Generating Eigenvalue Bounds Using Optimization

Henry Wolkowicz

Dedicated to the memory of Professor George Isac

Abstract This paper illustrates how optimization can be used to derive known and new theoretical results about perturbations of matrices and sensitivity of eigenvalues. More specifically, the Karush-Kuhn-Tucker conditions, the shadow prices, and the parametric solution of a fractional program are used to derive explicit formulae for bounds for functions of matrix eigenvalues.

1 Introduction

Many classical and new inequalities can be derived using optimization techniques. One first formulates the desired inequality as the maximum (minimum) of a function subject to appropriate constraints. The inequality, along with conditions for equality to hold, can then be derived and proved, provided that the optimization problem can be explicitly solved.

For example, consider the *Rayleigh principle*

$$\lambda_{\max} = \max\{\langle x, Ax\rangle : \ x \in \mathbb{R}^n, \|x\| = 1\}, \tag{1}$$

where A is an $n \times n$ Hermitian matrix, $\lambda_{\max}$ is the largest eigenvalue of A, $\langle \cdot, \cdot \rangle$ is the Euclidean inner product, and $\| \cdot \|$ is the associated norm. Typically, this principle is proved by maximizing the quadratic function $\langle x, Ax\rangle$ subject to the equality constraint, $\|x\|^2 = 1$. An explicit solution can be found using the classical and well known, Euler-Lagrange multiplier rule of calculus. (See Example 1 below). It is an

Henry Wolkowicz
Department of Combinatorics and Optimization, University of Waterloo, Waterloo, Ontario N2L 3G1, Canada (Research Report CORR 2009-02)

interesting coincidence that λ is the standard symbol used in the literature for both eigenvalues and Lagrange multipliers; and the eigenvalue and Lagrange multiplier coincide in the above derivation. Not so well known are the multiplier rules for inequality constrained programs. The *Holder inequality*

$$\langle x,y\rangle = \sum_{i=1}^{n} x_i y_i \le \left(\sum_{i=1}^{n} x_i^p\right)^{1/p} \left(\sum_{i=1}^{n} y_i^q\right)^{1/q},$$

where $x,y \in \mathbb{R}^n_+, p > 1, q = p/(p-1)$, can be proved by solving the optimization problem $h(x) := \max_y\{\sum_i x_i y_i : \sum_i y_i^q - 1 \le 0, y_i \ge 0, \forall i\}$. The John multiplier rule yields the explicit solution. (See [8] and Example 3 below). The classical *arithmetic-geometric mean inequality* $(\alpha_1 \ldots \alpha_n)^{1/n} \le \frac{1}{n}(\alpha_1 + \ldots + \alpha_n)$, where $\alpha_i > 0, i = 1,\ldots,n$, can be derived by solving the geometric programming problem

$$\max\{\Pi_i \alpha_1 : \sum_{i=1}^{n} \alpha_i = 1, \alpha_i \ge 0, \forall i\}.$$

Convexity properties of the functions, which arise when reformulating the inequalities as programming problems, can prove very helpful. For example, convexity can guarantee that sufficiency, rather than only necessity, holds in optimality conditions. The *quasi-convexity* of the function

$$\phi(f) = \int f d\nu \int (1/f) d\mu, \tag{2}$$

where μ and ν are two nontrivial positive measures on a measurable space X, can be used to derive the Kantorovich inequality, [1]. (We prove the Kantorovich inequality using optimization in Example 2.) We rely heavily on the convexity and pseudo-convexity of the functions.

Optimality conditions, such as the Lagrange and Karush-Kuhn-Tucker multiplier rules, are needed to numerically solve mathematical programming problems. The purpose of this paper is to show how to use optimization techniques to generate known, as well as new, explicit eigenvalue inequalities. Rather than include all possible results, we concentrate on just a few, which allow us to illustrate several useful techniques. For example, suppose that A is an $n \times n$ complex matrix with real eigenvalues $\lambda_1 \ge \ldots \ge \lambda_n$. A lower bound for λ_k can be found if we can explicitly solve the problem

$$\begin{array}{lll} \min & \lambda_k & \\ \text{subject to} & \sum_{i=1}^{n} \lambda_i = \operatorname{trace} A & \\ & \sum_{i=1}^{n} \lambda_i^2 \le \operatorname{trace} A^2 & \\ & \lambda_k - \lambda_i \le 0, & i = 1,\ldots,k-1 \\ & \lambda_i - \lambda_k \le 0, & i = k+1,\ldots,n. \end{array} \tag{3}$$

We can use the *Karush-Kuhn-Tucker* necessary conditions for optimality to find the explicit solution. (See Theorem 6.) Sufficiency guarantees that we actually have the solution. This yields the best lower bound for λ_k based on the known data. (Further results along these lines can be found in [4].)

In addition, the Lagrange multipliers, obtained when solving the program (3), provide *shadow prices*. These shadow prices are sensitivity coefficients with respect to perturbations in the right-hand sides of the constraints. We use these shadow prices to improve the lower bound in the case that we have additional information about the eigenvalues. (See e.g. Corollaries 2 and 3.)

1.1 Outline

In Section 2 we introduce the optimality conditions and use them to prove the well known: (i) Rayleigh Principle; (ii) Holder inequality; and (iii) Kantorovich inequality. In Section 3 we show how to use the convex multiplier rule (or the Karush-Kuhn-Tucker conditions) to generate bounds for functions of the eigenvalues of an $n \times n$ matrix A with real eigenvalues. Some of these results have appeared in [7, 12, 4]. Included are bounds for $\lambda_k, \lambda_k + \lambda_\ell$ and $\lambda_k - \lambda_\ell$. We also show how to use the Lagrange multipliers (shadow prices) to strengthen the bounds. Section 4 uses fractional programming techniques to generate bounds for the ratios $(\lambda_k - \lambda_\ell)/(\lambda_k + \lambda_\ell)$. Some of the inequalities obtained here are given in [7, 12] but with proofs using elementary calculus techniques rather than optimization.

2 Optimality Conditions

2.1 Equality Constraints

First, consider the program

$$\min\{f(x) : h_k(x) = 0, k = 1, \ldots, q,\ x \in U\}, \tag{4}$$

where U is an open subset of $\mathbb{R}^n$ and the functions f, h_k, $k = 1, \ldots, q$, are continuously differentiable. The function f is called the *objective function* of the program. The *feasible set*, denoted by $\mathscr{F}$, is the set of points in $\mathbb{R}^n$ which satisfy the constraints. Then, the classical Euler-Lagrange multiplier rule states, e.g. [8],

Theorem 1. *Suppose that* $a \in \mathbb{R}^n$ *solves* (4) *and that the gradients* $\bigtriangledown h_1(a), \ldots, \bigtriangledown h_q(a)$ *are linearly independent. Then,*

$$\bigtriangledown f(a) + \sum_{k=1}^{q} \lambda_k \bigtriangledown h_k(a) = 0, \tag{5}$$

for some (Lagrange multipliers) $\lambda_k \in \mathbb{R}, k = 1, \ldots, q$. ■

Example 1. Suppose that A is an $n \times n$ Hermitian matrix with eigenvalues $\lambda_1 \geq \ldots \geq \lambda_n$. To prove the Rayleigh Principle (1), consider the equivalent program

$$\text{minimize} \left\{ -\langle x, Ax \rangle : 1 - \sum_{i=1}^{n} x_i^2 = 0, x \in \mathbb{R}^n \right\}. \tag{6}$$

Since the objective function is continuous while the feasible set is compact, the minimum is attained at some $a \in \mathscr{F} \subset \mathbb{R}^n$. If we apply Theorem 1, we see that there exists a Lagrange multiplier $\lambda \in \mathbb{R}$ such that

$$2Aa - 2\lambda a = 0,$$

i.e. a is an eigenvector corresponding to the eigenvalue equal to the Lagrange multiplier λ. Since the objective function

$$\langle a, Aa \rangle = \lambda \langle a, a \rangle = \lambda,$$

we conclude that λ must be the largest eigenvalue and we get the desired result. If we now add the constraint that x be restricted to the $n-1$ dimensional subspace orthogonal to a, then we recover the second largest eigenvalue. Continuing in this manner, we get all the eigenvalues. More precisely, if $a_1, a_2, \ldots, a_k$ are k mutually orthonormal eigenvectors corresponding to the k largest eigenvalues of A, $\lambda_1 \geq \ldots \geq \lambda_k$, then we solve (6) with the added constraints

$$\langle x, a_i \rangle = 0, \;\; i = 1, \ldots, k.$$

The gradients of the constraints are necessarily linearly independent since the vectors x, and a_i, $\;i = 1, \ldots, k$, are (mutually) orthonormal. Now if a is a solution, then (5) yields

$$2Aa - 2\lambda a + \sum_{i=1}^{k} \alpha_i a_i = 0,$$

for some Lagrange multipliers $\lambda, \alpha_i, i = 1, \ldots, k$. However, taking the inner product with fixed a_i, and using the fact that

$$\langle Aa, a_i \rangle = \langle a, Aa_i \rangle = \lambda_i \langle a, a_i \rangle = 0,$$

we see that $\alpha_i = 0$, $\;i = 1, \ldots, k$, and so $Aa = \lambda a$, i.e. a is the eigenvector corresponding to the $(k+1)-st$ largest eigenvalue. This argument also shows that A necessarily has n (real) mutually orthonormal eigenvectors. ■

