Chapter 7. Constrained Optimization

- Algorithms for constrained optimization
- Frank-Wolfe method
- Alternating direction method (ADM)
- Linearized alternating direction method (LADM)
- Proximal Linearized alternating direction method (PLADM)
- Coordinate descent and block coordinate descent

Algorithms for constrained optimization

- Projections

$$\begin{array}{ll} \min & f(\mathbf{x}) \\ s.t. & \mathbf{x} \in \Omega. \end{array}$$

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \alpha_k \mathbf{d}^{(k)},$$

where $\mathbf{d}^{(k)}$ is typically a function of $\nabla f(\mathbf{x}^{(k)})$, may not work. A simple modification involves the introduction of a projection:

$$\mathbf{x}^{(k+1)} = \mathcal{P}_\Omega(\mathbf{x}^{(k)} + \alpha_k \mathbf{d}^{(k)}).$$

The projected version of the gradient algorithm has the form

$$\mathbf{x}^{(k+1)} = \mathcal{P}_\Omega(\mathbf{x}^{(k)} - \alpha_k \nabla f(\mathbf{x}^{(k)})).$$

We refer to the above as the *projected gradient algorithm.*

Algorithms for constrained optimization

- Projections - Example

$$\min \quad \frac{1}{2}\mathbf{x}^T\mathbf{Q}\mathbf{x}$$

$$s.t. \quad \|\mathbf{x}\|^2 = 1,$$

where $\mathbf{Q} = \mathbf{Q}^T \succ \mathbf{0}$. Suppose that we apply a fixed-step-size projected gradient algorithm to this problem.

Algorithms for constrained optimization

- Projected Gradient Methods with Linear Constraints

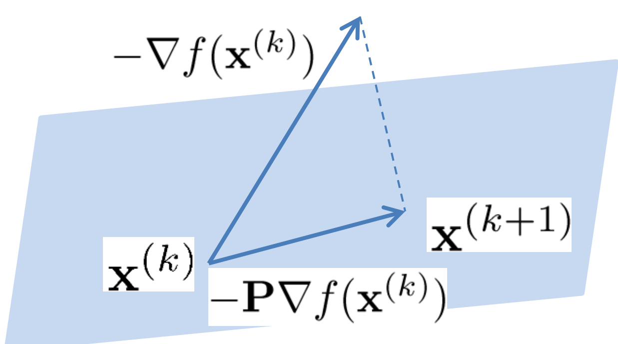
$$\begin{aligned} \min \quad & f(\mathbf{x}) \\ s.t. \quad & \mathbf{A}\mathbf{x} = \mathbf{b}, \end{aligned} \tag{1}$$

where $f : \mathbb{R}^n \to \mathbb{R}$, $\mathbf{A} \in \mathbb{R}^{m \times n}$, $m < n$, $\operatorname{rank} \mathbf{A} = m$, $\mathbf{b} \in \mathbb{R}^m$.

$$\mathbf{P} = \mathbf{I}_n - \mathbf{A}^T(\mathbf{A}\mathbf{A}^T)^{-1}\mathbf{A},$$

$$\mathcal{P}_\Omega(\mathbf{x}) = \mathbf{x} - \mathbf{A}^T(\mathbf{A}\mathbf{A}^T)^{-1}(\mathbf{A}\mathbf{x} - \mathbf{b}) = \mathbf{P}\mathbf{x} + \mathbf{A}^T(\mathbf{A}\mathbf{A}^T)^{-1}\mathbf{b}.$$

$$\begin{aligned} \mathbf{x}^{(k)} \in \Omega \Longrightarrow \mathbf{x}^{(k+1)} =& \mathcal{P}_\Omega(\mathbf{x}^{(k)} - \alpha_k \nabla f(\mathbf{x}^{(k)})) \\ =& [\mathbf{I} - \mathbf{A}^T(\mathbf{A}\mathbf{A}^T)^{-1}\mathbf{A}](\mathbf{x}^{(k)} - \alpha_k \nabla f(\mathbf{x}^{(k)})) \\ & + \mathbf{A}^T(\mathbf{A}\mathbf{A}^T)^{-1}\mathbf{b} \\ =& \mathbf{x}^{(k)} - \alpha_k \mathbf{P} \nabla f(\mathbf{x}^{(k)}). \end{aligned}$$



