Rescale-Invariant SVM for Binary Classification

Mojtaba Montazery and Nic Wilson
Insight Centre for Data Analytics
School of Computer Science and IT
University College Cork, Ireland
{mojtaba.montazery, nic.wilson}@insight-centre.org

Abstract
Support Vector Machines (SVM) are among the best-known machine learning methods, with broad use in different scientific areas. However, one necessary pre-processing phase for SVM is normalisation (scaling) of features, since SVM is not invariant to the scales of the features’ spaces, i.e., different ways of scaling may lead to different results. We define a more robust decision-making approach for binary classification, in which one sample strongly belongs to a class if it belongs to that class for all possible rescalings of features. We derive a way of characterising the approach for binary SVM that allows determining when an instance strongly belongs to a class and when the classification is invariant to rescaling. The characterisation leads to a computational method to determine whether one sample is strongly positive, strongly negative or neither. Our experimental results back up the intuition that being strongly positive suggests stronger confidence that an instance really is positive.

1 Introduction
Among the wide spectrum of (supervised) machine learning techniques, the Support Vector Machine approach [Burges, 1998] is one of the best-known. It is widely used in a variety of application domains from text classification [Tong and Koller, 2001] to biological sciences [Byvatov and Schneider, 2002].

However, scaling of features is a crucial requirement for SVM because, for example, a particular feature with very large values, compared with the other features, might effectively veto the effect of other features on the SVM objective function. Therefore, SVM is clearly sensitive to the way features are scaled [Stolcke et al., 2008; Ben-Hur and Weston, 2010]. The common practice for scalings is based on the properties of input instances [Jain and Dubes, 1988; Aksoy and Haralick, 2001; Tax and Duin, 2000]; as an example of a scaling method, the value of a feature is subtracted by the minimum of all values of that feature in the dataset and divided by the difference between the maximum and minimum. So, the scaling can make the learnt model sometimes highly sensitive to precisely which instances are received. There can also be subjective, and even rather arbitrary, choices in the scaling of the feature spaces; different ways lead to different results.

It is therefore natural to consider a more cautious classification in which an instance is positively (negatively) classified if it is labeled as positive (negative) for all choices of scaling; the other instances, whose classification can depend on the choice of scaling, are labelled as neutral. Thus, this method refines the set of positively (negatively) classified instances by SVM in order to improve the level of confidence in the classification decisions. This could be helpful in certain sensitive decision making applications such as disease diagnosis; e.g., the test for presence of a particular disease would fall into three categories, positive, negative, and requires further examination.

There are some studies attempting to account for the dependence on feature scaling in margin-based optimisation methods from a different perspective, see e.g., [Jebara and Shivaswamy, 2009; Dorkó and Schmid, 2003]. Also, note that the motivation behind this paper is different from studies, like [Xu et al., ], which are concerned with improving the robustness of SVM against outliers and noise, since we are specifically focusing on the uncertainty caused by rescaling of features.

The rest of the paper is organised as follows. In Section 2, we rephrase conventional binary SVM to facilitate our extension. Section 3 considers the effect of rescaling and defines strong classification. In Section 4, we characterise rescale-optimality, where the rescale-optimal vectors are those that can be made optimal in the SVM objective function for some rescaling. This characterisation leads to a method for computing strong classification, as described in Section 5. In Section 6, the presented approach is evaluated with 18 benchmarks, which are derived from six real data sets.1

1Because of the space restrictions, it was not possible to include all the proofs. See http://ucc.insight-centre.org/mwilson/RescaleInvarSVMClassLonger.pdf for the missing proofs. The longer document also contains a glossary of symbols.
2 Standard SVM for Binary Classification

In this section, we introduce some notation, and express standard SVM using that notation, along with some relevant results. This enables easy generalisation to the rescale-invariant case. In this paper, as an initial step, we just consider the case when the training set is consistent (i.e., the instances are linearly separable).

