OPTIMIZATION OF A QUADRATIC FUNCTION OVER A UNIT SPHERE

A PREPRINT

Mohammad Taheri
Computer Science Engineering Department
Shiraz University
Assistant Professor
mtaheri@cse.shirazu.ac.ir

Mohammad Amin Abbaszadeh
Computer Science Engineering Department
Shiraz University
Undergraduate Alumnus
abbaszadehmohammadamin@yahoo.com

September 2021

ABSTRACT

In this work, a closed-form solution based on eigenvalues and gradient is developed to find a global solution for a quadratic form subjected to a norm constraint. The study compared the time performance of the given method with another method named SSM which is iterative and based on sequential subspaces. The results confirmed the decrease in the run time of the gradient method(our proposed method) in comparison to the SSM method. The gradient method is implemented in both python and C languages.

Keywords Quadratic programming over a single sphere · Global Optimization · Constrained Quadratic Optimization · Non-convex Optimization

1 Introduction

We intend to solve the following constrained optimization problem:

$$\min_{x} f(x) = x^T Ax + b^T x + c \text{ subject to } \|x\|_2 = 1, x \in \mathbb{R}^n. \tag{1}$$

Here **A** is a n*n symmetric matrix, **b** and **x** are n*1 vectors, and **c** is a real number. This model is utilized in a class of non-linear optimization problems named Trust Region. In each step of the trust region method, a solution for minimizing a quadratic model subjected to a norm constraint should be found [1, 2, 3, 4, 5]. Other applications of this model are in different areas such as computer vision(You can see an example in [6]), regularization methods for ill-posed problems [7, 8], graph partitioning problems [9], and other engineering disciplines. There are several solutions to solve this model proposed by the authors. Some methods to solve are using Lagrange multiplier and solving a system of equations or by diagonalization of **A**, which are efficient specially for large scale problems. FORSYTHE and GOLUB found all stationary values of the function by solving a first order equation with the norm constraint with the help of eigenvalues [10, 11]. Other solutions based on the Lagrange multiplier use eigenvalue decomposition to reduce the problem to a so-called secular equation and finally use zero-finding methods to find the solution [12, 13, 14, 15, 16]. In [17] the author solved the model based on KKT (***Karush–Kuhn–Tucker***) conditions and used an iterative method to find the optimal solution. Another author solved the model by canonical duality theory [20]. Hager in [18, 19] demonstrated another approach for the problem. His solution is based on an iterative method, in which he improves the solution over a subspace that is adjusted in every iteration. His method is named the sequential subspace method (SSM). In the rest of the article, we first demonstrate the mathematical proof of our proposed approach in section (2), and we show our solution to solve the last equation of section (2) in section (2.1), after that our proposed algorithm is explained in section (2.2), and we compare the run time performance of our method with the SSM method in section(3), and finally, we explain shortly our future research following this work in section (4).

2 Mathematical Proof

We can divide each term in the objective function in (1), without sacrificing the generality of the problem, by the norm of vector **x** that is equal to 1. Hence, we can eliminate the constraint term; Therefore, we will derive the following new problem.

$$\min \frac{x^T Ax}{x^T x} + \frac{b^T x}{x^T x} + c \tag{2}$$

$$\|x\|_2 = \sqrt{x^T x} = 1 = x^T x \tag{3}$$

In order to find the stationary points of (2), we must calculate the derivative of the function with respect to **x**. At any stationary point, the derivation of the function is equal to zero.

$$\frac{[(2Ax)(x^T x)] - [2x(x^T Ax)]}{(x^T x)^2} + \frac{[b(x^T x)] - [\frac{x}{x^T x}(b^T x)]}{(x^T x)^2} = 0 \tag{4}$$

Here, we denote a vertical $1 * n$ vector with all entries equal to zero by 0. Regarding (3), we can drop the norm terms in (4) and simplify it to the following equation:

$$2Ax - 2x(x^T Ax) + b - xx^T b = 0 \tag{5}$$

To simplify more:

$$2Ax + b = x(2x^T Ax + x^T b) \tag{6}$$

Since $(2x^T Ax + x^T b)$ is a real number, we can substitute it by α , so we can deduce the following result:

