Gradient Projection for Optimizing a Function on Standard Simplex

Youwei Liang*

June 15, 2020

Abstract

When optimizing a function with standard simplex constraint using active set method, we need to project the gradient of the function to a hyperplane through the origin with sign constraints. We propose a novel algorithm to efficiently project the gradient for this purpose.

1 Introduction

Given a function \( f: \mathbb{R}^n \to \mathbb{R}^m \), suppose we want to minimize \( f \) on a constant-sum simplex.

\[
\begin{align*}
\text{minimize} & \quad f(x) \\
\text{subject to} & \quad e^\top x = c, \; x \geq 0
\end{align*}
\]

where \( e \) is an all-one vector and \( c \) is a constant. When \( c = 1 \), the constraint is a standard simplex (a.k.a. probability simplex). Intrinsically this leads to combinatorial optimization since we need to decide which elements in \( x \) should be 0 and which should be greater than 0. A special case is \( f \) is a quadratic function, which arises in various applications such as spectral clustering (Liang et al., 2019). A popular iterative approach to solve the problem is the active set method (Nocedal and Wright, 2006) with gradient projection (Birgin et al., 2000; Cristofari et al., 2017; Dai and Fletcher, 2006; di Serafino et al., 2018). An active set is a set determining which elements in \( x \) are fixed to 0 and which elements are free variables. With an active set, we optimize \( f \) with respect to free variables as if there were no constraint. A general optimization method working with active set method is gradient descent, which seeks to minimize \( f \) by taking a step along the opposite direction of the gradient of \( f \). In constrained optimization, however, directly taking a step with gradient might cause \( x \) to violate the constraints. Therefore, we need to project the gradient to a space where the constraints hold. Let \( g = \nabla f \) be the gradient of \( f \), then we want to take a step along the projected gradient \( \tilde{g} \), which should be as close to \( g \) as possible while the constraints (2) are satisfied. Then we have \( e^\top (x - \alpha \tilde{g}) = c, \; (x - \alpha \tilde{g}) \geq 0 \), where \( \alpha > 0 \) is the step size. Under the framework of active set method, some elements in \( x \) are fixed.

*College of Mathematics and Informatics, South China Agricultural University, Guangzhou 510642, China. Email: liangyouwei1@gmail.com
to 0 while other elements are free variables that can be changed. Therefore, \( \tilde{g} \) must satisfy \( e^\top \tilde{g} = 0 \) and \( \tilde{g}_i \leq 0 \) for \( i \in G \) where \( G \) is the active set. Then we are interested in projecting a vector \( g \) onto the hyperplane \( e^\top x = 0 \) with sign constraints on some elements of \( g \). Then the gradient projection is formulated as an optimization problem.

\[
\begin{align*}
\text{minimize} & \quad \|x - g\|^2 \\
\text{subject to} & \quad e^\top x = 0 \\
& \quad x_i \leq 0, \quad i \in G
\end{align*}
\] (3)

where \( G \) is an index set (the active set). Some similar projection problems where the inequalities (5) are imposed on all variables have been addressed by many authors (Duchi et al., 2008; Chen and Ye, 2011; Wang and Carreira-Perpiñán, 2013; Wang and Lu, 2015; Condat, 2016), while how to solve the problem with inequality (5) imposed on partial variables is not fully investigated. In this paper, we analyze the properties of the solution to Problem (3) and present an efficient algorithm based on our analysis.

2 Gradient Projection

2.1 Analysis of the Solution

Without loss of generality, we assume that that elements in \( g \) is in descending order such that \( g_1 \geq g_2 \geq \cdots \geq g_n \), where \( n \) is the number of elements in \( g \). Let \( I = \{1, \ldots, n\} \). An important property of the solution to the standard simplex projection is that it preserves the order of the elements in the vector being projected (Duchi et al., 2008, Lemma 1). However, when the vector being projected has sign constraints on only some of its elements, the property no long holds. Instead, we have the following lemma.

**Lemma 1.** Let \( x^* \) be the optimal solution of problem. Let \( I^+ \) and \( I^- \) denote the index set of the non-negative and non-positive elements of \( x^* \) respectively. If \( i, j \in I^+ \), \( i \leq j \), we have \( x^*_i \leq x^*_j \). Similarly, if \( i, j \in I^- \), \( i \leq j \), we have \( x^*_i \leq x^*_j \).