We define the input set $X$ for a binary classification task to be a set of samples where each sample is characterised by $n$ features. A sample is expressed as a pair of $(x, y)$, where $x \in \mathbb{R}^{n}$ is the feature vector (i.e., $x(k)$ being the score for $x$ regarding the $k$th feature$^{2}$), and $y \in \{+1, -1\}$ indicates the class label for that sample. So, $X \subseteq \mathbb{R}^{n} \times \{+1, -1\}$. The input set can be represented by two disjoint sets:

- $X^+ = \{x^+ : (x^+, +1) \in X\}$
- $X^- = \{x^- : (x^-, -1) \in X\}$

We say that $X$ is non-trivial if both $X^+$ and $X^-$ are non-empty. We define the following terms for any $X \subseteq \mathbb{R}^{n} \times \{+1, -1\}$ which are used throughout the paper, and illustrated in Example 1 below.

- $\Lambda(X) = \{x^+ - x^- : x^+ \in X^+, x^- \in X^-\}$
- $I(X) = \{1, \ldots, |X^+||X^-|\}$
- $G(X) = \{w \in \mathbb{R}^{n} : \forall \lambda \in \Lambda(X), w \cdot \lambda \geq 1\}$
- $MP_{w} = \min_{x^+ \in X^+} w \cdot x^+$
- $MN_{w} = \min_{x^- \in X^-} w \cdot x^-$

The dot product $u \cdot v$ is equal to $\sum_{j=1}^{n} u(j)v(j)$ for vectors $u, v \in \mathbb{R}^{n}$.

**Example 1.** Suppose that $n = 2$, $X^+ = \{(-1, 5), (5, -1), (7/5, 7/5)\}$ and $X^- = \{(1, 1)\}$, with the points marked in Figure 1(a). Then, $\Lambda(X) = \{(-1, 2), (2, -1), (1/5, 1/5)\}$.

$\Lambda(X)$ is determined by the set of vectors $x^+ - x^-$, with $x^+ \in X^+$ and $x^- \in X^-$. This set is the difference of $X^+$ and $X^-$. The dot product $u \cdot v$ is equal to $\sum_{j=1}^{n} u(j)v(j)$ for vectors $u, v \in \mathbb{R}^{n}$.

$\Lambda(X)$ is determined by the set of vectors $x^+ - x^-$, with $x^+ \in X^+$ and $x^- \in X^-$. This set is the difference of $X^+$ and $X^-$. The dot product $u \cdot v$ is equal to $\sum_{j=1}^{n} u(j)v(j)$ for vectors $u, v \in \mathbb{R}^{n}$.

$\Lambda(X)$ is determined by the set of vectors $x^+ - x^-$, with $x^+ \in X^+$ and $x^- \in X^-$. This set is the difference of $X^+$ and $X^-$. The dot product $u \cdot v$ is equal to $\sum_{j=1}^{n} u(j)v(j)$ for vectors $u, v \in \mathbb{R}^{n}$.

$\Lambda(X)$ is determined by the set of vectors $x^+ - x^-$, with $x^+ \in X^+$ and $x^- \in X^-$. This set is the difference of $X^+$ and $X^-$. The dot product $u \cdot v$ is equal to $\sum_{j=1}^{n} u(j)v(j)$ for vectors $u, v \in \mathbb{R}^{n}$.
classified if and only if there exists \( x^+ \in X^+ \) such that \( w^* \cdot (x^+ - \alpha) \leq 1 \).

Proof: Vector \( \alpha \) is positively classified if and only if \( w^* \cdot \alpha + b^* \geq 0 \), which is, by Theorem 2, if and only if \( MP_{w^*} \geq 0 \), i.e., \( MP_{w^*} \leq w^* \cdot \alpha + 1 \). This holds, from the definition of \( MP_{w^*} \), if and only if \( \min_{x^+ \in X^+} w^* \cdot x^+ \leq w^* \cdot \alpha + 1 \), which is if and only if there exists \( x^+ \in X^+ \) such that \( w^* \cdot x^+ \leq w^* \cdot \alpha + 1 \). This immediately leads to the result. \( \square \)

3 Rescaling SVM

In this section we consider how performing certain affine transformations, i.e., translations and rescalings, on the domain of each feature, affect the result of SVM. We define a vector as being strongly positively classified, if it is positively classified under all affine transformations of each feature domain.

(2) It is a known fact that the maximum margin hyperplane is essentially unaffected by a translation of feature space (see e.g., [Meila, 2003]); i.e., by moving the origin, the normal vector to the separating hyperplane and hence the result of SVM does not change. Lemma 4 states this formally.

