A simple yet efficient algorithm for multiple kernel learning under elastic-net constraints

Luca Citi LCITI@IEEE.ORG
School of Computer Science and Electronic Engineering
University of Essex
Colchester, CO4-3SQ, UK

Editor: N/A

1. Introduction

This report presents an algorithm for the solution of multiple kernel learning (MKL) problems with elastic-net constraints on the kernel weights. Please see Sun et al. (2013) and Yang et al. (2011) for a review on multiple kernel learning and its extensions. In particular Yang et al. (2011) introduced the generalized multiple kernel learning (GMKL) model where the kernel weights are subject to elastic-net constraints.

While Xu et al. (2010) presents an elegant algorithm to solve MKL problems with L_1-norm and L_p-norm ($p \geq 1$) constraints, a similar algorithm is lacking in the case of MKL under elastic-net constraints. In fact, the algorithm that Yang et al. (2011) propose for the solution of their GMKL model is implemented as an extensive piece of code that depends on large and possibly commercial libraries (e.g. MOSEK).

The algorithm presented in this report provides an extremely simple and efficient solution to the elastic-net constrained MKL (GMKL) problem. Because it can be implemented in few lines of code and does not depend on external libraries (except a conventional L_2-norm SVM solver), it has a wider applicability and can be readily included in existing open-source machine learning libraries.

2. Methods

2.1 Formulation of the generalized MKL problem

Given a set of labelled training data $\mathcal{D} = \{x_\ell, y_\ell\}_{\ell=1}^N$ where $x_\ell \in \mathcal{X}$ and $y_\ell \in \{-1, +1\}$, the learning problem corresponding to a generalized MKL classifier with elastic-net constraints (Yang et al., 2011) can be formulated as

$$\underset{\substack{\theta\in\Theta, b\in\mathbb{R},\\ \{f_k\in\mathcal{H}_k\}}}{\text{minimize}} \; \frac{1}{2}\sum_k \frac{\|f_k\|_2^2}{\theta_k} + C\sum_\ell L\Big(\sum_k f_k(x_\ell) - b, y_\ell\Big), \tag{1}$$

where $\mathcal{H}_k$ is the RKHS associated with the k-th kernel, $L(\cdot)$ is the hinge loss function, and

$$\Theta = \{\theta \in \mathbb{R}_{++}^Q : \eta\|\theta\|_1 + (1-\eta)\|\theta\|_2^2 \leq 1\} \tag{2}$$

represents the elastic-net constraint on the kernel weights. For simplicity of notation, here and in the following all summations involving ℓ go from 1 to N (the number of training instances) while those involving i, k go from 1 to Q (the number of kernels). Note that the minimization problem in (1) is a convex optimization problem because: a) the function to be minimized is jointly convex in its parameters θ, $\{f_k\}$, and b (Rakotomamonjy et al., 2007); and b) the search space is convex, in particular the elastic-net constraint Θ. It has been previously shown (Yang et al., 2011) that (1) always attains a global minimum at a point where the the elastic-net constraint is tight.

The method presented in this paper exploits this fact and seeks to minimize the following problem directly:

$$\underset{\substack{\theta\in\Omega, b\in\mathbb{R},\\ \{f_k\in\mathcal{H}_k\}}}{\text{minimize}} \; \frac{1}{2}\sum_k \frac{\|f_k\|_2^2}{\theta_k} + C\sum_\ell L\Big(\sum_k f_k(x_\ell) - b, y_\ell\Big), \tag{3}$$

where

$$\Omega = \{\theta \in \mathbb{R}^Q_{++} : \eta\|\theta\|_1 + (1-\eta)\|\theta\|_2^2 = 1\}. \tag{4}$$

A possible disadvantage of this approach is that the problem becomes non-convex and could in principle admit local non-global minimizers. This phenomenon occurs, for example, for the problem: $\text{minimize}_{x,y}\,(x-2)^2 + 5y^2$ subject to $x, y \in \mathbb{R}$, $x^2 + y^2 \leq 1$. In this form, it requires the minimization of a convex function on a convex domain and this guarantees against the existence of local minima other than the global minimizer at the point $(1, 0)$. Conversely, if we replace the inequality constraint with equality — i.e. we restrict the feasible region to the (non-convex) boundary of the unit circle — the function presents an additional local minimum in $(-1, 0)$. As a consequence, iterative methods such as gradient or coordinate descent can get stuck at this local minima and never approach the global optimum.