$$(\alpha I_{n*n} - 2A)x = b \tag{7}$$

Here, *I* is a n by n identity matrix. Next, we can perform eigenvalue decomposition on A since it is symmetric and substitute it in 7 with its factors. $A = Q_{n*n}\Lambda_{n*n}Q^T_{n*n}$. To elucidate more, Q is an orthonormal matrix ($Q^TQ = I$) comprises eigenvectors of A as its columns. Λ is a diagonal matrix that includes eigenvalues of A, i.e., $\Lambda_{ii} = \lambda_i$, which λ_i is the ith greatest eigenvalue of A. As $I = QIQ^{-1}$, we can rewrite (7) in the following form:

$$(Q\Lambda' Q^{-1})x = b \tag{8}$$

Here, Λ' is a diagonal matrix that $\Lambda'_{ii} = \alpha - 2\lambda_i$. We can conclude that:

$$(Q\Lambda'' Q^{-1})b = x \tag{9}$$

Here, Λ'' is a diagonal matrix that $\Lambda''_{ii} = \frac{1}{\alpha - 2\lambda_i}$. Bearing in mind that $x^Tx = 1$, we substitute (9) in the norm formula, so we yield the following equation.

$$b^TQ\Lambda''^T\Lambda''Q^Tb = 1 \tag{10}$$

if we summarize $b^TQ = y$ and expand (10) we can derive the following important equation that we will solve in the next section:

$$\Psi(\alpha) = \sum_{i=0}^{n-1} y_i^2(\frac{1}{\alpha - 2\lambda_i})^2 = 1 \tag{11}$$

So far, we have boiled down the main problem in (1) to solving the equation in (11). We can find all feasible points of our model by replacing the solutions of (11) in (9). We will yield the optimal **x** by the root of $\Psi(\alpha) - 1$ which is in the interval of $[-\infty, 2 * \lambda_{min}]$.

Lemma 2.1. *The minimizer of 1 is a* **x** *that is related to the root of* $\Psi(\alpha) - 1$ *in the interval of* $(-\infty, 2 * \lambda_{min})$.

Proof. To prove it by contradiction try and assume that the statement is false, proceed from there and at some point, you will arrive in a contradiction. We assume that there is a **x** related to a root of $\Psi(\alpha) - 1$ in the interval of $(2 * \lambda_{min}, +\infty)$; We name the root β and the related **x** is named $\mathbf{x}_\beta$. In the same way, we name another root in the interval of $(-\infty, 2 * \lambda_{min})$, γ and the related **x** is named $\mathbf{x}_\gamma$. Thereby, we must contradict this statement $f(x_\beta) < f(x_\gamma)$. □

2.1 Zero finding method

In this section, we are going to demonstrate our approach to find a solution of (11). With regard to the two following facts about the function of $\Psi(\alpha)$:

$$\lim_{x \to -\infty} \Psi(\alpha) = 0$$

$$\lim_{x \to 2*\lambda_i} \Psi(\alpha) = +\infty$$

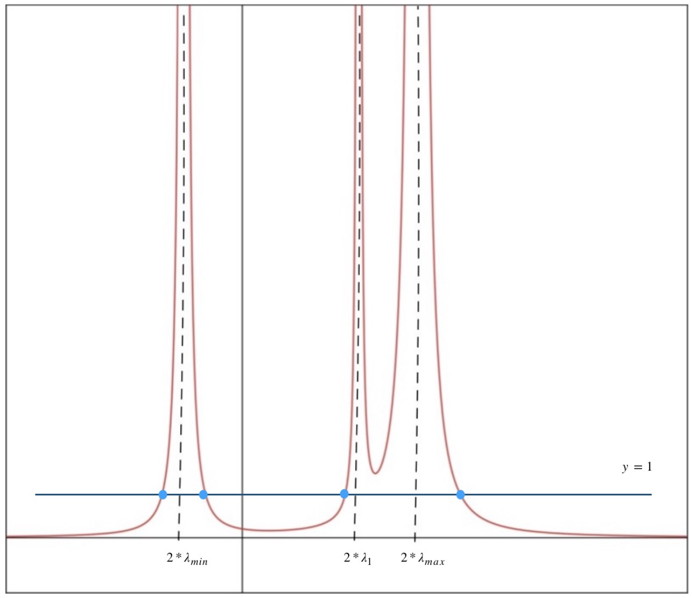

Figure 1: Graph of Function $\Psi(\alpha)$ and its intersection with $y = 1$

And knowing that the desired solution lies in the interval of $(-\infty, 2 * \lambda_{min})$, we can use the bisection method in order to find the root of $\Psi(\alpha) - 1$ related to the λ_{min}. In Figure 1 you can gain a general understanding of the graph of $\Psi(\alpha)$ when $n = 3$.