**Lemma 4.** Consider any finite and non-trivial \( X \subseteq \mathbb{R}^n \times \{+1, -1\} \), any \( \alpha, \delta \in \mathbb{R}^n \), and let \( X^\delta = \{(x + \delta, y) : (x, y) \in X\} \). Then, \( \alpha \) is positively classified with respect to \( X \) if and only if the vector \( \alpha + \delta \) is positively classified with respect to \( X^\delta \).

In contrast with translations, changing the scales of features may significantly affect the result of SVM. Firstly, we define \( u \circ v \) to be the vector in \( \mathbb{R}^n \) given by pointwise multiplication of \( u, v \in \mathbb{R}^n \), and thus, for all \( j = 1, \ldots, n \), \( (u \circ v)(j) = u(j) v(j) \). Operation \( \circ \) is commutative, associative and distributes over addition of vectors, and satisfies the property \( (u \circ v) \cdot w = u \cdot (v \circ w) \), for any \( u, v, w \in \mathbb{R}^n \). Also, let \( \mathbb{R}^n_{\geq 0} \) be the set of strictly positive elements in \( \mathbb{R}^n \), i.e., \( v \in \mathbb{R}^n \) such that \( v(j) > 0 \) for all \( j = 1, \ldots, n \).

Now, consider the effect of a rescaling \( \tau \in \mathbb{R}^n_{\geq 0} \), so that a feature vector \( x \in \mathbb{R}^n \) is transformed into \( x \circ \tau \). Therefore, \( X \) becomes \( X^\tau = \{(x \circ \tau, y) : (x, y) \in X\} \). We write the element of \( G(X^\tau) \) with minimal norm as \( w^\tau \).

**Example 2.** If we rescale \( X \) in Example 1 by \( \tau = (5, 1) \) then \( X^\tau \) will be \( \{(-5, 5), (25, -1), (7, 7/5)\} \), \( X^\tau \) becomes \( \{(5, 1)\} \), and consequently, \( A(X^\tau) = \{(-5, 2), (10, -1), (1, 1/5)\} \). It can be shown that \( w^\tau \) equals \( (5/5, 2) \) and \( b^\tau = 0 \). Now, let \( \alpha = (-1, 4) \) and so \( \alpha \circ \tau = (5, -4) \). Clearly, when there is no scaling, \( \alpha \) is positively classified since \( (2, 5, 2.5) \cdot (1, 4) - 6 = 1.5 > 0 \), but under scaling \( \tau \), \( \alpha \circ \tau \) is negatively classified because \( (5, 2) \cdot (5, -4) - 6 = -1 < 0 \).

For a rescaling vector \( \tau \in \mathbb{R}^n_{\geq 0} \), we say that \( \alpha \in \mathbb{R}^n \) is positively classified under rescaling \( \tau \) if \( \alpha \circ \tau \) is positively classified with respect to \( X^\tau \), which, by using Proposition 3, is if and only if there exists \( x^\tau \in X^\tau \) such that \( w^\tau \cdot (x^\tau - (\alpha \circ \tau)) \leq 1 \). This holds iff there exists \( x^\tau \in X^\tau \) such that \( w^\tau \cdot ((x^\tau - \alpha) \circ \tau) \leq 1 \), i.e., \( (w^\tau \circ \tau) \cdot (x^\tau - \alpha) \leq 1 \).

Let us say that \( \alpha \in \mathbb{R}^n \) is strongly positively classified if and only if it is positively classified under any rescaling \( \tau \in \mathbb{R}^n_{\geq 0} \) and any shift \( \delta \in \mathbb{R}^n \). This is if and only if it is positively classified under any rescaling and no shift (i.e., a shift of \( \delta = 0 \), the zero vector).

**Definition 1.** For \( \alpha \in \mathbb{R}^n \), we define \( Y_X(\alpha) = 1 \) if and only if \( \alpha \) is strongly positively classified; i.e., for all \( \tau \in \mathbb{R}^n_{\geq 0} \) there exists \( x^\tau \in X^\tau \) such that \( (w^\tau \circ \tau) \cdot (x^\tau - \alpha) \leq 1 \).