Algorithms for constrained optimization

- Projected Gradient Methods with Linear Constraints

Proposition 1. $-\mathbf{P}\nabla f(\mathbf{x}^{(k)})$ *points in the direction of maximum rate of decrease of* f *at* $\mathbf{x}^{(k)}$ *along the surface defined by* $\mathbf{A}\mathbf{x} = \mathbf{b}$.

Proposition 2. *If* $-\mathbf{P}\nabla f(\mathbf{x}^{(k)}) \neq \mathbf{0}$ *then it is a descent direction.*

Algorithms for constrained optimization

- Equality constrained convex quadratic minimization

$$\min_{\mathbf{x}} \frac{1}{2}\mathbf{x}^T\mathbf{P}\mathbf{x} + \mathbf{q}^T\mathbf{x} + r,$$
$$s.t.\ \mathbf{A}\mathbf{x} = \mathbf{b},$$

where $\mathbf{P} \in \mathbb{S}^n_+$. The optimality conditions are:

$$\mathbf{A}\mathbf{x}^* = \mathbf{b}, \quad \mathbf{P}\mathbf{x}^* + \mathbf{q} + \mathbf{A}^T\boldsymbol{\nu}^* = \mathbf{0},$$

which can be written as

KKT matrix

$$\boxed{\boxed{\begin{pmatrix} \mathbf{P} & \mathbf{A}^T \\ \mathbf{A} & \mathbf{0} \end{pmatrix}} \begin{pmatrix} \mathbf{x}^* \\ \boldsymbol{\nu}^* \end{pmatrix} = \begin{pmatrix} -\mathbf{q} \\ \mathbf{b} \end{pmatrix}}.$$

KKT system

Algorithms for constrained optimization

- Equality constrained convex quadratic minimization

Nonsingularity of the KKT matrix

Recall our assumption that $\mathbf{P} \in \mathbb{S}_+^n$ and $\operatorname{rank} \mathbf{A} = p < n$. There are several conditions equivalent to nonsingularity of the KKT matrix:

- $\mathcal{N}(\mathbf{P}) \cap \mathcal{N}(\mathbf{A}) = \{\mathbf{0}\}$, i.e., $\mathbf{P}$ and $\mathbf{A}$ have no nontrivial common nullspace.
- $\mathbf{A}\mathbf{x} = \mathbf{0}$, $\mathbf{x} \neq \mathbf{0} \Longrightarrow \mathbf{x}^T \mathbf{P} \mathbf{x} > 0$, i.e., $\mathbf{P}$ is positive definite on the nullspace of $\mathbf{A}$.
- $\mathbf{F}^T \mathbf{P} \mathbf{F} \succ \mathbf{0}$, where $\mathbf{F} \in \mathbb{R}^{n \times (n-p)}$ is a matrix for which $\mathcal{R}(\mathbf{F}) = \mathcal{N}(\mathbf{A})$.

As an important special case, we note that if $\mathbf{P} \succ \mathbf{0}$, the KKT matrix must be nonsingular.

Algorithms for constrained optimization

- Equality constrained convex quadratic minimization

Solving KKT systems

$$\begin{pmatrix} \mathbf{H} & \mathbf{A}^T \\ \mathbf{A} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{v} \\ \mathbf{w} \end{pmatrix} = - \begin{pmatrix} \mathbf{g} \\ \mathbf{h} \end{pmatrix}. \tag{2}$$

Here we assume $\mathbf{H} \in \mathbb{S}_+^n$ and $\mathbf{A} \in \mathbb{R}^{p \times n}$ with $\operatorname{rank} \mathbf{A} = p < n$.