We now show that this unfortunate situation cannot occur for our elastic-net MKL problem. We start by noticing that the Lagrangian $\mathcal{L}$ associated with the two optimization problems, (3) and (1), is identical with the only difference that the Lagrange multiplier λ corresponding to the elastic-net constraint is defined in $\mathbb{R}$ for the constraint Ω, and on $\mathbb{R}_+$ for Θ. A local optimizer $(\theta', \{f'_k\}, b')$ for (3) must satisfy the necessary conditions for local optimality, including:

$$0 = \nabla_{\theta_k}\mathcal{L}(\theta', \ldots, \lambda', \ldots) = \eta\lambda' + 2(1-\eta)\theta'_k\lambda' - \|f_k\|_2^2\,(\theta')^{-2}, \tag{5}$$

which, incidentally, implies that $\lambda' \geq 0$. This inequality, together with the remaining conditions on the Lagrangian and the fact that the problem in (1) is convex, represent sufficient conditions for the global optimality of this local minimum on Θ and, as a result, on Ω. This shows that a local minimizer on Ω is necessarily also globally optimal.

2.2 Overview of the proposed algorithm

The approach proposed in this manuscript consists of a two-step block coordinate descent alternating between the optimization of the SVM classifiers and the optimization of the kernel weights.

In the first step, problem (3) is minimized with respect to $\{f_k\}$ and b for fixed values of the kernel weights. As previously noted by others (Rakotomamonjy et al., 2007; Xu et al., 2010; Yang et al., 2011), this problem is equivalent to the standard SVM problem with a composite kernel matrix $K(x_\ell, x_{\ell'}) = \sum_k \theta_k K_k(x_\ell, x_{\ell'})$ and can be efficiently solved using existing SVM solvers. The second step consists in minimizing (3) for $\theta \in \Omega$ while keeping $\{f_k\}$ and b constant. Since the only term that depends on θ is the regularizer, we can define $\beta_k = \|f_k\|_2^2$ and attack this sub-problem as an instance of the more general problem of minimizing a positive-weighted sum of reciprocals bound to elastic-net constraints:

$$\begin{aligned} \underset{\theta \in \mathbb{R}^Q_{++}}{\text{minimize}} \quad & \sum_k \frac{\beta_k}{\theta_k} \\ \text{subj. to} \quad & \eta\|\theta\|_1 + (1-\eta)\|\theta\|_2^2 = 1 \end{aligned} \tag{6}$$

where $\beta_k > 0$, $\forall k$. In the special case $\eta = 1$, the constraint is a lasso constraint and the problem has a straightforward closed-form solution (Xu et al., 2010). In this manuscript, a novel, simple and efficient algorithm for the solution of this optimization problem when $\eta \in [0, 1)$ is presented. Since the proposed solution to this sub-problem represents the novelty and main contribution of this paper, the remaining of this section will be entirely devoted to explaining this algorithm in detail.

2.3 Re-scaled objective function

This and the following sections pertain to the solution of the optimization problem (6) and, in a sense, they abstract from the original MKL learning problem. Please notice that in the following x and y simply denote vectors in $\mathbb{R}^Q_{++}$ (rather than training instances and labels like in the previous sections).

As a first step in attacking the problem (6), we introduce an equivalent optimization problem. By the change of variable $\theta = x/s(x)$, the original problem (6) can be transformed into the following equivalent one:

$$\underset{x \in \mathbb{R}^Q_{++}}{\text{minimize}} \quad h(x) = s(x)\, g(x) \tag{7}$$

where

$$g(x) = \sum_i \frac{\beta_i}{x_i}, \tag{8}$$

$$s(x) = \frac{\eta \left(\sum_i x_i \right) + \sqrt{\eta^2 \left(\sum_i x_i \right)^2 + 4\,(1-\eta) \sum_i x_i^2}}{2}, \tag{9}$$

and $\beta \in \mathbb{R}^Q_{++}$, $\eta \in [0, 1)$. The new problem implicitly accounts for the elastic-net equality constraint (Ω) by means of the re-scaling function s which re-normalizes any $x \in \mathbb{R}^Q_{++}$ such that the vector $\theta = x/s(x)$ satisfies the constraint.