Example 2. Consider the Kantorovich inequality, e.g. [1, 3],

$$1 \le \langle x, Ax\rangle \langle x, A^{-1}x\rangle \le \frac{1}{4}\left(\sqrt{\frac{\lambda_1}{\lambda_n}}+\sqrt{\frac{\lambda_n}{\lambda_1}}\right)^2, \tag{7}$$

where A is an $n \times n$ positive definite Hermitian matrix with eigenvalues $\lambda_1 \ge \ldots \ge \lambda_n > 0, x \in \mathbb{R}^n$, and $\|x\| = 1$. This inequality is useful in obtaining bounds for the rate of convergence of the method of steepest descent, e.g. [5]. To prove the inequality we consider the following (two) optimization problems

$$\begin{array}{ll} \min(\max) & f_1(a) := \left(\sum_{i=1}^n a_i^2 \lambda_i\right)\left(\sum_{i=1}^n a_i^2 \lambda_i^{-1}\right) \\ \text{subject to} & g(a) := 1 - \sum_{i=1}^n a_i^2 = 0, \end{array} \tag{8}$$

where $a = (a_i) \in \mathbb{R}^n, a_i = \langle x, u_i\rangle$ and $u_i, i = 1, \ldots, n$, is an orthonormal set of eigenvectors of A corresponding to the eigenvalues $\lambda_i,\ \ i = 1, \ldots, n$, respectively. Thus, $f_1(a)$ is the middle expression in (7). Suppose that the vector $a = (a_i)$ solves (8). Then, the necessary conditions of optimality state that (μ is the Lagrange multiplier)

$$a_i\lambda_i\left(\sum_j a_j^2\lambda_j^{-1}\right) + a_i\lambda_i^{-1}\left(\sum_j a_j^2\lambda_j\right) - \mu a_i = 0, i = 1, \ldots, n; \quad \sum_i a_i^2 = 1. \tag{9}$$

Thus,

$$f_2(a) := \lambda_i\left(\sum_j a_j^2\lambda_j^{-1}\right) + \lambda_i^{-1}\left(\sum_j a_j^2\lambda_j\right) = \mu, \text{ if } a_i \neq 0. \tag{10}$$

On the other hand, if we multiply (9) by a_i and sum over i, we get

$$\mu = 2\left(\sum_j a_j^2\lambda_j\right)\left(\sum_j a_j^2\lambda_j^{-1}\right) = 2f_1(a). \tag{11}$$

By (10) and (11), we can replace $f_1(a)$ in (8) by the middle expression in (10), i.e. by $f_2(a)$. The new necessary conditions for optimality (with μ playing the role of the Lagrange multiplier again and $a_i \neq 0$) are

$$a_j\frac{\lambda_i}{\lambda_j} + a_j\frac{\lambda_j}{\lambda_i} - a_j\mu = 0, \quad j = 1, \ldots, n.$$

Now, if both $a_j \neq 0, a_i \neq 0$, we get

$$f_3(a) := \frac{\lambda_i}{\lambda_j} + \frac{\lambda_j}{\lambda_i} = \mu. \tag{12}$$

And, multiplying (12) by a_j and summing over j yields

$$\mu = \sum_j a_j^2\left(\frac{\lambda_i}{\lambda_j} + \frac{\lambda_j}{\lambda_i}\right) = f_2(a).$$

Thus, we can now replace $f_2(a)$ (and so $f_1(a)$) in (8) by $f_3(a)$. Note that a does not appear explicitly in $f_3(a)$. However, the i and j must correspond to $a_i \neq 0$ and $a_j \neq 0$. Consider the function

$$h(x,y) = \frac{x}{y} + \frac{y}{x}, \tag{13}$$

where $0 < \alpha \leq x \leq y \leq \beta$. Since

$$\frac{d}{dx}h(x,y) = \frac{y(x^2 - y^2)}{(xy)^2} \leq 0 \quad (< 0 \text{ if } x \neq y),$$

and, similarly $\frac{d}{dy}h(x,y) \geq 0$ (> 0 if $x \neq y$)), we see that h attains its maximum at $x = \alpha$ and $y = \beta$, and it attains its minimum at $x = y$. This shows that $2 \leq f_3(a)$ and that f_3 attains its maximum at $\frac{\lambda_1}{\lambda_n} + \frac{\lambda_n}{\lambda_1}$, i.e. at $a_1 \neq 0$ and $a_n \neq 0$. The left-hand side of (7) now follows from $2f_1(a) = f_2(a) = f_3(a)$. Now, to have $f_3(a) = f_2(a)$, we must choose $a_1 = a_n = \frac{1}{2}$, and $a_i = 0, \forall 1 < i < n$. Substituting this choice of a in $f_1(a)$ yields the right-hand side of (7). ■

2.2 Equality and Inequality Constraints

Now suppose that program (4) has, in addition, the inequality constraints (continuously differentiable)

$$g_i(x) \leq 0, \;\; i = 1, \ldots, m. \tag{14}$$

Then, we obtain the John necessary conditions of optimality. (See e.g. [8].)

Theorem 2. *Suppose that $a \in \mathbb{R}^n$ solves (4) with the additional constraints* (14). *Then, there exist Lagrange multiplier vectors $\lambda \in \mathbb{R}^{m+1}_+, \alpha \in \mathbb{R}^q$, not both zero, such that*

$$\begin{array}{c} \lambda_0 \bigtriangledown f(a) + \sum_{i=1}^m \lambda_i \bigtriangledown g_i(a) + \sum_{j=1}^q \alpha_j \bigtriangledown h_j(a) = 0, \\ \lambda_i g_i(a) = 0, \;\; i = 1, \ldots, m. \end{array} \tag{15}$$

■

The first condition in (15) is *dual feasibility*. The second Condition in (15) is called *complementary slackness*. It shows that either the multiplier $\lambda_i = 0$ or the constraint is *binding*, i.e. $g_i(a) = 0$. The Karush-Kuhn-Tucker conditions (e.g. [8]) assume a *constraint qualification* and have $\lambda_0 = 1$.

Example 3. Holder's inequality states that if $x, y \in \mathbb{R}^n_{++}$, are (positive) vectors, $p > 1$, and $q = p/(p-1)$, then

$$\langle x,y\rangle = \sum_{i=1}^{n} x_i y_i \le \left(\sum_i x_i^p\right)^{1/p} \left(\sum_i y_i^q\right)^{1/q} = \|x\|_p \|y\|_q.$$

We now include a proof of this inequality using the John Multiplier Rule. (This proof corrects the one given in [8].)

Fix $y = (y_i) \in \mathbb{R}^n_{++}$ and consider the program

$$\begin{array}{rll} \min & f(x) := -\sum_{i=1}^n x_i y_i & \\ \text{subject to} & g(x) := \sum_{i=1}^n x_i^p - 1 & \le 0 \\ & h_i(x) := -x_i & \le 0,\ i = 1,\ldots,n. \end{array}$$

Holder's inequality follows if the optimal value is $-\|y\|_q$. Since the feasible set is compact, the minimum is attained at say $a = (a_i) \in \mathbb{R}^n_+$. Then, there exist constants (Lagrange multipliers) $\lambda_0 \ge 0, \lambda_1 \ge 0, \gamma_i \ge 0$, not all zero, such that

$$\begin{array}{l} -\lambda_0 y_i + \lambda_1 p a_i^{p-1} - \gamma_i = 0,\ \gamma_i \ge 0, \forall i \\ \lambda_1 g(a) = 0, \quad \gamma_i a_i = 0, \forall i. \end{array}$$

This implies that, for each i we have

$$\begin{array}{rl} -\lambda_0 y_i + \lambda_1 p a_i^{p-1} = \gamma_i = 0, & \text{if } a_i > 0, \\ -\lambda_0 y_i = \gamma_i \ge 0, & \text{if } a_i = 0. \end{array}$$

Since $y_i \ge 0$ and $\lambda_0 \ge 0$, we conclude that $\lambda_0 y_i = \gamma_i = 0$, if $a_i = 0$. Therefore, we get

$$-\lambda_0 y_i + \lambda_1 p a_i^{p-1} = \gamma_i = 0, \forall i. \tag{16}$$

The remainder of the proof now follows as in [8]. More precisely, since not all the multipliers are 0, if $\lambda_0 = 0$, then $\lambda_1 > 0$. This implies that

$$g(a) = 0, \tag{17}$$

and, by (16) that $a = 0$, contradiction. On the other hand, if $\lambda_1 = 0$, then $\lambda_0 > 0$ which implies $y = 0$, contradiction. Thus, both λ_0 and λ_1 are positive and we can assume, without loss of generality, that $\lambda_0 = 1$. Moreover, we conclude that (17) holds. From (16) and (17) we get

$$\begin{aligned} -f(a) &= \textstyle\sum_{i=1}^n a_i y_i \\ &= \lambda_1 p \textstyle\sum_{i=1}^n a_i^p \\ &= \lambda_1 p. \end{aligned}$$

Since $q = p/(p-1)$, (16) and (17) now imply that

$$\sum_{i=1}^n y_i^q = \sum_{i=1}^n (\lambda_1 p)^q a_i^q = (\lambda_1 p)^q = -f(a)^q.$$

$\blacksquare$

2.3 Sensitivity Analysis

Consider now the *convex (perturbed) program*

$$(P_\varepsilon) \qquad \begin{array}{rl} \mu(\varepsilon) = \min & f(x) \\ \text{subject to} & g_i(x) \le \varepsilon_i, \ \ i = 1, \dots, m, \\ & h_j(x) = \varepsilon_j, \ \ j = m+1, \dots, q, \\ & x \in U, \end{array} \tag{18}$$

where U is an open subset of $\mathbb{R}^n$, and the functions f and $g_i, \ i = 1, \dots, m$, are convex and $h_j, \ j = m+1, \dots, q$, are affine. The *generalized Slater Constraint Qualification (CQ)* for (P_ε) states that

$$\begin{array}{c} \text{there exists } \hat{x} \in \operatorname{int} U \text{ such that} \\ g_i(\hat{x}) < \varepsilon_i, i = 1, \dots, m, \text{ and } h_j(\hat{x}) = \varepsilon_j, j = m+1, \dots, q. \end{array} \tag{19}$$

We can now state the convex multiplier rule and the corresponding shadow price interpretation of the multipliers. (See e.g. [8, 9].)