Solving KKT systems

One straightforward approach is to simply solve the KKT system (1), which is a set of $n+p$ linear equations in $n+p$ variables. The KKT matrix is symmetric, but may not be positive definite, so a good way to do this is to use an LDL^T factorization. If no structure of the matrix is exploited, the cost is $(1/3)(n+p)^3$ flops. This can be a reasonable approach when the problem is small (i.e., n and p are not too large), or when $\mathbf{A}$ and $\mathbf{H}$ are sparse.

Algorithms for constrained optimization

- Equality constrained convex quadratic minimization

Solving KKT system via elimination

We start by describing the simplest case, in which $\mathbf{H} \succ \mathbf{0}$. Starting from the first of the KKT equations

$$\mathbf{H}\mathbf{v} + \mathbf{A}^T\mathbf{w} = -\mathbf{g}, \quad \mathbf{A}\mathbf{v} = -\mathbf{h},$$

we solve for $\mathbf{v}$ to obtain

$$\mathbf{v} = -\mathbf{H}^{-1}(\mathbf{g} + \mathbf{A}^T\mathbf{w}).$$

Substituting this into the second KKT equation yields $\mathbf{A}\mathbf{H}^{-1}(\mathbf{g} + \mathbf{A}^T\mathbf{w}) = \mathbf{h}$, so we have

$$\mathbf{w} = (\mathbf{A}\mathbf{H}^{-1}\mathbf{A}^T)^{-1}(\mathbf{h} - \mathbf{A}\mathbf{H}^{-1}\mathbf{g}).$$

These formulae give us a method for computing $\mathbf{v}$ and $\mathbf{w}$.

Algorithms for constrained optimization

- Equality constrained convex quadratic minimization

Solving KKT system by block elimination.
Given: KKT system with $\mathbf{H} \succ \mathbf{0}$.

1. Form $\mathbf{H}^{-1}\mathbf{A}^T$ and $\mathbf{H}^{-1}\mathbf{g}$.
2. Form Schur complement $\mathbf{S} = -\mathbf{A}\mathbf{H}^{-1}\mathbf{A}^T$.
3. Determine $\mathbf{w}$ by solving $\mathbf{S}\mathbf{w} = \mathbf{A}\mathbf{H}^{-1}\mathbf{g} - \mathbf{h}$.
4. Determine $\mathbf{v}$ by solving $\mathbf{H}\mathbf{v} = -\mathbf{A}^T\mathbf{w} - \mathbf{g}$.

Algorithms for constrained optimization

- Equality constrained convex quadratic minimization

Step 1 can be done by a Cholesky factorization of $\mathbf{H}$, followed by $p+1$ solves, which costs $f + (p + 1)s$, where f is the cost of factoring $\mathbf{H}$ and s is the cost of an associated solve. Step 2 requires a $p \times n$ by $n \times p$ matrix multiplication. If we exploit no structure in this calculation, the cost is p^2n flops. (Since the result is symmetric, we only need to compute the upper triangular part of $\mathbf{S}$.) In some cases special structure in $\mathbf{A}$ and $\mathbf{H}$ can be exploited to carry out step 2 more efficiently. Step 3 can be carried out by Cholesky factorization of $-\mathbf{S}$, which costs $(1/3)p^3$ flops if no further structure of $\mathbf{S}$ is exploited. Step 4 can be carried out using the factorization of $\mathbf{H}$ already calculated in step 1, so the cost is $2np + s$ flops. The total flop count, assuming that no structure is exploited in forming or factoring the Schur complement, is

$$f + ps + p^2n + (1/3)p^3$$

flops (keeping only dominant terms). If we exploit structure in forming or factoring $\mathbf{S}$, the last two terms are even smaller.