Our new task is therefore to find a global minimum of h in the positive orthant. Although h is not a convex function, we will be able to prove a weaker result — pseudoconvexity —

which is still very useful in practice because critical points of pseudoconvex functions are also global minima. In order to show that h is pseudoconvex, the following theorem and its corollary are introduced (proofs in Appendix A.1).

Theorem 1 *Let $A \subseteq \mathbb{R}^n$ be an open convex cone and $g, s : A \to \mathbb{R}_{++}$ be differentiable convex functions such that $s(cx) = cs(x)$ and $g(cx) = g(x)/c$ for all $c \in \mathbb{R}_{++}$ and $x \in A$. Their pointwise product $h(x) = s(x)\, g(x)$ is a pseudoconvex function in A.*

Corollary 2 *Under the conditions of Theorem 1, points $x = cx^*$ with $c \in \mathbb{R}_{++}$ and x^* satisfying $\nabla s(x^*) = -\nabla g(x^*)$ are global minima for the function h, where it takes value $h(cx^*) = s^2(x^*) = g^2(x^*)$. If at least one of s or g is strictly convex, then x^* is unique.*

The functions s and g defined in (8) and (9) satisfy the requirements for Theorem 1. In fact, they are positive-valued differentiable functions in the positive orthant $\mathbb{R}^Q_{++}$ (which is an open convex cone) and they can be shown to be strictly convex through some simple calculus. As a result of Theorem 1, h is pseudoconvex which gives us hope that it may be efficiently minimized by using nonlinear programming algorithms (Bertsekas, 1999).

2.4 Iterative minimization algorithm

The problem (7) can be minimized using the following novel iterative algorithm. Given the current iterate $x^{(m)}$, the next iterate $x^{(m+1)}$ is generated as:

$$x_i^{(m+1)} = \sqrt{\frac{\beta_i}{q_i^{(m)}}} \tag{10a}$$

where

$$q_i^{(m)} = \nabla_i s(x^{(m)}) = \left.\frac{\mathrm{d}s(x)}{\mathrm{d}x_i}\right|_{x=x^{(m)}}. \tag{10b}$$

The algorithm is iterated until a stopping condition is met (more on this later), at which point the last iterate $\hat{x}$ is re-scaled to obtain the solution to the problem (6) as $\hat{\theta} = \hat{x}/s(\hat{x})$.

While a full proof of the convergence of the algorithm is provided in Section 2.5, the intuition behind it is sketched here. For ease of notation, we will hereafter drop the iteration superscript and refer to the current iterate as $w \triangleq x^{(m)}$ and to the next one as $z \triangleq x^{(m+1)}$. The new iterate z generated from (10) can be interpreted as the solution to the problem:

$$\begin{aligned} \underset{x \in \mathbb{R}^Q_{++}}{\text{minimize}} \quad & \sum_i \frac{\beta_i}{x_i} \\ \text{subj. to} \quad & q^\mathsf{T} x = p, \end{aligned} \tag{11}$$

where $p = \sum_i \sqrt{\beta_i q_i}$. In other words, the new iterate is generated by minimizing the function g on a hyperplane which is perpendicular to the gradient of s at w. The specific choice of the offset constant, i.e. p in (11), has an interesting geometrical interpretation. Because the functions s and g satisfy the requirements for Theorem 1, for any positive c the point $y = cw$ is such that $s(y)g(y) = s(w)g(w)$ and also that $p = q^\mathsf{T} x = \nabla s(y)^\mathsf{T} x \leq s(x)\ \forall x$ (see

Theorem 4 below). Choosing c such that $s(y) = cs(w) = p$ and substituting (10) in (8), it is easy to show that the hyperplane $q^\mathsf{T} x = p$ has the following properties:

$$q = \nabla s(y) = -\nabla g(z), \tag{12a}$$
$$p = s(y) \leq s(x) \qquad \text{and} \tag{12b}$$
$$p = g(z) \leq g(x) \qquad \forall x : q^\mathsf{T} x = p. \tag{12c}$$

In other words, this hyperplane is externally tangent to the level sets of s and g of the same value, p. Asymptotically, the algorithm finds the hyperplane that is tangent to the two level sets at the same point. Although there is no guarantee that each step decreases both g and s, the next section will show that their product h decreases monotonically at each step and that the algorithm, in fact, convergences towards the solution.