Theorem 3. *Suppose that the CQ in* (19) *holds for* (P_0) *in* (18). *Then,*

$$\mu(0) = \min\{f(x) + \sum_{i=1}^{m} \lambda_i g_i(x) + \sum_{j=m+1}^{q} \lambda_j h_j(x) : \ \ x \in U\}, \tag{20}$$

for some $\lambda_j \in R, \ \ j = m+1, \dots q$, *and* $\lambda_i \ge 0, \ \ i = 1, \dots, m$. *If* $a \in \mathscr{F}$ *solves* (P_0), *then in addition*

$$\lambda_i g_i(a) = 0, \quad i = 1, \dots, m. \tag{21}$$

Theorem 4. *Suppose that* $a \in \mathscr{F}$. *Then, (20) and (21) imply that* a *solves* (P_0).

Theorem 5. *Suppose that* a^1 *and* a^2 *are solutions to* (P_{ε^1}) *and* (P_{ε^2}), *respectively, with corresponding multiplier vectors* λ^1 *and* λ^2. *Then,*

$$(\varepsilon^2 - \varepsilon^1, \lambda^2) \le f(a^1) - f(a^2) \le (\varepsilon^2 - \varepsilon^1, \lambda^1). \tag{22}$$

■

Note that since the functions are convex and the problem (20) is an unconstrained minimization problem, we see that if $a \in \mathscr{F}$ solves (P_0), then (20) and (21) are equivalent to the system

$$\begin{array}{l} \bigtriangledown f(a) + \sum_{i=1}^{m} \lambda_i \bigtriangledown g_i(a) + \sum_{j=m+1}^{q} \lambda_j \bigtriangledown h_j(a) = 0 \\ \lambda_i \ge 0, \ \ \lambda_i g_i(a) = 0, \ \ i = 1, \dots, m. \end{array} \tag{23}$$

Moreover, since $f(a^i) = \mu(\varepsilon^i)$, when a^i solves (P_{ε^i}), (22) implies that $-\lambda^i \in \partial\mu(\varepsilon^i)$, i.e. the negative of the multiplier λ^i is in the subdifferential of the perturbation function $\mu(\varepsilon)$ at ε^i. In fact (see [9])

$$\partial\mu(0) = \{-\lambda : \lambda \text{ is a multiplier vector for } (P_0)\}.$$

If λ is unique, this implies that μ is differentiable at 0 and $\bigtriangledown\mu(0) = -\lambda$. Note that

$$\partial\mu(a) = \{\phi \in \mathbb{R}^n : \ (\phi, \eta - a) \le \mu(\eta) - \mu(a)\}.$$

We will apply the convex multiplier rule in the sequel. Note that the necessity of (23) requires a constraint qualification, such as Slater's condition, while sufficiency does not. Thus, in our applications we do not have to worry about any constraint qualification. For, as soon as we can solve (23), the sufficiency guarantees optimality. Note that necessity is used in numerical algorithms.

3 Generating Eigenvalue Bounds

We consider the $n \times n$ matrix A which has real eigenvalues $\lambda_1 \ge \ldots \ge \lambda_n$. We have seen how to apply optimization techniques in order to prove several known inequalities. Now suppose that we are given several facts about the matrix A, e.g. $n, \mathrm{trace} A$ and/or $\det A$ etc... In order to find upper (lower) bounds for $f(\lambda)$, a function of the eigenvalues, we could then maximize (minimize) $f(\lambda)$ subject to the constraints corresponding to the given facts about A. An explicit solution to the optimization problem would then provide the, previously unknown, best upper (lower) bounds to $f(\lambda)$ given these facts. To be able to obtain an explicit solution we must choose simple enough constraints and/or have a lot of patience.

Suppose we wish to obtain a lower bound for λ_k, the k-th largest eigenvalue, given the facts that

$$K := \mathrm{trace} A, \quad m := \frac{K}{n}, \quad L := \mathrm{trace} A^2, \quad s^2 := \frac{L}{n} - m^2.$$

Then we can try and solve the program

$$\begin{array}{ll} \min & \lambda_k \\ \text{subject to} & (a)\ \sum_{i=1}^n \lambda_i = K, \\ & (b)\ \sum_{i=1}^n \lambda_i^2 \le L, \\ & (c)\ \lambda_k - \lambda_i \le 0, \ \ i = 1, \ldots, k-1, \\ & (d)\ \lambda_i - \lambda_k \le 0, \ \ i = k+1, \ldots, n. \end{array} \tag{24}$$

This is a program in the variables λ_i with n, k, K and L fixed. We have replaced the constraint $\sum \lambda_i^2 = L$ with $\sum \lambda_i^2 \le L$. This increases the feasible set of vectors $\lambda = (\lambda_i)$ and so the solution of (24) still provides a lower bound for λ_k. However, the program now becomes a convex program. Note that $(\mathrm{trace} A)^2 = (\sum \lambda_i)^2 \le n \sum \lambda_i^2 =$

$n\mathrm{trace}A^2$, by the Cauchy-Schwartz inequality, with equality if and only if $\lambda_1 = \lambda_2 = \ldots = \lambda_n$. Thus, if $(\mathrm{trace}A)^2 = n\mathrm{trace}A^2$, then we can immediately conclude that $\lambda_i = \mathrm{trace}A/n,\ i = 1,\ldots,n$. Moreover, if $nL \neq K^2$, then $nL > K^2$, and we can always find a feasible solution to the constraints which strictly satisfies $\sum \lambda_i^2 < L$, and hence we can always satisfy the generalized Slater CQ.

Theorem 6. *If $K^2 < nL$ and $1 < k \leq n$, then the (unique) explicit solution to* (24) *is*

$$\begin{array}{l} \lambda_1 = \ldots = \lambda_{k-1} = m + s\left(\frac{n-k+1}{k-1}\right)^{\frac{1}{2}}, \\ \lambda_k = \ldots = \lambda_n = m - s\left(\frac{k-1}{n-k+1}\right)^{\frac{1}{2}}, \end{array} \tag{25}$$

with Lagrange multipliers for the constraints (a) to (d) in (24) *being*

$$\begin{array}{l} \alpha = \frac{-m}{ns}\left(\frac{k-1}{n-k+1}\right)^{\frac{1}{2}} - \frac{1}{n}, \\ \beta = \left(\frac{k-1}{n-k+1}\right)^{\frac{1}{2}} \frac{1}{2ns}, \\ \gamma_i = 0,\ \ i = 1,\ldots,k-1, \\ \gamma_i = \frac{1}{n-k+1},\ \ i = k+1,\ldots,n, \end{array} \tag{26}$$

respectively.

Proof. Since (24) is a convex program, the Karush-Kuhn-Tucker conditions are sufficient for optimality. Thus, we need only verify that the above solution satisfies both the constraints and (23). However, let us suppose that the solution is unknown beforehand, and show that we can use the necessity of (23) to find it. We get

$$\alpha + 2\beta\lambda_i - \gamma_i = 0,\ \ i = 1,\ldots,k-1 \tag{27a}$$

$$1 + \alpha + 2\beta\lambda_k + \sum_{i=1}^{k-1}\gamma_i - \sum_{i=k+1}^{n}\gamma_i = 0 \tag{27b}$$

$$\alpha + 2\beta\lambda_i + \gamma_i = 0, i = k+1,\ldots,n, \tag{27c}$$

$$\alpha \in R, \beta \geq 0, \beta\left(\sum_{i=1}^{n}\lambda_i^2 - L\right) = 0, \gamma_i \geq 0, \gamma_i(\lambda_i - \lambda_k) = 0, i = 1,\ldots,n. \tag{27d}$$

Now, if $\beta = 0$, then

$$\alpha = \gamma_i = -\gamma_j, i = 1,\ldots,k-1, j = k+1,\ldots,n.$$

This implies that they are all 0, (or all > 0 if $k = n$) which contradicts (27b). Thus, $\beta > 0$ and, by (27d),

$$\sum_i \lambda_i^2 = L. \tag{28}$$

From (27a) to (27d), we now have

$$\lambda_i = \frac{-\alpha}{2\beta} + \frac{\gamma_i}{2\beta}, i = 1,\ldots,k-1,$$

$$\lambda_j = \frac{-\alpha}{2\beta} - \frac{\gamma_j}{2\beta}, j = k+1, \ldots, n,$$

$$\lambda_k = \frac{-\alpha}{2\beta} - \frac{1}{2\beta} - \sum_{i=1}^{k-1} \frac{\gamma_i}{2\beta} + \sum_{j=k+1}^{n} \frac{\gamma_j}{2\beta}.$$