Algorithms for constrained optimization

- Equality constrained convex quadratic minimization

If $\mathbf{H}$ can be factored efficiently, then block elimination gives us a flop count advantage over directly solving the KKT system using an LDL^T factorization. For example, if $\mathbf{H}$ is diagonal (which corresponds to a separable objective function), we have $f = 0$ and $s = n$, so the total cost is $p^2n + (1/3)p^3$ flops, which grows only linearly with n. If $\mathbf{H}$ is banded with bandwidth $k \ll n$, then $f = nk^2$, $s = 4nk$, so the total cost is around $nk^2 + 4nkp + p^2n + (1/3)p^3$ which still grows only linearly with n. Other structures of $\mathbf{H}$ that can be exploited are block diagonal (which corresponds to block separable objective function), sparse, or diagonal plus low rank.

Examples: Equality constrained analytic center, Minimum length piecewise-linear curve subject to equality constraints, Locally linear embedding (LLE)

Algorithms for constrained optimization

- Equality constrained convex quadratic minimization

Elimination with singular H

The block elimination method described above obviously does not work when $\mathbf{H}$ is singular, but a simple variation on the method can be used in this more general case. The more general method is based on the following result: The KKT matrix is nonsingular iff $\mathbf{H} + \mathbf{A}^T\mathbf{Q}\mathbf{A} \succ \mathbf{0}$ for some $\mathbf{Q} \succeq \mathbf{0}$, in which case, $\mathbf{H} + \mathbf{A}^T\mathbf{Q}\mathbf{A} \succ \mathbf{0}$ for all $\mathbf{Q} \succ \mathbf{0}$. We conclude, for example, that if the KKT matrix is nonsingular, then $\mathbf{H} + \mathbf{A}^T\mathbf{A} \succ \mathbf{0}$.

Let $\mathbf{Q} \succeq \mathbf{0}$ be a matrix for which $\mathbf{H} + \mathbf{A}^T\mathbf{Q}\mathbf{A} \succ \mathbf{0}$. Then the KKT system (2) is equivalent to

$$\begin{pmatrix} \mathbf{H} + \mathbf{A}^T\mathbf{Q}\mathbf{A} & \mathbf{A}^T \\ \mathbf{A} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{v} \\ \mathbf{w} \end{pmatrix} = -\begin{pmatrix} \mathbf{g} + \mathbf{A}^T\mathbf{Q}\mathbf{h} \\ \mathbf{h} \end{pmatrix}, \tag{3}$$

which can be solved using elimination since $\mathbf{H} + \mathbf{A}^T\mathbf{Q}\mathbf{A} \succ \mathbf{0}$.

Examples: Equality constrained analytic centering, Optimal network flow, Optimal network flow, Analytic center of a linear matrix inequality

Algorithms for constrained optimization

- Eliminating equality constraints

One general approach to solving the equality constrained problem is to eliminte the equality constraints and then solve the resulting unconstrained problem using methods for unconstrained minimization. We first find a matrix $\mathbf{F} \in \mathbb{R}^{n\times(n-p)}$ and vector $\hat{\mathbf{x}} \in \mathbb{R}^n$ that parameterize the (affine) feasible set:

$$\{\mathbf{x}|\mathbf{A}\mathbf{x} = \mathbf{b}\} = \{\mathbf{F}\mathbf{z} + \hat{\mathbf{x}}|\mathbf{z} \in \mathbb{R}^{n-p}\}.$$

Here $\hat{\mathbf{x}}$ can be chosen as any particular solution of $\mathbf{A}\mathbf{x} = \mathbf{b}$ and $\mathbf{F} \in \mathbb{R}^{n\times(n-p)}$ is any matrix whose range is the nullspace of $\mathbf{A}$. We then form the reduced or eliminated optimization problem:

$$\min_{\mathbf{z}} \ \tilde{f}(\mathbf{z}) \triangleq f(\mathbf{F}\mathbf{z} + \hat{\mathbf{x}}), \tag{4}$$

which is an unconstrained problem with variable $\mathbf{z} \in \mathbb{R}^{n-p}$. From its solution $\mathbf{z}^*$, we can find the solution of the equality constrained problem as $\mathbf{x}^* = \mathbf{F}\mathbf{z}^* + \hat{\mathbf{x}}$.