2.5 Convergence analysis

A fixed point for the iterative map (10) is the point x^* satisfying the conditions of Corollary 2. In fact, substituting $q_i^* = \nabla_i s(x^*) = -\nabla_i g(x^*) = \beta_i/(x_i^*)^2$ in the iterate update (10a) makes it an identity. By Corollary 2, this fixed point is a global minimum for h and, therefore, a solution for (7).

To show that the algorithm (10) can be used to solve (7), it remains to be proven that the iterative map (10) converges to its fixed point x^* for all starting points $x^{(0)} \in \mathbb{R}^Q_{++}$. To do so, we will make use of convergence results of descent algorithms (Zangwill, 1969; Meyer, 1976; Bertsekas, 1999; Luenberger and Ye, 2008) and in particular of Zangwill's Global Convergence Theorem (Luenberger and Ye, 2008, p. 205), restated here for convenience.

Theorem 3 (Global Convergence Theorem) *Let $\mathcal{A}$ be an algorithm on A, and suppose that, given $x^{(0)}$, the sequence $\{x^{(m)}\}_{m=0}^{\infty}$ is generated satisfying $x^{(m+1)} \in \mathcal{A}(x^{(m)})$. Let a solution set $\Gamma \subset A$ be given, and suppose:*

1. *all points $x^{(m)}$ are contained in a compact set $S \subset A$,*
2. *there is a continuous function ζ on A such that:*
   (a) *if $x \notin \Gamma$, then $\zeta(z) < \zeta(x)$ for all $z \in \mathcal{A}(x)$,*
   (b) *if $x \in \Gamma$, then $\zeta(z) \leq \zeta(x)$ for all $z \in \mathcal{A}(x)$,*
3. *the mapping $\mathcal{A}$ is closed at points outside Γ.*

Then the limit of any convergent subsequence of $\{x^{(m)}\}$ is a solution.

The following will show that Theorem 3 applies to the mapping $\{x^{(m+1)}\} = \mathcal{A}(x^{(m)})$ corresponding to (10). This mapping is defined in $A = \mathbb{R}^Q_{++}$ and has solution set $\Gamma = \{x^*\}$.

Since s is a differentiable convex function in the open convex set $\mathbb{R}^Q_{++}$, it is actually continuously differentiable in $\mathbb{R}^Q_{++}$ (Rockafellar, 1970, Corollary 25.5.1). The specific choice of s in (9) is such that $q_i^{(m)}$ is also strictly positive and, therefore, the iteration (10) defines a continuous function (point-to-point mapping) from $x^{(m)}$ to $x^{(m+1)}$. Since for a point-to-point mapping continuity implies closedness (Luenberger and Ye, 2008, p. 206), the third condition of Zangwill's theorem is satisfied.

Through some simple algebra, it is easy to show that $\nabla_i s(x) \in [\eta/(2-\eta), 1]\ \forall x, i$, which implies that $x^{(m+1)} \in [\min_i \sqrt{\beta_i},\ \sqrt{(2-\eta)/\eta}\ \max_i \sqrt{\beta_i}]^Q$. Because all the points of the sequence (with the immaterial possible exception of $x^{(0)}$) are contained in this closed and bounded subset of $\mathbb{R}^Q_{++}$, the first condition of the theorem is also satisfied.

As a first step towards verifying the second condition, the following theorem is introduced (proof provided in Appendix A.2).

Theorem 4 *Given a norm $s : \mathbb{R}^n \to \mathbb{R}_+$ of the form $s(x) = d_0\|x\|_1 + \sqrt{d_1\|x\|_1^2 + d_2\|x\|_2^2}$ with $d_0, d_1, d_2 \geq 0$, the following property holds:*

$$\nabla s(y)^\mathsf{T} x \leq s(x) \leq \sqrt{x^\mathsf{T} \Lambda_y\, x} \quad \forall x, y \in \mathbb{R}^n, \tag{13}$$

where Λ_y is a diagonal matrix whose i-th diagonal element is $s(y)\nabla_i s(y)/y_i$.