**Proof.** Suppose \( x \) is the minimizer of problem (3) and \( g_i > g_j \). Suppose \( x_i < x_j \leq 0 \), i.e., \( x_i, x_j \in I^- \). Switch \( x_i \) and \( x_j \) to get a new solution \( \tilde{x} \) where \( \tilde{x}_i = x_j, \tilde{x}_j = x_i \) and \( \tilde{x}_k = x_k \) for \( k \neq i, j \). Note that \( \tilde{x} \) satisfies the constraints (4) and (5). Then

\[
\Delta f(x) = \|\tilde{x} - g\|^2 - \|x - g\|^2 = (x_j - g_i)^2 + (x_i - g_j)^2 - (x_i - g_i)^2 - (x_j - g_j)^2 = 2(x_j - x_i)(g_j - g_i) < 0
\]

This contradicts that \( x \) is the minimizer. Thus we conclude \( x_i \geq x_j \). The analysis for \( x_i, x_j \in I^+ \) is similar and omitted here. Note that if \( g_i = g_j \), \( \Delta f(x) = 0 \) and thus there might be multiple optimal solutions. We choose to adopt the optimal solution that obeys Lemma 1. \( \square \)
Noting that minimizing \(\|x - g\|^2\) is equivalent to minimizing \(x^\top x - 2x^\top g\), we construct a Lagrangian function \(L(x, \lambda, \mu) = x^\top x - 2x^\top g - \lambda e^\top x + \mu^\top x\), where \(\mu\) is a vector defined as \(\mu_i = 0\) if \(i \in G^c = \{k \mid 1 \leq k \leq n, k \notin G\}\). KKT conditions imply
\[
\frac{\partial L}{\partial x} = 2x - 2g - \lambda e + \mu = 0 \quad (6)
\]
\[
e^\top x = 0 \quad (7)
\]
\[
\mu \geq 0 \quad (8)
\]
\[
\mu_i x_i = 0, \quad i \in G \quad (9)
\]

Let \(J\) denote the index set of the non-zero elements of \(x^*\). For any \(j \in J, x_j^* \neq 0\), by (9) and the definition of \(\mu\) (note that it is possible that \(j \in G^c\)), we have \(\mu_j = 0\). By (6),\[
2x_j^* - 2g_j - \lambda = 0.
\]
Since \(\sum_{j \in J} x_j^* = 0\), \(\sum_{j \in J}(2x_j^* - 2g_j - \lambda) = \sum_{j \in J}(-2g_j - \lambda) = 0\). Let \(\bar{g}_J = \sum_{j \in J} g_j / m\), i.e., the average of \(g_J = \{g_j \mid j \in J\}\). Thus
\[
\lambda = -\frac{2\sum_{j \in J} g_j}{m} = -2\bar{g}_J \quad (10)
\]
where \(m\) is the number of elements in \(J\). For all \(i \in G \setminus J\), we have \(x_i = 0\). Thus by (6) we have
\[
\mu_i = 2g_i + \lambda = 2g_i - \frac{2\sum_{j \in J} g_j}{m} = 2(g_i - \bar{g}_J) \quad (11)
\]
For all \(i \in J \cup G^c\), we have \(\mu_i = 0\). Thus by (6) we have
\[
x_i^* = g_i + \frac{\lambda}{2} = g_i - \frac{\sum_{j \in J} g_j}{m} = g_i - \bar{g}_J \quad (12)
\]
Let \(\bar{g}_I = \sum_{i \in J} g_i / n\), i.e., the average of \(g_I\) (which is also \(g\)). Let \(x\) be defined as \(x_i = g_i - \bar{g}_I\) for \(i \in I\). If for \(i \in G, x_i \leq 0\), then \(x\) is the optimal solution since it is the optimal solution to the problem without inequality constraints (5) (this can be checked by the optimal conditions for the equality-constrained problem).

If for some \(i \in G, x_i > 0\), let \(A = \{i \in G \mid x_i > 0\}\) and \(a = \max(A)\). Note that
\[
x_a = g_a - \bar{g}_I > 0 \quad (13)
\]
Since \(a \in G\), only one of the two situation can happen: \(x_a^* = 0\) or \(x_a^* < 0\).

Suppose \(x_a^* < 0\), then \(a \in J\) and by (12) we have \(g_a - \bar{g}_I < 0\). Thus \(\bar{g}_J > \bar{g}_I\). Let \(F = \{i \mid 1 \leq i \leq n, g_i < \bar{g}_I\}\), i.e., the index set of \(g_i\)’s which are smaller than the average of \(g_I\). \(\bar{g}_J > \bar{g}_I\) implies that for some \(f \in F, f \in J^c = I \setminus J\). Thus, \(x_f^* = 0\). Note that \(f, a \in I^c\). By (13) and definition of \(F\), \(g_f < \bar{g}_I < g_a\), by Lemma 1, \(x_f^* \leq x_a^*\). But \(x_f^* = 0 > x_a^*\) causes a contradiction. Thus it is impossible that \(x_a^* < 0\) and only \(x_a^* = 0\) can be true. In the analysis we set \(a = \max(A)\) only for ease of introducing our algorithm. In fact for all \(i \in A\), \(x_i^* = 0\).

Since \(x_a^* = 0\), we can remove \(g_a\) from \(g\) and construct a reduced problem. Formally, let \(I' = I \setminus \{a\}\) and \(G' = G \setminus \{a\}\) be the reduced index sets. The reduced problem is
\[
\min_x \|x_{I'} - g_{I'}\|^2 \quad (14)
\]
\[
s.t. \quad e^\top x_{I'} = 0 \quad (15)
\]
\[
x_i \leq 0, \quad i \in G' \quad (16)
\]
Repeating the same analysis for \( g'_I \) and \( x'_I \). Either the inequality constraints (16) are satisfied or a zero element in \( x^*_I \) is determined. Repeat the procedures until \( x^* \) is found.

