Approximating Concavely Parameterized Optimization Problems

Authored by:

Joachim Giesen
Jens Mueller
Soeren Laue
Sascha Swiercy

Abstract

We consider an abstract class of optimization problems that are parameterized concavely in a single parameter, and show that the solution path along the parameter can always be approximated with accuracy $varepsilon ¿0$ by a set of size $O(1/sqrt{varepsilon})$. A lower bound of size $Omega (1/sqrt{varepsilon})$ shows that the upper bound is tight up to a constant factor. We also devise an algorithm that calls a step-size oracle and computes an approximate path of size $O(1/sqrt{varepsilon})$. Finally, we provide an implementation of the oracle for soft-margin support vector machines, and a parameterized semi-definite program for matrix completion.

1 Paper Body

Problem description. Let D be a set, I ? R an interval, and f : I ? D ? R such that (1) f (t, ?) is bounded from below for every t ? I, and (2) f (?, x) is concave for every x ? D. We study the parameterized optimization problem h(t) = minx?D f (t, x). A solution x?t ? D is called optimal at parameter value t if f (t, x?t) = h(t), and x ? D is called an ?-approximation at t if ?(t, x) := f (t, x) ? h(t) ? ?. Of course it holds ?(t, x?t) = 0. A subset P ? D is called an ?-path if P contains an ?-approximation for every t ? I. The size of a smallest ?-approximation path is called the ?-path complexity of the parameterized optimization problem. The aim of this paper is to derive upper and lower bounds on the path complexity, and to provide efficient algorithms to compute ?-paths. Motivation. The rather abstract problem from above is motivated by regularized optimization problems that are abundant in machine learning, i.e., by problems of the form min f (t, x) := r(x) + t ? l(x),
x?D

where r(x) is a regularization- and l(x) a loss term. The parameter t controls the trade-off between regularization and loss. Note that here f (?, x) is always linear and hence concave in the parameter t. 1

Previous work. Due to the widespread use of regularized optimization methods in machine learning regularization path following algorithms have become an active area of research. Initially, exact path tracking methods have been developed for many machine learning problems [16, 18, 3, 9] starting with the algorithm for SVMs by Hastie et al. [10]. Exact tracking algorithms tend to be slow and numerically unstable as they need to invert large matrices. Also, the exact regularization path can be exponentially large in the input size [5, 14]. Approximation algorithms can overcome these problems [4]. Approximation path algorithms with approximation guarantees have been developed for SVMs with square loss [6], the LASSO [14], and matrix completion and factorization problems [8, 7]. ? Contributions. We provide a structural upper bound in O(1/ ?) for the ?-path complexity for the abstract problem class described above. We?show that this bound is tight up to a multiplicative constant by constructing a lower bound in ?(1/ ?). Finally, we devise a generic algorithm to compute ?-paths that calls a problem specific oracle providing a step-size ? certificate. If such a certificate exists, then the algorithm computes a path of complexity in O(1/ ?). Finally, we demonstrate the implementation of the oracle for standard SVMs and a matrix completion problem. ? Resulting in the first algorithms for both problems that compute ?-paths of complexity in O 1/ ? . Previously, no approximation path algorithms have been known for standard SVMs but only a heuristic [12] and an approximation algorithm for square loss SVMs [6] with complexity in O(1/?). The best approximation path algorithm for matrix completion also has complexity in ? O(1/?). To our knowledge, the

only known approximation path algorithm with complexity in O 1/ ? is [14] for the LASSO.

2

Upper Bound

Here we show that any problem that fits the problem definition from the introduction for a compact ? interval I = [a, b] has an ?-path with complexity in O(1/ ?). Let (a, b) be the interior of [a, b] and let g : (a, b) ? R be concave, then g is continuous and has a 0 0 (t), respectively, at every point t ? I (see for example [15]). (t) and g+ left- and right derivative g? Note that f (?, x) is concave by assumption and h is concave as the minimum over a family of concave functions. 0 0 Lemma 1. For all t ? (a, b), h0? (t) ? f? (t, x?t) ? f+ (t, x?t) ? h0+ (t).

