An Equivalence between the Lasso and Support Vector Machines

Martin Jaggi JAGGI@CMAP.POLYTECHNIQUE.FR
CMAP, École Polytechnique, Palaiseau, France

Abstract

We investigate the relation of two fundamental tools in machine learning, that is the support vector machine (SVM) for classification, and the Lasso technique used in regression. We show that the resulting optimization problems are equivalent, in the following sense: Given any instance of an ℓ_2-loss soft-margin (or hard-margin) SVM, we construct a Lasso instance having the same optimal solutions, and vice versa.

In consequence, many existing optimization algorithms for both SVMs and Lasso can also be applied to the respective other problem instances. Also, the equivalence allows for many known theoretical insights for SVM and Lasso to be translated between the two settings. One such implication gives a simple kernelized version of the Lasso, analogous to the kernels used in the SVM setting. Another consequence is that the sparsity of a Lasso solution is equal to the number of support vectors for the corresponding SVM instance, and that one can use screening rules to prune the set of support vectors. Furthermore, we can relate sublinear time algorithms for the two problems, and give a new such algorithm variant for the Lasso.

1. Introduction

Large margin classification and kernel methods, and in particular the support vector machine (SVM) (Cortes & Vapnik, 1995), are among the most popular standard tools for classification. On the other hand, ℓ_1-regularized least squares regression, i.e. the Lasso estimator (Tibshirani, 1996), is one of the most widely used tools for robust regression and sparse estimation.

Along with the many successful practical applications of SVM and the Lasso in various fields, there is a vast amount of existing literature[1] on the two methods themselves, considering both theory and also algorithms for each of the two. However, the two research topics developed largely independently and were not much set into context with each other so far.

In this paper, we attempt to better relate the two problems, with two main goals in mind: We want to show that on the algorithmic side, many of the existing algorithms for each of the two problems can be set into comparison, and can be applied to the other respective problem. As a particular example of this idea, we can apply the recent sublinear time SVM algorithm by (Clarkson et al., 2010) also to any Lasso problem, resulting in a new alternative sublinear time algorithm variant for the Lasso.

As a second goal, we would like to relate and transfer theoretical results between the existing literature for SVMs and the Lasso. In this spirit, we propose a simple kernelized variant of the Lasso, being equivalent to the well-researched use of kernels in the SVM setting. Furthermore, we observe that by using our equivalence, the sparsity of a Lasso solution is equal to the number of support vectors for the corresponding SVM instance. Finally, we point out that screening rules, which are a way of pre-processing the input data in order to identify inactive variables for the Lasso, can also be applied to SVMs, to eliminate potential support vectors and thereby reducing the problem size.

Support Vector Machines. In this work, we focus on SVM large margin classifiers whose dual optimization problem is of the form

$$\min_{x \in \triangle} \|Ax\|_2^2 \ . \tag{1}$$

This includes the commonly used soft-margin SVM with ℓ_2-loss (for one or two classes, with regularized or no offset, with or without using a kernel). Here the matrix $A \in \mathbb{R}^{d \times n}$ contains all n datapoints as its columns, and $\triangle$ is the unit simplex in $\mathbb{R}^n$, being the

[1] As of October 2012, Google Scholar returned nearly 200′000 publications containing the term "Support Vector Machine", and over 10′000 for Lasso regression.

set of non-negative vectors summing up to one (i.e. probability vectors). We will explain the large margin interpretation of this optimization problem in more detail in Section 2.

Lasso. On the other hand, the Lasso (Tibshirani, 1996), is given by the quadratic program

$$\min_{x \in \blacklozenge} \|Ax - b\|_2^2 \ , \tag{2}$$

also known as the constrained variant of ℓ_1-regularized least squares regression. Here the right hand side b is a fixed vector $b \in \mathbb{R}^d$, and $\blacklozenge$ is the ℓ_1-unit-ball in $\mathbb{R}^n$. Note that if the desired ℓ_1-regularization constraint is not $\|x\|_1 \le 1$, but $\|x\|_1 \le r$ for some $r > 0$ instead, then it is enough to simply re-scale the input matrix A by a factor of $\frac{1}{r}$, in order to obtain our above formulation (2) for any general Lasso problem.

In applications of the Lasso, it is important to distinguish two alternative interpretations of the data matrix A, which defines the problem instance (2): On one hand, in the setting of *sparse regression*, the matrix A is usually called the dictionary matrix, with its columns $A_{:j}$ being the dictionary elements, and the goal being to approximate the single vector b by a combination of few dictionary vectors. On the other hand if the Lasso problem is interpreted as *feature-selection*, then each row $A_{i:}$ of the matrix A is interpreted as an input vector, and for each of those, the Lasso is approximating the response b_i to input $A_{i:}$. See e.g. (Bühlmann & van de Geer, 2011) for a recent overview of Lasso-type methods.

The Equivalence. We will prove that the two problems (1) and (2) are indeed equivalent, in the following sense: For any Lasso instance given by (A, b), we construct an equivalent SVM instance, having the same optimal solution. This will be a simple reduction preserving all objective values. On the other hand, the task of finding an equivalent Lasso instance for a given SVM appears to be a harder problem. Here we show that there always exists such an equivalent Lasso instance, and furthermore, if we are given an weakly-separating vector for the SVM, then we can explicitly construct the equivalent Lasso instance. This reduction also applies to the ℓ_2-loss soft-margin SVM, where we show that a weakly-separating vector is trivial to obtain. So our reduction does *not* require that the SVM input data is separable.

On the way to this goal, we will also explain the relation to the "non-negative" Lasso variant when the variable vector x is required to lie in the simplex, i.e.

$$\min_{x \in \triangle} \|Ax - b\|_2^2 \ . \tag{3}$$

It turns out the equivalence of the optimization problems (1) and (3) is straightforward to see. Our main contribution is to explain the relation of these two optimization problems to the original Lasso problem (2), and to study some the implications of the equivalence.

Related Work. The early work of (Girosi, 1998) has already significantly deepened the joint understanding of kernel methods and the sparse coding setting of the Lasso. Despite its title, (Girosi, 1998) is not addressing SVM classifiers, but in fact the ε-insensitive loss variant of support vector regression (SVR), which the author proves to be equivalent to a Lasso problem where ε then becomes the ℓ_1-regularization. Unfortunately, this reduction does not apply anymore when $\varepsilon = 0$, which is the case of interest for standard hinge-loss SVR, and also for SVMs in the classification setting, which are the focus of our work here.