2.2 Proposed algorithm

Algorithm 1: Our Proposed Algorithm To Find Global Optimizer

Data: $A_{n*n}, b_{n*1}, c_{n*1}$
Result: x^*
$Q\Lambda Q^{-1} \leftarrow Eigenvalue\_Decomposition(A)$;
$\lambda_{min} \leftarrow min(Diag(\Lambda))$;
$root \leftarrow Bisection(\Psi, -\infty, 2 * \lambda_{min}, 1)$; `/* Find an` α `that` $\Psi(\alpha)$ `= 1 and` $\alpha \in (-\infty, 2 * \lambda_{min})$ `*/`
$x^* \leftarrow Produce\_x(root)$; `/* Equation (9) */`
return x*

In the algorithm, min(vector) function finds the minimum entry of a given vector. Produce_x(α) is the equation in (9) that computes the vector **x** based on eigenfactors of **A** and the vector **b**.

2.2.1 Time Complexity of the algorithm

Here, We demonstrate the time and space complexity of the proposed algorithm in the worst case. The bottleneck of the algorithm is eigenvalue decomposition that the best known order of it is $O(n^3 + n^2 \log^2 n \log b)$ for an approximation within 2^{-b}. The order of min() is generally $O(n)$ but when the given argument is sorted the order is $O(1)$ like in our algorithm. The order of Bisection is $O(\log m * n)$, where m is the width of the initial interval. Produce_x(x) function do matrix multiplications, which has the order of $O(n^3)$. To sum up, the order of our algorithm is $O(n^3 + n^2 \log^2 n \log b + \log n^3 + \log m * n + n)$.

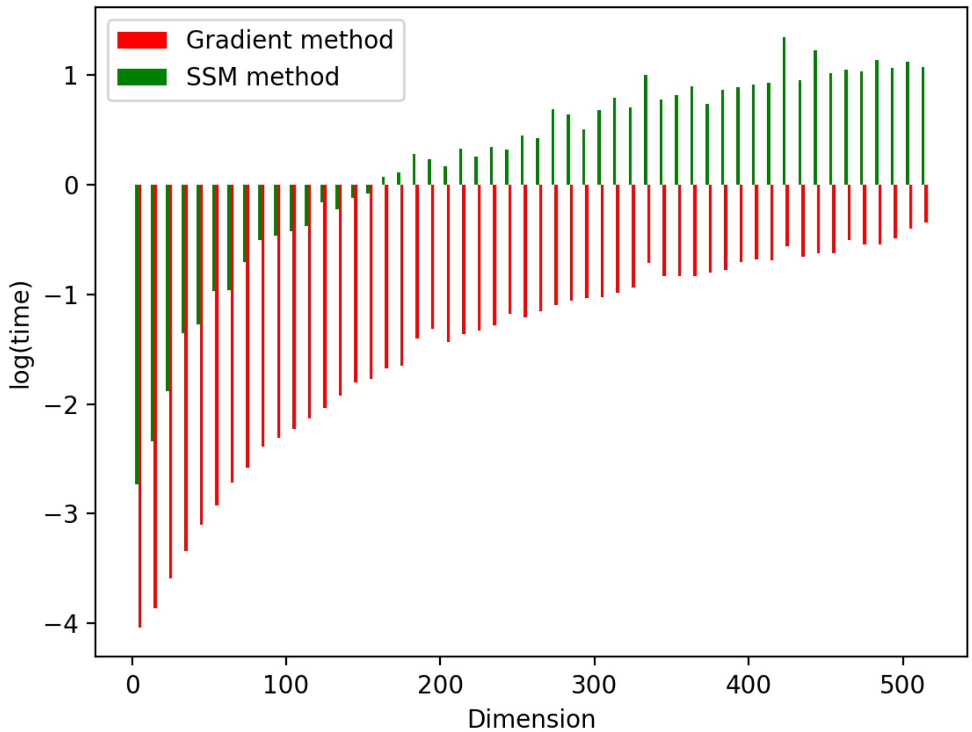

Figure 2: Runtime comparison of our method (Gradient method) with SSM method

2.2.2 Space Complexity of the algorithm

The algorithm works with matrices and vectors, so it should access the data in memory. Since the matrix Q as well as the eigenvalues, λ_i, should be saved in the memory, the space complexity of the algorithm is $O(n^2)$.