Proof. For all t0 ¡ t it holds h(t0) ? f (t0 , x?t) and hence h(t) ? h(t0) ? f (t, x?t) ? f (t0 , x?t) which implies h(t) ? h(t0) f (t, x?t) ? f (t0 , x?t) 0 h0? (t) := lim ? lim =: f? (t, x?t). t0 ?t t0 ?t t ? t0 t ? t0 0 0 0 The inequality f+ (t, x?t) ? h0+ (t) follows analogously, and f? (t, x?t) ? f+ (t, x?t) follows after ? some algebra from the concavity of f (?, xt) and the definition of the derivatives (see [15]).

Definition 2. Let I = [a, b] be a compact interval, ? ¿ 0, and t0 = a. Let

Tk = t — t ? (tk?1 , b] such that ?(t, x?tk?1) := f (t, x?tk?1) ? h(t) = ?

, and tk = min Tk for all integral k ¿ 0 such that Tk 6= ?. Finally, let P ? = {x?tk — k ? N such that Tk 6= ?}. ?1 2 Lemma 3. Let s1 , . . . , sn ? R¿0 , then (s1 + . . . + sn)(s?1 1 + . . . + sn) ? n .

Proof. The claim holds for n = 1 as s1 s?1 = 1 = 12 . Assume the claim holds for n ? 1 and 1 ?1 ?1 let a = s1 + . . . + sn?1 and b = s1 + . . . + s?1 n?1 . The rectangle with side lengths asn and ?1 ?1 bsn has circumference 2(asn + bsn) and area asn bsn ? = ab. Since the square minimizes the circumference for a given area we have 2(as?1 + bs) ? 4 ab. The claim for n now follows from n n ? ? ?1 2 2 2 (a + sn)(b + s?1 n) = ab + asn + bsn + 1 ? ab + 2 ab + 1 = (ab + 1) ? ((n ? 1) + 1) = n .

2

Lemma 4. The size of P ? is at most

q

? (b ? a)(h0? (a) ? h0? (b)) /? ? O 1/ ? .

Proof. Let a = t0 ? t1 ? . . . be the sequence from Definition 2. Define ?k = tk+1 ? tk and ?k = h0? (tk) ? h0? (tk+1). We have 0 ?k ?k ? (f? (tk , x?tk) ? h0? (tk+1))(tk+1 ? tk)

f (tk+1 , x?tk) ? f (tk , x?tk) h(tk+1) ? h(tk) ? ? (tk+1 ? tk) tk+1 ? tk tk+1 ? tk = f (tk+1 , x?tk) ? h(tk+1) = ?(tk+1 , x?tk),

where the first inequality follows from Lemma 1 and the second inequality follows from concavity and the definition of derivatives (see [15]). Thus, there exists sk ¿ 0 such that ?k ? ?sk and ?k ? s?1 k . It follows from Lemma 3 that ?n2 ? ?

?1 ?(s1 + . . . + sn)(s?1 1 + . . . + sn)

(b ? a)(?1 + . . . + ?n)

?

(b ?

?

(?1 + . . . + ?n)(?1 0 a)(h? (a) ? h0? (b)),

+ . . . + ?n)

? h0? (t) for t ? b (which can be proved from conwhere the last inequality follows from cavity, see again [15]). Hence, the size of P ? must be finite, or more q the sequence (tk) and thus

specifically n is bounded by (b ? a)(h0? (a) ? h0? (b)) /?. h0? (b)

Theorem 5. P ? is an ?-path for I = [a, b]. Proof. For any x ? D, ?(?, x) is a continuous function. Hence, x?tk is an ?-approximation for all t ? [tk , tk+1], because if there would be t ? (tk , tk+1] with ?(t, x?tk) ¿ ?, then by continuity, there would be also t0 ? (tk , tk+1) with ?(t, x?tk) = ? which contradicts the minimality of tk+1 . The claim of the theorem follows since the proof of Lemma 4 shows that the sequence (tk) is finite and hence the intervals [tk , tk+1] cover the whole [a, b].

3

Lower Bound

Here we show that there exists a problem that fits the problem description from the introduction ? whose ?-path complexity is in ?(1/ ?). This shows that the upper bound from the previous section is tight up to a constant. Let I = [a, b], D = R, f (t, x) = 12 x2 ? tx and thus

2 1 2 1 1 h(t) = min x ? tx = x?t ? tx?t = ? t2 , x?R 2 2 2 where the last equality follows from the convexity and differentiability of f (t, x) in x which together ? ? imply ?f ?x (t, xt) = xt ? t = 0.

