11. Special classes of convex optimization

- quadratic optimization
- semidefinite programs
- geometric programming
- quasiconvex optimization

Quadratic optimization

Quadratic program (quadratic optimization problem)

$$\begin{array}{ll} \text{minimize} & (1/2)\boldsymbol{x}^T Q \boldsymbol{x} + \boldsymbol{r}^T \boldsymbol{x} \\ \text{subject to} & A\boldsymbol{x} \leq \boldsymbol{b} \\ & G\boldsymbol{x} = \boldsymbol{h}, \end{array}$$

where $Q \in \mathbb{R}^{n \times n}$ is positive semidefinite, $\boldsymbol{r} \in \mathbb{R}^n$, $A \in \mathbb{R}^{m \times n}$, $G \in \mathbb{R}^{p \times n}$, $\boldsymbol{h} \in \mathbb{R}^p$, and $\boldsymbol{b} \in \mathbb{R}^m$

Quadratically constrained quadratic problem (QCQP)

$$\begin{array}{ll} \text{minimize} & (1/2)\boldsymbol{x}^T Q_0 \boldsymbol{x} + \boldsymbol{q}_0 \boldsymbol{x} + \boldsymbol{r}_0 \\ \text{subject to} & (1/2)\boldsymbol{x}^T Q_i \boldsymbol{x} + \boldsymbol{r}_i^T \boldsymbol{x} \leq 0, \quad i = 1, \ldots, p \\ & A\boldsymbol{x} = \boldsymbol{b}, \end{array}$$

where Q_i ($i = 0, 1 \ldots, m$) are positive semidefinite

Examples

Least squares:

$$\text{minimize} \quad \|A\boldsymbol{x} - \boldsymbol{b}\|^2 = \boldsymbol{x}^T A^T A \boldsymbol{x} - 2\boldsymbol{b}^T A \boldsymbol{x} + \boldsymbol{b}^T \boldsymbol{b}$$

Constrained least squares: when linear constraints are added, the problem is called *constrained least-square* for example,

$$\begin{array}{ll} \text{minimize} & \|A\boldsymbol{x} - \boldsymbol{b}\|^2 \\ \text{subject to} & G\boldsymbol{x} = \boldsymbol{h} \\ & l_i \leq x_i \leq u_i, \quad i = 1, \ldots, n \end{array}$$

this problem has no simple analytical solution

Example: Power distribution (aggregator model)

- in electricity markets, an aggregator
  - buys wholesale p units of power (Megawatt) from power distribution utilities
  - and resells this power to a group of n business or industrial customers
- the ith customer, $i = 1, \ldots, n$, would ideally wants p_i Megawatts
- the customer i does not want to receive more or less power than needed; the customer dissatisfaction can be modeled as

$$f_i(x_i) = c_i(x_i - p_i)^2, \quad i = 1, \ldots, n,$$

where x_i is the power given to customer i by the aggregator, and c_i is a given customer parameter

- the aggregator problem is finding the power allocations x_i, $i = 1, \ldots, n$, such that
  - the average customer dissatisfaction is minimized,
  - the whole power p is sold,
  - and that the dissatisfaction level is no greater than a contract level, say d
- the aggregator problem is

$$\begin{array}{ll} \text{minimize} & \displaystyle\frac{1}{n}\sum_{i=1}^{n} c_i(x_i - p_i)^2 \\ \text{subject to} & \displaystyle\sum_{i=1}^{n} x_i = p, \\ & c_i(x_i - p_i)^2 \leq d, \quad i = 1, \ldots, n \\ & x_i \geq 0, \quad i = 1, \ldots, n \end{array}$$

this is a QCQP

Example: portfolio optimization

an investor wants to select a set of assets (stocks) to achieve a good return on the investment while minimizing risks of losses