We can now write the following chain of inequalities showing that h is non-increasing at each step:

$$h(x^{(m+1)}) \equiv h(z) \leq s^2(z) \leq z^\mathsf{T}\Lambda_y\, z = \sum_i \sqrt{\frac{\beta_i}{q_i}}\, s(y) \frac{q_i}{y_i} \sqrt{\frac{\beta_i}{q_i}} = h(y) = h(w) \equiv h(x^{(m)}), \tag{14}$$

where the first inequality follows from (12) while the second from Theorem 4. Unfortunately, the fact that h is constant along rays out of the origin makes it unsuitable as function ζ for Theorem 3 (the strict inequality in condition 2.(a) is violated for points $x = cx^*$ with $c \in \mathbb{R}_{++}$). Instead, we consider the function:

$$\zeta(x) = 2\, h(x) + [s(x) - g(x)]^2 = g^2(x) + s^2(x), \tag{15}$$

for which we can obtain the following inequality readily from (12), (14), and (15):

$$\zeta(z) = g^2(z) + s^2(z) \leq 2\, s^2(z) \leq 2\, h(w) \leq \zeta(w). \tag{16}$$

Importantly, as prescribed by 2.(a) the expression (16) holds with equality only if the starting point of the iteration (w in our case) is in the solution set Γ. This can be shown by first noticing that $\zeta(z) = \zeta(w)$ implies $g(z) = s(z) = g(w) = s(w)$. From the definition of y, we see that $s(w) = g(z) \ \Rightarrow\ y \equiv w$. Since the restriction of g along $q^\mathsf{T}x = p$ is strictly convex, the inequality in (12c) holds as equality only at the minimum, i.e. $g(z) = g(y) \ \Rightarrow\ z \equiv y$. Putting these together, we obtain that $z \equiv w$, which substituted in (12a) finally yields $\nabla s(w) = -\nabla g(w)$, the condition defining the fixed point x^*.

In conclusion, we have proven that the algorithm defined by the iteration (10) satisfies the conditions of Zangwill's theorem. Also, because the solution set Γ consists of a single point x^*, the sequence $\{x^{(m)}\}$ converges to x^* (Luenberger and Ye, 2008, p. 206).

2.6 Stopping conditions

We now establish a lower bound on the optimal value of h, which will be used to provide a non-heuristic stopping criterion for the iterative algorithm in (10). Given the solution x^* and the new iterate $x^{(m+1)}$ obtained as described in Section 2.4, we observe that, since q and

x^* lie in the (strictly) positive orthant, there always exists $c \in \mathbb{R}_{++}$ such a that $q^\mathsf{T}(cx^*) = p$. Therefore, (12c) implies $p \leq g(cx^*)$ and Theorem 4 yields $p = q^\mathsf{T}(cx^*) \leq s(cx^*)$. Combining these two inequalities gives $p^2 \leq g(cx^*)\, s(cx^*)$, which can be rewritten as $g^2(x^{(m+1)}) \leq h(x^*)$ where the equality only holds at the solution x^*.

As a result, $h(x^{(m+1)}) - g^2(x^{(m+1)})$ bounds how suboptimal the iterate is, even without knowing the exact value of $h(x^*)$. The algorithm presented in this paper uses the following stopping condition to guarantee a predefined relative accuracy $\epsilon_{\text{rel}} > 0$:

$$\frac{h(x^{(m+1)}) - g^2(x^{(m+1)})}{g^2(x^{(m+1)})} = \frac{s(x^{(m+1)})}{g(x^{(m+1)})} - 1 \leq \epsilon_{\text{rel}}. \tag{17}$$

The algorithm terminates after an ϵ_{rel}-suboptimal iterate is produced, i.e. when (17) is satisfied, which guarantees that $h(x^{(m+1)}) - h(x^*) \leq \epsilon_{\text{rel}}\, h(x^*)$.

3. Conclusions

This technical report focuses on an algorithm for the minimization of a positive-weighted sum of reciprocals bound to elastic-net constraints. This algorithm, explained in detail in Sections 2.3–2.6, can be used to optimize the kernel weights within a two-step block coordinate descent alternating between the optimization of the SVM classifiers and the optimization of the kernel weights. Preliminary tests (not reported) of the computational cost of the algorithm show that it compares very favourably to existing and alternative approaches. Finally, because it does not depend on external libraries, it has a wide applicability and can be readily included in existing open-source machine learning libraries.