3 Runtime Performance Comparison

As a part of our work, we compared the runtime of our program with the program that Hager developed by SSM method (http://users.clas.ufl.edu/hager/papers/Software/). The program that Hager developed is in C language, so for an accurate comparison, we developed our method in C language. We added some snippets of code in the Hager program to compute the runtime of the program. For the eigenvalue decomposition function in our algorithm, we used the Accelerate Framework developed by Apple company. We run both programs on MacBook Pro 2016 (CPU: 2.7 GHz Quad-Core Intel Core i7). The comparison result is shown in figure(2). As you can see, our method performs much faster than the SSM method. Refer to this Github repository for the source codes.

4 Future Works

Following this work, we aim to utilize the quadratic optimization spherically constrained model to solve a subproblem related to constrained clustering in the field of machine learning. Our main intention is to accelerate the performance of some related machine learning algorithms.

References

[1] R. H. Byrd, R. B. Schnabel, and G. A. Schultz, A trust region algorithm for nonlinearly constrained optimization, SIAM J. Numer. Anal., 24 (1987), pp. 1152–1170.

[2] M. R. Celis, J. E. Dennis, and R. A. Tapia, A trust region strategy for nonlinear equality constrained optimization, in Numerical Optimization 1984, P. T. Boggs, R. H. Byrd, and R. B. Schnabel, eds., SIAM, Philadelphia, PA, 1985, pp. 71–82.

[3] M. El-Alem, A global convergence theory for the Celis–Dennis–Tapia trust-region algorithm for constrained optimization, SIAM J. Numer. Anal., 28 (1991), pp. 266–290.

[4] J. J. More , Recent developments in algorithms and software for trust region methods, in Mathematical Programming: State of the Art, A. Bachem, M. Grotschel, and B. Korte, eds., Springer-Verlag, Berlin, 1983, pp. 258–287.

[5] M. J. D. Powell and Y. Yuan, A trust region algorithm for equality constrained optimization, Math. Programming, 49 (1991), pp. 189–211.

[6] Phan, AH., Yamagishi, M., Mandic, D. et al. Quadratic programming over ellipsoids with applications to constrained linear regression and tensor decomposition. Neural Comput Applic 32, 7097–7120 (2020).

[7] W. Menke, Geophysical Data Analysis: Discrete Inverse Theory, Academic Press, San Diego, CA, 1989.

[8] A. Tarantola, Inverse Problem Theory, Elsevier, Amsterdam, The Netherlands, 1987.

[9] W. W. Hager and Y. Krylyuk, Graph partitioning and continuous quadratic programming, SIAM J. Discrete Math., 12 (1999), pp. 500–523.

[10] George E. Forsythe and Gene H. Golub. 1965. Maximizing a second-degree polynomial on the unit sphere. Technical Report. Stanford University, Stanford, CA, USA.

[11] Forsythe, G. E., Golub, G. H. (1965). On the Stationary Values of a Second-Degree Polynomial on the Unit Sphere. Journal of the Society for Industrial and Applied Mathematics, 13(4), 1050–1068.

[12] Burrows, J. W. (1966). Maximization of a Second-Degree Polynomial on the Unit Sphere. Mathematics of Computation, 20(95), 441–444.

[13] Lyle, S., Szularz, M. Local minima of the trust region problem. J Optim Theory Appl 80, 117–134 (1994).

[14] W. Gander, G. H. Golub, and U. von Matt, “A constrained eigenvalue problem,” Linear Algebra and its Applications, vol. 114–115, pp. 815–839, 1989.

[15] Phan, AH., Yamagishi, M., Mandic, D. et al. Quadratic programming over ellipsoids with applications to constrained linear regression and tensor decomposition. Neural Comput Applic 32, 7097–7120 (2020).

[16] Golub, G.H., von Matt, U. Quadratically constrained least squares and quadratic problems. Numer. Math. 59, 561–580 (1991).

[17] Vavasis, Stephen A. and Richard Zippel. “Proving Polynomial-Time for Sphere-Constrained Quadratic Programming.” (1990).

[18] W. W. Hager, “Minimizing a Quadratic Over a Sphere,” SIAM Journal on Optimization, vol. 12, no. 1, pp. 188–208, 2001, doi: 10.1137/S1052623499356071.

[19] Hager, William W., and Soonchul Park. "Global Convergence of SSM for Minimizing a Quadratic over a Sphere." Mathematics of Computation, vol. 74, no. 251, American Mathematical Society, 2005, pp. 1413–23. http://www.jstor.org/stable/4100186

[20] Yi Chen and David Y. Gao. Global solutions to large-scale spherical constrained quadratic minimization viacanonical dual approach, 2013.