- we have n stocks and let $x_i \geq 0$ be the proportion of investment on stock i
- let r_i be the return for stock i; we assume that the expected returns are known,

$$\mu_j = \mathbb{E}\left(r_j\right), \quad j = 1, 2, \ldots, n,$$

  and that the covariances of all the pairs of variables are also known,

$$\sigma_{i,j}^2 = \mathbb{E}\,(r_i - \mu_i)(r_j - \mu_j), \quad i, j = 1, 2, \ldots, n$$

  (typically, the mean and variance are estimated from historical data)

  - a high variance indicates high risk; a low variance indicates low risk
  - positive covariance $\sigma_{ij}^2 > 0$ means that the stocks i and j prices move in the same general direction while a negative one moves in opposite direction

- the overall return is the random variable $R = \sum_{j=1}^{n} x_j r_j$ whose expectation and variance are given by

$$\mathbb{E}(R) = \boldsymbol{\mu}^T \boldsymbol{x}, \quad \text{Var}(R) = \boldsymbol{x}^T \Sigma \boldsymbol{x},$$

- $\boldsymbol{\mu} = (\mu_1, \mu_2, \ldots, \mu_n)$
- Σ is the covariance matrix whose elements are given by $\Sigma_{i,j} = \sigma_{i,j}$ for all $i, j = 1, \ldots, n$
- the covariance matrix is always positive semidefinite

Portfolio problem QP formulation:

$$\begin{array}{ll} \text{minimize} & \boldsymbol{x}^T \Sigma \boldsymbol{x} \\ \text{subject to} & \boldsymbol{\mu}^T \boldsymbol{x} \geq \alpha \\ & \mathbf{1}^T \boldsymbol{x} = 1 \\ & \boldsymbol{x} \geq \mathbf{0} \end{array}$$

where α is the minimal return value

Portfolio problem QCQP formulation:

$$\begin{array}{ll} \text{maximize} & \boldsymbol{\mu}^T \boldsymbol{x} \\ \text{subject to} & \boldsymbol{x}^T \Sigma \boldsymbol{x} \leq \beta \\ & \mathbf{1}^T \boldsymbol{x} = 1 \\ & \boldsymbol{x} \geq \mathbf{0} \end{array}$$

where β is the upper bound on the risk

Outline

- quadratic optimization
- **semidefinite programs**
- geometric programming
- quasiconvex optimization

Linear matrix inequalities

a *linear matrix inequality* (LMI) constrains a vector of variables $\boldsymbol{x} \in \mathbb{R}^n$ as

$$F(\boldsymbol{x}) = F_0 + \sum_{i=1}^{m} x_i F_i \leq 0 \qquad (11.1)$$

- symmetric coefficient matrices $F_0, \ldots, F_n$ of size $m \times m$
- often, LMI constraints apply directly to matrix variables rather than vector variables

Semidefinite program

a *semidefinite program* (SDP) is a particular type of convex optimization problem:

$$\begin{array}{ll} \text{minimize} & \boldsymbol{c}^T\boldsymbol{x} \\ \text{subject to} & F(\boldsymbol{x}) \leq 0 \end{array} \tag{11.2}$$

with

$$F(\boldsymbol{x}) = F_0 + \sum_{i=1}^{n} x_i F_i$$

- $\boldsymbol{x} \in \mathbb{R}^n$ is the optimization variable and $\boldsymbol{c} \in \mathbb{R}^n$
- each F_i for $i = 0, \ldots, n$ is a known $m \times m$ symmetric matrices
- if $F_0, F_1, \ldots, F_m$ are diagonal matrices, the LMI simplifies to n linear inequalities, and thus, the SDP (11.2) becomes a linear program

General form SDPs:

$$\begin{array}{ll} \text{minimize} & \boldsymbol{c}^T\boldsymbol{x} \\ \text{subject to} & F^{(i)}(\boldsymbol{x}) = x_1F_1^{(i)} + \cdots + x_nF_n^{(i)} + F_0^{(i)} \preceq 0, \quad i = 1, \ldots, K \\ & G\boldsymbol{x} \preceq \boldsymbol{h} \\ & A\boldsymbol{x} = \boldsymbol{b} \end{array}$$

these problems can be equivalently represented as an SDP by constructing a block diagonal LMI using the given LMIs and linear inequalities:

$$\begin{array}{ll} \text{minimize} & \boldsymbol{c}^T\boldsymbol{x} \\ \text{subject to} & \mathrm{diag}(G\boldsymbol{x} - \boldsymbol{h}, F^{(1)}(\boldsymbol{x}), \ldots, F^{(K)}(\boldsymbol{x})) \preceq 0 \\ & A\boldsymbol{x} = \boldsymbol{b} \end{array}$$