4 Characterisations Using Rescale Optimality

Here we borrow the notion of rescale-optimality from [Wilson and Montazer, 2016a], and make use of general mathematical results from there. This leads to the computational technique in Section 5 for testing if a vector is strongly positively classified.

We define \( RO(X) \) to be \( \{w^\tau \circ \tau : \tau \in \mathbb{R}^n_{\geq 0}\} \). Hence, it is clear, from Definition 1, that \( Y_X(\alpha) = 1 \) if and only if \( \forall \alpha \in \mathbb{R}^n \).

**Definition 2** (rescale-optimal). For any non-trivial finite \( X \subseteq \mathbb{R}^n \times \{+1, -1\} \), and \( w \in G(X) \), let us say that \( w \) is rescale-optimal in \( G(X) \) if there exists \( \tau \in \mathbb{R}^n_{\geq 0} \) such that for all \( u \in G(X) \), \( \|u \circ \tau\|^2 \geq \|w \circ \tau\|^2 \).

It can be seen intuitively that every element of the line segment between \( (3, 2) \) and \( (2, 3) \) in Figure 1(b) is rescale-optimal; if \( \tau(1) > \tau(2) \) (i.e., with the ratio \( \frac{\tau(1)}{\tau(2)} \) being increased) then \( w^\tau \circ \tau \) moves from \( (2, 5, 2.5) \) towards \( (3, 2) \). Similarly, increasing the ratio \( \frac{\tau(2)}{\tau(1)} \) from 1 moves \( w^\tau \circ \tau \) from \( (2, 5, 2.5) \) towards \( (2, 3) \). For instance, in Example 2, \( \tau(1) = 5 \) leads to \( w^\tau \circ \tau = (3, 2) \).

The following proposition, which follows immediately from Proposition 1 in [Wilson and Montazer, 2016a], states that the set of all rescale-optimal elements of \( G(X) \) is \( RO(X) \). As a result of this equivalence, we can say \( Y_X(\alpha) = 1 \) if and only if for every rescale-optimal element \( w \) in \( G(X) \), there exists \( x^\tau \in X^\tau \) such that \( w \cdot (x^\tau - \alpha) \leq 1 \).

**Proposition 5.** Consider any non-trivial finite \( X \subseteq \mathbb{R}^n \times \{+1, -1\} \), and any \( w \in \mathbb{R}^n \). Then, \( w \) is rescale-optimal in \( G(X) \) if and only if \( w \) is in \( RO(X) \).

In Section 4.2 below we characterise the rescale-optimal elements, which leads to a computational procedure for strong classification. First, in Section 4.1, we characterise the situations in which rescaling the values of the features makes no difference to the result of the classification, in which case strong classification is the same as the standard classification.

4.1 Determining Invariance to Rescaling

Theorem 6 below shows that rescaling the features vector makes no difference in the classification when there is a unique rescale-optimal vector in \( G(X) \) (and the vector thus has minimal norm in \( G(X) \)).

**Theorem 6.** Consider any non-trivial finite \( X \subseteq \mathbb{R}^n \times \{+1, -1\} \). Let \( Pos(X) \) be the set of all \( \alpha \in \mathbb{R}^n \) that are positively classified, and let \( SPos(X) \) be the set of all \( \alpha \) that are strongly positively classified. Then \( Pos(X) = SPos(X) \) if and only if there exists a unique rescale-optimal element in \( G(X) \), i.e., \( RO(X) \) is a singleton set.
Proof: For \( w \in \mathbb{R}^n \), let \( \text{Pos}_{w^*} \) be the set of all \( \alpha \in \mathbb{R}^n \) such that there exists \( x^+ \in X^+ \) such that \( w \cdot (x^+ - \alpha) \leq 1 \). Then, by Proposition 3, \( \text{Pos}(X) = \text{Pos}_{w^*} \), and using Proposition 5, \( S\text{Pos}(X) \) is the intersection of \( \text{Pos}_{w^*} \) over all rescale-optimal \( w \) in \( G(X) \). Note that \( w^* \) is rescale optimal in \( G(X) \) (using the identity rescaling). Thus, if there is a single rescale-optimal \( w \) then \( w = w^* \) and so \( \text{Pos}(X) = S\text{Pos}(X) \).