For ? ¿ 0 and x ? R let Ix = t ? [a, b] ?(t, x) := 21 x2 ? tx + 12 t2 ? ? , which is an interval ? since 12 x2 ? tx + 12 t2 is a quadratic function in t. The length of this interval is 2 2? independent ? of x. Hence, the ?-path complexity for the problem is at least (b ? a)/2 2?. Let us compare this lowerqbound with the upper from the previous section which gives for the q

(b?a)2 0 0 ? . Hence the upper specific problem at hand, (b ? a)(h? (a) ? h? (b)) /? = = b?a ? ? ? bound is tight up to constant of at most 2 2.

4

Generic Algorithm

So far we have only discussed structural complexity ? bounds for ?-paths. Now we give a generic algorithm to compute an ?-path of complexity in O(1/ ?). When applying the generic algorithm to 3

a specific problem a plugin-subroutine PATH P OLYNOMIAL needs to be implemented for the specific problem. The generic algorithm builds on the simple idea that has been introduced in [6] to compute an (?/?)-approximation (for ? ¿ 1) and only update this approximation along the parameter interval I = [a, b] when it fails to be an ?-approximation. The plugin-subroutine PATH P OLYNOMIAL provides a bound on the step-size for the algorithm, i.e., a certificate for how long the approximation is valid along the interval I. Hence we describe the idea behind the construction of this certificate first. 4.1

Step-size certificate and algorithm

We always consider a problem that fits the problem description from the introduction. Definition 6. Let P be the set of all concave polynomials p : I ? R of degree at most 2. For t ? I, x ? D and ? ¿ 0 let Pt (x, ?) := {p ? P — p ? h, f (t, x) ? p(t) ? ?}, 0

where p ? h means p(t) ? h(t0) for all t0 ? I. Note that P contains constant and linear polynomials with second derivative p00 = 0 and quadratic polynomials with constant second derivative p00 ¡ 0. If Pt (x, ?) 6= ?, then x is an ?-approximation at parameter value t, because there exists p ? P such that ?(t, x) ? f (t, x) ? p(t) ? ?. Definition 7. [Step-size] For t ? I = [a, b], p ? P, ? ¿ 0, and ? ¿ 1, let ?t := t ? a and ?t (p, ?) =

? , if p00 ¡ 0 and ?t ¿ 0. ? ?t2 —p00 —

The step-size is given as ? (1) ? : p00 = 0 ? ?t (p) (2) ?t (p, ?) = ?t (p, ?) : p00 ¡ 0, ?t (p, ?) ? ? ? (3) ?t (p, ?) : p00 ¡ 0, ?t (p, ?) ? where

(1)

?t (p)

1 2 1 2

?t (? ? 1) s

2

2? 1 1 2 = + ?t ?t (p, ?) ? ? ?t ?t (p, ?) + —p00 — 2 2 s

2? 1 = 1? ? 00 —p — ?

