Line Search and Gradient Method for Solving Constrained Optimization Problems

Mohammed A. Hasan
Department of Electrical & Computer Engineering
University of Minnesota Duluth
E.mail:mhasan@d.umn.edu

Abstract

The problem of optimizing functionals with linear or orthogonal constraints arises in many applications in engineering and applied sciences. In this paper, a unified framework involving constrained optimization using gradient descent in conjunction with exact or approximate line search is developed. In this framework, the optimality conditions are enforced at each step while optimizing along the direction of the gradient of the Lagrangian of the problem. Among many applications, this paper proposes learning algorithms which extract principal and minor components, reduced rank Wiener filter, and the first few minimum or maximum singular vectors of rectangular matrices. The main attraction of these algorithms is that they are matrix inverse free and thus are computationally efficient for large dimensional problems.

1. Introduction

Constrained optimization over linear and orthogonal (or unitary) constraints arises in many applications in applied physics, control theory, and signal processing. For example, optimization of symmetric Rayleigh quotient over the unit sphere yields the minimum and maximum eigenvalue of a symmetric matrix. In the signal processing field, there are numerous problems that can be formulated as optimization problems over orthogonal constraints. These problems include: minor and principle subspace computation [1], minor and principal subspace tracking [2], adaptive subspace computation, canonical correlation analysis [3], and reduced rank Wiener filtering [4,5]. Numerical methods for constrained optimization can be found in [6, 7].

We present in this paper new methods of computing and solving optimization problems using constrained gradient descent of the Lagrangian in conjunction with exact and approximate line search. Thus these approaches may be considered as constrained iterative gradient descent methods.

2. Problem Formulation

Consider the following optimization problem

$$\text{Optimize} \quad F(x) \text{ subject to } x^T x = I_r, \tag{1}$$

where F is at least twice continuously differentiable real valued function, $x \in \mathbb{R}^{m\times r}$, I_r stands for the identity matrix of size r, and $\{.\}^T$ denotes matrix transpose. Define the Lagrangian as

$$\mathcal{L}(x,\lambda) = F(x) - trace\{(x^T x - I_r)\frac{\lambda}{2}\}, \tag{2}$$

where λ is a matrix of Lagrange multipliers. The necessary condition for optimality is that $\nabla\mathcal{L} = 0$, where

$$\nabla\mathcal{L} = \begin{bmatrix} \nabla_x F(x) - x\lambda \\ x^T x - I_r \end{bmatrix}. \tag{3a}$$

At an optimal solution x, the Lagrange multiplier λ may be expressed as

$$\lambda = x^T \nabla_x F(x). \tag{3b}$$

Substituting this expression in (3a) yields

$$\nabla_x\mathcal{L} = \nabla_x F(x) - xx^T\nabla_x F(x) = (I_r - xx^T)\nabla_x F(x). \tag{4}$$

Note that $I_r - xx^T$ is projection on the sphere defined by the constraint $x^T x = I_r$. Thus in any application of gradient descent, one may use $(I_r - xx^T)\nabla_x F(x)$ as a constrained gradient. Now assume that an approximate solution matrix x is given and assume that λ has been computed as in (3b). Then, for a given nonzero direction matrix h, we are interested in computing $\alpha \in \mathbb{R}^{r\times r}$ so that $\mathcal{L}(x+h\alpha)$ is minimum. Clearly, the Taylor expansion of $\mathcal{L}(x+h\alpha)$ around x is given by

$$\mathcal{L}(x+h\alpha) = F(x) + D_x F(x)h\alpha + \frac{1}{2}\alpha^T h^T \nabla_x^2 F(x)h\alpha + h.o.t. \\ - trace\{x^T x\frac{\lambda}{2} + \alpha^T h^T x\frac{\lambda}{2} + x^T h\alpha\frac{\lambda}{2} + \alpha^T h^T h\alpha\frac{\lambda}{2}\}, \tag{5a}$$

where h.o.t. stands for cubic and higher order terms. Ignoring the higher order terms, it follows that

$$(\frac{\partial\mathcal{L}(x+h\alpha)}{\partial\alpha})^T = D_x F(x)h + h^T\nabla_x^2 F(x)h\alpha - h^T x\alpha - h^T h\alpha\lambda. \tag{5b}$$