2.2 An Algorithm

At last, we present an algorithm for solving problem (3) based on our analysis.

\begin{algorithm}
\textbf{Algorithm 1} Gradient Projection onto Simplex

\textbf{Input:} \( g, G, n \)

\textbf{Output:} \( x \)

1: Sort \( g \) into descending order such that \( g_1 \geq g_2 \geq \cdots \geq g_n \). Reset the indices in \( G \) to match the indices of sorted \( g \). And sort \( G \) into ascending order.

2: \( a = \text{mean}(g), \ s = \text{sum}(g) \quad \triangleright \text{the mean and sum of } g \)

3: \( m = |G| \quad \triangleright \text{number of inequality constraints} \)

4: \( H = \{ \} \)

5: for \( i = 1, \ldots, m \) do

6: \( \text{if } g_{G_i} > a \text{ then} \)

7: \( s \leftarrow s - g_{G_i} \)

8: \( a \leftarrow s/(n-i) \)

9: \( H \leftarrow H \cup \{G_i\} \quad \triangleright \text{add } G_i \text{ to the index set } H \)

10: \( \text{else} \)

11: \( \text{Break} \)

12: \( \text{end if} \)

13: end for

14: \( x_i \leftarrow g_i - a \text{ for } i \in I \setminus H \)

15: \( x_i \leftarrow 0 \text{ for } i \in H \)

16: Reorder the elements in \( x \) to match the original order of \( g \) before sorting

\end{algorithm}

In the for loop in Algorithm 1, since \( g_{G_i} > a \) and \( g_{G_i} \) is removed from the sum \( s \), \( a \) is decreasing during the procedure. When the algorithm terminates, \( J = I \setminus H \) and \( a \) equals \( \bar{g}_J \). Thus for \( i \in H \), \( g_i > a = \bar{g}_J \), and the KKT multiplier for \( x^*_i \) is \( \mu_i = 2(g_i - \bar{g}_J) > 0 \), satisfying the KKT condition (8).

In Algorithm 1, the computation bottleneck lies in the sorting of the input vector, and thus the time complexity is the same as that of the sorting algorithm. Many sorting algorithms have \( O(n \log n) \) time complexity, and thus Algorithm 1 can be run efficiently in \( O(n \log n) \) time.

References

E. G. Birgin, J. M. Martínez, and M. Raydan. Nonmonotone spectral projected gradient methods on convex sets. \textit{SIAM Journal on Optimization}, 10(4):1196–1211, 2000.

Y. Chen and X. Ye. Projection onto a simplex. \textit{CoRR}, abs/1101.6081, 2011. URL \url{http://arxiv.org/abs/1101.6081}. 
L. Condat. Fast projection onto the simplex and the $\ell_1$ ball. *Math. Program.*, 158 (12):575585, July 2016. ISSN 0025-5610. doi: 10.1007/s10107-015-0946-6. URL https://doi.org/10.1007/s10107-015-0946-6.

A. Cristofari, M. De Santis, S. Lucidi, and F. Rinaldi. An active-set algorithmic framework for non-convex optimization problems over the simplex. *arXiv preprint arXiv:1703.07761*, 2017.

Y.-H. Dai and R. Fletcher. New algorithms for singly linearly constrained quadratic programs subject to lower and upper bounds. *Mathematical Programming*, 106(3):403–421, 2006.

D. di Serafino, G. Toraldo, M. Viola, and J. Barlow. A two-phase gradient method for quadratic programming problems with a single linear constraint and bounds on the variables. *SIAM Journal on Optimization*, 28(4):2809–2838, 2018.

J. Duchi, S. Shalev-Shwartz, Y. Singer, and T. Chandra. Efficient projections onto the $l_1$-ball for learning in high dimensions. In *Proceedings of the 25th International Conference on Machine Learning*, ICML 08, page 272279, New York, NY, USA, 2008. Association for Computing Machinery. ISBN 9781605582054. doi: 10.1145/1390156.1390191. URL https://doi.org/10.1145/1390156.1390191.

Y. Liang, D. Huang, and C.-D. Wang. Consistency meets inconsistency: A unified graph learning framework for multi-view clustering. In *Proceedings of the IEEE International Conference on Data Mining*, 2019.

J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer New York, 2006. URL https://doi.org/10.1007/978-0-387-40065-5.

W. Wang and M. Á. Carreira-Perpiñán. Projection onto the probability simplex: An efficient algorithm with a simple proof, and an application. *CoRR*, abs/1309.1541, 2013. URL http://arxiv.org/abs/1309.1541.

W. Wang and C. Lu. Projection onto the capped simplex. *CoRR*, abs/1503.01002, 2015. URL http://arxiv.org/abs/1503.01002.