To prove the converse, suppose that \( \text{Pos}(X) = S\text{Pos}(X) \), which implies that for all rescale-optimal \( w \) in \( G(X) \), \( \text{Pos}_{w^*} \) contains \( \text{Pos}_{w} \). Now, we can write \( \text{Pos}_{w^*} \) as \( \{ \alpha \in \mathbb{R}^n : \alpha \cdot w \geq MP_w - 1 \} \), and thus \( w^* \) is a half-space with normal vector \( w \). Hence, since \( \text{Pos}_{w^*} \) contains \( \text{Pos}_{w} \), there exists real scalar \( q > 0 \) with \( w^* = qw \). Then for any \( \tau \in \mathbb{R}_+^n \), \( \|w^* \cdot \tau\|^2 = q^2 \|w \cdot \tau\|^2 \). By Definition 2, \( q = 1 \), since both \( w \) and \( w^* \) are rescale-optimal, and hence \( w = w^* \). This shows that there is a unique rescale-optimal element in \( G(X) \). \( \square \)

Definition 3 (pointwise dominance). For \( u, w \in \mathbb{R}^n \), \( w \) pointwise dominates \( u \) if \( u \neq w \) and for all \( j \in \{1, \ldots, n\} \), either \( 0 \leq w(j) \leq u(j) \) or \( 0 \geq w(j) \geq u(j) \) (i.e., \( w(j) \) is closer to zero than \( u(j) \)).

Example 3. Consider modifying Example 1 by just removing \((7/5,7/5)\) from \( X^+ \). Then, \( \Lambda(X) = \{(-1,2),(2,-1)\} \), and \( G(X) \) becomes the intersection of the half-spaces \(-x_1+2x_2 \geq 1 \) and \( 2x_1-x_2 \geq 1 \), with a single extremal point \((1,1)\). This point dominates every other element in \( G(X) \) (see Figure 1(b)). Pointwise dominating every other element, from Theorem 7 below, means that \((1,1)\) will be the unique element of \( G(X) \) that is rescale-optimal. Using Theorem 6, this implies that the classification is invariant to the rescaling (in this example).

Theorem 7 below characterises when there is a unique rescale-optimal element, and Corollary 8 leads to a polynomial algorithm to determine that unique element. These results follow from Theorem 2 and Corollary 1 in [Wilson and Montazery, 2016a; 2016b].

Theorem 7. Consider any non-trivial finite \( X \subseteq \mathbb{R}^n \times \{+1,-1\} \) and any \( w \in G(X) \). Then, \( w \) is the unique element of \( G(X) \) that is rescale-optimal if and only if \( w \) pointwise dominates every element in \( G(X) \) \( \setminus \{w\} \).

Corollary 8. Consider any non-trivial finite \( X \subseteq \mathbb{R}^n \times \{+1,-1\} \) and any \( w \in G(X) \). Define \( w \in \mathbb{R}^n \) as follows. For each \( j \in \{1, \ldots, n\} \): (i) If \( v(j) = 0 \) then define \( w(j) = 0 \). (ii) If \( v(j) > 0 \) then define \( w(j) = \inf \{u(j) : u \in G(X), u(j) \geq 0\} \). (iii) If \( v(j) < 0 \) then define \( w(j) = \sup \{u(j) : u \in G(X), u(j) \leq 0\} \). If \( w \in G(X) \) then \( w \) is uniquely rescale-optimal in \( G(X) \). Also, there exists a uniquely rescale-optimal element in \( G(X) \) if and only if \( w \in G(X) \).

4.2 Characterising Rescale-Optimality
Here, we characterise the set of rescale-optimal elements in \( G(X) \), which leads to a computational method for strong classification.

Definition 4 (agreeing on signs). For \( w, \mu \in \mathbb{R}^n \), we say that \( w \) and \( \mu \) agree on signs if for all \( j = 1, \ldots, n \), (i) \( w(j) = 0 \iff \mu(j) = 0 \); (ii) \( w(j) > 0 \iff \mu(j) > 0 \); and thus also: (iii) \( w(j) < 0 \iff \mu(j) < 0 \).