When F is quadratic function of x, the Kronecker product may be used to obtain an exact solution to the system; $\frac{\partial \mathcal{L}(x+h\alpha)}{\partial \alpha} = 0$. Specifically, $vec(\alpha)$ is a solution for the linear system:

$$((I_r \otimes h^T D_x^2 F(x) h) - (\lambda^T \otimes h^T h))\mathrm{vec}(\alpha) = \mathrm{vec}(h^T x \lambda - h^T \nabla_x F(x)). \tag{6}$$

Here *vec* stands for the operation of stacking the columns of a matrix into one column, and $\otimes$ denotes the Kronecker product. The expression in (6) can be further simplified by choosing $h = \nabla_x \mathcal{L}(x, \lambda)$, in which case $x^T h = 0$. Once α is computed, then x can be updated as

$$x' = x + h\alpha. \tag{7a}$$

If $r = 1$, i.e., x is a vector, then the scalar α can be obtained as

$$\alpha = -(h^T \nabla_x^2 \mathcal{L} h)^{-1} h^T h, \tag{7b}$$

where $\nabla_x^2 \mathcal{L} = \nabla_x^2 F(x) - \lambda I_r$. If α is chosen to be fixed at each stage, then the above procedure reduces to the constrained gradient descent.

3. Applications

In this section we present a few signal processing applications where the proposed methods can be utilized.

Application 1: Generalized Minimum Subspace Computation

Given two square matrices A and B, the generalized eigenvalue problem consists of finding a nonzero vector x and corresponding λ such that

$$Ax = Bx\lambda.$$

Clearly, $\lambda = \frac{x^T Ax}{x^T Bx}$ and hence the maximum and minimum generalized eigenpairs can be obtained by solving the constrained optimization problem:

$$\text{Optimize } \{x^T Ax \text{ subject to } x^T Bx = 1\}.$$

There are many situations where it is required to obtain only a small number of lowest or largest generalized eigenpairs. One can repeatedly solve the above optimization problem restricting x to the space orthogonal to the previous subspace. It is also possible to compute the r smallest or largest generalized eigenpairs simultaneously by considering the corresponding subspace as a whole. This is especially advisable in situations where the problem has multiple or clustered eigenvalues in which case it is recommended to compute the whole invariant subspace spanned by the corresponding generalized eigenvectors.

The general form of the generalized multi-dimensional extremum subspace computation can be expressed as an optimization problem:

$$\text{Optimize}_x \quad Trace\{x^T Ax, \text{ subject to } x^T Bx = I_r, \tag{8}$$

where A is symmetric and B is positive definite of size m. This problem can be shown to be equivalent to optimizing $trace\{(x^T Ax)(x^T Bx)^{-1}\}$ over all non zero vectors x.

Let $\mathcal{L}(x, \lambda) = \frac{1}{2} trace\{x^T Ax\} - trace\{(x^T Bx - I_r)\frac{\lambda}{2}\}$ be the Lagrangian, then a necessary condition for optimality is that

$$\nabla_x \mathcal{L}(x, \lambda) = \begin{bmatrix} Ax - Bx\lambda \\ x^T Bx - I_r \end{bmatrix} = 0.$$

A steepest descent method would search a new minimum along h, $x' = x + h\alpha$ with a (small) coefficient matrix $\alpha \in \mathbb{R}^{r \times r}$. The matrix α is required to minimize $\mathcal{L}(x + h\alpha, \lambda)$ with respect to $r \times r$ matrix α, where $\lambda = (x^T Ax)(x^T Bx)^{-1}$. The exact solution for α can be computed from the equation

$$h^T Ax + h^T Ah\alpha - h^T Bx\lambda - h^T Bh\alpha\lambda = 0.$$

A natural choice for h is $h = Ax - Bx(x^T Bx)^{-1} x^T Ax$. The updated solution is then given by $x_1 = x + h\alpha$. This solution must then be normalized to a new matrix y so that $y^T By = I$. One approach is to use

$$y = x_1 (x_1^T Bx_1)^{\frac{-1}{2}}.$$

Note that the matrix $(x_1^T Bx_1)$ is of relatively small size $r \times r$. When $r = 1$, i.e., x is one dimensional, then $\alpha = -\frac{h^T (A - \lambda B)x}{h^T (A - \lambda B)h}$.