Examples

Maximum eigenvalue minimization

$$\text{minimize} \quad \lambda_{\max}(F(\boldsymbol{x}))$$

generally, the function $\lambda_{\max}(\cdot)$ is nonconvex, but this problem can be equivalently reformulated as:

$$\begin{array}{ll} \text{minimize} & t \\ \text{subject to} & F(\boldsymbol{x}) - tI \leq 0, \end{array}$$

where the variables are $\boldsymbol{x} \in \mathbb{R}^n$ and $t \in \mathbb{R}$

this formulation is a specific instance of a semidefinite program (SDP) in the augmented (vector) variable:

$$\hat{\boldsymbol{x}} = \left[\begin{array}{c} t \\ \boldsymbol{x} \end{array} \right], \quad \hat{\boldsymbol{c}} = (1, 0, \ldots, 0)\,, \quad \hat{F}(\hat{\boldsymbol{x}}) = F(\boldsymbol{x}) - tI$$

Spectral matrix norm minimization:

$$\text{minimize} \quad \|A(\boldsymbol{x})\|_2$$

where $A(\boldsymbol{x}) \in \mathbb{R}^{p\times m}$ represented as:

$$A(\boldsymbol{x}) = A_0 + x_1 A_1 + \cdots + x_n A_n$$

this problem is equivalent to the following SDP:

$$\begin{array}{ll} \text{minimize} & t \\ \text{subject to} & \begin{bmatrix} tI_m & A^T(\boldsymbol{x}) \\ A(\boldsymbol{x}) & tI_p \end{bmatrix} \geq 0 \end{array}$$

with decision variables $\boldsymbol{x} \in \mathbb{R}^n$ and $t \in \mathbb{R}$ ($t \geq 0$)

- to show this, recall that the spectral norm is equal to the largest singular value of $A(\boldsymbol{x})$:

$$\|A(\boldsymbol{x})\|_2 = \sqrt{\lambda_{\max}\left(A^T(\boldsymbol{x})A(\boldsymbol{x})\right)}$$

- it follows that

$$\|A(\boldsymbol{x})\|_2 \leq t \implies \|A(\boldsymbol{x})\|_2^2 \leq t^2 \implies \lambda_{\max}\left(A^T(\boldsymbol{x})A(\boldsymbol{x})\right) \leq t^2$$

- this is satisfied if the eigenvalue condition holds for every i:

$$\lambda_i\left(A^T(\boldsymbol{x})A(\boldsymbol{x})\right) \leq t^2, \quad \forall\, i = 1, \ldots, m$$

- the above can be equivalently stated as:

$$A^T(\boldsymbol{x})A(\boldsymbol{x}) - t^2 I_m \preceq 0$$

- using the Schur complement rule, this matrix inequality can be transformed to the LMI form with variables t^2 and $\boldsymbol{x}$:

$$\begin{bmatrix} t^2 I_m & A^T(\boldsymbol{x}) \\ A(\boldsymbol{x}) & I_p \end{bmatrix} \geq 0$$

since $t = 0$ if and only if $A(\boldsymbol{x}) = 0$

- we can represent the above LMI as:

$$\begin{bmatrix} tI_m & A^T(\boldsymbol{x}) \\ A(\boldsymbol{x}) & tI_p \end{bmatrix} \geq 0,$$

obtained through congruence transformation with the diagonal matrix

$$\text{diag}(1/\sqrt{t}I_m, \sqrt{t}I_p)$$

for $t > 0$

Frobenius norm minimization

$$\text{minimize} \quad \|A(\boldsymbol{x})\|_F^2$$

the objective can be represented in the SDP format:

$$\begin{array}{ll} \text{minimize} & \text{trace}(Y) \\ \text{subject to} & \begin{bmatrix} Y & A(\boldsymbol{x}) \\ A^T(\boldsymbol{x}) & I_m \end{bmatrix} \geq 0 \end{array}$$

where the decision variables are $\boldsymbol{x} \in \mathbb{R}^n$ and $Y \in \mathbb{R}^{p \times p}$ is positive semidefinite