In a different line of research, (Li et al., 2005) have studied the relation of a dual variant of the Lasso to the primal of the so called *potential SVM* originally proposed by (Hochreiter & Obermayer, 2004), which is not a classifier but a specialized method of feature selection.

In the application paper (Ghosh & Chinnaiyan, 2005) in the area of computational biology, the authors already suggested to make use of the "easier" direction of our reduction, reducing the Lasso to a very particular SVM instance. Here, the idea is to use barycentric coordinates to represent points in the ℓ_1-ball. Alternatively, this can also be interpreted as considering an SVM defined by all Lasso dictionary vectors together with their negatives ($2n$ many points). We formalize this interpretation more precisely in Section 3.1. (Ghosh & Chinnaiyan, 2005) does not address the SVM regularization parameter.

Notation. The unit simplex, the filled simplex as well as the ℓ_1-unit-ball in $\mathbb{R}^n$ are central for our investigations, and will be denoted by

$$\begin{aligned} \triangle &:= \{x \in \mathbb{R}^n \mid x \ge \mathbf{0},\ \textstyle\sum_i x_i = 1\} \ , \\ \blacktriangle &:= \{x \in \mathbb{R}^n \mid x \ge \mathbf{0},\ \textstyle\sum_i x_i \le 1\} \ , \\ \blacklozenge &:= \{x \in \mathbb{R}^n \mid \|x\|_1 \le 1\} \ . \end{aligned}$$

For a given matrix $A \in \mathbb{R}^{d \times n}$, we write $A_i \in \mathbb{R}^d$, $i \in [1..n]$ for its columns. By $\mathbf{0}$ and $\mathbf{1}$ we denote the all-zero and all-ones vectors in $\mathbb{R}^n$, and $\mathbf{I}_n$ is the $n \times n$ identity matrix. We write $(A|B)$ for the horizontal concatenation of two matrices A, B.

2. Properties of the SVM Problem

Linear classification using large margin is usually introduced the setting where we are given n datapoints $X_i \in \mathbb{R}^d$, together with their binary labels $y_i \in \pm 1$, for $i \in [1..n]$. The linear classifier separating the two point classes by the best possible margin is found by the following SVM optimization problem

$$\max_{w\in\mathbb{R}^d} \min_i \; y_i X_i^T \frac{w}{\|w\|_2} \ ,$$

i.e. maximizing the minimum distance of the points to the hyperplane given by the normal vector w, passing through the origin. In our case, the *dual* of SVM problem (1) is exactly such a large margin formulation, or more formally

$$\max_{x\in\triangle} \min_i \; A_i^T \frac{Ax}{\|Ax\|_2} \ . \tag{4}$$

This can be proved in two ways, when starting from the non-squared version of the SVM problem (1), that is $\min_{x\in\triangle} \|Ax\|_2$. Computing the simple "linearization" dual as in (Jaggi, 2013) (which can be seen as a special case of Fenchel duality) directly gives (4). Alternatively, taking the standard Lagrange dual (Boyd & Vandenberghe, 2004, Section 5) and plugging-in the KKT conditions also leads to the same expression, see e.g. (Gärtner & Jaggi, 2009, Appendix A).

Because $x \in \triangle$, any feasible point x readily gives us a candidate classifier vector $w = Ax$, represented as a convex combination of the datapoints. The datapoints corresponding to non-zero entries in x are called the *support vectors*.

A crucial and widely used observation is that both optimization problems (1) and (4) are formulated purely in terms of the inner products of the datapoints $A_i := y_i X_i$, meaning that they can directly be optimized in the *kernel* case (Cortes & Vapnik, 1995), were we only have access to the entries of the matrix $A^TA \in \mathbb{R}^{n\times n}$, but not the explicit features $A \in \mathbb{R}^{d\times n}$.

It is natural to measure the quality of an approximate solution x to the SVM problem as the attained margin, which is precisely the attained value in the above dual problem (4).

Definition 1. *A vector $w \in \mathbb{R}^d$ is called* σ-weakly-separating *for the SVM instance (1) or (4) respectively, for a parameter $\sigma \geq 0$, if it holds that*

$$A_i^T \frac{w}{\|w\|_2} \geq \sigma \quad \forall i \ ,$$

meaning that w attains a margin of separation of σ.

Any such attained margin value in (4) for some $Ax = w$ directly gives a certificate on the *duality gap* as the difference from the corresponding value $\|Ax\|$ in problem (1), making this quantity a useful stopping criterion for SVM optimizers, see e.g. (Gärtner & Jaggi, 2009; Clarkson et al., 2010). The simple perceptron algorithm (Rosenblatt, 1958) is known to return a σ-weakly-separating solution to the SVM after $O\big(\frac{1}{\varepsilon^2}\big)$ iterations, for $\varepsilon := \sigma^* - \sigma$ being the additive error, if σ^* is the optimal solution to (1) and (4).

2.1. Soft-Margin SVMs

For the successful practical application of SVMs, the soft-margin concept of tolerating outliers is of central importance. Here we recall that also the soft-margin SVM variants using ℓ_2-loss, with regularized offset or no offset, both in the one-class and the two-class case, can be formulated in the form (1). This fact is known in the SVM literature (Schölkopf & Smola, 2002; Keerthi et al., 2000; Tsang et al., 2005), and can be formalized as follows:

The two-class soft-margin SVM with squared loss is given by the optimization problem

$$\begin{aligned} \min_{\substack{\bar w\in\mathbb{R}^d,\ \rho\in\mathbb{R},\\ \xi\in\mathbb{R}^n}} \quad & \tfrac{1}{2}\|\bar w\|_2^2 - \rho + \tfrac{C}{2}\textstyle\sum_i \xi_i^2 \\ s.t. \quad & y_i \cdot \bar w^T X_i \geq \rho - \xi_i \quad \forall i \in [1..n] \ . \end{aligned} \tag{5}$$

Here $C > 0$ is the regularization parameter, and $\rho/\|\bar w\|$ is the attained margin of separation. Note that in the classical SVM formulation, the margin parameter ρ is usually fixed to one instead, while ρ is explicitly used in the equivalent ν-SVM formulation known in the literature, see e.g. (Schölkopf & Smola, 2002). The equivalence of the soft-margin SVM dual problem to the optimization problem (1) is stated in the following Lemma:

Lemma 2. *The dual of the soft-margin SVM (5) is an instance of the classifier formulation (1), that is $\min_{x\in\triangle} \|Ax\|_2^2$, with*

$$A := \begin{pmatrix} Z \\ \frac{1}{\sqrt{C}}\mathbf{I}_n \end{pmatrix} \in \mathbb{R}^{(d+n)\times n}$$

where the data matrix $Z \in \mathbb{R}^{d\times n}$ consists of the n columns $Z_i := y_i X_i$.