A Special Case: If $B = I$, then a formulation for computing an orthogonal basis of a principle subspace of rank r can be obtained by solving the optimization problem:

$$\text{Optimize } \{x^T Ax \text{ subject to } x_i^T x_j = \delta_{ij}, \ i, j = 1, \cdots, r.\} \tag{9}$$

A Lagrangian of this problem is

$$\mathcal{L}(x, \lambda) = \frac{1}{2} trace\{x^T Ax\} - \sum_{i=1}^{r} \sum_{j=1}^{r} \frac{\lambda_{ij}}{2} (x_i^T x_j - \delta_{ij}.) \tag{10}$$

It can be shown that $\nabla_{x_j} \mathcal{L} = Ax_j - \sum_{i=1}^{r} x_i \lambda_{ji}$, for $j = 1, \cdots, r$. At optimal solutions, the following relations hold:

$$\lambda_{ij} = x_i^T Ax_j.$$

If $\mathcal{L}(x, \lambda)$ is optimized along the directions h_i by optimizing $\mathcal{L}(x_i + \alpha h_i, \lambda_{ij})$, then α can be obtained by solving the equation:

$$\begin{aligned} &\alpha \sum_{i=1}^{r} h_i^T Ah_i + \frac{1}{2} \sum_{i=1}^{r} h_i^T Ax_i + \frac{1}{2} \sum_{i=1}^{r} x_i^T Ah_i - 2\alpha \sum_{i=1}^{r} \sum_{j \neq i}^{r} \frac{\lambda_{ij}}{2} h_i^T h_j \\ &- \sum_{i=1}^{r} \sum_{j \neq i}^{r} \frac{\lambda_{ij}}{2} h_i^T x_j - \sum_{i=1}^{r} \sum_{j \neq i}^{r} \frac{\lambda_{ij}}{2} x_i^T h_j - 2\alpha \sum_{i=1}^{r} \frac{\lambda_{ii}}{2} h_i^T h_i \\ &- 2 \sum_{i=1}^{r} \frac{\lambda_{ii}}{2} h_i^T x_i = 0. \end{aligned} \tag{11}$$

The above development can be summarized in the following algorithm.

Algorithm 1

1. Let $x_1(0), x_2(0)...., x_r(0)$ be a random set of orthogonal unit vectors
2. For $j = 1, 2, \cdots, r$, let $\lambda_{ij} = x_j^T(k)Ax_j(k)$ and set $h_j = \nabla_{x_j(k)}\mathcal{L} = Ax_j(k) - \sum_{i=1}^{r} x_i(k)\lambda_{ji}$, for $j = 1, \cdots, r$.
3. Solve (11) for α and set $x_j(k+1) = x_j(k) + h_j\alpha$.
4. Let $B = [x_1(k+1)\ x_2(k+1)\ ...\ x_r(k+1)]$ and use Gram-Schmidt process to orthogonalize B.
5. Stop if convergence is satisfactory, otherwise go to 2.

Note that Step 4 can accomplished in many different ways. For example $B(B^TB)^{-\frac{1}{2}}$ is an orthogonal matrix. Another way is to compute the QR factorization of B so that $B = QR$, where Q is an orthogonal matrix.

Application 2: Singular Value Decomposition

Singular value decomposition (SVD) is one of the most important tools of matrix algebra that has been applied to a number of areas including principal component analysis, canonical correlation, the determination of the MoorePenrose generalized inverse, and low rank approximation of matrices. Many computational aspects of SVD are discussed in [8].

Let the singular triplet (u, v, σ) denote the right, left singular vectors corresponding to the singular value σ of a given a matrix $A \in \mathcal{R}^{M\times N}$. The maximum and minimum singular triplets (u, v, σ) can be obtained as a solution of the optimization problem

$$\text{Optimize } \{\frac{u^TAv}{\sqrt{u^Tu}\sqrt{v^Tv}} \quad u \neq 0, v \neq 0,\} \tag{12a}$$

or equivalently,

$$\text{Optimize } \{u^TAv \ \textit{subject to}\ u^Tu = 1, v^Tv = 1\} \tag{12b}$$