Appendix A. Proofs of theorems

The proofs of the theorems given in the text are reported in this appendix in the form of structured proofs as advocated by Leslie Lamport (2012). Each assertion follows from previously stated facts, which are explicitly named to tell the reader exactly which ones are being used at each step.

A.1 Proofs of Theorem 1 and Corollary 2

Theorem 1 *Let $A \subseteq \mathbb{R}^n$ be an open convex cone and $g, s : A \to \mathbb{R}_{++}$ be differentiable convex functions such that $s(cx) = cs(x)$ and $g(cx) = g(x)/c$ for all $c \in \mathbb{R}_{++}$ and $x \in A$. Their pointwise product $h(x) = s(x)\, g(x)$ is a pseudoconvex function in A.*

Proof

1. To show that the differentiable function $h : A \to \mathbb{R}_{++}$ defined in an open convex set is pseudoconvex, it suffices to assume for the remaining of this proof that:

   1.1. $y, z \in A$,

   1.2. $h(z) < h(y)$,

   and prove that $\nabla h(y)^\mathsf{T}(z - y) < 0$.

   Proof: By the definition of pseudoconvex function (Cambini and Martein, 2008, definition 3.2.1).

2. $\forall\, x \in A : \nabla g(x)^{\mathsf{T}} x = -g(x)$.

   Proof: By differentiating $g(cx) = g(x)/c$ w.r.t. c and evaluating it for $c = 1$.

3. $\forall\, x \in A : \nabla s(x)^{\mathsf{T}} x = s(x)$.

   Proof: By differentiating $s(cx) = cs(x)$ w.r.t. c and evaluating it for $c = 1$.

4. $\forall\, x \in A : \nabla h(x)^{\mathsf{T}} x = g(x)\nabla s(x)^{\mathsf{T}} x + s(x)\nabla g(x)^{\mathsf{T}} x = 0$.

   Proof: Follows directly from 2 and 3.

5. Given z and y as in 1.1, $\exists\, c \in \mathbb{R}_{++}$ such that the point $z' = cz$ satisfies $h(z') = h(z)$ and $s(z') = s(y)$.

   Proof: For any positive c the corresponding z' is in A (because A is a cone) and satisfies the first condition: $h(z') = s(cz)\, g(cz) = cs(z)\, g(z)/c = h(z)$. We choose $c = s(y)/s(z)$ which also satisfies the second condition: $s(z') = s(y)/s(z)\, s(z) = s(y)$.

6. $\forall\, x, x' \in A : \nabla s(x)^{\mathsf{T}} x' \leq s(x')$.

   Proof: The first-order conditions for convexity (Boyd and Vandenberghe, 2009, ch 3.1.3) imply $s(x') \geq s(x) + \nabla s(x)^{\mathsf{T}}(x' - x)$. Substituting 3 and rearranging yields 6.

7. $\forall\, x, x' \in A : \nabla g(x)^{\mathsf{T}} x' \leq g(x') - 2\, g(x)$.

   Proof: The first-order conditions for convexity imply $g(x') \geq g(x) + \nabla g(x)^{\mathsf{T}}(x' - x)$. Substituting 2 and rearranging yields 7.

8. $\nabla h(y)^{\mathsf{T}} z' < 0$.

   Proof: By 6, 7, 5 and 1.2, we have:

$$\begin{aligned} \nabla h(y)^{\mathsf{T}} z' &= g(y)\, \nabla s(y)^{\mathsf{T}} z' + s(y)\, \nabla g(y)^{\mathsf{T}} z' \\ &\leq g(y)\, s(z') + [s(y)\, g(z') - 2\, s(y)\, g(y)] \\ &= h(y) + h(z') - 2\, h(y) \\ &= h(z') - h(y) = h(z) - h(y) < 0. \end{aligned}$$

9. Q.E.D.

   Proof: By 8, 5 and 4, we have:

$$c\nabla h(y)^{\mathsf{T}} z < 0 \;\Rightarrow\; \nabla h(y)^{\mathsf{T}} z = \nabla h(y)^{\mathsf{T}}(z - y) < 0.$$

   By 1, the latter proves the theorem.