Proof. Given in Appendix B for completeness, using standard Lagrange duality. □

Obtaining a Weakly-Separating Vector for the ℓ_2-loss Soft-Margin SVM. By the above lemma, we observe that a weakly-separating vector is trivial to obtain for the ℓ_2-loss SVM. This holds without any assumptions on the original input data (X_i, y_i). We set $w := \big(\begin{smallmatrix}\mathbf{0}\\ \frac{1}{\sqrt{n}}\mathbf{1}\end{smallmatrix}\big) \in \mathbb{R}^{d+n}$ to the all-one vector only on the second block of coordinates, rescaled to unit length. Clearly, this direction w attains a separation margin of $A_i^T \frac{w}{\|w\|_2} = \begin{pmatrix} y_i X_i \\ \frac{1}{\sqrt{C}} \mathbf{e}_i \end{pmatrix}^T \begin{pmatrix} \mathbf{0} \\ \frac{1}{\sqrt{n}} \mathbf{1} \end{pmatrix} = \frac{1}{\sqrt{nC}} > 0$ for all points i in Definition 1.

Incorporating an Offset Term. Our above SVM formulation also allows the use of an *offset* variable $b \in \mathbb{R}$ to obtain a classifier that does not necessarily pass through the origin. Formally, the separation constraints then become $y_i \cdot (w^T X_i + b) \geq \rho - \xi_i \ \ \forall i \in [1..n]$. This formulation can easily be seen to be equivalent to (5), by the standard trick of increasing the dimensionality of X_i and w by one, and adding a fixed value of one as the last coordinate to each of the datapoints X_i, see e.g. (Keerthi et al., 2000; Tsang et al., 2005). As a side-effect, the offset b^2 is then also regularized in the new term $\|w\|_2^2$. Nevertheless, if desired, the effect of this additional regularization can be made arbitrary weak by re-scaling the fixed additional feature value of from one to a larger value.

One-class SVMs. All mentioned properties in this section also hold for the case of *one-class SVMs*, by setting all labels to one, resulting in the same form of optimization problems (1) and (4).

3. The Equivalence

Before we investigate the "real" Lasso problem (2) in the next two subsections, we will warm-up by considering the non-negative variant (3). It is a simple observation that the non-negative Lasso (3) is directly equivalent to the SVM problem (1) by a translation:

Translations, and the Equivalence of (1) and (3). By translating each column vector of the matrix A by the vector $-b$, any instance of (3) becomes precisely an SVM instance (1), with the data matrix being $\tilde{A} := A - b\mathbf{1}^T \in \mathbb{R}^{d \times n}$. Here we have crucially used the simplex domain, ensuring that $b\mathbf{1}^T x = b$ for any $x \in \triangle$. Note that the translation precisely preserves the objective values of all x. The reduction in the other direction is trivial by choosing $b := \mathbf{0}$.

3.1. (Lasso $\preceq$ SVM): Given a Lasso instance, construct an equivalent SVM instance

(This reduction is significantly easier than the other direction.)

Parameterizing the ℓ_1-Ball as a Convex Hull. In order to represent the ℓ_1-ball $\blacklozenge$ by a simplex $\triangle$, the standard concept of *barycentric* coordinates comes to help, stating that every polytope can be represented as the convex hull of its vertices (Ziegler, 1995). The ℓ_1-ball $\blacklozenge$ is the convex hull of its $2n$ vertices, which are $\{\pm \mathbf{e}_i \mid i \in [1..n]\}$, illustrating why $\blacklozenge$ is also called the cross-polytope.

The barycentric representation of the ℓ_1-ball therefore amounts to using two non-negative variables to "represent" a real variable, which can be formalized as follows: Any n-vector $x_\diamond \in \blacklozenge$ can be written as $x_\diamond = (\mathbf{I}_n \,|\, {-\mathbf{I}_n}) x_\triangle$ for $x_\triangle \in \mathbb{R}^{2n}, x_\triangle \in \triangle$. Note that the barycentric representation is usually not a bijection, as there might be several $x_\triangle \in \triangle$ representing the same $x_\diamond \in \mathbb{R}^n$.

The Equivalent SVM Instance. Given a Lasso instance of the form (2), that is, $\min_{x \in \blacklozenge} \|Ax - b\|_2^2$, we can directly parameterize the ℓ_1-ball by the $2n$-dimensional simplex as described above. By writing $(\mathbf{I}_n \,|\, {-\mathbf{I}_n}) x_\triangle$ for any $x \in \blacklozenge$, the objective function becomes $\|(A \,|\, {-A}) x_\triangle - b\|_2^2$. This means we have obtained the equivalent non-negative regression problem of the form (3) over the domain $x_\triangle \in \triangle$ which, by our above remark on translations, is equivalent to the SVM formulation (1), i.e.

$$\min_{x_\triangle \in \triangle} \left\| \tilde{A} x_\triangle \right\|_2^2 ,$$

where the data matrix is given by $\tilde{A} := (A \,|\, {-A}) - b\mathbf{1}^T \in \mathbb{R}^{d \times 2n}$. The additive rank-one term $b\mathbf{1}^T$ for $\mathbf{1} \in \mathbb{R}^{2n}$ again just means that the vector b is subtracted from each original column of A and $-A$. So we have obtained an equivalent SVM instance consisting of $2n$ points in $\mathbb{R}^d$.

Note that this equivalence not only means that the optimal solutions of the Lasso and the SVM coincide, but indeed gives us the one-to-one correspondence of all feasible solutions, preserving the objective values: For any feasible solution $x \in \mathbb{R}^n$ to the Lasso, we have a feasible SVM solution $x_\triangle \in \mathbb{R}^{2n}$ of the *same* objective value, and vice versa.