Algorithms for constrained optimization

- Eliminating equality constraints

Example: Optimal allocation with resource constraint. We consider the problem:

$$\min_{\mathbf{x}} \sum_{i=1}^{n} f_i(x_i),$$
$$s.t. \sum_{i=1}^{n} x_i = b,$$

where the functions $f_i : \mathbb{R} \to \mathbb{R}$ are convex and twice differentiable, and $b \in \mathbb{R}$ is a problem parameter. We interpret this as the problem of optimally allocating a single resource, with a fixed total amount b (the budget) to n otherwise independent activities.

Algorithms for constrained optimization

- Solving equality constrained problems via the dual

Another approach to solving (1) is to solve the dual, and then recover the optimal primal variable $\mathbf{x}^*$. The dual function of (1) is:

$$\begin{aligned} g(\boldsymbol{\nu}) &= -\mathbf{b}^T\boldsymbol{\nu} + \inf_{\mathbf{x}}(f(\mathbf{x}) + \boldsymbol{\nu}^T\mathbf{A}\mathbf{x}) \\ &= -\mathbf{b}^T\boldsymbol{\nu} - \sup_{\mathbf{x}}((-\mathbf{A}^T\boldsymbol{\nu})^T\mathbf{x} - f(\mathbf{x})) = -\mathbf{b}^T\boldsymbol{\nu} - f^*(-\mathbf{A}^T\boldsymbol{\nu}), \end{aligned}$$

where f^* is the conjugate of f. So the dual problem is:

$$\max_{\boldsymbol{\nu}} -\mathbf{b}^T\boldsymbol{\nu} - f^*(-\mathbf{A}^T\boldsymbol{\nu}).$$

Since by assumption there is an optimal point, the problem is strictly feasible, so Slater's condition holds. Therefore strong duality holds, and the dual optimum is attained, i.e., there exists a $\boldsymbol{\nu}^*$ with $g(\boldsymbol{\nu}^*) = p^*$.

Once we find an optimal dual variable $\boldsymbol{\nu}^*$, we reconstruct an optimal primal solution $\mathbf{x}^*$ from it.

Algorithms for constrained optimization

- Solving equality constrained problems via the dual

Example: Equality constrained analytic center. We consider the problem

$$\min_{\mathbf{x}} \ f(\mathbf{x}) = -\sum_{i=1}^{n} \log x_i, \quad s.t. \ \mathbf{A}\mathbf{x} = \mathbf{b}, \tag{4}$$

where $\mathbf{A} \in \mathbb{R}^{p \times n}$, with implicit constraint $\mathbf{x} > \mathbf{0}$. Using

$$f^*(\mathbf{y}) = \sum_{i=1}^{n}(-1 - \log(-y_i)) = -n - \sum_{i=1}^{n} \log(-y_i)$$

(with $\mathrm{dom} f^* = -\mathbb{R}^n_{++}$), the dual problem is

$$\max_{\boldsymbol{\nu}} g(\boldsymbol{\nu}) = -\mathbf{b}^T \boldsymbol{\nu} + n + \sum_{i=1}^{n} \log(\mathbf{A}^T \boldsymbol{\nu})_i, \tag{5}$$

with implicit constraint $\mathbf{A}^T \boldsymbol{\nu} > \mathbf{0}$.

Algorithms for constrained optimization

- Solving equality constrained problems via the dual

Here we can easily solve the dual feasibility equation, i.e., find the $\mathbf{x}$ that minimizes $L(\mathbf{x}, \boldsymbol{\nu})$:

$$\nabla f(\mathbf{x}) + \mathbf{A}^T \boldsymbol{\nu} = -(1/x_1, \cdots, 1/x_n) + \mathbf{A}^T \boldsymbol{\nu} = \mathbf{0},$$

and so

$$x_i(\boldsymbol{\nu}) = 1/(\mathbf{A}^T \boldsymbol{\nu})_i. \tag{6}$$

To solve the equality constrained analytic centering problem (4), we solve the (unconstrained) dual problem (5), and then recover the optimal solution of (4) via (6).