Suppose $\gamma_{i_0} > 0$, where $1 \le i_0 \le k-1$. Then (27a) and (27d) imply that $\lambda_k = \lambda_{i_0} = \frac{-\alpha}{2\beta} + \frac{\gamma_{i_0}}{2\beta}$. On the other hand, since we need

$$\lambda_i = \frac{-\alpha}{2\beta} + \frac{\gamma_i}{2\beta} \ge \lambda_k = \frac{-\alpha}{2\beta} + \frac{\gamma_{i_0}}{2\beta} > \frac{-\alpha}{2\beta}, i = 1, \ldots, k-1,$$

we must have $\gamma_i > 0, i = 1, \ldots, k-1$ and, by complementary slackness,

$$\lambda_i = \lambda_k = \frac{-\alpha + \gamma_i}{2\beta}, i = 1, \ldots, k-1. \tag{29}$$

But then

$$\lambda_k = \frac{-\alpha}{2\beta} - \frac{1}{2\beta} - \sum_{j=1}^{k-1} \frac{\gamma_j}{2\beta} + \sum_{j=k+1}^{n} \frac{\gamma_j}{2\beta} = \frac{-\alpha}{2\beta} + \frac{\gamma_i}{2\beta}, i = 1, \ldots, k-1,$$

which implies $\sum_{j=k+1}^{n} \gamma_j > 0$. But $\gamma_j > 0$ implies $\lambda_j = \lambda_k$. This yields $\lambda_1 = \ldots = \lambda_k = \ldots = \lambda_n$, a contradiction since we assumed $nL > K^2$. Thus, we conclude that

$$\gamma_i = 0, \quad i = 1, \ldots, k-1.$$

Now if $\gamma_{j_0} > 0$, for some $k+1 \le j_0 \le n$, then $\lambda_{j_0} = \frac{-\alpha}{2\beta} - \frac{1}{2\beta} + \sum_{j=k+1}^{n} \frac{\gamma_j}{2\beta}$. Since $\lambda_j = \frac{-\alpha}{2\beta} - \frac{\lambda_j}{2\beta} \le \lambda_k$, we must have $\gamma_j > 0$, for all $j = k+1, \ldots, n$. Note that $\gamma_j = 0$ for all $j = k+1, \ldots, n$, leads to a contradiction since then $\lambda_j = \frac{-\alpha}{2\beta} > \lambda_k = \frac{-\alpha}{2\beta} - \frac{1}{2\beta}$. Thus we have shown that the λ_i's split into two parts,

$$\lambda_1 = \ldots = \lambda_{k-1} > \lambda_k = \ldots = \lambda_n. \tag{30}$$

The Lagrange multipliers also split into two parts,

$$\gamma_1 = \ldots = \gamma_{k-1} = 0, \gamma_{k+1} = \ldots = \gamma_n = \gamma.$$

We now explicitly solve for $\lambda_1, \lambda_k, \alpha, \beta$, and γ. From the first two constraints and (28) we get

$$\begin{aligned} (k-1)\lambda_1 + (n-k+1)\lambda_k &= K, \\ (k-1)\lambda_1^2 + (n-k+1)\lambda_k^2 &= L. \end{aligned} \tag{31}$$

Eliminating one of the variables in (31) and solving the resulting quadratic yields (25). Uniqueness of (25) follows from the necessity of the optimality conditions. It also follows from the strict convexity of the quadratic constraint in the program (24). Using the partition in (30), we can substitute (27c) in (27b) to get

$$1+\alpha+2\beta(\frac{-\alpha-\gamma}{2\beta})-(n-k)\gamma=0,$$

i.e.

$$\gamma=\frac{1}{n-k+1}. \tag{32}$$

In addition, $\lambda_1-\lambda_k=\frac{-\alpha}{2\beta}-\left(\frac{-\alpha}{2\beta}-\frac{\gamma}{2\beta}\right)$ implies

$$\beta=\frac{\gamma}{2(\lambda_1-\lambda_k)}=\left(\frac{k-1}{n-k+1}\right)^{\frac{1}{2}}\frac{1}{2ns}, \tag{33}$$

while

$$\alpha=-2\lambda_1\beta=\frac{-m}{ns}\left(\frac{k-1}{n-k+1}\right)^{\frac{1}{2}}-\frac{1}{n}. \tag{34}$$

■

In the above, we have made use of the necessity of the Karush-Kuhn-Tucker (**KKT**) conditions to eliminate non-optimal feasible solutions. Sufficiency of the **KKT** conditions in the convex case, then guarantees that we have actually found the optimal solution and so we need not worry about any constraint qualification. We can verify our solution by substituting into (27).

The explicit optimal solution yields the lower bound as well as conditions for it to be attained.

Corollary 1. *Let* $1<k\leq n$. *Then*

$$\lambda_k\geq m-\left(\frac{k-1}{n-k+1}\right)^{\frac{1}{2}}s, \tag{35}$$

with equality if and only if $\lambda_1=\ldots=\lambda_{k-1},\lambda_k=\ldots=\lambda_n$. ■

The above Corollary is given in [12] but with a different proof. From the proof of Theorem 6, we see that $\beta=0$ if $k=1$ and so the quadratic constraint $\sum\lambda_i^2\leq L$ may not be binding at the optimum. Thus the solution may violate the fact that $\sum\lambda_i^2=\operatorname{trace}A^2$. This suggests that we can do better if we replace the inequality constraint by the equality constraint $\sum\lambda_i^2=L$. We, however, lose the nice convexity properties of the problem. However, applying the John conditions, Theorem 2, and using a similar argument to the proof of Theorem 6, yields the explicit solution

$$\begin{aligned}\lambda_1=\ldots=\lambda_{n-1}&=m+s/(n-1)^{\frac{1}{2}}\\ \lambda_n&=m-(n-1)^{\frac{1}{2}}s,\end{aligned}$$

i.e. we get the lower bound

$$\lambda_1 \geq m + s/(n-1)^{\frac{1}{2}},$$

with equality if and only if $\lambda_1 = \ldots = \lambda_{n-1}$. (This result is also given in [12] but with a different proof.)

The Lagrange multipliers obtained in Theorem 6 also provide the sensitivity coefficients for program (24). (In fact, the multipliers are unique and so the perturbation function is differentiable.) This helps in obtaining further bounds for the eigenvalues when we have some additional information, e.g. from Gerŝgorin discs. We can now improve our lower bound and also obtain lower bounds for other eigenvalues.

Corollary 2. *Let* $1 < k < n$. *Suppose that we know*

$$\lambda_{k+i} - \lambda_k \leq -\varepsilon_i, \tag{36}$$

where $\varepsilon_i \geq 0, i = 1, \ldots, n-k$. *Then*

$$\lambda_k \geq m - \left(\frac{k-1}{n-k+1}\right)^{\frac{1}{2}} s + \frac{1}{n-k+1} \sum_{i=1}^{n-k} \varepsilon_i. \tag{37}$$

Proof. The result follows immediately from the left-hand side of (22), if we perturb the constraints in program (24) as given in (36) and use the multipliers $\gamma_i = \frac{1}{n=k+1}$. Note that λ_k remains the k-th largest eigenvalue. ■

Corollary 3. *Let* $1 < k < n$. *Suppose that we know*

$$\lambda_{k+i-1} - \lambda_{k+t} \leq \varepsilon_i,$$

for some $\varepsilon_i \geq 0, i = 1, \ldots, t$. *Then*

$$\lambda_{k+t} \geq m - \left(\frac{k-1}{n-k+1}\right)^{\frac{1}{2}} s - \frac{1}{n-k+1} \sum_{i=1}^{t} \varepsilon_i.$$

Proof. Suppose that we perturb the constraints in program (24) to obtain

$$\lambda_{k+i} - \lambda_k \leq \varepsilon_i, i = 1, \ldots, t. \tag{38}$$

Since $\varepsilon_i \geq 0$, this allows a change in the ordering of the λ_i, for then we can have $\lambda_{k+i} = \lambda_k + \varepsilon_i > \lambda_k$. Thus the perturbation in the hypothesis is equivalent to (38). From (22), the result follows, since the k-th ordered λ_i has become the $(k+t)$-th order λ_i. ■

The results obtained using perturbations in the above two Corollaries can be approached in a different way. Since the perturbation function μ is convex (see e.g. [9]) we are obtaining a lower estimate of the perturbed value $\mu(\varepsilon)$ by using the multiplier whose negative is an element of the sub-differential $\partial\mu(\varepsilon)$. We can however obtain better estimates by solving program (24) with the new perturbed constraints.