3.2. (SVM $\preceq$ Lasso): Given an SVM instance, constructing an equivalent Lasso instance

This reduction is harder to accomplish than the other direction we explained before. Given an instance of an SVM problem (1), we suppose that we have a (possibly non-optimal) σ-weakly-separating vector $w \in \mathbb{R}^d$ available, for some (small) value $\sigma > 0$. Given w, we will demonstrate in the following how to construct an equivalent Lasso instance (2).

Perhaps surprisingly, such a weakly-separating vector w is trivial to obtain for the ℓ_2-loss soft-margin SVM, as we have observed in Section 2.1 (even if the SVM input data is not separable). Also for other SVM variants, finding such a weakly-separating vector for a small σ is still significantly easier than the final goal of obtaining a near-perfect $(\sigma^* - \varepsilon)$-separation for a small precision ε. It corresponds to running an SVM solver (such as the perceptron algorithm) for only constantly many iterations. In contrast, obtaining a better ε-accurate solution by the same algorithm would require $O\big(\frac{1}{\varepsilon^2}\big)$ iterations, as mentioned in Section 2.

The Equivalent Lasso Instance. Formally, we define the Lasso instance $(\tilde{A}, \tilde{b})$ as the translated SVM datapoints $\tilde{A} := \Big\{ A_i + \tilde{b} \;\Big|\; i \in [1..n] \Big\}$ together with the right hand side $\tilde{b} := -\frac{w}{\|w\|_2} \cdot \frac{D^2}{\sigma}$.

Here $D > 0$ is a strict upper bound on the length of the original SVM datapoints, i.e. $\|A_i\|_2 < D \quad \forall i$. By definition of $\tilde{A}$, the resulting new Lasso objective function is

$$\begin{aligned} \Big\|\tilde{A}x - \tilde{b}\Big\|_2 &= \quad \Big\|(A + \tilde{b}\mathbf{1}^T)x - \tilde{b}\Big\|_2 \\ &= \Big\|Ax + (\mathbf{1}^T x - 1)\tilde{b}\Big\|_2 . \end{aligned} \tag{6}$$

Therefore, this objective coincides with the original SVM objective (1), for any $x \in \triangle$ (meaning that $\mathbf{1}^T x = 1$). However, this does not necessarily hold for the larger part of the Lasso domain when $x \in \blacklozenge \setminus \triangle$. In the following discussion and the main Theorem 3, we will prove that all those candidates $x \in \blacklozenge \setminus \triangle$ can be discarded from the Lasso problem, as they do not contribute to any optimal solutions.

As a side-remark, we note that the quantity $\frac{D}{\sigma}$ that determines the magnitude of our translation is a known parameter measuring the "difficulty" of the SVM instance, essentially its VC-dimension (Burges, 1998; Schölkopf & Smola, 2002).

Geometric Intuition. Geometrically, the Lasso problem (2) is to compute the smallest Euclidean distance of the set $A\blacklozenge$ to the point $b \in \mathbb{R}^d$. On the other hand the SVM problem — after translating by b — is to minimize the distance of the smaller set $A\triangle \subset A\blacklozenge$ to the point b. Here we have used the notation $AS := \{Ax \mid x \in S\}$ for subsets $S \subseteq \mathbb{R}^d$ and linear maps A (it is easy to check that linear maps do preserve convexity of sets, so that $\operatorname{conv}(AS) = A \operatorname{conv}(S)$).

Intuitively, the main idea of our reduction is to mirror our SVM points A_i at the origin, such that both the points and their mirrored copies — and therefore the entire larger polytope $A\blacklozenge$ — do end up lying "behind" the separating SVM margin. The hope is that the resulting Lasso instance will have all its optimal solutions being non-negative, and lying in the simplex. Surprisingly, this can be done, and we will show that all SVM solutions are preserved (and no new solutions are introduced) when the feasible set $\triangle$ is extended to $\blacklozenge$. In the following we will formalize this precisely, and demonstrate how to translate along our known weakly-separating vector w such that the resulting Lasso problem will have the same solution as the original SVM.

Properties of the Constructed Lasso Instance. The following theorem shows that for our constructed Lasso instance, all interesting feasible solutions are contained in the simplex $\triangle$. By our previous observation (6), we already know that all those candidates are feasible for both the Lasso (2) and the SVM (1), and obtain the same objective values in both problems.

In other words, we have a one-to-one correspondence between all feasible points for the SVM (1) on one hand, and the subset $\triangle \subset \blacklozenge$ of feasible points of our constructed Lasso instance (2), preserving all objective values. Furthermore, we have that in this Lasso instance, all points in $\blacklozenge \setminus \triangle$ are strictly worse than the ones in $\triangle$. Therefore, we have also shown that all optimal solutions must coincide.

Theorem 3. *For any candidate solution $x_\diamond \in \blacklozenge$ to the Lasso problem (2) defined by $(\tilde{A}, \tilde{b})$, there is a feasible vector $x_\triangle \in \triangle$ in the simplex, of the same or better Lasso objective value γ.*
Furthermore, this $x_\triangle \in \triangle$ attains the same objective value γ in the original SVM problem (1).

The proof of the following two main propositions is given in Appendix A, and makes use of the defined the translation $\tilde{b}$ along a weakly separating vector w.

Proposition 4 (Flipping improves the objective)**.** *Consider the Lasso problem (2) defined by $(\tilde{A}, \tilde{b})$, and assume that $x_\diamond \in \blacklozenge$ has some negative entries.*

Then there is a strictly better solution $x_\blacktriangle \in \blacktriangle$ having only non-negative entries.

Proposition 5 (Scaling up improves for non-negative vectors)**.** *Consider the Lasso problem (2) defined by $(\tilde{A}, \tilde{b})$, and assume that $x_\blacktriangle \in \blacktriangle$ has $\|x_\blacktriangle\|_1 < 1$.*

Then we obtain a strictly better solution $x_\triangle \in \triangle$ by linearly scaling $x_\blacktriangle$.