■

Corollary 2 *Under the conditions of Theorem 1, points $x = cx^*$ with $c \in \mathbb{R}_{++}$ and x^* satisfying $\nabla s(x^*) = -\nabla g(x^*)$ are global minima for the function h, where it takes value $h(cx^*) = s^2(x^*) = g^2(x^*)$. If at least one of s or g is strictly convex, then x^* is unique.*

Proof

10. $\nabla s(x^*) = -\nabla g(x^*) \;\Rightarrow\; s(x^*) = g(x^*)$.

    Proof: Follows immediately from the statements 2 and 3 of the proof of Theorem 1.

11. x^* is a critical point for h.

    Proof: From the condition $\nabla s(x^*) = -\nabla g(x^*)$ and from statement 10: $\nabla h(x^*) = g(x^*)\,\nabla s(x^*) + s(x^*)\,\nabla g(x^*) = 0$.

12. x^* is a global minimum of h.

    Proof: Because x^* is a critical point (statement 12) of a pseudoconvex function (Theorem 1), it is also a global minimum (Cambini and Martein, 2008, theorem 3.2.5).

13. If at least one of s or g is strictly convex, then x^* is unique.

    Proof: Aiming for a contradiction, let us assume that there is a point $\hat{x} \in A \backslash \{x^*\}$ such that $\nabla s(\hat{x}) = -\nabla g(\hat{x})$. By using the same reasoning as in 10, this implies $s(\hat{x}) = g(\hat{x})$. Without loss of generality, let us assume that $s(x^*) \geq s(\hat{x})$ and that s is strictly convex. From the first-order conditions for (strict) convexity, we obtain:

$$s(\hat{x}) > s(x^*) + \nabla s(x^*)^\mathsf{T}(\hat{x} - x^*) \quad \Rightarrow \quad \nabla s(x^*)^\mathsf{T}(\hat{x} - x^*) < 0, \tag{18}$$

$$g(\hat{x}) \geq g(x^*) + \nabla g(x^*)^\mathsf{T}(\hat{x} - x^*) \quad \Rightarrow \quad \nabla s(x^*)^\mathsf{T}(\hat{x} - x^*) \geq 0, \tag{19}$$

    which is obviously a contradiction.

14. Q.E.D.

    Proof: From 10, 12, 13, and the definitions of s, g, and h in the statement of Theorem 1.

■

A.2 Proof of Theorem 4

Theorem 4 *Given a norm $s : \mathbb{R}^n \to \mathbb{R}_+$ of the form $s(x) = d_0\|x\|_1 + \sqrt{d_1\|x\|_1^2 + d_2\|x\|_2^2}$ with $d_0, d_1, d_2 \geq 0$, the following property holds:*

$$\nabla s(y)^\mathsf{T} x \leq s(x) \leq \sqrt{x^\mathsf{T} \Lambda_y\, x} \quad \forall x, y \in \mathbb{R}^n, \tag{13}$$

where Λ_y is a diagonal matrix whose i-th diagonal element is $s(y)\nabla_i s(y)/y_i$.

Proof

1. $\forall\, y \in \mathbb{R}^n : \nabla s(y)^\mathsf{T} y = s(y)$.

   Proof: Since s is a norm, $s(cy) = |c|s(y)$. By differentiating both sides w.r.t. c and evaluating it for $c = 1$, we obtain the statement 1.

2. $\forall\, y, x \in \mathbb{R}^n : \nabla s(y)^\mathsf{T} x \leq s(x)$.

   Proof: The first-order conditions for convexity (Boyd and Vandenberghe, 2009, ch 3.1.3) imply $s(x) \geq s(y) + \nabla s(y)^\mathsf{T}(x - y)$. Substituting 1 and rearranging yields 2.