=
(2)
?t (p, ?) (3)
?t (p, ?)
To simplify the notation we will skip the argument ? of the step-size ?t whenever the value of ? is obvious from the context. (2)
(3)
Observation 8. If ?t (p, ?) = 1/2, then ?t (p) = ?t (p), because ?t (p, ?) = 1/2 implies ?t = q 2? ? —p00— . Lemma 9. For t ? (a, b), x ? D, ? ¿ 0 and ? ¿ 1. If there exists p ? Pt (x, ?/?), then x is an ?-approximation for all t0 ? [t, b] with t0 ? t + ?t (p). Proof. Let g : [a, b] ? R be the following linear function, g(t0) = (t0 ? t)
p(t) + ?/? ? p(a) ? + p(t) + . t?a ?
Then, for all t0 ? [t, b], f (t0 , x) ? (t0 ? t)
f (t, x) ? f (a, x) p(t) + ?/? ? p(a) ? + f (t, x) ? (t0 ? t) + p(t) + = g(t0) t?a t?a ? 4
where the first inequality follows from the concavity of f (?, x), and the second inequality follows from f (t, x) ? p(t) ? ?/? and from p(a) ? h(a) ? f (a, x). Thus, x is an ?-approximation for all t0 ? [t, b] that satisfy g(t0) ? p(t0) ? ? because ?(t0 , x) = f (t0 , x) ? h(t0) ? f (t0 , x) ? p(t0) ? g(t0) ? p(t0) ? ?. We finish the proof by considering three cases. (i) If p00 = 0, then g(t0) ? p(t0) is a linear function in t0 , and g(t0) ? p(t0) ? ? solves to t0 ? t ? (1) ?t (? ? 1) = ?t (p) = ?t (p). (ii) If p00 ¡ 0, then g(t0) ? p(t0) is a quadratic polynomial in t0 with second derivative ?p00 ¿ 0, (2) and the equation g(t0)?p(t0) ? ? solves to t0 ?t ? ?t (p). Note that we do not need the condition ?t (p) ? 1/2 here. (iii) The caseq p00 ¡ 0 and ?t (p) ? 1/2 can q be reduced to Case (ii). From ?t (p) ? 1/2 we obtain 2? and thus a ? t ? ?. Let p? the restriction of p onto the interval t ? a = ?t ? —p2? 00 —? —p00 —? =: a
[? a, b] and ??t = t ? a ?, then p?00 = p00 , and thus ?t (? p) = ?/ ? ??2 —? p00 — = 1 . Hence by Observation 8, t
(3)
(3)
2
(2)
?t (p) = ?t (? p) = ?t (? p). The claim follows from Case (ii). Assume now that we have an oracle PATH P OLYNOMIAL available that on input t ? (a, b) and ?/? ¿ 0 returns x ? D and p ? Pt (x, ?/?), then the following algorithm G ENERIC PATH returns an ?-path if it terminates. Algorithm 1 G ENERIC PATH Input: f : [a, b] ? D ? R that fits the problem description, and ? ¿ 0 Output: ?-path for the interval [a, b] choose t? ? (a, b) P := C OMPUTE PATH (f, t?, ?) define f? : [a, b] ? D ? R, (t, x) 7? f (a + b ? t, x) [then f? also fits the problem description] P := P ? C OMPUTE PATH (f?, a + b ? t?, ?) return P Algorithm 2 C OMPUTE PATH Input: f : [a, b] ? D ? R that fits the problem description, t? ? (a, b) and ? ¿ 0 Output: ?-path for the interval [t?, b] t := t? and P := ? while t ? b do

(x, p) := PATH P OLYNOMIAL t, ?/? P := P ? {x}
t := min b, t + ?t (p) end while return P 4.2
Analysis of the generic algorithm
The running time of the algorithm G ENERIC PATH is essentially determined by the complexity of the computed path times the cost of the oracle PATH ? P OLYNOMIAL. In the following we show that the complexity of the computed path is at most O(1/ ?). ? Observation 10. For c ? R let ?c : R ? ? R, x 7? x2 + c ? x. Then we have ¿
—c—
1. limx?? ?c (x) = 0 2. ?0c (x) = ?xx2 +c ? 1 for the derivative of ?c . Thus, ?0c (x) ¿ 0 for c ¡ 0 and ?c is monotonously increasing. 5
(2)
Furthermore, ?t (p) =
q r
2? —p00 —
+ ?t2 ?t (p) ?
1 2
2
1 2
2
?t2 ?t (p) + ? ?
=
r
?t2 ?t (p) + ? ?
=
1 2 2
1 2
+ ?t2 ?(1 ? ?) ? ?t ?t (p) +
1 2
+ ?t2 ?(1 ? ?) ? ?t ?t (p) + ? ?
= ??2 ?(1??) ?t ?t (p) + ? ? t
? ?t ?t (p) +
1 2
1 2
+ ?t (? ? 1)
+ ?t (? ? 1).
Lemma 11. Given t ? I and p ? P, then ?t (p) is continuous in —p00 —.
(2)
(3)
Proof. The continuity for —p00 — ¿ 0 follows from the definitions of ?t (p) and ?t (p), and from Observation 8. Since ?t (p) ¿ 1/2 for small —p00 — the continuity at —p00 — = 0 follows from Observation 10, because (2)
(1)
lim ?t (p) = lim ??t2 ?(1??) (?t ? (?t (p) + ? ? 1/2)) + ?t (? ? 1) = ?t (? ? 1) = ?t (p), 00
—p00 —?0