- the equivalence of this formulation can be established by noting the relationship:

$$\|A(\boldsymbol{x})\|_F^2 = \text{trace}(A(\boldsymbol{x})A^T(\boldsymbol{x}))$$

- using the Schur complement, the matrix condition can be written as:

$$\begin{bmatrix} Y & A(\boldsymbol{x}) \\ A^T(\boldsymbol{x}) & I_m \end{bmatrix} \geq 0 \iff A(\boldsymbol{x})A^T(\boldsymbol{x}) \leq Y$$

  this validation links the original objective with the SDP representation

Outline

- quadratic optimization
- semidefinite programs
- **geometric programming**
- quasiconvex optimization

Monomials and posynomials

Monomial

$$f(\boldsymbol{x}) = cx_1^{a_1}x_2^{a_2}\dots x_n^{a_n}$$

- $f : \mathbb{R}^n \to \mathbb{R}$ with domain $\mathrm{dom}\, f = \mathbb{R}^n_{++}$
- $c > 0$ and each $a_i \in \mathbb{R}$

Posynomial

$$f(\boldsymbol{x}) = \sum_{k=1}^{K} c_k x_1^{a_{1k}} x_2^{a_{2k}} \dots x_n^{a_{nk}}$$

each $c_k > 0$

Example

wireless cellular network with n paired transmitters and receivers

- $p_1, \dots, p_n$ are the transmit powers for these pairs
- each transmitter i is intended to communicate with its corresponding receiver i
- the signal to interference plus noise ratio (SINR) for each receiver is:

$$\gamma_i = \frac{S_i}{l_i + \sigma_i}, \quad i = 1, \dots, n,$$

  - S_i represents the power of the desired signal received from transmitter i
  - l_i is the combined interference from all other transmitters
  - σ_i is the receiver's noise power

the Rayleigh fading model suggests that the received power S_i is a linear function of the transmitted powers $p_1, \ldots, p_n$:

$$S_i = G_{ii} p_i, \quad i = 1, \ldots, n,$$

and

$$l_i = \sum_{j \neq i} G_{ij} p_j,$$

where G_{ij} are the known path gains from transmitter j to receiver i

therefore, the SINR expressions in terms of the powers $p_1, \ldots, p_n$ are:

$$\gamma_i(\boldsymbol{p}) = \frac{G_{ii} p_i}{\sigma_i + \sum_{j \neq i} G_{ij} p_j}, \quad i = 1, \ldots, n,$$

while the SINR functions aren't posynomials, their inverses are posynomial functions of the powers:

$$\gamma_i^{-1}(\boldsymbol{p}) = \frac{\sigma_i}{G_{ii}} p_i^{-1} + \sum_{j \neq i} \frac{G_{ij}}{G_{ii}} p_j p_i^{-1}, \quad i = 1, \ldots, n$$

Generalized posynomials

a generalized posynomial is obtained from posynomials through various operations like addition, multiplication, pointwise maximum, and raising to a specific power

Example: consider the function $f : \mathbb{R}^3_{++} \to \mathbb{R}$ defined as:

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

this function qualifies as a generalized posynomial

Introducing variables

Max of posynomial: consider a posynomial f expressed as:

$$f(\boldsymbol{x}) = \max\left(f_1(\boldsymbol{x}), f_2(\boldsymbol{x})\right),$$

where both f_1 and f_2 are posynomials, then for some $t > 0$, the inequality $f(\boldsymbol{x}) \leq t$ can be broken down into two posynomial inequalities in $(\boldsymbol{x}, t)$:

$$f_1(\boldsymbol{x}) \leq t \quad \text{and} \quad f_2(\boldsymbol{x}) \leq t$$

Power of posynomial: for a given $t > 0$ and $a > 0$, the power constraint

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

with f being a regular posynomial and $\alpha > 0$, is equivalent to:

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

or

$$g(\boldsymbol{x}, t) = t^{-1/\alpha} f(\boldsymbol{x}) \leq 1$$

Geometric program

an optimization problem is defined as a *geometric program (GP)* if it is structured as:

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

- $f, g_1 \dots, g_m$ being posynomials
- $h_1, \dots, h_p$ as monomials
- its domain is inherently set as $\mathcal{D} = \mathbb{R}^n_{++}$ (implicit constraint $\boldsymbol{x} > 0$).

Example 11.1

consider the optimization problem:

$$\begin{array}{ll} \text{maximize} & x/y \\ \text{subject to} & 2 \leq x \leq 3 \\ & x^2 + 3y/z \leq \sqrt{y} \\ & x/z = z^2, \end{array}$$

where $x, y, z \in \mathbb{R}$ and implicitly $x, y, z > 0$

the problem can be recast into the standard GP form:

$$\begin{array}{ll} \text{minimize} & x^{-1}y \\ \text{subject to} & 2x^{-1} \leq 1, \quad (1/3)x \leq 1 \\ & x^2 y^{-1/2} + 3y^{1/2}z^{-1} \leq 1 \\ & xy^{-1}z^{-2} = 1 \end{array}$$

Geometric program in convex form

geometric programs are generally not convex optimization problems, but they can be recast into convex forms through suitable transformations

Change of variable: by defining $y_i = \log x_i$ such that $x_i = e^{y_i}$, monomial functions of $\boldsymbol{x}$ represented as

$$f(\boldsymbol{x}) = c x_1^{a_1} x_2^{a_2} \cdots x_n^{a_n}$$

can be transformed to

$$f(\boldsymbol{y}) = e^{\boldsymbol{a}^T \boldsymbol{y} + \log c}$$

where $\boldsymbol{a}^T \boldsymbol{y}$ is an affine function of $\boldsymbol{y}$

similarly, for posynomials defined as

$$f(\boldsymbol{x}) = \sum_{k=1}^{K} c_k x_1^{a_{1k}} x_2^{a_{2k}} \cdots x_n^{a_{nk}}$$

the transformation yields

$$f(\boldsymbol{x}) = \sum_{k=1}^{K} e^{\boldsymbol{a}_k^T y + \log c_k}$$

with $\boldsymbol{a}_k = (a_{1k}, \ldots, a_{nk})$

now, consider the standard geometric program, which in terms of $\boldsymbol{y}$ is expressed as:

$$\begin{array}{ll} \text{minimize} & \sum_{k=1}^{K_0} e^{\boldsymbol{a}_{0k}^T \boldsymbol{y} + b_{0k}} \\ \text{subject to} & \sum_{k=1}^{K_i} e^{\boldsymbol{a}_{ik}^T \boldsymbol{y} + b_{ik}} \leq 1, \quad i = 1, \ldots, m \\ & e^{\boldsymbol{h}_i^T \boldsymbol{y} + d_i} = 1, \quad i = 1, \ldots, p \end{array}$$

Equivalent convex form

by applying the logarithm to the objective and constraint functions, the problem morphs into:

$$\begin{array}{ll} \text{minimize} & \bar{f}(\boldsymbol{y}) = \log\left(\sum_{k=1}^{K_0} e^{\boldsymbol{a}_{0k}^T \boldsymbol{y} + b_{0k}}\right) \\ \text{subject to} & \bar{g}_i(\boldsymbol{y}) = \log\left(\sum_{k=1}^{K_i} e^{\boldsymbol{a}_{ik}^T \boldsymbol{y} + b_{ik}}\right) \leq 0, \quad i = 1, \ldots, m \\ & \bar{h}_i(\boldsymbol{y}) = \boldsymbol{h}_i^T \boldsymbol{y} + d_i = 0, \quad i = 1, \ldots, p \end{array}$$