Algorithms for constrained optimization

- Dual Derivatives and Subgradients – Model Problem

We focus on the primal problem:

$$\min_{\mathbf{x}} f(\mathbf{x}) \tag{10}$$
$$s.t.\ \mathbf{x} \in \mathcal{X}, g_j(\mathbf{x}) \leq 0, j = 1, \cdots, r,$$

and its dual

$$\max_{\boldsymbol{\mu}} q(\boldsymbol{\mu}) \tag{11}$$
$$s.t.\ \boldsymbol{\mu} \geq \mathbf{0},$$

where $f : \mathbb{R}^n \to \mathbb{R}$ and $g_j : \mathbb{R}^n \to \mathbb{R}$ are given functions, $\mathcal{X}$ is a subset of $\mathbb{R}^n$, and

$$q(\boldsymbol{\mu}) = \inf_{\mathbf{x}\in\mathcal{X}} L(\mathbf{x}, \boldsymbol{\mu}) = \inf_{\mathbf{x}\in\mathcal{X}} \left\{ f(\mathbf{x}) + \boldsymbol{\mu}^T \mathbf{g}(\mathbf{x}) \right\}$$

is the dual function.

Algorithms for constrained optimization

- Dual Derivatives and Subgradients - Pros

It is worth reflecting on the potential incentives for solving the dual problem in place of the primal. These are:

a) The dual is a concave problem (concave cost, convex constraint set). By contrast, the primal need not be convex.

b) The dual may have smaller dimension and/or simpler constraints than the primal.

Algorithms for constrained optimization

- Dual Derivatives and Subgradients - Pros

c) If their is no duality gap and the dual is solved exactly to yield a Lagrange multiplier $\boldsymbol{\mu}^*$, all optimal primal solutions can be obtained by minimizing the Lagrangian $L(\mathbf{x}, \boldsymbol{\mu}^*)$ over $\mathbf{x} \in \mathcal{X}$ (however, there may be additional minimizers of $L(\mathbf{x}, \boldsymbol{\mu}^*)$ that the primal-infeasible). Furthermore, if the dual is solved approximately to yield an approximate Lagrange multiplier $\boldsymbol{\mu}$, and $\mathbf{x}_\mu$ minimizes $L(\mathbf{x}, \boldsymbol{\mu})$ over $\mathbf{x} \in \mathcal{X}$, then $\mathbf{x}_\mu$ also solves

$$\min_{\mathbf{x}} \; f(\mathbf{x}) \tag{10}$$
$$s.t. \; \mathbf{x} \in \mathcal{X}, g_j(\mathbf{x}) \leq g_j(\mathbf{x}_\mu), j = 1, \cdots, r.$$

Thus if the constraint violations $g_j(\mathbf{x}_\mu)$ are not much larger than zero, $\mathbf{x}_\mu$ may be an acceptable practical solution.

d) Even if there is a duality gap, for every $\boldsymbol{\mu} \geq \mathbf{0}$, the dual value $q(\boldsymbol{\mu})$ is a lower bound to the optimal primal value. This lower bound may be useful in the context of discrete optimization and branch and bound procedures.