Theorem 9. Consider any finite non-trivial \( X \subseteq \mathbb{R}^n \times \{+1,-1\} \), and any non-zero vector \( w \in G(X) \). Then, \( w \) is rescale-optimal in \( G(X) \) if and only if there exists \( \mu \in \mathbb{R}^n \) and non-negative reals \( r_i \) for each \( i \in I(X) \), such that \( \mu \) agrees on signs with \( w \): (b) \( \mu = \sum_{i \in I} r_i \lambda_i \); and (c), for each \( i \in I(X) \) either \( r_i = 0 \) or \( \lambda_i \cdot w = 1 \), where \( \lambda_i \) is the \( i \)th element of \( \Lambda(X) \).

Proof: Theorem 5 of [Wilson and Montazery, 2016a] implies that \( w \) is rescale-optimal in \( G(X) \) if and only if there exists \( \mu \in \mathbb{R}^n \) and non-negative reals \( r_i \) (for each \( i \in I(X) \)) such that conditions (a), (b), (c) and (d) hold, where (a), (b) and (c) are the conditions above, and (d) is the condition that \( w \cdot \mu = 1 \).

\( \Rightarrow \): this follows immediately from Theorem 5 of [Wilson and Montazery, 2016a].

\( \Leftarrow \): Assume that there exists \( \mu \in \mathbb{R}^n \) and non-negative reals \( r_i \) (for each \( i \in I(X) \)) such that conditions (a), (b), (c) hold, \( w \) is not the zero vector, \( w \in G(X) \). Condition (a), that \( \mu \) agrees on signs with \( w \), implies that \( \mu \cdot w > 0 \). Let \( \mu' = \frac{\mu}{\mu_+} \), and for each \( i \in I(X) \) define \( r_i' = \frac{r_i}{\mu_+} \). Then, (a) \( \mu' \) agrees on signs with \( w \), (b) \( \mu' = \sum_{i \in I} r_i' \lambda_i \), (c) for each \( i \in I(X) \) either \( r_i' = 0 \) or \( \lambda_i \cdot w = 1 \), and (d) \( w \cdot \mu' = 1 \). Theorem 5 of [Wilson and Montazery, 2016a] then implies that \( w \) is rescale-optimal in \( G(X) \). \( \square \)

5 Computation of Strong Classification
Here we express if an instance is strongly positively classified in terms of a set of constraints.

For a set \( X \subseteq \mathbb{R}^n \times \{+1,-1\} \) and arbitrary \( \alpha \in \mathbb{R}^n \), we would like to determine if \( Y_X(\alpha) = 1 \), i.e., if \( \alpha \) is strongly positively classified. We can infer from Propositions 3 and 5 that \( Y_X(\alpha) \neq 1 \) if and only if there exists a rescale-optimal element \( w \) in \( G(X) \) such that for all \( x^+ \in X^+ \), \( w \cdot (x^+ - \alpha) > 1 \). By taking into account Theorem 9 for rescale-optimising classification, we obtain a set of inequalities to determine if \( Y_X(\alpha) \neq 1 \) as follows.

Let \( \lambda_i \) be the \( i \)th element of \( \Lambda(X) \) where \( i \in I(X) \). Now, \( Y_X(\alpha) \neq 1 \) if and only if there exists \( \alpha \in \mathbb{R}^n \) and \( \mu \in \mathbb{R}^n \), and non-negative reals \( r_i \) for each \( i \in I(X) \) such that

- \( \forall x^+ \in X^+ \), \( w \cdot (x^+ - \alpha) > 1 \);
- \( \forall i \in I(X), \mu \cdot \lambda_i \geq 1 \); (i.e., \( w \in G(X) \))
- \( \forall i \in I(X), w \cdot \lambda_i = 1 \) or \( r_i = 0 \);
- \( \forall j = 1, \ldots, n, w(j) = 0 \iff \mu(j) = 0 \), and \( w(j) > 0 \iff \mu(j) > 0 \); (i.e., agreeing on signs);
- \( \mu = \sum_{i \in I(X)} r_i \lambda_i \).