Theorem 7. *Under the hypotheses of Corollary 2, we get*

$$\lambda_k \geq m - \left(\frac{k-1}{n-k+1}\right)^{\frac{1}{2}} s_\varepsilon + \frac{1}{n-k+1}\sum_{j=1}^{t}\varepsilon_j, \tag{39}$$

where

$$s_\varepsilon^2 = s^2 - \frac{(n-k+1)\sum_{j=1}^t \varepsilon_j^2 - \left(\sum_{j=1}^t \varepsilon_j\right)^2}{n(n-k+1)}.$$

Equality holds if and only if

$$\lambda_1 = \ldots = \lambda_{k-1}; \lambda_{k+i} - \lambda_k = -\varepsilon_i, i = 1,\ldots,t.$$

Proof. We replace the last set of constraints in program (24) by the perturbed constraints (36), for $i = k+1,\ldots,k+t$. The arguments in the proof of Theorem 6 show that the solution must satisfy (28) and

$$\lambda_1 = \ldots = \lambda_{k-1}; \lambda_{k+j} - \lambda_k = -\varepsilon_j, j = 1,\ldots,t.$$

We can assume that $k+t = n$, since we must have $\lambda_{k+t+j} \leq \lambda_{k+t}$ and so we can add the constraints

$$\lambda_{k+t+j} - \lambda_k \leq -\varepsilon_t, j > 1,$$

if required. This leads to the system

$$\begin{array}{l}(k-1)\lambda_1 + \sum_{j=1}^t(\lambda_k - \varepsilon_j) = K,\\ (k-1)\lambda_1^2 + \sum_{j=1}^t(\lambda_k - \varepsilon_j)^2 = L.\end{array} \tag{40}$$

Let $\varepsilon := \sum_{j=1}^t \varepsilon_j$ and $\bar{\varepsilon} := \sum_{j=1}^t \varepsilon_j^2$. Then (40) reduces to

$$\begin{array}{l}(k-1)\lambda_1 + (n-k+1)\lambda_k = K_\varepsilon := K + \varepsilon,\\ (k-1)\lambda_1^2 + (n-k+1)\lambda_k^2 - 2\varepsilon\lambda_k = L_\varepsilon := L - \bar{\varepsilon}.\end{array}$$

Then

$$\lambda_k = (K_\varepsilon - (k-1)\lambda_1)/(n-k+1).$$

Substituting for λ_k yields the quadratic

$$n(k-1)\lambda_1^2 - 2(k-1)(K_\varepsilon - \varepsilon)\lambda_1 + K_\varepsilon^2 - 2\varepsilon K_\varepsilon - (n-k+1)L_\varepsilon = 0,$$

which implies

$$\lambda_1 = \frac{K}{n} + \left(\frac{n-k+1}{k-1}\right)^{\frac{1}{2}} \left\{ \frac{(n-k+1)L_\varepsilon + \varepsilon^2}{n(n-k+1)} - \left(\frac{K}{n}\right)^2 \right\}^{\frac{1}{2}} \tag{41}$$

and

$$\lambda_k = \frac{K}{n} + \frac{\varepsilon}{n-k+1} - \left(\frac{k-1}{n-k+1}\right)^{\frac{1}{2}} \left\{ \frac{L}{n} - \frac{(n-k+1)\bar{\varepsilon} + \varepsilon^2}{n(n-k+1)} - \left(\frac{K}{n}\right)^2 \right\}^{\frac{1}{2}} . \tag{42}$$

■

Note that the partial derivative with respect to $-\varepsilon_j$, at $\varepsilon_j = 0$, of the lower bound for λ_k in (39) is $-1/(n-k+1)$. This agrees with the fact that the corresponding multiplier is $\gamma_j = 1/(n-k+1)$.

Corollary 3 can be improved in the same way that Theorem 7 improves Corollary 2. We need to consider the program (24) with the new constraints

$$\lambda_{k-i} - \lambda_k \leq \varepsilon_i, i = 1, \ldots, t,$$

where $\varepsilon_i \geq 0$ and k has replaced $k+t$. Further improvements can be obtained if more information is known. For example, we might know that

$$\lambda_{t+i} - \lambda_t \leq -\varepsilon_i, \quad i = 1, \ldots, s,$$

where $l+s < t+s < k$ or $k+s < t+s < n$. In these cases we would obtain a result as in Theorem 7.

In the remainder of this section we consider bounds for $\lambda_k + \lambda_\ell$ and $\lambda_k - \lambda_\ell$. To obtain a lower bound for $\lambda_k + \lambda_\ell$ we consider the program

$$\begin{array}{lll} \text{minimize} & & \lambda_k + \lambda_\ell \\ \text{subject to} & (a) & \sum \lambda_i = K, \\ & (b) & \sum \lambda_i^2 \leq L, \\ & (c) & \lambda_i - \lambda_k \leq 0, i = k+1, \ldots, \ell \\ & (d) & \lambda_j - \lambda_\ell \leq 0, j = \ell+1, \ldots, n. \end{array} \tag{43}$$

Note that we have ignored the constraints $\lambda_i - \lambda_k \geq 0, i = 1, \ldots, k-1$. From our previous work in the proof of Theorem 6, we see that the Lagrange multipliers for these constraints should all be 0, i.e. we can safely ignore these constraints without weakening the bound.

Theorem 8. *Suppose that $K^2 < nL$ and $1 \leq k < \ell \leq n$. Then the explicit solution to* (43) *is*

1. If $n-\ell > \ell-k-1$, then

$$\begin{array}{l} \lambda_1 = \ldots = \lambda_{k-1} = m + s\left(\frac{n-k+1}{k-1}\right)^{\frac{1}{2}}, \\ \lambda_k = \ldots = \lambda_n = m - s\left(\frac{k-1}{n-k+1}\right)^{\frac{1}{2}}, \end{array} \tag{44}$$

with Lagrange multipliers for the constraints

$$\begin{array}{l}\alpha=-2\beta\lambda_k-\frac{2}{(n-k+1)},\\ \beta=\frac{1}{ns}\left(\frac{k-1}{n-k+1}\right)^{\frac{1}{2}},\\ \gamma_i=\delta_j=2/(n-k+1), i=k+1,\ldots,\ell-1, j=\ell+1,\ldots,n,\\ \gamma_\ell=\frac{2(n-\ell+1)}{n-k+1}-1.\end{array} \tag{45}$$

2. *If $n-\ell\le\ell-k-1$, then λ_1 is the solution of the quadratic* (51) *and*

$$\begin{array}{l}\lambda_1=\ldots=\lambda_{k-1},\\ \lambda_k=\ldots=\lambda_{\ell-1}=\lambda_1+\frac{K-n\lambda_1}{2(\ell-k)}\\ \lambda_\ell=\ldots=\lambda_n=\lambda_1+\frac{K-n\lambda_1}{2(n-\ell+1)}\end{array} \tag{46}$$

with Lagrange multipliers for the constraints being

$$\begin{array}{l}\alpha=-2\beta\lambda_1,\\ \beta=\frac{1}{n\lambda_1-K},\\ \gamma_i=1/(\ell-k), i=k+1,\ldots,\ell-1, \gamma_\ell=0,\\ \delta_j=1/(n-\ell+1), j=\ell+1,\ldots,n.\end{array}$$

Proof. To simplify notation, we let $\beta\leftarrow 2\beta$. The Karush-Kuhn-Tucker conditions for (43) yield

$$\begin{array}{llllll}(a) & \alpha+\beta\lambda_i & & & =0, & i=1,\ldots,k-1,\\ (b) & 1+\alpha+\beta\lambda_k-\sum_{i=k+1}^{\ell}\gamma_i & & & =0, & \\ (c) & \alpha+\beta\lambda_i+ & \gamma_i & & =0, & i=k+1,\ldots,\ell-1,\\ (d) & 1+\alpha+\beta\lambda_\ell+ & \gamma_\ell & -\sum_{j=\ell+1}^{n}\delta_j & =0, & \\ (e) & \alpha+\beta\lambda_j & & +\quad\delta_j & =0, & j=\ell+1,\ldots,n,\\ (f) & \beta,\gamma_i,\delta_j\ge 0, & \beta\left(\sum_1^n\lambda_t^2-L\right)=0, \forall i,j, & & & \\ (g) & \gamma_i(\lambda_i-\lambda_k)=0, & \delta_j(\lambda_j-\lambda_\ell)=0, \forall i,j. & & & \end{array} \tag{47}$$

First suppose that $\beta=0$. If $k>1$, we get that $\alpha=0$ and so $\gamma_i=\delta_j=0$, for all i,j. This contradicts (47)(b). If $k=1$, we get $\alpha=-\delta_j=-\gamma_i$, for all i,j. So if $\alpha\ne 0$, we must have $\lambda_1=\ldots=\lambda_n=m$. So we can let $k>1$ and assume that $\beta>0$. Then we get

$$\begin{array}{llllll}\lambda_i= & -\frac{\alpha}{\beta} & & & & i=1,\ldots,k-1\\ \lambda_k=\frac{-1}{\beta} & -\frac{\alpha}{\beta} & +\sum_{i=k+1}^{\ell}\frac{\gamma_i}{\beta} & & & \\ \lambda_i= & -\frac{\alpha}{\beta} & -\frac{\gamma_i}{\beta} & & & i=k+1\ldots,\ell-1\\ \lambda_\ell=\frac{-1}{\beta} & -\frac{\alpha}{\beta} & -\frac{\gamma_\ell}{\beta} & +\frac{\sum_{j=\ell+1}^{n}\delta_j}{\beta} & & \\ \lambda_j= & \frac{-\alpha}{\beta} & & -\frac{\delta_j}{\beta} & & j=\ell+1,\ldots,n\end{array}$$

To simplify notation, the index i will now refer to $i=k+1,\ldots,\ell-1$ while the index j will refer to $j=\ell+1,\ldots,n$. Since $\lambda_{i_0}\le\lambda_k, i_0=k+1,\ldots,\ell-1$, we get

$$\sum \gamma_i \geq 1 - \gamma_{i_0}. \tag{48}$$

Therefore, there exists at least one $\gamma_{i_0} > 0$. This implies that $\lambda_{i_0} = \lambda_k$, and

$$\gamma_{i_0} = 1 - \sum_{i=k+1}^{\ell} \gamma_i.$$

Now if $\gamma_{i_1} = 0$, then

$$\lambda_{i_1} = \frac{-\alpha}{\beta} > \frac{-\alpha}{\beta} - \frac{\gamma_{i_0}}{\beta} = \lambda_{i_0} = \lambda_k,$$

which is a contradiction. We conclude

$$\lambda_{i_0} = \lambda_k, \gamma_{i_0} = 1 - \textstyle\sum_{i=k+1}^{\ell} \gamma_i, \quad i_0 = k+1, \ldots, \ell - 1.$$