Proof of Theorem 3. By Propositions 4 and 5, and assumed that $x_\diamond$ does not already lie in the simplex, we have $x_\triangle \in \triangle$, of a strictly better objective value γ for problem (3). By the observation (6) about the Lasso objective, we know that the original SVM objective attained by this $x_\triangle$ is equal to γ. $\square$

4. Implications & Remarks

4.1. Some Implications for the Lasso

Using only the "easy" direction (Lasso $\preceq$ SVM) of our reduction in Section 3.1, we obtain the following insights into the Lasso problem:

Sublinear Time Algorithms. The recent breakthrough SVM algorithm of (Clarkson et al., 2010; Hazan et al., 2011) in time $O(\varepsilon^{-2}(n+d)\log n)$ returns an ε-accurate solution to problem (1). Here ε-accurate means $(\sigma^* - \varepsilon)$-weakly-separating. The running time of the algorithm is remarkable since it is significantly smaller than even the size of the input matrix, being $d \cdot n$. therefore the algorithm does not read the full input matrix $\tilde{A}$. More precisely, (Clarkson et al., 2010, Corollary III.2) proves that the algorithm provides (w.h.p.) a solution $p^* \in \triangle$ of additive error at most ε to

$$\min_{p\in\triangle} \max_{\substack{w\in\mathbb{R}^d,\\ \|w\|_2\le 1}} w^T\tilde{A}p\ .$$

This is a reformulation of $\min_{p\in\triangle} p^T\tilde{A}^T\tilde{A}p$, which is exactly our SVM problem (1), since for given p, the inner maximum is attained when $w = \tilde{A}p$. Therefore, using our simple trick from Section 3.1 of reducing any Lasso instance (2) to an SVM (1) (with its matrix $\tilde{A}$ having twice the number of columns as A), we directly obtain a sublinear time algorithm for the Lasso. Note that since the algorithm of (Clarkson et al., 2010; Hazan et al., 2011) only accesses the matrix $\tilde{A}$ by simple entry-wise queries, it is not necessary to explicitly compute and store $\tilde{A}$ (which is a preprocessing that would need linear time and storage). Instead, every entry $\tilde{A}_{ij}$ that is queried by the algorithm can be provided on the fly, by returning the corresponding (signed) entry of the Lasso matrix A, minus b_i.

It will be interesting to compare this alternative algorithm to the recent more specialized sublinear time Lasso solvers in the line of work of (Cesa-Bianchi et al., 2011; Hazan & Koren, 2012), which are only allowed to access a constant fraction of the entries (or features) of each row of A. If we use our proposed reduction here instead, the resulting algorithm from (Clarkson et al., 2010) has more freedom: it can (randomly) pick arbitrary entries of A, without necessarily accessing an equal number of entries from each row.

A Lasso in Kernel Space. Traditional kernel regression techniques (Smola & Schölkopf, 2004; Saunders et al., 1998; Roth, 2004) try to learn a real-valued function f from the space $\mathbb{R}^d$ of the datapoints, such that the resulting real value for each datapoint approximates some observed value. The function f is chosen to be a linear combination the (kernel) inner products to few existing datapoints in the kernel space.

Here, as we present a kernelization of the Lasso that is in complete analogy to the classical kernel trick for SVMs, our goal is different. We are not trying to approximate n many individual real values (one for each datapoint, or row of A), but instead we are searching for a linear combination of our points in kernel space, such that the resulting combination is *close* to the lifted point b, measured in the kernel space norm. Formally, suppose our kernel space $\mathcal{H}$ is given by an inner product $\kappa(y,z) = \langle\Psi(y),\Psi(z)\rangle$ for some implicit mapping $\Psi : \mathbb{R}^d \to \mathcal{H}$. Then we define our kernelized variant of the Lasso as

$$\min_{x\in\blacklozenge} \left\| \sum_i \Psi(A_i)x_i - \Psi(b) \right\|_{\mathcal{H}}^2\ . \tag{7}$$

Nicely, analogous to the SVM case, also this objective function here is determined purely in terms of the pairwise (kernel) inner products $\kappa(.,.)$.

An alternative way to see this is to observe that our simple "mirror-and-translate" trick from Section 3.1 also works the very same way in any kernel space $\mathcal{H}$. Here, the equivalent SVM instance is given by the $2n$ new points $\{\pm\Psi(A_i) - \Psi(b) \mid i \in [1..n]\} \subset \mathcal{H}$. The crucial observation is that the (kernel) inner product of any two such points is

$$\begin{aligned} & \langle s_i\Psi(A_i) - \Psi(b), s_j\Psi(A_j) - \Psi(b)\rangle \\ = \ & s_is_j\kappa(A_i,A_j) - s_i\kappa(A_i,b) - s_j\kappa(A_j,b) + \kappa(b,b)\ . \end{aligned}$$

Here $s_i, s_j \in \pm 1$ are the signs corresponding to each point. Therefore we have completely determined the resulting $2n\times 2n$ kernel matrix K that defines the kernelized SVM (1), namely $\min_{x\in\triangle} x^TKx$, which solves our equivalent Lasso problem (7) in the kernel space $\mathcal{H}$.

Discussion. While traditional kernel regression corresponds to a lifting of the *rows* of the Lasso matrix A into the kernel space, our approach (7) by contrast is lifting the *columns* of A (and the r.h.s. b). We note that it seems indeed counter-intuitive to make the regression "more difficult" by artificially increasing the dimension of b. Using e.g. a polynomial kernel, this means that we also want the higher moments of b to be well approximated by our estimated x. On the other hand, increasing the dimension of b naturally corresponds to adding more data rows (or measurements) to a classical Lasso instance (2).

In the light of the success of the kernel idea for the classification case with its existing well-developed theory, we think it will be interesting to relate these results to the above proposed kernelized version of the Lasso, and to study how different kernels will affect the solution x for applications of the Lasso. Independently of our work, (Thiagarajan et al., 2012) has very recently proposed a similar kernelization idea for the application of image retrieval.

4.2. Some Implications for SVMs

The more complicated direction of our reduction (SVM $\preceq$ Lasso) from Section 3.2 also gives a promising approach towards some new insights into SVMs:

Structure and Sparsity of the Support Vectors, in the View of Lasso Sparsity. There has been a large amount of new literature studying the sparsity of solutions to the Lasso and related ℓ_1-regularized methods, in particular the study of the sparsity of x when A and b are from distributions with certain properties. For example, in the setting known as *sparse recovery*, the goal is to approximately recover a sparse solution x using instances A, b (consisting of only a small number of rows), where b is interpreted as a noisy or corrupted measurement of Ax, see e.g. (Chen et al., 2001; Porat & Strauss, 2012).