- given that $\bar{f}$ and $\bar{g}_i$ functions are convex, and $\bar{h}_i$ functions are affine, this optimization problem is convex
- we label it as the *geometric program in convex form*; for clarity, the original form is termed the *geometric program in posynomial form*

Example 11.2

consider a cylindrical liquid storage tank characterized by its height, h, and diameter, d

- unlike the main body of the tank, its base is made from a distinct material
- we assume that the height of the base remains unchanged irrespective of the tank's height
- V_{tank} is the volume of the tank
- V_{supp} is the volume supplied within a designated time frame
- the total costs associated with manufacturing and operating the tank over a set duration (e.g., a year) is divided into filling cost and construction cost

Filling costs:

$$C_{\text{fill}}(d,h) = \alpha_1 \frac{V_{\text{supp}}}{V_{\text{tank}}} = c_1 h^{-1} d^{-2},$$

where α_1 is a positive constant (in dollars), and $c_1 = \frac{4\alpha_1 V_{\text{supp}}}{\pi}$

- this cost is tied to supplying a certain volume, V_{supp}, of a liquid (like water) within the designated time-frame
- $V_{\text{supp}}/V_{\text{tank}}$ determines the frequency of tank refilling; hence its cost
- therefore, as the volume of the tank diminishes relative to the supply volume, filling costs rise

Construction costs:

$$C_{\text{constr}}(d,h) = c_2 d^2 + c_3 dh,$$

where $c_2 = \alpha_2 \frac{\pi}{4}$ and $c_3 = \alpha_3 \pi$ (α_2 and α_3 are positive dollar-per-square-meter constants)

- include the expenses of constructing the tank's and its base
- the base's cost is proportional to its area, $\frac{\pi d^2}{4}$, whereas the tank's cost correlates with its surface area, πdh

Total cost

$$C_{\text{total}}(d,h) = C_{\text{fill}}(d,h) + C_{\text{constr}}(d,h) = c_1 h^{-1} d^{-2} + c_2 d^2 + c_3 dh$$

this posynomial objective function is subject to constraints such as upper and lower limits on the diameter and height, represented as:

$$0 < d \le d_{\max}, \quad 0 < h \le h_{\max}$$

GP formulation

$$\begin{array}{ll} \text{minimize} & c_1 h^{-1} d^{-2} + c_2 d^2 + c_3 dh \\ \text{subject to} & 0 < d_{\max}^{-1} d \le 1 \\ & 0 < h_{\max}^{-1} h \le 1 \end{array}$$

with variables d, h

Outline

- quadratic optimization
- semidefinite programs
- geometric programming
- **quasiconvex optimization**

Quasiconvex function

function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is defined as *quasiconvex* if, for every real number γ, its domain and all of its sublevel sets

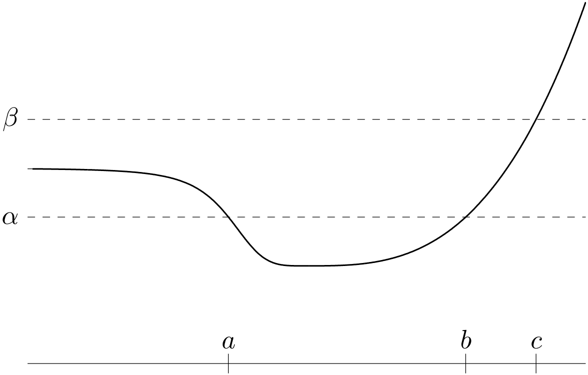
$$\mathcal{S}_\gamma = \{\boldsymbol{x} \mid f(\boldsymbol{x}) \leq \gamma\}$$

are convex

- although every convex function naturally possesses convex level sets, the opposite isn't always true; indeed, there exist non-convex functions that have convex level sets
- a function is termed *quasiconcave* if its negative counterpart $(-f)$ is quasiconvex (for every superlevel set $\{\boldsymbol{x} \mid f(\boldsymbol{x}) \geq \alpha\}$, the set is convex)
- a function that's both quasiconvex and quasiconcave is called *quasilinear*; for these functions, both their domain and each level set $\{\boldsymbol{x} \mid f(\boldsymbol{x}) = \alpha\}$ are convex