3. Define $r : \mathbb{R}^n \to \mathbb{R}_+$ as $r(y) = \sqrt{d_1^2\|y\|_1^2 + d_2^2\|y\|_2^2}$.

4. $\forall\, y, x \in \mathbb{R}^n : \sqrt{\dfrac{d_1^2\|x\|_1^2 + d_2^2\|x\|_2^2}{\|x\|_1^2}} \leq \dfrac{1}{2}\left[\dfrac{r(y)}{\|y\|_1} + \dfrac{d_1^2\|x\|_1^2 + d_2^2\|x\|_2^2}{\|x\|_1^2}\dfrac{\|y\|_1}{r(y)}\right]$.

   Proof: From the inequality $\sqrt{z} \leq \frac{1}{2}[\sqrt{z_0} + z/\sqrt{z_0}]$, which in turn results from the concavity of the square root function.

5. $\forall\, y, x \in \mathbb{R}^n : s^2(x) \leq s(y)\left[\dfrac{d_0}{\|y\|_1}\|x\|_1^2 + \dfrac{d_1}{r(y)}\|x\|_1^2 + \dfrac{d_2}{r(y)}\|x\|_2^2\right]$.

   Proof: Follows from writing out the lhs explicitly using the definition of s and then exploiting the statement in 4.

6. $\forall\, y, x \in \mathbb{R}^n : \dfrac{d_0}{\|y\|_1}\|x\|_1^2 + \dfrac{d_1}{r(y)}\|x\|_1^2 + \dfrac{d_2}{r(y)}\|x\|_2^2 \leq \sum_i \dfrac{d_0}{|y_i|}x_i^2 + \sum_i \dfrac{d_1\|y\|_1}{r(y)|y_i|}x_i^2 + \sum_i \dfrac{d_2}{r(y)}x_i^2$.

   Proof: The last term of each side of the inequality is identical. Applying Radon's inequality it is easy to show that the each one of the first two terms of the lhs is bounded by the corresponding term in the rhs.

7. $\forall\, y, x \in \mathbb{R}^n : s^2(x) \leq x^\mathsf{T}\Lambda_y\, x$.

   Proof: Follows from combining 5 and 6, then using the definition of Λ_y.

8. Q.E.D.

   Proof: Combining 2 and 7 proves the theorem.

■

References

Dimitri P. Bertsekas. *Nonlinear Programming*. Athena Scientific, Belmont, Mass, 2nd edition edition, September 1999. ISBN 978-1-886-52900-7.

Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2009. ISBN 978-0-521-83378-3.

Alberto Cambini and Laura Martein. *Generalized convexity and optimization: Theory and applications*, volume 616. Springer, 2008. ISBN 978-3-540-70875-9.

Leslie Lamport. How to write a 21st century proof. *Journal of Fixed Point Theory and Applications*, 11(1):43–63, 2012. doi: 10.1007/s11784-012-0071-6.

David G. Luenberger and Yinyu Ye. *Linear and Nonlinear Programming*. Springer Science & Business Media, June 2008. ISBN 978-0-387-74503-9.

Robert R. Meyer. Sufficient conditions for the convergence of monotonic mathematical programming algorithms. *Journal of Computer and System Sciences*, 12(1):108–121, February 1976. doi: 10.1016/S0022-0000(76)80021-9.

Alain Rakotomamonjy, Francis Bach, Stéphane Canu, and Yves Grandvalet. More efficiency in multiple kernel learning. In *Proceedings of the 24th international conference on Machine learning*, pages 775–782. ACM, 2007.

Ralph T. Rockafellar. *Convex Analysis*. Princeton University Press, 1970. ISBN 0-691-08069-0.

Tao Sun, Licheng Jiao, Fang Liu, Shuang Wang, and Jie Feng. Selective multiple kernel learning for classification with ensemble strategy. *Pattern Recognition*, 46(11):3081–3090, November 2013. doi: 10.1016/j.patcog.2013.04.003.

Zenglin Xu, Rong Jin, Haiqin Yang, Irwin King, and Michael R. Lyu. Simple and efficient multiple kernel learning by group lasso. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pages 1175–1182, 2010.

Haiqin Yang, Zenglin Xu, Jieping Ye, I King, and M R Lyu. Efficient sparse generalized multiple kernel learning. *IEEE Transactions on Neural Networks*, 22(3):433–446, March 2011. doi: 10.1109/TNN.2010.2103571.

Willard I. Zangwill. *Nonlinear programming: a unified approach*. Prentice-Hall, 1969. ISBN 978-0-136-23579-8.