Using our construction of the equivalent Lasso instance for a given SVM, such results then directly apply to the sparsity pattern of the solution to our original SVM (i.e. the pattern and the number of support vectors). More precisely, any result giving a distribution of matrices A and corresponding b for which the Lasso sparsity is well characterized, will also characterize the patterns of support vectors for the equivalent SVM instance (and in particular the number of support vectors). This assumes that a Lasso sparsity result is applicable for the type of translation b that we have used here in order to construct our equivalent Lasso instance. However, this is not hopeless, since the only special property of our constructed Lasso instance is that the r.h.s. b is strongly correlated with all columns of A. It remains to investigate which distributions and corresponding sparsity results would be of most interest for the SVM perspective.

Screening Rules for Support Vector Machines. For the Lasso, *screening rules* have been developed recently. This approach consists of a single preprocessing pass though the data A, in order to immediately discard those predictors A_i that can be guaranteed to be inactive for the optimal solution (Ghaoui et al., 2010; Tibshirani et al., 2011). Translated to the SVM setting by our reduction, any such Lasso screening rule can be used to permanently discard input points before the SVM optimization is started. The screening rule then guarantees that any discarded point will not be a support vector, so the resulting optimal classifier remains unchanged. We are not aware of screening rules in the SVM literature so far.

5. Conclusions

We have investigated the relation between the Lasso and SVMs, and constructed equivalent instances of the respective other problem. While obtaining an equivalent SVM instance for a given Lasso is straightforward, the other direction is slightly more involved (but efficient e.g. for ℓ_2-loss SVMs). The two reductions allow us to better relate and compare many existing algorithms for both problems. In the future, by transferring more of the existing rich theory between the two popular methods, we hope that the understanding of both of them can be further deepened.

References

Boyd, Stephen P and Vandenberghe, Lieven. *Convex optimization*. 2004.

Bühlmann, Peter and van de Geer, Sara. *Statistics for High-Dimensional Data - Methods, Theory and Applications*. Springer Series in Statistics, 2011. ISBN 978-3-642-20191-2.

Burges, Christopher. A Tutorial on Support Vector Machines for Pattern Recognition. *Data Mining and Knowledge Discovery*, 2(2):121–167, 1998.

Cesa-Bianchi, Nicolò, Shalev-Shwartz, Shai, and Shamir, Ohad. Efficient Learning with Partially Observed Attributes. *The Journal of Machine Learning Research*, 12, 2011.

Chen, Scott Shaobing, Donoho, David L, and Saunders, Michael A. Atomic Decomposition by Basis Pursuit. *SIAM Review*, 43(1):129–159, 2001.

Clarkson, Kenneth L, Hazan, Elad, and Woodruff, David P. Sublinear Optimization for Machine Learning. *FOCS 2010 - 51st Annual IEEE Symposium on Foundations of Computer Science*, 2010.

Cortes, Corinna and Vapnik, Vladimir. Support-Vector Networks. *Machine Learning*, 20(3):273–297, 1995.

Gärtner, Bernd and Jaggi, Martin. Coresets for polytope distance. *SCG '09: Proceedings of the 25th annual Symposium on Computational Geometry*, 2009.

Ghaoui, Laurent El, Viallon, Vivian, and Rabbani, Tarek. Safe Feature Elimination for the LASSO and Sparse Supervised Learning Problems. *arXiv.org*, cs.LG, 2010.

Ghosh, Debashis and Chinnaiyan, Arul M. Classification and Selection of Biomarkers in Genomic Data Using LASSO. *Journal of Biomedicine and Biotechnology*, 2005(2):147–154, 2005.

Girosi, Federico. An Equivalence Between Sparse Approximation and Support Vector Machines. *Neural Computation*, 10(6):1455–1480, 1998.

Hazan, Elad and Koren, Tomer. Linear Regression with Limited Observation. In *ICML 2012: Proceedings of the 29th International Conference on Machine Learning*, 2012.

Hazan, Elad, Koren, Tomer, and Srebro, Nathan. Beating SGD: Learning SVMs in Sublinear Time. In *NIPS*, 2011.

Hochreiter, Sepp and Obermayer, Klaus. Gene Selection for Microarray Data. In Schölkopf, B, Vert, J-P, and Tsuda, K (eds.), *Kernel Methods in Computational Biology*, pp. 319. MIT Press, 2004. ISBN 0262195097.

Jaggi, Martin. Revisiting Frank-Wolfe: Projection-Free Sparse Convex Optimization. In *ICML 2013: Proceedings of the 30th International Conference on Machine Learning*, 2013.

Keerthi, S Sathiya, Shevade, Shirish K, Bhattacharyya, Chiranjib, and Murthy, K R K. A fast iterative nearest point algorithm for support vector machine classifier design. *IEEE Transactions on Neural Networks*, 11(1): 124–136, 2000.

Li, Fan, Yang, Yiming, and Xing, Eric P. From Lasso regression to Feature vector machine. In *NIPS*, 2005.

Porat, Ely and Strauss, Martin J. Sublinear time, measurement-optimal, sparse recovery for all. In *SODA '12: Proceedings of the Twenty-Third Annual ACM-SIAM Symposium on Discrete Algorithms*. SIAM, 2012.

Rosenblatt, Frank. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386–408, 1958.

Roth, Volker. The Generalized LASSO. *IEEE Transactions on Neural Networks*, 15(1):16–28, 2004.

Saunders, Craig, Gammerman, Alexander, and Vovk, Volodya. Ridge Regression Learning Algorithm in Dual Variables. In *ICML '98: Proceedings of the Fifteenth International Conference on Machine Learning*, 1998.

Schölkopf, Bernhard and Smola, Alex J. *Learning with kernels*. support vector machines, regularization, optimization, and beyond. The MIT Press, 2002. ISBN 9780262194754.

Smola, Alex J and Schölkopf, Bernhard. A tutorial on support vector regression. *Statistics and Computing*, 14: 199–222, 2004.

Thiagarajan, Jayaraman J, Ramamurthy, Karthikeyan Natesan, and Spanias, Andreas. Local Sparse Coding for Image Classification and Retrieval. Technical report, 2012.

Tibshirani, Robert. Regression Shrinkage and Selection via the Lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 267–288, 1996.