Graphical illustration

quasiconvex function that is non-convex



for a specific α, its α-sublevel set, denoted as S_α, is convex

Examples

- $f(x) = \sqrt{|x|}$ is nonconvex, but it is quasiconvex; when $\gamma < 0$, we observe that $\mathcal{S}_\gamma = \emptyset$; for $\gamma \geq 0$, the sublevel set is given by:

$$\mathcal{S}_\gamma = \{x \mid \sqrt{|x|} \leq \gamma\} = \{x \mid |x| \leq \gamma^2\} = [-\gamma^2, \gamma^2]$$

- *Logarithm:* $\log x$ over $\mathbb{R}_{++}$ is both quasiconvex and quasiconcave, making it quasilinear

- *Ceiling function:* $\operatorname{ceil}(x) = \inf\{z \in \mathbb{Z} \mid z \geq x\}$, is quasiconvex and quasiconcave

- *Linear-over-linear:* the function

$$f(\boldsymbol{x}) = \frac{\boldsymbol{a}^T\boldsymbol{x} + b}{\boldsymbol{c}^T\boldsymbol{x} + d},$$

defined over the domain $\operatorname{dom} f = \{\boldsymbol{x} \in \mathbb{R}^n \mid \boldsymbol{c}^T\boldsymbol{x} + d > 0\}$ and $\boldsymbol{c} \neq 0$ is quasiconvex since

$$\mathcal{S}_\gamma = \{\boldsymbol{x} \mid f(\boldsymbol{x}) \leq \gamma\} = \{\boldsymbol{x} \in \mathbb{R}^n \mid (\boldsymbol{a} - \gamma\boldsymbol{c})^T\boldsymbol{x} + (b - \gamma d) \leq 0\}$$

is a convex set

- $f(x_1, x_2) = x_1 x_2$ has Hessian matrix

$$\nabla^2 f(\boldsymbol{x}) = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix},$$

indicating its indefinite nature; yet, f is quasiconcave on $\mathbb{R}^2_+$ due to its convex superlevel sets, but not on $\mathbb{R}^2$

- *Distance ratio function:* given points $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{R}^n$, the function

$$f(\boldsymbol{x}) = \frac{\|\boldsymbol{x} - \boldsymbol{a}\|_2}{\|\boldsymbol{x} - \boldsymbol{b}\|_2}$$

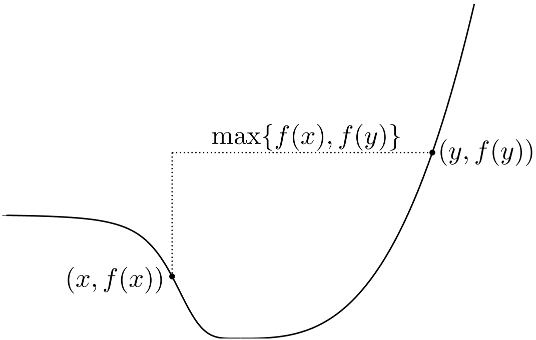
is quasiconvex since its sublevel set represents the halfspace where the distance to $\boldsymbol{a}$ is less than or equal to the distance to $\boldsymbol{b}$

A characterization of quasiconvex function

f is quasiconvex if and only if its domain, $\mathrm{dom}\, f$, is convex and for any $\boldsymbol{x}, \boldsymbol{y} \in \mathrm{dom}\, f$ with $0 \leq \theta \leq 1$,

$$f(\theta \boldsymbol{x} + (1-\theta)\boldsymbol{y}) \leq \max\{f(\boldsymbol{x}), f(\boldsymbol{y})\},$$

signifying that on any line segment, the function's value will not exceed the maximum of its endpoints