Let

$$\mathcal{L} = u^TAv - (u^Tu - 1)\frac{\lambda_1}{2} - (v^Tv - 1)\frac{\lambda_2}{2} \tag{13}$$

be the Lagrangian, then a necessary condition for optimality is that

$$\nabla_x\mathcal{L}(u, v, \lambda_1, \lambda_2) = \begin{bmatrix} Av - u\lambda_1 \\ A^Tu - v\lambda_2 \end{bmatrix} = 0. \tag{14}$$

If $(u, v, \lambda_1, \lambda_2)$ is an optimal solution, then

$$\begin{aligned} \lambda_1 &= u^TAv \\ \lambda_2 &= v^TA^Tu. \end{aligned} \tag{15}$$

Now given an approximate singular vectors u and v, let $h_1 = Av - u(u^TAv) = (I - uu^T)Av$ and $h_2 = A^Tu - v(v^TA^Tu) = (I - vv^T)A^Tu$ be descent directions. A better approximation can be obtained by minimizing $\mathcal{L}(u + \alpha h_1, v + \alpha h_2)$ over α

$$\begin{aligned} \mathcal{L}(u + \alpha h_1, v + \alpha h_2) &= (u + \alpha h_1)^TA(v + \alpha h_2) \\ &- (u + \alpha h_1)^T(u + \alpha h_1)\frac{\lambda_1}{2} - (v + \alpha h_2)^T(v + \alpha h_2)\frac{\lambda_2}{2}. \end{aligned} \tag{16}$$

Note that $v^Th_2 = u^Th_1 = 0$. It can be shown that

$$\alpha = -\frac{h_1^Th_1 + h_2^Th_2}{2h_1^TAh_2 - h_1^Th1\lambda_1 - h_2^Th_2\lambda_2}$$

Hence the updated singular vectors are given by:

$$\begin{aligned} u' &= u + \alpha h_1, \\ v' &= v + \alpha h_2. \end{aligned}$$

This process can be repeated until convergence.

There are different applications where only right or left singular vectors are required. Assume that we would like to compute the maximum singular vector u. In this case, u is an eigenvector of AA^T, i.e., $AA^Tu = \sigma^2u$ for some σ. Thus u is the solution of the problem

$$\text{Optimize } \{u^TAA^Tu \text{ subject to } u^Tu = 1,\}$$

which can be solved using the method outlined in Application 1.

Remark 1: If u is a left singular vector, then A^Tu is an eigenvector of A^TA. This can be shown by multiplying both sides of $AA^Tu = \lambda u$ by A^T so that $(A^TA)A^Tu = \lambda A^Tu$. This means that if u is an eigenvector of $AA^Tu = \lambda u$, then A^Tu is an eigenvector of A^TA, i.e., $v = \frac{A^Tu}{\sqrt{u^TAA^Tu}}$. The significance of this observation is that A^TA is $N \times N$ matrix while AA^T is $M \times M$. The following algorithm exploit the above remark to extract the largest singular value and left singular vector of A. The most significant feature of this method is that it is inverse free and relies only on matrix-vector multiplication.

Algorithm 2

The input is a symmetric matrix A and the output is the largest singular value and singular vector of A.

```
$Step\ 1:$  Given a unit vector $0 \neq \mathrm{x}(1)$, randomly generated
$Step\ 2: for\ k = 1, 2, \cdots\ compute$
    $u(k) = Ax(k)$
    $\lambda(k) = u(k)^Tu(k)$
    $h(k) = A^Tu(k) - x(k)\lambda(k)$
    $v(k) = Ah(k)$
      $a_0 = -2(h^Th)(u^Tv),$
      $a_1 = v^Tv - (u^Tu)(h^Th),$
      $a_2 = 2u^Tv$
      $\alpha = \frac{-a_1 - \sqrt{a_1^2 - a_0a_2}}{a_0}$
$Step\ 3: y(k) = x(k) + h(k)\alpha$
$Step\ 4: x(k+1) = \frac{y(k)}{\sqrt{y(k)^Ty(k)}}$
```

Remark 2: If α in Algorithm 2 is computed as $\alpha = \frac{-a_1+\sqrt{a_1^2-a_0a_2}}{a_0}$, then $x(k)$ converges to the minimum singular vector of A.