Tibshirani, Robert, Bien, Jacob, Friedman, Jerome, Hastie, Trevor, Simon, Noah, Taylor, Jonathan, and Tibshirani, Ryan J. Strong rules for discarding predictors in lasso-type problems. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 74(2): 245–266, 2011.

Tsang, Ivor W, Kwok, James T, and Cheung, Pak-Ming. Core Vector Machines: Fast SVM Training on Very Large Data Sets. *Journal of Machine Learning Research*, 6:363–392, 2005.

Ziegler, Günter M. *Lectures on Polytopes*, volume 152 of *Graduate Texts in Mathematics*. Springer, 1995. ISBN 978-0-387-94329-9.

A. Proof of the Reduction from SVM to Lasso

As described in Section 3.2, here we assume that $w \in \mathbb{R}^d$ is some σ-weakly-separating vector for the original SVM problem (1) for $\sigma > 0$. Then we study the Lasso instance $(\tilde{A}, \tilde{b})$ defined above, where the translation vector is defined as $\tilde{b} := -\frac{w}{\|w\|_2} \cdot \frac{D^2}{\sigma}$.

Proposition 4 (Flipping improves the objective). *Consider the Lasso problem (2) defined by $(\tilde{A}, \tilde{b})$, and assume that $x_\diamond \in \blacklozenge$ has some negative entries.*

Then there is a strictly better solution $x_\blacktriangle \in \blacktriangle$ having only non-negative entries.

Proof of Proposition 4. We are given $x_\diamond \neq 0$, having at least one negative coordinate. Define $x_\blacktriangle \neq \mathbf{0}$ as the vector you get by flipping all the negative coordinates in $x_\diamond$. We define $\delta \in \blacktriangle$ to be the difference vector corresponding to this flipping, i.e. $\delta_i := -(x_\diamond)_i$ if $(x_\diamond)_i < 0$, and $\delta_i := 0$ otherwise, so that $x_\blacktriangle := x_\diamond + 2\delta$ gives $x_\blacktriangle \in \blacktriangle$. We want to show that with respect to the quadratic objective function, $x_\blacktriangle$ is strictly better than $x_\diamond$. We do this by showing that the following difference in the objective values is strictly negative:

$$\begin{aligned} & \left\|\tilde{A}x_\blacktriangle - \tilde{b}\right\|_2^2 - \left\|\tilde{A}x_\diamond - \tilde{b}\right\|_2^2 \\ = \; & \|c+d\|_2^2 - \|c\|_2^2 \\ = \; & c^Tc + 2c^Td + d^Td - c^Tc \; = \; (2c+d)^Td \\ = \; & 4(\tilde{A}x_\diamond - \tilde{b} + \tilde{A}\delta)^T\tilde{A}\delta \\ = \; & 4(\tilde{A}(x_\diamond + \delta) - \tilde{b})^T\tilde{A}\delta \end{aligned}$$

where in the above calculations we have used that $\tilde{A}x_\blacktriangle = \tilde{A}x_\diamond + 2\tilde{A}\delta$, and we substituted $c := \tilde{A}x_\diamond - \tilde{b}$ and $d := 2\tilde{A}\delta$. Interestingly, $x_\diamond + \delta \in \blacktriangle$, since this addition just sets all previously negative coordinates to zero.

The proof then follows from Lemma 8 below. □

Proposition 5 (Scaling up improves for non-negative vectors). *Consider the Lasso problem (2) defined by $(\tilde{A}, \tilde{b})$, and assume that $x_\blacktriangle \in \blacktriangle$ has $\|x_\blacktriangle\|_1 < 1$.*

Then we obtain a strictly better solution $x_\triangle \in \triangle$ by linearly scaling $x_\blacktriangle$.

Proof of Proposition 5. The proof follows along similar lines as the above proposition. We are given $x_\blacktriangle \neq 0$ with $\|x_\blacktriangle\|_1 < 1$. Define $x_\triangle$ as the vector we get by scaling up $x_\triangle := \lambda x_\blacktriangle$ by $\lambda > 1$ such that $\|x_\triangle\|_1 = 1$. We want to show that with respect to the quadratic objective function, $x_\triangle$ is strictly better than $x_\blacktriangle$. As in the previous proof, we again do this by showing that the following difference in the objective values is strictly negative:

$$\begin{aligned} & \left\|\tilde{A}x_\triangle - \tilde{b}\right\|_2^2 - \left\|\tilde{A}x_\blacktriangle - \tilde{b}\right\|_2^2 \\ = \; & \|c+d\|_2^2 - \|c\|_2^2 \\ = \; & c^Tc + 2c^Td + d^Td - c^Tc \; = \; (2c+d)^Td \\ = \; & \lambda'(2\tilde{A}x_\blacktriangle - 2b + \lambda'\tilde{A}x_\blacktriangle)^T\tilde{A}x_\blacktriangle \\ = \; & 2\lambda'(\tilde{A}\big(1 + \tfrac{\lambda'}{2}\big)x_\blacktriangle - \tilde{b})^T\tilde{A}x_\blacktriangle \end{aligned}$$

where in the above calculations we have used that $\tilde{A}x_\triangle = \lambda\tilde{A}x_\blacktriangle$ for $\lambda > 1$, and we substituted $c := \tilde{A}x_\blacktriangle - \tilde{b}$ and $d := \tilde{A}x_\triangle - \tilde{A}x_\blacktriangle = (\lambda - 1)\tilde{A}x_\blacktriangle =: \lambda'\tilde{A}x_\blacktriangle$ for $\lambda' := \lambda - 1 > 0$. Note that $x_\triangle := (1 + \lambda')x_\blacktriangle \in \triangle$ so $\big(1 + \frac{\lambda'}{2}\big)x_\blacktriangle \in \blacktriangle$.

The proof then follows from Lemma 8 below. □

Definition 6. *For a given axis vector $w \in \mathbb{R}^d$, the* cone *with axis w, angle $\alpha \in (0, \frac{\pi}{2})$ with tip at the origin is defined as* $\text{cone}(w, \alpha) := \left\{x \in \mathbb{R}^d \;\middle|\; \measuredangle(x, w) \leq \alpha\right\}$*, or equivalently $\frac{x^Tw}{\|x\|_2\|w\|_2} \geq \cos\alpha$. By $\mathring{\text{cone}}(w, \alpha)$ we denote the interior of the convex set $\text{cone}(w, \alpha)$, including the tip $\mathbf{0}$.*

Lemma 7 (Separation). *Let w be some σ-weakly-separating vector for the SVM (1) for $\sigma > 0$. Then*

i) $A\blacktriangle \subseteq \mathring{\text{cone}}(w, \arccos(\frac{\sigma}{D}))$

ii) Any vector in $\text{cone}(w, \arcsin(\frac{\sigma}{D}))$ is still σ'-weakly-separating for A for some $\sigma' > 0$.