Examples

- the cardinality of a vector $\boldsymbol{x} \in \mathbb{R}^n$, represented as $\mathrm{card}(\boldsymbol{x})$, is the count of its non-zero components. It is intriguing to note that the function *card* is quasiconcave on $\mathbb{R}^n_+$ but not on $\mathbb{R}^n$; this stems from the fact:

$$\mathrm{card}(\boldsymbol{x} + \boldsymbol{y}) \geq \min\{\mathrm{card}(\boldsymbol{x}), \mathrm{card}(\boldsymbol{y})\},$$

  valid for non-negative vectors $\boldsymbol{x}, \boldsymbol{y}$

- the rank function, represented as $\mathrm{rank}\, X$, demonstrates quasiconcavity on positive semidefinite matrices; this fact is attributed to the inequality:

$$\mathrm{rank}(X + Y) \geq \min\{\mathrm{rank}\, X, \mathrm{rank}\, Y\},$$

  applicable to positive semidefinite matrices X, Y

Quasiconvex optimization

a quasiconvex optimization problem in standard form is represented as

$$
\begin{array}{ll}
\text{minimize} & f(\boldsymbol{x}) \\
\text{subject to} & g_i(\boldsymbol{x}) \leq \mathbf{0}, \quad i = 1, \ldots, m \\
 & A\boldsymbol{x} = \boldsymbol{b} \quad i = 1, \ldots, p
\end{array}
\tag{11.3}
$$

- the objective f is quasiconvex
- g_i are convex

Quasiconvex optimization via convex feasibility problems

consider a set of convex functions, $\phi_t : \mathbb{R}^n \to \mathbb{R}$ for $t \in \mathbb{R}$, that adhere to:

$$f_0(\boldsymbol{x}) \leq t \iff \phi_t(\boldsymbol{x}) \leq 0,$$

where for every $\boldsymbol{x}$, we have $\phi_s(\boldsymbol{x}) \leq \phi_t(\boldsymbol{x})$ for any $s \geq t$

given $p^\star$ as the optimal solution of the quasiconvex optimization problem (11.3), if the feasibility problem

$$\begin{array}{ll} \text{find} & \boldsymbol{x} \\ \text{subject to} & \phi_t(\boldsymbol{x}) \leq 0 \\ & f_i(\boldsymbol{x}) \leq 0, \quad i = 1, \ldots, m \\ & A\boldsymbol{x} = \boldsymbol{b}, \end{array} \tag{11.4}$$

holds, then $p^\star \leq t$; however, if infeasible, it establishes $p^\star \geq t$; this problem is a convex feasibility problem; hence, the optimal value $p^\star$ being greater or lesser than a value t can be identified by this problem

Bisection for quasiconvex problems

this insight can anchor a bisection-based algorithm listed in to resolve the quasiconvex optimization problem (11.4)

assuming feasibility, one would initiate with an interval $[l, u]$ containing $p^\star$; by evaluating the problem at the midpoint $t = \frac{l+u}{2}$, we deduce whether $p^\star$ lies in the upper or lower interval half and adjust the interval

---

Algorithm BISECTION FOR QUASICONVEX PROBLEMS

---

given: $l \leq p^\star, u \geq p^\star$ and a tolerance $\epsilon > 0$

repeat

1. $t := \frac{l+u}{2}$.
2. evaluate the convex feasibility problem (11.4)
3. **if** (11.4) is feasible, update: if feasible, set $u := t$; **else**, set $l := t$

until $u - l \leq \epsilon$

References and further readings

- Stephen Boyd and Lieven Vandenberghe. *Convex Optimization,* Cambridge University Press, 2004, chapters 2.2.1, 2.2.4, 4.3.
- G. C. Calafiore and L. El Ghaoui. *Optimization Models,* Cambridge University Press, 2014, chapter 9 (9.3, 9.5).