Application 3: Reduced Rank Wiener Filtering

Given two signals $x(n)$ and $y(n)$, the aim here is to find a rank r minimizer W_r that minimizes

$$Q_{xx}(W_r) = E(||x(n) - W_r y(n)||^2) \quad W_r \text{ has rank r.}$$

The computation of reduced rank Wiener filter $W_r = \sigma u v^T$ of rank 1 can be posed as a constrained maximization problem as

$$\text{Maximize} \quad \{u^T R_{xy} v : u^T u = 1,\ v^T R_{yy}^2 v = 1\}. \tag{17}$$

The Lagrangian of this problem is given by

$$\mathcal{L} = u^T R_{xy} v - (u^T u - 1)\frac{\lambda_1}{2} - (v^T R_{yy}^2 v - 1)\frac{\lambda_2}{2}$$

The optimality conditions are

$$\begin{aligned} R_{xy} v - \lambda_1 u &= 0 \\ R_{yx} u - \lambda_2 R_{yy}^2 v &= 0, \end{aligned}$$

for some Lagrange multipliers λ_1 and λ_2. There are many versions for the maximization problem (17). For example $u^T R_{xy} v$ can be replaced by $d(u^T R_{xy} v)^l$, where $d = 1$ if $l > 0$ and $d = -1$ if $l < 0$. Another choice is $\ln(u^T R_{xy} v)$.

Algorithm 3:

1. Given initial guesses u, v, compute

$$\begin{aligned} \lambda_1 &= u^T R_{xy} v \\ \lambda_2 &= \frac{v^T R_{yx} u}{v^T R_{yy}^2 v} \end{aligned}$$

2. Set

$$\begin{aligned} h &= R_{xy} v - \lambda_1 u \\ k &= R_{yx} u - \lambda_2 R_{yy}^2 v \end{aligned}$$

3. Compute $\alpha = -\frac{h^T h + k^T k}{2h^T R_{xy} k - \lambda_1 h^T h - \lambda_2 k^T R_{yy}^2 k}$.
4. Update u and v so that

$$\begin{aligned} u' &= u + h\alpha \\ v' &= v + k\alpha \end{aligned}$$

5. Repeat step 2-4 until convergence.

4. Conclusion

In this paper we proposed a number of computational tools for solving optimization problems over spheres. These include, among many other problems, reduced rank Wiener filters, reduced rank principal and minor component analysis, and singular value decomposition. The main motivation of this work is the desire to solve linear systems of equations arising from the necessary conditions of optimality of these problems without inverting large scale matrices. We should also emphasized that the derivation in Application 1 is applied to non-definite symmetric matrices. Additionally, Algorithm 2 can be modified so that all singular vectors of a given matrix can be computed. Simulations have been conducted to examine the performance of each of the proposed methods, however, we did not include them here due to space limitation. Finally, the proposed approaches can be extended to complex-valued functions with minor modifications.

References

[1] S. Y. Kung, K. I. Diamantaras, J. S. Taur, ”Adaptive Principal component EXtraction (APEX) and applications,” Signal Processing, IEEE Transactions on, Volume: 42 Issue: 5, May 1994, Page(s): 1202-1217.

[2] Dowling, E.M.; Ammann, L.P.; DeGroat, R.D.; ”An adaptive TQR-SVD for angle and frequency tracking,” Signals, Systems and Computers, 1992. 1992 Conference Record of The Twenty-Sixth Asilomar Conference on , 26-28 Oct. 1992, vol.1, pp. 555-560.

[3] T. W. Anderson, An Introduction to Multivariate Statistical Analysis, 2nd ed. New York: Wiley, 1984.

[4] Goldstein, J.S.; Reed, I.S.; Scharf, L.L.; ”A multistage representation of the Wiener filter based on orthogonal projections,” Information Theory, IEEE Transactions on, Volume: 44 Issue: 7 , Nov. 1998, pp.2943-2959.

[5] L. L. Scharf, Statistical Signal Processing, Detection, Estimation, and Time Series Analysis, Addison-Wesley, 1991.

[6] 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.

[7] 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.

[8] G. H. Golub and C. F. Van Loan, Matrix Computations, 2nd ed. Baltimore, MD: Johns Hopkins Univ. Press, 1989.