Proof. i) Definition 1 of weakly separating, and using that $\|A_i\|_2 < D$.

ii) For any unit length vector $v \in \text{cone}(w, \arcsin(\frac{\sigma}{D}))$, every other vector having a zero or negative inner product with this v must have angle at least $\frac{\pi}{2} - \arcsin(\frac{\sigma}{D}) = \arccos(\frac{\sigma}{D})$ with the cone axis w. However, by using i), we have $A\triangle \subseteq \mathring{\text{cone}}(w, \arccos(\frac{\sigma}{D}))$, so every column vector of A must have strictly positive inner product with v, or in other words v is σ'-weakly-separating for A (for some $\sigma' > 0$). □

Lemma 8. *Let w be some σ-weakly-separating vector for the SVM for $\sigma > 0$. Then we claim that the translation by the vector $\tilde{b} := -\frac{w}{\|w\|_2} \cdot \frac{D^2}{\sigma}$ has the following properties: For any pair of vectors $x, \delta \in \blacktriangle, \delta \neq \mathbf{0}$, we have that $(\tilde{A}x - \tilde{b})^T(-\tilde{A}\delta) > 0$.*

Proof. By definition of the translation $\tilde{b}$, we have that the entire Euclidean ball of radius D around the point $\tilde{b}$ — and therefore also the point set $-\tilde{A}\blacktriangle$ and in particular $v := -\tilde{A}\delta$ — is contained in $\text{cone}(w, \arcsin(\frac{D}{\sigma}))$.

Therefore by Lemma 7 ii), v is separating for A, and by translation v also separates $\tilde{A}$ from $\tilde{b}$. This establishes the result $(\tilde{A}x - \tilde{b})^T v > 0$ for any $x \in \triangle$.

To extend this to the case $x \in \blacktriangle$, we observe that by definition of $\tilde{b}$, also the point $\mathbf{0} - \tilde{b}$ has strictly positive inner product with v. Therefore the entire convex hull of $\tilde{A}\triangle \cup \mathbf{0}$ and thus the set $\tilde{A}\blacktriangle$ has the desired property. □

B. Some Soft-Margin SVM Variants that are Equivalent to (1)

We include the derivation of the dual formulation to the ℓ_2-loss soft-margin SVM (5) for n datapoints $X_i \in \mathbb{R}^d$, together with their binary class labels $y_i \in \pm 1$, for $i \in [1..n]$, as defined above in Section 2.1.

The equivalence to (1) directly extends to the one- and two-class case, without or with (regularized) offset term, and as well for the hard-margin SVM. These equivalent formulations have been known in the SVM literature, see e.g. (Schölkopf & Smola, 2002; Keerthi et al., 2000; Tsang et al., 2005; Gärtner & Jaggi, 2009), and the references therein.

Lemma 2. *The dual of the soft-margin SVM (5) is an instance of the classifier formulation (1), that is* $\min_{x \in \triangle} \|Ax\|_2^2$ *, with*

$$A := \begin{pmatrix} Z \\ \frac{1}{\sqrt{C}} \mathbf{I}_n \end{pmatrix} \in \mathbb{R}^{(d+n) \times n}$$

where the data matrix $Z \in \mathbb{R}^{d \times n}$ *consists of the* n *columns* $Z_i := y_i X_i$.

Proof. The Lagrangian (Boyd & Vandenberghe, 2004, Section 5) of the soft-margin SVM formulation (5) with its n constraints can be written as

$$\begin{aligned} L(w, \rho, \xi, \alpha) := \quad & \tfrac{1}{2} \|w\|_2^2 - \rho + \tfrac{C}{2} \textstyle\sum_i \xi_i^2 \\ & + \textstyle\sum_i \alpha_i \left(-w^T Z_i + \rho - \xi_i \right) \ . \end{aligned}$$

Here we introduced a non-negative Lagrange multiplier $\alpha_i \geq 0$ for each of the n constraints. Differentiating L with respect to the primal variables, we obtain the KKT optimality conditions

$$\begin{aligned} \mathbf{0} \stackrel{!}{=} \ & \tfrac{\partial}{\partial w} & = \ & w - \textstyle\sum_i \alpha_i Z_i \\ 0 \stackrel{!}{=} \ & \tfrac{\partial}{\partial \rho} & = \ & 1 - \textstyle\sum_i \alpha_i \\ \mathbf{0} \stackrel{!}{=} \ & \tfrac{\partial}{\partial \xi} & = \ & C\xi - \alpha \ . \end{aligned}$$

When plugged into the Lagrange dual problem $\max_\alpha \min_{w,\rho,\xi} L(w, \rho, \xi, \alpha)$, these give us the equivalent formulation

$$\begin{aligned} \max_\alpha \quad & \tfrac{1}{2} \alpha^T Z^T Z \alpha - \rho + \tfrac{C}{2} \tfrac{1}{C^2} \alpha^T \alpha \\ & \quad - \alpha^T Z^T Z \alpha + \rho - \tfrac{1}{C} \alpha^T \alpha \\ & = -\tfrac{1}{2} \alpha^T Z^T Z \alpha - \tfrac{1}{2C} \alpha^T \alpha \ . \end{aligned}$$

In other words the dual is

$$\begin{aligned} \min_\alpha \quad & \alpha^T \left(Z^T Z + \tfrac{1}{C} \mathbf{I}_n \right) \alpha \\ s.t. \quad & \alpha \geq 0 \\ & \alpha^T \mathbf{1} = 1 \ . \end{aligned}$$

This is directly an instance of our first SVM formulation (1) used in the introduction, if we use the extended matrix

$$A := \begin{pmatrix} Z \\ \frac{1}{\sqrt{C}} \mathbf{I}_n \end{pmatrix} \in \mathbb{R}^{(d+n) \times n} \ .$$

□

Note that any optimal primal solution w can directly be obtained from the dual optimum α by using the optimality condition $w = A\alpha$.