In CPLEX, a disjunctive constraint such as \( [w \cdot \lambda_i = 0 \text{ or } r_i = 0] \) can be expressed as \( (w \cdot \lambda_i = 0) + (r_i = 0) \geq 1 \) (each logical proposition is treated as an integer; 0 for false and 1 for true). The number of constraints here is \( 2|X^+||X^-| + |X^+| + n + 2 \).
The experiments make use of the UCI machine learning repository\(^3\) from which six real data sets are chosen, namely \textit{Breast Cancer Wisconsin} [Street \textit{et al.}, 1993], \textit{Pima Indians Diabetes}, \textit{Blood Transfusion Service Center} [Yeh \textit{et al.}, 2009], \textit{Indian Liver Patient}, \textit{Fertility} [Méndez \textit{et al.}, 2012], and \textit{Banknote Authentication}. In fact, we include data sets that meet three criteria: (i) the data set has only two classes, (ii) the data set consists of only numeric features, and (iii) number of features is at most 10. The first two criteria ensure that the data set complies with the proposed method in this paper, and the third made the computation especially fast. Thereafter, 18 benchmarks are derived from those data sets. Table 1 contains specifications of these benchmarks.

For constructing a benchmark, a random selector creates two disjoint sets from a data set, one for learning (i.e., \(X\)) and one for testing. However, a pre-processing phase deletes some elements of the learning set in order to make it consistent (i.e., \(C(X) \neq \emptyset\)), since in this paper we consider only the consistent case.

For each instance in the testing set (let’s say \(\alpha\)), the rescaling method determines—based on the learning set \(X\)— either (i) it is strongly positive (\(Y_X(\alpha) = 1\)), or (ii) it is strongly negative (\(Y_X(\alpha) = -1\)), or (iii) it is neutral (\(Y_X(\alpha) \neq 1\) and \(Y_X(\alpha) \neq -1\)).

The number and percentage of neutral instances for each benchmark can be found in Table 2. The ratio of neutral instances for the benchmarks (i.e., those instances being classified differently under different rescaling) varies from 6% to 95% with a mean of 55.9%. For more than half of the instances, rescaling the feature values can change the result of the SVM classification. That points out the unreliability of standard SVM—specifically for larger numbers like 95%—with respect to being sensitive to the way the features are scaled.

\(^3\)http://archive.ics.uci.edu/ml/
Table 3: A comparison, using 18 benchmarks, between the NPV of SVM and the rescaling method. The undefined values, labelled -, are excluded from the mean.

<table>
<thead>
<tr>
<th>Bench.</th>
<th>SVM</th>
<th>Rescaling Method</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>FP</td>
<td>TP</td>
</tr>
<tr>
<td>1.</td>
<td>15</td>
<td>439</td>
</tr>
<tr>
<td>2.</td>
<td>3</td>
<td>71</td>
</tr>
<tr>
<td>3.</td>
<td>9</td>
<td>64</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4.</td>
<td>25</td>
<td>48</td>
</tr>
<tr>
<td>5.</td>
<td>38</td>
<td>53</td>
</tr>
<tr>
<td>6.</td>
<td>8</td>
<td>15</td>
</tr>
<tr>
<td>7.</td>
<td>25</td>
<td>6</td>
</tr>
<tr>
<td>8.</td>
<td>7</td>
<td>4</td>
</tr>
<tr>
<td>9.</td>
<td>27</td>
<td>9</td>
</tr>
<tr>
<td>10.</td>
<td>15</td>
<td>13</td>
</tr>
<tr>
<td>11.</td>
<td>35</td>
<td>15</td>
</tr>
<tr>
<td>12.</td>
<td>44</td>
<td>23</td>
</tr>
<tr>
<td>13.</td>
<td>15</td>
<td>2</td>
</tr>
<tr>
<td>14.</td>
<td>19</td>
<td>1</td>
</tr>
<tr>
<td>15.</td>
<td>11</td>
<td>3</td>
</tr>
<tr>
<td>16.</td>
<td>3</td>
<td>45</td>
</tr>
<tr>
<td>17.</td>
<td>1</td>
<td>42</td>
</tr>
<tr>
<td>18.</td>
<td>3</td>
<td>44</td>
</tr>
</tbody>
</table>

Avg. 56.56 62.06

Table 4: A comparison, using 18 benchmarks, between the NPV of SVM and the rescaling method. The undefined values, labelled -, are excluded from the mean.