—p —?0

where we have used ?t (p) ? ? as —p00 — ? 0. Lemma 12. Given t ? I and p1 , p2 ? P, then ?t (p1) ? ?t (p2) if —p001 — ? —p002 —. Proof. The claim is that ?t (p) is monotonously decreasing in —p00 —. Since ?t is continuous in —p00 — (1) (2) (3) by Lemma 11 it is enough to check the monotonicity of ?t (p), ?t (p) and ?t (p). The mono(1) (3) tonicity of ?t (p) and ?t (p) follows directly from the definitions of the latter. The monotonicity (2) of ?t (p) follows from Observation 10 since we have

1 (2) ?t (p) = ??t2 ?(1??) ?t ?t (p) + ? ? + ?t (? ? 1), 2 (2)

and thus ?t (p) is monotonously decreasing in —p00 — because ?t2 ?(1 ? ?) ¡ 0 and ?t (p) is monotonously decreasing in —p00 —. Lemma 13. Given t ? I and p ? P, then ?t (p) is monotonously increasing in ?t and hence in t. Proof. Since ?t (p) is continuous in ?t by Observation 8 it is enough to check the monotonicity of (1) (2) (3) (1) (3) ?t (p), ?t (p) and ?t (p). The monotonicity of ?t (p) and ?t (p) follows directly from the (2) definitions of the latter. It remains toshow the monotonicity of ?t (p) for ?t (p) ? 21 . For c ? 0

let ??1 : R¿0 ? R, y 7?

1 2

?c (??1 c (y)) = y. Apparently,

c y ? y . The notation is justified because for ??1 c is monotonously decreasing, and we have

??1 c (y) ¿ 0 we have

(2)

?1 ?1 ?t (p) = ?c1 (??1 c2 (?t)) ? ?t = ?c1 (?c2 (?t)) ? ?c2 (?c2 (?t)), 1 with c1 = —p2?00 — and c2 = c?1 . Note that ??1 c2 (?t) ¿ 0 since ?t (p) ? 2 , and c2 ¡ c1 since ? ¿ 1. 0 0 ?1 Because ?c1 ? ?c2 ¡ 0 for c1 ¿ c2 , both ?c2 and ?c1 ? ?c2 are monotonously decreasing in their (2) respective arguments. Hence, ?t (p) is monotonously increasing in ?t .

Theorem 14. If there exists p ? P and ?? ¿ 0 such that —q 00 — ? —p00 — for all q that are returned by the oracle?PATH 1 terminates after at most P OLYNOMIAL on input t ? [a, b] and ? ? ??. Then Algorithm ? O 1/ ? steps, and thus returns an ?-path of complexity in O(1/ ?). Proof. For all t ? [t?, b], where t? ? (a, b) is chosen in algorithm G ENERIC PATH, we have ?t (q) ? ?t (p) ? ?t?(p). Here the first inequality is due to Lemma 12 and the second inequality is due to Lemma 13. Hence, the number of steps in the first call of C OMPUTE PATH is upper bounded by (b? t?)/(min{?t?(p), b? t?})+1. Similarly, the number of steps in the second call of C OMPUTE PATH is upper bounded by (t? ? a)/(min{?a+b?t?(p), t? ? a}) + 1. 6

(1)

For the asymptotic behavior, observe that ?t?(p) = ?t? (p) does not depend on ? for p00 = 0. For —p00 — ¿ 0 observe that lim??0 ?t?(p, ?) = 0. Hence, there exists ?? ¿ 0 such that ?t?(p, ?) ¡ 1/2 and (3) ?t? (p, ?) ? b ? t? for all ? ¡ ??, and thus r

? ? —p00 — b ? t? 1 b ? t? ? ? + 1 = (3) (b ? t) + 1 ? O . +1 = ? 2? ? ? 1 ? min{?t?(p), b ? t?} ?t? (p) ? Analogously, (t? ? a)/(min{?a+b?t?(p), t? ? a}) + 1 ? O 1/ ? , which completes the proof.