Note that if $\gamma_\ell = 0$, we get

$$\gamma_i = 1/(\ell - k), i = k+1, \ldots, \ell - 1.$$

Similarly, since $\lambda_{j_0} \leq \lambda_\ell, j_0 = \ell + 1, \ldots, n$, we get

$$\sum \delta_j - \gamma_\ell \geq 1 - \delta_j,$$

i.e. at least one $\delta_{j_0} > 0$ and so $\lambda_{j_0} = \lambda_\ell$. But if $\delta_{j_1} = 0$, then

$$\lambda_{j_1} = \frac{-\alpha}{\beta} > \frac{-\alpha}{\beta} - \frac{\delta_{j_0}}{\beta} = \lambda_{j_0} = \lambda_\ell,$$

a contradiction. We conclude

$$\lambda_{j_0} = \lambda_\ell, \delta_{j_0} = 1 - \textstyle\sum_{l+1}^{n} \delta_j + \gamma_\ell, \quad j_0 = \ell + 1, \ldots, n.$$

So that if $\gamma_\ell = 0$, we also have

$$\delta_j = 1/(n - \ell + 1), j = \ell + 1, \ldots, n.$$

There now remains two cases to consider:

$$\gamma_\ell = 0 \text{ and } \gamma_\ell > 0.$$

Since $\lambda_k \geq \lambda_\ell$, we must have

$$\sum_{i=k+1}^{\ell-1} \gamma_i + 2\gamma_\ell \geq \sum_{j=\ell+1}^{n} \delta_j.$$

Moreover

$$\lambda_j \leq \lambda_\ell \leq \lambda_k = \lambda_i, \text{ for all } i, j,$$

which implies that

$$\delta_j \geq \gamma_i, \text{ for all } i, j.$$

So that if $\gamma_\ell = 0$, we must have

$$\ell - k - 1 > n - \ell.$$

From the expressions for γ_i, δ_j, we get

$$\begin{array}{rl} \lambda_k = & \frac{-1}{\beta} - \frac{\alpha}{\beta} + \frac{\ell-k-1}{(\ell-k)\beta}, \\ = \lambda_i = & \frac{-\alpha}{\beta} - \frac{1}{(\ell-k)\beta}, \quad i = k+1, \ldots, \ell-1, \\ \lambda_\ell = & \frac{-1}{\beta} - \frac{\alpha}{\beta} + \frac{n-\ell}{(n-\ell+1)\beta}, \\ = \lambda_j = & \frac{-\alpha}{\beta} - \frac{1}{(n-\ell+1)\beta}, \; j = \ell+1, \ldots, n. \end{array}$$

Thus

$$\begin{array}{l} \lambda_k = \lambda_1 - \frac{1}{\beta(\ell-k)}, \\ \lambda_\ell = \lambda_1 - \frac{1}{\beta(n-\ell+1)}. \end{array} \tag{49}$$

After substitution, this yields

$$\frac{-1}{\beta} = \frac{K - n\lambda_1}{2}. \tag{50}$$

Since $\beta > 0$, we can apply complementary slackness and substitute for λ_k and λ_ℓ. We get the quadratic

$$(k-1)\lambda_1^2 + (\ell-k)\left(\lambda_1 + \frac{K - n\lambda_1}{2(\ell-k)}\right)^2 + (n-\ell+1)\left(\lambda_1 + \frac{K - n\lambda_1}{2(n-\ell+1)}\right)^2 = L, \tag{51}$$

or equivalently

$$\begin{array}{l} \left\{4(\ell-k)(n-\ell+1)(k-1) + (n-l+1)(2(\ell-k)-n)^2 + (l-k)(2(n-\ell+1)-n)^2\right\}\lambda_1^2 \\ \quad +2K\left\{(n-l+1)(2(\ell-k)-n) + (l-k)(2(n-\ell+1)-n)\right\}\lambda_1 \\ \quad\quad + \left\{(n-l+1)K^2 + (l-k)K^2 - 4(\ell-k)(n-\ell+1)L\right\} = 0. \end{array}$$

Note that the above implies

$$\lambda_k + \lambda_\ell = 2\lambda_1 + \frac{K - n\lambda_1}{2}\left(\frac{1}{\ell-k} + \frac{1}{n-\ell+1}\right). \tag{52}$$

In the case that $\ell - k - 1 < n - \ell$, we get $\gamma_\ell > 0$. Thus, $\lambda_\ell = \lambda_k$ and

$$\lambda_i = \lambda_k, i = k+1, \ldots, n. \tag{53}$$

Substitution yields the desired optimal values for λ. Moreover,

$$\gamma_i = \delta_j = 1 - \sum_{t=k+1}^{\ell} \gamma_t = 1 - \sum_{s=\ell+1}^{n} \delta_s + \gamma_\ell.$$

Let $\gamma = \gamma_i$ and $\delta = \delta_j$, then we get

$$\gamma = \delta = 1 - (\ell - k - 1)\gamma - \gamma_\ell = 1 - (n - \ell)\delta + \gamma_\ell.$$

This implies

$$\begin{array}{l} \gamma = \delta = 2/(n-k+1) \\ \gamma_\ell = \frac{2(n-\ell+1)}{n-k+1} - 1. \end{array} \tag{54}$$

Now if $k > 1$, we see that

$$\beta = \frac{\gamma_i}{\lambda_1 - \lambda_i} = 2\left(\frac{k-1}{n-k+1}\right)^{\frac{1}{2}} /(ns).$$

Then

$$\alpha = -\beta\lambda_k - \gamma_i.$$

■

To obtain an upper bound for $\lambda_k - \lambda_\ell$, we consider the program

$$\begin{array}{ll} \text{minimize} & -\lambda_k + \lambda_\ell \\ \text{subject to} & \sum \lambda_i = K, \\ & \sum \lambda_i^2 \le L, \\ & \lambda_k - \lambda_i \le 0, i = 1, \ldots, k-1, \\ & \lambda_j - \lambda_\ell \le 0, j = \ell+1, \ldots, n \end{array} \tag{55}$$

Theorem 9. *Suppose that $K^2 < nL$ and $1 < k < \ell < n$. Let $\bar{m} = m, \bar{L} = L - (l-k-1)\bar{m}$, and $\bar{s}^2 = \frac{\bar{L}}{k+n-l+1} - \bar{m}^2$. Then the explicit solution to program* (55) *is*

$$\begin{array}{rcll} \lambda_1 = \ldots = \lambda_k & = & m + \frac{n-\ell+1}{k}\bar{s} & = m + \frac{1}{2k\beta}, \\ \lambda_{k+1} = \ldots = \lambda_{\ell-1} & = & \bar{m} & = m, \\ \lambda_\ell = \ldots = \lambda_n & = & m + \frac{k}{n-\ell+1}\bar{s}, & = m - \frac{1}{2(n-\ell+1)\beta}, \end{array} \tag{56}$$

with Lagrange multipliers for the four sets of constraints being

$$\begin{array}{rcl} \alpha & = & -2m\beta, \\ \beta & = & \frac{\sqrt{\frac{1}{k} + \frac{1}{n-l+1}}}{2\sqrt{ns}} \\ \gamma_i & = & 1/k, i = 1, \ldots, k-1, \\ \delta_j & = & 1/(n-\ell+1), j = \ell+1, \ldots, n, \end{array} \tag{57}$$

respectively.

Proof. The proof is similar to that in Theorem 8. Alternatively, sufficiency of the **KKT** can be used. ■

The Theorem yields the upper bound

$$\lambda_k - \lambda_\ell \leq n^{\frac{1}{2}} s \left(\frac{1}{k} + \frac{1}{n-\ell+1} \right)^{\frac{1}{2}}.$$

4 Fractional Programming

We now apply techniques from the theory of *fractional programming* to derive bounds for the *Kantorovich ratio*

$$\frac{\lambda_k - \lambda_\ell}{\lambda_k + \lambda_\ell}. \tag{58}$$

This ratio is useful in deriving rates of convergence for the accelerated steepest descent method, e.g. [6].

Consider the *fractional program* (e.g. [10, 11])

$$\max \left\{ \frac{f(x)}{g(x)} : x \in \mathscr{F} \right\}. \tag{59}$$

If f is concave and g is convex and positive, then $h = \frac{f}{g}$ is a *pseudo-concave* function, i.e. $h : \mathbb{R}^n \to R$ satisfies $(y-x)^t \bigtriangledown h(x) \leq 0$ implies $h(y) \leq h(x)$. The convex multiplier rules still hold if the objective function is pseudo-convex. We could therefore generate bounds for the ratio (58) as was done for λ_k in Section 3. However, it is simpler to use the following parametric technique. Let

$$h(q) := \max \left\{ f(x) - qg(x) : x \in \mathscr{F} \right\}. \tag{60}$$

Lemma 1 ([2]). *Suppose that $g(x) > 0$, for all $x \in \mathscr{F}$, and that q is a zero of $h(q)$ with corresponding solution $\bar{x} \in \mathscr{F}$. Then $\bar{x}$ solves* (59).