<table>
<thead>
<tr>
<th>Bench.</th>
<th>SVM</th>
<th>Rescaling Method</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>FN</td>
<td>TN</td>
</tr>
<tr>
<td>1.</td>
<td>10</td>
<td>219</td>
</tr>
<tr>
<td>2.</td>
<td>1</td>
<td>25</td>
</tr>
<tr>
<td>3.</td>
<td>0</td>
<td>27</td>
</tr>
<tr>
<td>4.</td>
<td>22</td>
<td>105</td>
</tr>
<tr>
<td>5.</td>
<td>14</td>
<td>95</td>
</tr>
<tr>
<td>6.</td>
<td>21</td>
<td>56</td>
</tr>
<tr>
<td>7.</td>
<td>17</td>
<td>52</td>
</tr>
<tr>
<td>8.</td>
<td>18</td>
<td>71</td>
</tr>
<tr>
<td>9.</td>
<td>9</td>
<td>55</td>
</tr>
<tr>
<td>10.</td>
<td>19</td>
<td>53</td>
</tr>
<tr>
<td>11.</td>
<td>13</td>
<td>37</td>
</tr>
<tr>
<td>12.</td>
<td>27</td>
<td>106</td>
</tr>
<tr>
<td>13.</td>
<td>6</td>
<td>47</td>
</tr>
<tr>
<td>14.</td>
<td>6</td>
<td>55</td>
</tr>
<tr>
<td>15.</td>
<td>8</td>
<td>63</td>
</tr>
<tr>
<td>16.</td>
<td>2</td>
<td>50</td>
</tr>
<tr>
<td>17.</td>
<td>1</td>
<td>56</td>
</tr>
<tr>
<td>18.</td>
<td>2</td>
<td>51</td>
</tr>
</tbody>
</table>

Avg. 86.00 91.47

Note that our approach is not intended to be a competitor of SVM in terms of classification accuracy. What it does is to highlight certain instances where we can have greater confidence, with the other instances having reduced confidence because the result of the classification could be changed if a different scaling of the features were used. This is why PPV and NPV are appropriate measures for the experimental results.

The approach discussed in Section 5 was implemented using the solver CPLEX 12.6.2. Determining whether an instance is strongly positive or strongly negative or neutral for any of benchmarks takes less than a couple of seconds, making use of a computer facilitated by a Core i7 2.60 GHz processor and 8 GB RAM memory.

7 Discussion

The scaling of individual features, before applying an SVM method, can be subjective and arbitrary. However, the way features are scaled can make a difference; in fact, for a little more than half of the instances in our experiments, rescaling the feature values can change the result of the SVM classification. Based on this fact, we say an instance strongly belongs to a class if it belongs to that class for all rescalings of features. This new definition boosts the confidence of labeling instances by excluding those instances which are classified differently under different scalings.

Building on the general mathematical results of [Wilson and Montazery, 2016a], we have developed a computational procedure that can test if an instance is strongly positive, i.e., labelled positive for every affine way of rescaling the values of each feature (and similarly for the negative case). We also have a polynomial algorithm for determining when rescaling makes no difference, i.e., when, for a given training set, the classification of any possible instance is not affected by rescaling. Our experiments also showed a slight improvement in the value of prediction, i.e., the likelihood that an instance is classified as a particular class, it truly belongs to that class.

There are many potential future directions following from this work; for instance, extending the method to the case when there is inconsistency in the input data (i.e., soft margin SVM); attempting to develop the computational method for certain kernels; performing multi-class classification by making use of repeated binary-class classification (e.g., using one-vs-all or one-vs-one approaches); and computing variations of the method where the user can limit the rescaling range for different features.

Acknowledgments

This publication has emanated from research conducted with the financial support of Science Foundation Ireland (SFI) under Grant Number SFI/12/RC/2289. Thanks to the reviewers for their comments, which helped improve the final version of the paper.
References


[Xu et al., 2009] Linli Xu, Koby Crammer, and Dale Schuurmans. Robust support vector machine training via convex outlier ablation. In Proceedings of The Twenty-First National Conference on Artificial Intelligence and the Eighteenth Innovative Applications of Artificial Intelligence Conference, July 16-20, 2006, Boston, Massachusetts, USA.