5

Applications

Here we demonstrate on two examples that Lagrange duality can be a tool for implementing the oracle PATH P OLYNOMIAL in the generic path algorithm. This approach obtains the step-size certificate from an approximate solution that has to be computed anyway. 5.1

Support vector machines

Given data points xi ? Rd together with labels yi ? {?1} for i = 1, . . . , n. A support vector machine (SVM) is the following parameterized optimization problem ! n X 1 2 T min kwk + t max{0, 1 ? yi (w xi + b)} =: f (t, w) 2 w?Rd ,b?R i=1 parameterized in the regularization parameter t ? [0, ?). The Lagrangian dual of the SVM is given as

1 s.t. 0 ? ?i ? t, y T ? = 0, maxn ? ?T K? + 1T ? =: d(?) ??R 2 where K = AT A, A = (y1 x1 , . . . , yn xn) ? Rd?n and y = (y1 , . . . , yn) ? Rn . Algorithm 3 PATH P OLYNOMIAL SVM Input: t ? (0, ?) and ? ¿ 0 Output: w ? Rd and p ? Pt (w, ?) compute a primal solution w ? Rd and a dual solution ? ? Rn such that f (t, w) ? d(?) ¡ ? define p : I ? R, t0 7? d ?t0 /t return (w, p) Lemma 15. Let (w, p) be the output of PATH P OLYNOMIAL SVM on input t ¿ 0 and ? ¿ 0, then p ? Pt (w, ?) and —p00 — ? max0???1 ? ?T K ? ? . [Hence, Theorem 14 applies here.] ? Proof. Let ? be the dual solution computed by PATH P OLYNOMIAL SVM and p be the polynomial defined in PATH P OLYNOMIAL SVM. Then, 2

t0 1 T t0 1 ? K? + 1T ? and thus p00 (t0) = ? 2 ?T K? ? 0 2 t 2 t t since K is positive semidefinite. Hence, p ? P. For p ? Pt (w, ?), it remains to show that p ? h = minw?Rd f (?, w) and f (t, w) ? p(t) ? ?. The latter follows immediately from p(t) = d(?). For t0 ¿ 0 let ?0 = ?t0 /t, then ?0 is feasible for the dual SVM at parameter value t0 since ? is feasible for the dual SVM at t. It follows, p(t0) = d(?0) ? h(t0) = minw?Rd f (?, w). Finally, observe that ? ?T K ? ?. ?i ? t implies —p00 — = t12 ?T K? ? max0???1 ? p(t0) = ?

The same results hold when using any positive kernel K. In the kernel case one has the following primal SVM (see [2]), (!) ! n n X X 1 T min

??Rm ,b

2

? K? + t ?

max

0, 1 ? yi

i=1

?j yj Kij + b

j=1

7

=: f (t, ?)

.

We have implemented the algorithm G ENERIC PATH for SVMs in Matlab using LIBSVM [1] as the SVM solver. To assess the practicability of the proposed algorithm we ran it on several datasets taken from the LIBSVM website. For each dataset we have measured the size of the computed ?-path (number of nodes) for t ? [0.1, 10] and ? ? {2?i — i = 2, . . . , 10}. Figure 5.1 shows the size of paths as a function of ? using double logarithmic plots. A straight line plot with slope ? 21 ? corresponds to an empirical path complexity that follows the function 1/ ?. 1/sqrt(epsilon) a1a duke fourclass scale mushrooms w1a

nodes

nodes

1/sqrt(epsilon) a1a duke fourclass scale mushrooms w1a

1

10

?3

10

?2

?1

10

?3

10

10

epsilon

?2

?1

10

10 epsilon

(a) Path complexity for a linear SVM

5.2

1

10

(b) Path complexity for a SVM with Gaussian kernel exp(??ku ? vk22) for ? = 0.5

Matrix completion

Matrix completion asks for a completion X of an (n ? m)-matrix Y that has been observed only at the indices in ? ? {1, . . . , m} ? {1, . . . , n}. The problem can be solved by the following convex semidefinite optimization approach, see [17, 11, 13],

X

2 1 A X 0. min Xij ? Yij + t ? tr(A) + tr(B) s.t. XT B 2 X?Rn?m , A?Rn?n , B?Rm?m (i,j)??

The Lagrangian dual of this convex semidefinite program is given as

X 1 tI ? max ? 0, and ?ij = 0 if (i, j) ? / ?. ?2ij + ?ij Yij s.t. ?T tI 2 ??Rn?m (i,j)??

? for X ? = (X, A, B) be the primal objective function at parameter value t, and d(?) be Let f (t, X) the dual objective function. Analogously to the SVM case we have the following: Algorithm 4 PATH P OLYNOMIAL M ATRIX C OMPLETION Input: t ? (0, ?) and ? ¿ 0 ? and p ? Pt (X, ? ?) Output: X ? and a dual solution ? ? Rn?m such that f (t, X) ? ? d(?) ¡ ? compute a primal solution X

0 0 define p : I ? R, t 7? d t /t ? ? p) return (X, ? p) be the output of PATH P OLYNOMIAL M ATRIXCOMPLETION on inLemma 16. Let (X, ? ?) and —p00 — ? max ? ? 2 put t ¿ 0 and ? ¿ 0, then p ? Pt (X, ??F1 k?kF , where

tI ?