Proof. Suppose not. Then there exists $x \in \mathscr{F}$ such that

$$q = \frac{f(\bar{x})}{g(\bar{x})} < \frac{f(x)}{g(x)},$$

which yields $0 < f(x) - qg(x)$. This contradicts the definition of q. ■

We also need the following

Lemma 2 ([12]). *Let $w, \lambda \in \mathbb{R}^n$ be real, nonzero vectors, and let*

$$m = \lambda^T e/n \text{ and } s^2 = \lambda^T C\lambda/n,$$

where e is the $n \times 1$ vector of ones, and the centering matrix $C = I - ee^T/n$. Then

$$-s(nw^T Cw)^{\frac{1}{2}} \leq w^T \lambda - mw^T e = w^T C\lambda \leq s(nw^T Cw)^{\frac{1}{2}}.$$

Equality holds on the left (resp. right) if and only if

$$\lambda = aw + be$$

for some scalars a and b, where $a \leq 0$ (resp. $a \geq 0$). ∎

We now use the above techniques to derive an upper bound for the *Kantorovich ratio* in (58). Consider the program

$$\begin{array}{rl} \max & \gamma_{k_\ell} = \frac{\lambda_k - \lambda_\ell}{\lambda_k + \lambda_\ell} \\ \text{subject to} & \sum \lambda_i = K \\ & \sum \lambda_i^2 \leq L \\ & \lambda_k - \lambda_i \leq 0, i = 1, \ldots, k-1 \\ & \lambda_i - \lambda_\ell \leq 0, i = \ell+1, \ldots, n. \end{array} \tag{61}$$

Theorem 10. *Suppose that $1 < k < \ell < n, K^2 < nL$, and Theorem 8 guarantees $\lambda_k + \lambda_\ell > 0$. Then the explicit solution to* (61) *is*

$$\begin{array}{rl} \lambda_1 = \ldots = \lambda_k = & \bar{p}\frac{(n-\ell+1+k)-(n-\ell+1)(1-\hat{p}^{\frac{1}{2}})}{k(n-\ell+1+k)} \\ \lambda_{k+1} = \ldots = \lambda_{\ell-1} = & \frac{\operatorname{trace} A^2}{\operatorname{trace} A} \\ \lambda_\ell = \ldots = \lambda_n = & \bar{p}\frac{1-\hat{p}^{\frac{1}{2}}}{n-\ell+1+k}, \end{array} \tag{62}$$

$$\gamma_{k_\ell} = \frac{(p+k)(n-\ell+1-p)^{\frac{1}{2}}(n-\ell+1+k)}{2(p+k)(k(n-\ell+1))^{\frac{1}{2}} + \{(p+k)(n-\ell+1-p)\}^{\frac{1}{2}}(n-\ell+1+k)},$$

where

$$\begin{array}{l} p := \frac{K^2}{L} - (\ell - 1) \\ \bar{p} := K - (\ell - k - 1)\frac{L}{K} \\ \hat{p} := 1 - \frac{k}{n-\ell+1}(n-\ell+1+k)\left(\frac{1}{k} + \frac{\ell-k-1}{\bar{p}^2}\left(\frac{L}{K}\right)^2 - \frac{L}{\bar{p}^2}\right). \end{array}$$

Proof. Let $\mathscr{F}$ denote the feasible set of (61), i.e. the set of $\lambda = (\lambda_i) \in \mathbb{R}^n$ satisfying the constraints. We consider the following parametric program

$$(P_q) \qquad h(q) := \max\{(\lambda_k - \lambda_\ell) - q(\lambda_k + \lambda_\ell) : \;\; \lambda \in \mathscr{F}\}.$$

Then $h(q)$ is a strictly decreasing function of q and, if λ^* solves (P_q) with $h(q) = 0$, then, by the above Lemma 1, λ^* solves the initial program (61) also.

The objective function of (P_q) can be rewritten as $\min -(1-q)\lambda_k + (1+q)\lambda_\ell$. The Karush-Kuhn-Tucker conditions for (P_q) now yield (with $\beta \leftarrow 2\beta$ again):

$$
\begin{array}{r} \\ \\ \\ k-th \\ \\ \\ l-th \\ \\ \\ \end{array}
\begin{pmatrix} \cdots \\ \cdots \\ \cdots \\ -(1-q) \\ \cdots \\ \cdots \\ 1+q \\ \cdots \\ \cdots \end{pmatrix}
+\alpha \begin{pmatrix} 1 \\ \cdots \\ \cdots \\ \cdots \\ \cdots \\ \cdots \\ \cdots \\ \cdots \\ 1 \end{pmatrix}
+\beta \begin{pmatrix} \lambda_1 \\ \cdots \\ \cdots \\ \cdots \\ \cdots \\ \cdots \\ \cdots \\ \cdots \\ \lambda_n \end{pmatrix}
+\ldots+ \begin{pmatrix} \cdots \\ -\delta_i \\ \cdots \\ \cdots \\ \cdots \\ \cdots \\ \cdots \\ \cdots \\ \cdots \end{pmatrix}
+ \begin{pmatrix} \cdots \\ \cdots \\ \cdots \\ \sum_{i=1}^{k-1} \delta_i \\ \cdots \\ \cdots \\ \cdots \\ \cdots \\ \cdots \end{pmatrix}
$$

$$
+ \begin{pmatrix} \cdots \\ \cdots \\ \cdots \\ \cdots \\ \cdots \\ \cdots \\ \sum_{i=\ell+1}^{n} \gamma_i \\ \cdots \\ \cdots \end{pmatrix}
+\ldots+\ldots \begin{pmatrix} \cdots \\ \cdots \\ \cdots \\ \cdots \\ \cdots \\ \cdots \\ \cdots \\ \gamma_j \\ \cdots \end{pmatrix} = 0
$$

$$
\begin{gathered}
\beta \geq 0; \delta_i \geq 0, \forall i = 1,\ldots,k-1; \gamma_j \geq 0, \forall j = \ell+1,\ldots,n; \lambda \in \mathscr{F}; \\
\beta\left(\sum \lambda_i^2 - K^2\right) = 0; \delta_i\left(\lambda_k - \lambda_i\right) = 0, \forall i; \gamma_j\left(\lambda_j - \lambda_\ell\right) = 0, \forall j.
\end{gathered}
$$

Since $\lambda_k \geq \lambda_\ell$ and we seek q such that $h(q) = 0$, we need only consider $q > 0$. Further, if $\beta = 0$, then we get the following cases:

$$
\begin{aligned}
k < i < \ell &: 0 = \alpha + \beta\lambda_i \text{ implies } \alpha = 0 \\
i < k &: 0 = \alpha + \beta\lambda_i - \delta_i \text{ implies } \alpha = \delta_i = 0 \\
\ell < i &: 0 = \alpha = \beta\lambda_i + \gamma_i \text{ implies } \alpha = -\gamma_i = 0 \\
i = k &: 0 = -(1-q) + \alpha + \beta\lambda_k + \sum \delta_i \text{ implies } \alpha = -\sum \delta_i + 1 - q \\
\ell = i &: 0 = +(1+q) + \alpha + \beta\lambda_\ell - \sum \delta_i \text{ implies } \alpha = \sum \gamma_i - (1+q).
\end{aligned} \tag{63}
$$

These equations are inconsistent. Therefore, we can assume $\beta > 0$, which implies that $\sum \lambda_i^2 = L$.

Now, for $i < k$, either $\lambda_k = \lambda_i$ or $\delta_i = 0$ which implies that $\lambda_i = -\alpha/\beta$. Similarly, for $\ell < i$, $\lambda_\ell = \lambda_i$ or $\lambda_i = -\alpha/\beta$. And, for $k < i < \ell$, $\lambda_i = -\alpha/\beta$. We can therefore see that our solution must satisfy

$$
\begin{aligned}
\lambda_i &= \lambda_k,\ i = 1,\ldots,k \\
\lambda_i &= \lambda,\ \ i = k+1,\ldots,\ell-1 \\
\lambda_i &= \lambda_\ell,\ i = \ell,\ldots,n.
\end{aligned}
$$

Now rather than continuing in this way, we can apply Lemma 2. Let $w = (w_i)$, with

$$\begin{aligned} w_i &= \tfrac{1-q}{k}, \quad i=1,\dots,k \\ w_i &= 0, \quad\;\; i=k+1,\dots,\ell-1 \\ w_i &= \tfrac{-(1+q)}{n-\ell+1},\ i=\ell,\dots,n. \end{aligned}$$

Then

$$\begin{aligned} (1-q)\lambda_k-(1+q)\lambda_\ell &= \tfrac{(1-q)}{k}\sum_{i=1}^k \lambda_i - \tfrac{(1+q)}{n-\ell+1}\sum_{i=\ell}^n \lambda_i = w^T\lambda; \\ mw^T e &= m(1-q-1-q) = -2mq; \\ w^T C w &= w^T I w - \tfrac{1}{n} w^T e e^T w \\ &= \tfrac{(1-q)^2}{k} + \tfrac{(1+q)^2}{n-\ell+1} - \tfrac{1}{n}(4q^2) \\ n w^T C w &= \tfrac{n(1-q)^2}{k} + \tfrac{n(1+q)^2}{n-\ell+1} - 4q^2. \end{aligned}$$

Therefore, Lemma 2 yields

$$(1-q)\lambda_1-(1+q)\lambda_n \le -2mq + s\left\{\frac{n(1-q)^2}{k}+\frac{n(1+q)^2}{n-\ell+1}-4q^2\right\}^{\frac{1}{2}}, \tag{64}$$

with equality if and only if

$$\lambda = aw+be,$$

for some scalars a and b with $a \ge 0$. And, the right hand side of (64) equals $h(q)$, the maximum value of (P_q).

