as a penalty. This is called *least absolute shrinkage and selection operator* (LASSO) problem (often, it is also used to refer to problem (4.8)).

$$\begin{aligned} \min_{x \in \mathbb{R}^n} \quad & \|Ax - y\|_2^2 \\ \text{s.t.}: \quad & \|x\|_1 \leq \alpha \end{aligned}$$

- The LASSO problem can be formulated in the form of minimization of $\|x\|_1$ subject to a constraint on the residual norm

$$\begin{aligned} \min_{x \in \mathbb{R}^n} \quad & \|x\|_1 \\ \text{s.t.}: \quad & \|Ax - y\|_2 \leq \epsilon \end{aligned}$$

which can also be cast as QCQP. All these variations on the LASSO problem yield convex optimization models that can be solved by standard efficient algorithms for QCQP, at least in principle.

4.7 Geometric Programs

Geometric programming (GP) is an optimization model where the variables are non-negative, and the objective and constraints are sums of powers of those variables, with non-negative weights. This arises naturally in the context of geometric design, or with models of processes that are well approximated with power laws. Although GPs are not convex, we can transform them, via a change of variables, into convex problems. In its convex form, GP can be seen as a natural extension of LP.

4.7.1 Monomials and Posynomials

Monomials

- **Monomials**. A function $f : \mathbb{R}^n \to \mathbb{R}$ is *monomial* if its domain is R_{++} (the set of vectors with positive components) and its value take the form

$$f(x) = c x_1^{a_1} x_2^{a_2} \cdots x_n^{a_n}, \quad x \in \mathbb{R}^n > 0, c > 0, a \in \mathbb{R}^n$$

where we follow the *power law notation*.

- **Log-linearity and Power Laws**. Monomials are closely related to linear or affine functions. If f is a monomial in variable x, the $\log f$ is affine in the vector. Hence monomial functions could be called *log-linear*.

- Just as linear models are important in (approximate) models between general variables, monomials play an ubiquituous role for modeling relationships between **positive variables**, such as prices, concentrations, energy, or geometric data such as length, area and volume, etc.

Posynomials

- **Posynomial**. A *posynomial* is defined as a function $f : \mathbb{R}^n_{++} \to \mathbb{R}$ which is a **non-negative linear combination** of positive monomials:

$$f(x) = \sum_{i=1}^{K} c_i x^{a_{(i)}}, x > 0$$

  where $c_i > 0$ and $a_{(i)} \in \mathbb{R}^n$.

- **Generalized posynomial**. A *generalized posynomial* is any function obtained from posynomials via addition, multiplication, pointwise maximum, and raising to a constant power. For example,

$$f(x) = \max\left(2x_1^{2.3}x_2^7, x_1x_2x_3^{3.14}, \sqrt{x_1 + x_2^3}\right)$$

4.7.2 Convex Representation of Posynomials

Monomials and (generalized) posynomials are not convex. Consider a posynomial function f. Instead of the original (positive) variables, we use the new variable $y_i = \log x_i, i = 1, \cdots, n$. We then take the logarithm of the function f.

- **Monomial**. For a monomial $f(x) = cx_1^{a_1}x_2^{a_2} \cdots x_n^{a_n}$ where $a \in \mathbb{R}^n, x \in \mathbb{R}^n_{++}$ and $c > 0$, taking a **logarithmic change of variables**

$$y_i = \log x_i$$

we have

$$\begin{aligned}
\tilde{g(y)} &= f(x(y)) = ce^{a_1 y_1} \cdots e^{a_n y_n} = ce^{a_1 y_1 + \cdots + a_n y_n} \\
[\text{letting } b \doteq \log c] &= e^{a^T y + b} \\
\log f(x) &= a^T y + b
\end{aligned}$$

where $y_i = \log x_i$ and $b = \log c$. The transformation yields an *affine* function.

- **Posynomial**. For a posynomial $f(x) = \sum_{k=1}^{K} c_k x_{a_k}$ where $c > 0$,

  $$\log f(x) = \log\left(\sum_{k=1}^{K} e^{a_k^T y + b_k}\right)$$

  where $b_k = \log c_k$.

- **Log-sum-exp**. The above can be written to

  $$\log f(x) = \text{lse}(Ay + b)$$

  where A is the $K \times n$ matrix with rows $a_1, \cdots, a_K, b \in R^K$, and *lse* is the log-sum-exp function, which is convex. We can view a posynomial as the **log-sum-exp function of an affine combination of the logarithm** of the original variables.

Convex Representation of Generalized Posynomials

We can transform generalized posynomail inequalitites into convex by adding variables and taking logarithmic change of variables.

- **Example**. Consider the posynomial

  $$f(x) = \max(f_1(x), f_2(x)), \mathbb{R}^n_{++} \to \mathbb{R}$$

  where f_1, f_2 are two posynomials. For $t > 0$, the constraint $f(x) \leq t$ can be expressed as **two** posynomials constraints in (x, t), $f1(x) \leq t$, $f_2(x) \leq t$.

- **Example**. For $t > 0, \alpha > 0$, consider the power constraint

$$(f(x)^{\alpha}) \leq t$$

where f is an ordinary posynomial. Since $\alpha > 0$, the above is equivalent to

$$f(x) \leq t^{1/a}$$

which is equivalent to the posynomial constraint in (x, t)

$$g(x,t) \doteq t^{-1/a} f(x) \leq 1$$

Hence, by adding as many variables as necessary, we can express a generalized posynomial constraint as a set of ordinary posynomial ones.

4.7.3 Standard Forms of GP

- **Standard form**. A geometric program (GP) involves generalized posynomial objective and inequality constraints, and (possibly) monomial equality constraints.

$$\begin{aligned} \min_{x} \quad & f_0(x) \\ \text{s.t.:} \quad & f_i(x) \leq 1, \quad i = 1, \cdots, m \\ & h_i(x) = 1, \quad i = 1, \cdots, p \end{aligned}$$

where $f_0, \cdots, f_m$ are generalized posynomials, and $h_i, i = 1, \cdots, p$ are positive monomials.

- **Standard posynomials standard form**. Assuming for simplicity that the $f_0, \cdots, f_m$ are standard posynomials, we can express the above GP as

$$\begin{aligned} \min_{x} \quad & \sum_{k=1}^{K_0} c_{k_0} x^{a_{(k_0)}} \\ \text{s.t.:} \quad & \sum_{k=1}^{K_i} c_{k_i} x^{a_{(k_i)}} \leq 1, \quad i = 1, \cdots, m \\ & g_i x^{r_{(i)}} = 1, \quad i = 1, \cdots, p \end{aligned}$$

where $a_{(k_0)}, \cdots, a_{(k_m)}, r_{(1)}, \cdots, r_{(p)}$ are vectors in $\mathbb{R}^n$, and c_{k_i}, g_i are positive scalars.

- **Convex form**. Using the logarithmic transformation, we can rewrite the above non-convex GP into an equivalent convex formulation,

$$\begin{aligned} \min_{y} \quad & \text{lse}(A_0 y + b_0) \\ \text{s.t.:} \quad & \text{lse}(A_i y + b_i) \leq 0, \quad i = 1, \cdots, m \\ & Ry + h = 0 \end{aligned}$$

  where A_i is a matrix with rows $a_{1_i}^T, \cdots, a_{K_i i}^T$, b_i is a vector with elements $c_{1_i}, \cdots, c_{K_i i}$. R is a matrix with rows $r_{(1)}^T, \cdots, r_{(p)}^T$, and h is a vector with elements $\log g_1, \cdots, \log g_p$.