Ft = ? ? Rn?m 0, ?ij = 0, ?(i, j) ? /? .

?T tI The proof for Lemma 16 is similar to the proof of Lemma 15, and Lemma 16 shows that Theorem 14 can be applied here. Acknowledgments schaft (GI-711/3-2).

This work has been supported by a grant of the Deutsche Forschungsgemein-

8

2 References

[1] Chih-Chung Chang and Chih-Jen Lin. Libsvm: A library for support vector machines. ACM Trans. Intell. Syst. Technol., 2(3):27:1?27:27, 2011. [2] Olivier Chapelle. Training a Support Vector Machine in the Primal. Neural Computation, 19(5):1155?1178, 2007. [3] Alexandre d?Aspremont, Francis R. Bach, and Laurent El Ghaoui. Full Regularization Path for Sparse Principal Component Analysis. In Proceedings of the International Conference on Machine Learning (ICML), pages 177?184, 2007. [4] Jerome Friedman, Trevor Hastie, Holger H?ofling, and Robert Tibshirani. Pathwise Coordinate Optimization. The Annals of Applied Statistics, 1(2):302?332, 2007. [5] Bernd G?artner, Martin Jaggi, and Clement Maria. An Exponential Lower Bound on the Complexity of Regularization Paths. arXiv.org, arXiv:0903.4817v, 2010. [6] Joachim Giesen, Martin Jaggi, and S?oren Laue. Approximating Parameterized Convex Optimization Problems. In Proceedings of Annual European Symposium on Algorithms (ESA), pages 524?535, 2010. [7] Joachim Giesen, Martin Jaggi, and S?oren Laue. Optimizing over the Growing Spectrahedron. In Proceedings of Annual European Symposium on Algorithms (ESA), pages 503?514, 2012. [8] Joachim Giesen, Martin Jaggi, and S?oren Laue. Regularization Paths with Guarantees for Convex Semidefinite Optimization. In Proceedings International Conference on Artificial Intelligence and Statistics (AISTATS), pages 432?439, 2012. [9] Bin Gu, Jian-Dong Wang, Guan-Sheng Zheng, and Yue cheng Yu. Regularization Path for ?Support Vector Classification. IEEE Transactions on Neural Networks and Learning Systems, 23(5):800?811, 2012. [10] Trevor Hastie, Saharon Rosset, Robert Tibshirani, and Ji Zhu. The entire regularization path for the support vector machine. The Journal of Machine Learning Research, 5:1391?1415, 2004. [11] Martin Jaggi and Marek Sulovsk?y. A Simple Algorithm for Nuclear Norm Regularized Problems. In Proceedings of the International Conference on Machine Learning (ICML), pages 471?478, 2010. [12] Masayuki Karasuyama and Ichiro Takeuchi. Suboptimal Solution Path Algorithm for Support Vector Machine. In Proceedings of the International Conference on Machine Learning (ICML), pages 473?480, 2011. [13] S?oren Laue. A hybrid algorithm for convex semidefinite optimization. In Proceedings of the International Conference on

Machine Learning (ICML), 2012. [14] Julien Mairal and Bin Yu. Complexity Analysis of the Lasso Regularization Path. In Proceedings of the International Conference on Machine Learning (ICML), 2012. [15] A. Wayne Roberts and Dale Varberg. Convex functions. Academic Press, New York, 1973. [16] Saharon Rosset and Ji Zhu. Piecewise linear regularized solution paths. Annals of Statistics, 35(3):1012?1030, 2007. [17] Nathan Srebro, Jason D. M. Rennie, and Tommi Jaakkola. Maximum-Margin Matrix Factorization. In Proceedings of Advances in Neural Information Processing Systems 17 (NIPS), 2004. [18] Zhi-li Wu, Aijun Zhang, Chun-hung Li, and Agus Sudjianto. Trace Solution Paths for SVMs via Parametric Quadratic Programming. In KDD Worskshop: Data Mining Using Matrices and Tensors, 2008.

9