We now need to find q such that $h(q)=0$, i.e.

$$4m^2q^2 = s^2\left\{\frac{n(1-q)^2}{k}+\frac{n(1+q)^2}{n-\ell+1}-4q^2\right\};$$

$$k(n-\ell+1)4m^2q^2 = (n-\ell+1)s^2n(1-q)^2 + ks^2n(1+q)^2 - k(n-\ell+1)s^2 4q^2;$$

$$\begin{aligned} &\left(-k(n-\ell+1)4m^2 + s^2n(n-\ell+1+k) - k(n-\ell+1)s^2 4\right)q^2 \\ &\quad +2s^2n(-(n-\ell+1)+k)q \\ &\quad\quad +s^2n((n-\ell+1)+k) = 0 \end{aligned}$$

$$[(ns^2-4km^2-4s^2k)(n-\ell+1)+ns^2k]q^2 + 2ns^2(k-(n-\ell+1))q + ns^2(n-\ell+1+k) = 0$$

$$q = \frac{-ns^2(k-(n-\ell+1)) - n^2s^4(k-(n-\ell+1))^2 - [\text{as above}]ns^2(n-\ell+1+k)^{\frac{1}{2}}}{[\text{as above}]}.$$

We have chosen the negative radical for the root, since the quantity in [ ] is negative and we need $q>0$. The conditions for equality in (64) yield:

$$\begin{aligned} \lambda_i &= a\tfrac{(1-q)}{k}+b, i=1,\dots,k \\ \lambda_i &= b, i=k+1,\dots,\ell-1 \\ \lambda_i &= \tfrac{-a(1+q)}{n-\ell+1}+b, i=\ell,\dots,n \end{aligned}$$

or

$$\begin{aligned}\frac{\lambda_k-\lambda_\ell}{\lambda_k+\lambda_\ell} &= \frac{a\left(\frac{1-q}{k}\right)+b+\frac{a(1+q)}{n-\ell+1}-b}{\frac{a(1-q)}{k}-\frac{a(1+q)}{n-\ell+1}+2b}\\ &= \frac{a(1-q)(n-\ell+1)+ak(1+q)}{a(n-\ell+1)(1-q)-a(1+q)k+2bk(n-\ell+1)}.\end{aligned}$$

We now solve for a and b by substituting for λ_i in $\sum\lambda_i = K$ and $\sum\lambda_i^2 = L$:

$$k\left(\frac{a(1-q)}{k}\right)+b+(\ell-k-1)b+(n-\ell+1)\left(\frac{-a(1+q)}{n-\ell+1}\right)+b=K$$

or

$$\begin{aligned}(k+\ell-k-1+n-\ell+1)b &= K-a(1-q)+a(1+q)\\ b &= \frac{2aq+K}{n}.\end{aligned}$$

And

$$k\left(\frac{a(1-q)}{k}\right)+b^2+(\ell-k-1)b^2+(n-\ell-1)\left(\frac{-a(1-q)}{n-\ell+1}+b^2\right)=K^2$$

or

$$[k+\ell-k-1+n-\ell+1]b^2+[2a(1-q)-2a(1+q)]b+\frac{a^2(1-q)^2}{k}+\frac{a^2(1+q)^2}{n-\ell+1}-K=0$$

$$k\left(a\frac{1-q}{k}+\frac{2aq+K}{n}\right)^2+(\ell-k-1)\left(\frac{2aq+K}{n}\right)^2+(n-\ell+1)\left(\frac{-a(1+q)}{n-\ell+1}+\frac{2aq+K}{n}\right)-K=0$$

$$\begin{gathered}k\left(a\left(\frac{1-q}{k}+\frac{2q}{n}\right)+\frac{K}{n}\right)^2+(\ell-k-1)\left(\frac{2aq}{n}+\frac{K}{n}\right)^2\\ +(n-\ell+1)\left(+a\left(\frac{-(1+q)}{n-\ell+1}+\frac{2q}{n}\right)+\frac{K}{n}\right)^2-L=0;\\ [k\left(\frac{1-q}{k}+\frac{2q}{n}\right)^2+(\ell-k-1)\frac{4q^2}{n^2}+(n-\ell+1)\left(\frac{-(1-q)}{n-\ell+1}+\frac{2q}{n}\right)^2]a^2\\ +a2[k\left(\frac{1-q}{k}+\frac{2q}{n}\right)\frac{K}{n}+(\ell-k-1)\frac{2q}{n}\frac{K}{n}+(n-\ell+1)\left(\frac{2q}{n}-\frac{(1+q)}{n-\ell+1}\right)\frac{K}{n}]\\ +k\left(\frac{K}{n}\right)^2+(\ell-k-1)\left(\frac{K}{n}\right)^2+(n-\ell+1)\left(\frac{K}{n}\right)^2-L=0\\ \left[\frac{-4q^2}{n}+\frac{(1-q)^2}{k}+\frac{(1+q)^2}{n-\ell+1}\right]a^2+nm^2-L=0\\ a=\frac{-nm^2+L}{\frac{-4q^2}{n}+\frac{(1-q)^2}{k}+\frac{(1+q)^2}{(n-\ell+1)}}^{\frac{1}{2}}\end{gathered}$$

Substitution for the λ_i yields the desired results. ■

Let $\gamma = \lambda_k/\lambda_\ell$. Then

$$\gamma_{k\ell} = \frac{\gamma - 1}{\gamma + 1}$$

and

$$\frac{d_{\gamma_{k\ell}}}{d_\gamma} = \frac{2}{(\gamma+1)^2} > 0.$$

Thus γ is isotonic to $\gamma_{k\ell}$. This yields an upper bound to $\gamma_{k\ell}$, see [13]: If (to guarantee $\lambda_\ell > 0$) we have $(\ell - 1)L < L$, then

$$\frac{\lambda_k}{\lambda_\ell} \leq \frac{c+k+\left\{\frac{n-\ell+1}{k}(c+k)(n-\ell+1-c)\right\}^{\frac{1}{2}}}{c+k-\left\{\frac{k}{n-\ell+1}(c+k)(n-\ell+1-c)\right\}^{\frac{1}{2}}},$$

where

$$c = \frac{(K)^2}{L} - (\ell - 1).$$

(These inequalities are also given in [7].) Note that

$$\frac{\gamma+1}{\gamma-1} = \frac{\lambda_k + \lambda_\ell}{\lambda_k - \lambda_\ell},$$

is reverse isotonic to γ. Thus we can derive a lower bound for this ratio.

5 Conclusion

We have used optimization techniques to derive bounds for functions of the eigenvalues of an $n \times n$ matrix A with real eigenvalues. By varying both the function to be minimized (maximized) and the constraints of a properly formulated program we have been able to derive bounds for the k-th largest eigenvalue, as well as for sums, differences and ratios of eigenvalues. Additional information about the eigenvalues was introduced to improve the bounds using the shadow prices of the program. Many more different variations remain to be tried.

The results obtained are actually about ordered sets of numbers $\lambda_1 \geq \ldots \geq \lambda_n$ and do not depend on the fact that these numbers are the eigenvalues of a matrix. We can use this to extend the bounds to complex eigenvalues. The constraints on the traces can be replaced by

$$\sum v_i = \operatorname{trace} T, \sum (v_i)^2 \leq \operatorname{trace} T^* T,$$

where v_i can take on the real, imaginary, and modulus of the eigenvalues λ_i, and the matrix T can become $(A + A^*)/2,\ (A - A^*)/2i$. Further improvements can be made by using improvements of the Schur inequality $\sum (v_i)^2 \leq \operatorname{trace} T^*T$. This approach is presented in [12] and [13].

Acknowledgements Research supported by The Natural Sciences and Engineering Research Council of Canada. The author is grateful to Wai Lung Yeung, University of Waterloo, for his help with this paper. His MATLAB file that verifies many of the inequalities in the paper is available at URL: orion.uwaterloo.ca/˜hwolkowi/henry/reports/ABSTRACTS.html

References

1. A. CLAUSING. Kantorovich-type inequalities. *Amer. Math. Monthly*, 89(5):314, 327–30, 1982.
2. W. DINKELBACH. On nonlinear fractional programming. *Management Sci.*, 13:492–498, 1967.
3. L.V. KANTOROVICH and G.P. AKILOV. *Functional analysis*. Pergamon Press, Oxford, second edition, 1982. Translated from the Russian by Howard L. Silcock.
4. R. Kumar. Bounds for eigenvalues. Master's thesis, University of Alberta, 1984.
5. D.G. Luenberger. *Optimization by Vector Space Methods*. John Wiley, 1969.
6. D.G. Luenberger. Algorithmic analysis in constrained optimization. In *Nonlinear programming (Proc. Sympos., New York, 1975)*, pages 39–51. SIAM–AMS Proc., Vol. IX, Providence, R. I., 1976. Amer. Math. Soc.
7. J. Merikoski, G.P.H. Styan, and H. Wolkowicz. Bounds for ratios of eigenvalues using traces. *Linear Algebra Appl.*, 55:105–124, 1983.
8. B.H. Pourciau. Modern multiplier methods. *American Mathematical Monthly*, 87(6):433–451, 1980.
9. R.T. Rockafellar. *Convex analysis*. Princeton Landmarks in Mathematics. Princeton University Press, Princeton, NJ, 1997. Reprint of the 1970 original, Princeton Paperbacks.
10. S. SCHAIBLE. Fractional programming—state of the art. In *Operational research '81 (Hamburg, 1981)*, pages 479–493. North-Holland, Amsterdam, 1981.
11. S. SCHAIBLE. Fractional programming. In *Handbook of global optimization*, volume 2 of *Nonconvex Optim. Appl.*, pages 495–608. Kluwer Acad. Publ., Dordrecht, 1995.
12. H. Wolkowicz and G.P.H. Styan. Bounds for eigenvalues using traces. *Linear Algebra Appl.*, 29:471–506, 1980.
13. H. Wolkowicz and G.P.H Styan. More bounds for eigenvalues using traces. *Linear Algebra Appl.*, 31:1–17, 1980.