Algorithms for constrained optimization

- Dual Derivatives and Subgradients - Cons

We should also consider some of the difficulties in solving the dual problem. The most important ones are the following:

a) To evaluate the dual function at any $\boldsymbol{\mu}$ requires minimization of the Lagrangian $L(\mathbf{x}, \boldsymbol{\mu})$ over $\mathbf{x} \in \mathcal{X}$. In effect, this restricts the utility of dual methods to problems where this minimization can either be done in closed form or else is relatively simple; for example, when there is special structure that allows decomposition, as in the separable problems and the monotropic programming problems.

b) In many types of problems, the dual function is nondifferentialbe, in which algorithms for smooth objective functions do not apply.

c) Even if we find an optimal dual solution $\boldsymbol{\mu}^*$, it may be difficult to obtain a primal feasible vector $\mathbf{x}$ from the minimization of $L(\mathbf{x}, \boldsymbol{\mu}^*)$ over $\mathbf{x} \in \mathcal{X}$ as required by the primal-dual optimality conditions, since this minimization can also yield primal-infeasible vectors.

Algorithms for constrained optimization

- Dual Derivatives and Subgradients

For a given $\boldsymbol{\mu} \in \mathbb{R}^r$, suppose that $\mathbf{x}_\mu$ minimizes the Lagrangian $L(\mathbf{x}, \boldsymbol{\mu})$,

$$\mathbf{x}_\mu = \operatorname*{argmin}_{\mathbf{x} \in \mathcal{X}} L(\mathbf{x}, \boldsymbol{\mu}) = \operatorname*{argmin}_{\mathbf{x} \in \mathcal{X}} \left\{ f(\mathbf{x}) + \boldsymbol{\mu}^T \mathbf{g}(\mathbf{x}) \right\}.$$

An important fact is that $\mathbf{g}(\mathbf{x}_\mu)$ is a subgradient of the dual function q at $\boldsymbol{\mu}$: obtained *essentially at no cost*

$$q(\bar{\boldsymbol{\mu}}) \leq q(\boldsymbol{\mu}) + (\bar{\boldsymbol{\mu}} - \boldsymbol{\mu})^T \mathbf{g}(\mathbf{x}_\mu), \quad \forall \bar{\boldsymbol{\mu}} \in \mathbb{R}^r. \tag{10}$$

To see this, we use the definition of q and $\mathbf{x}_\mu$ to write for all $\bar{\boldsymbol{\mu}} \in \mathbb{R}^r$,

$$\begin{aligned} q(\bar{\boldsymbol{\mu}}) = \inf_{\mathbf{x} \in \mathcal{X}} \left\{ f(\mathbf{x}) + \bar{\boldsymbol{\mu}}^T \mathbf{g}(\mathbf{x}) \right\} &\leq f(\mathbf{x}_\mu) + \bar{\boldsymbol{\mu}}^T \mathbf{g}(\mathbf{x}_\mu) \\ &= f(\mathbf{x}_\mu) + \boldsymbol{\mu}^T \mathbf{g}(\mathbf{x}_\mu) + (\bar{\boldsymbol{\mu}} - \boldsymbol{\mu})^T \mathbf{g}(\mathbf{x}_\mu) \\ &= q(\boldsymbol{\mu}) + (\bar{\boldsymbol{\mu}} - \boldsymbol{\mu})^T \mathbf{g}(\mathbf{x}_\mu). \end{aligned}$$

Note that this calculation is valid for all $\boldsymbol{\mu} \in \mathbb{R}^r$ for which there is a minimizing vector $\mathbf{x}_\mu$, regardless of whether $\boldsymbol{\mu} \geq \mathbf{0}$.

Algorithms for constrained optimization

- Dual Derivatives and Subgradients

Proposition 1. *Let $\mathcal{X}$ be a compact set, and let f and $\mathbf{g}$ be continuous over $\mathcal{X}$. Assume also that for every $\boldsymbol{\mu} \in \mathbb{R}^r$, $L(\mathbf{x}, \boldsymbol{\mu})$ is minimized over $\mathbf{x} \in \mathcal{X}$ at a unique point $\mathbf{x}_\mu$. Then q is everywhere continuously differentiable and*

$$\nabla q(\boldsymbol{\mu}) = \mathbf{g}(\mathbf{x}_\mu), \quad \forall \boldsymbol{\mu} \in \mathbb{R}^r.$$