On the Problem of Finding the Least Number of Features by L1-Norm Minimisation

Sascha Klement and Thomas Martinetz

Institute for Neuro- and Bioinformatics, University of Lübeck,
Ratzeburger Allee 160, 23538 Lübeck, Germany
{klement,martinetz}@inb.uni-luebeck.de

Abstract. Recently, the so-called Support Feature Machine (SFM) was proposed as a novel approach to feature selection for classification. It relies on approximating the zero-norm minimising weight vector of a separating hyperplane by optimising for its one-norm. In contrast to the L1-SVM it uses an additional constraint based on the average of data points. In experiments on artificial datasets we observe that the SFM is highly superior in returning a lower number of features and a larger percentage of truly relevant features. Here, we derive a necessary condition that the zero-norm and 1-norm solution coincide. Based on this condition the superiority can be made plausible.

Keywords: Support feature machine, L1-SVM, feature selection, zero norm minimisation, classification.

1 Introduction

The ever increasing complexity of real-world machine learning tasks requires more and more sophisticated methods to deal with datasets that contain only very few relevant features but many irrelevant noise dimensions. In practise, these scenarios often arise in the analysis of biological datasets, such as tissue classification using microarrays [2], identification of disease-specific genome mutations or distinction between mental states using functional magnetic resonance imaging [3]. It is well-known that a large number of irrelevant features may distract state-of-the-art methods, such as the support vector machine. Thus, feature selection is a fundamental preprocessing step to achieve proper classification results, to improve runtime, and to make the training results more interpretable.

The recently proposed Support Feature Machine [5,4] relies on approximating the zero-norm of a separating hyperplane. As zero-norm optimisation is computationally infeasible for real world datasets, the SFM approach uses an iterative optimisation scheme based on the one-norm that is closely related to the SVM-based method proposed by Weston et. al [6]. However, in artificial experiments it has been shown that the SFM approach is superior, i.e. it returns a significantly lower number of features and a larger number of truly relevant features. The reason is not obvious, so here, we derive plausibility considerations to explain why the SFM approach finds the zero-norm more frequently.

T. Honkela et al. (Eds.): ICANN 2011, Part I, LNCS 6791, pp. 315–322, 2011.

The following sections are organised as follows. First, we outline the mathematical formulation of the Support Feature Machine and related methods. Then, we compare its performance with the L1-SVM on an artificial dataset. Finally, for the SFM and Weston's method we derive a coincidence condition, i.e. a condition in which zero-norm and one-norm minimising solution coincide. Unfortunately, it is not possible to decide for a specific dataset whether this condition is fulfilled or not. However, we compare both methods in a simple scenario to give a plausible explanation for the superior performance of the SFM.

2 Feature Selection by Zero-Norm Minimisation

We make use of the common notations used in classification and feature selection frameworks, i.e. the training set $\mathcal{D} = \{\boldsymbol{x}_i, y_i\}_{i=1}^n$ consists of feature vectors $\boldsymbol{x}_i \in \mathbb{R}^d$ and corresponding class labels $y_i \in \{-1, +1\}$. We assume the dataset $\mathcal{D}$ to be linearly separable without bias, i.e.

$$\exists\, \boldsymbol{w} \in \mathbb{R}^d \quad \text{with} \quad y_i \boldsymbol{x}_i^{\mathrm{T}} \boldsymbol{w} \geq 0 \;\forall\, i \quad \text{and} \quad \boldsymbol{w} \neq \boldsymbol{0}\,, \tag{1}$$

where the normal vector $\boldsymbol{w} \in \mathbb{R}^d$ describes the separating hyperplane except for a constant factor. Analogous formulations including bias can be found in [5] and [4]. In general, there is no unique solution to (1). A common approach in feature selection is to find a weight vector $\boldsymbol{w}$ which solves

$$\text{minimise} \quad \|\boldsymbol{w}\|_0^0 \quad \text{subject to} \quad y_i \boldsymbol{x}_i^{\mathrm{T}} \boldsymbol{w} \geq 0 \quad \text{and} \quad \boldsymbol{w} \neq \boldsymbol{0} \tag{2}$$

with $\|\boldsymbol{w}\|_0^0 = \operatorname{card}\{w_i | w_i \neq 0\}$. Hence, solutions to (2) solve the classification problem (1) using the least number of features. Some attempts have been made to approximate the above problem with a variant of the Support Vector Machine (SVM), e.g. by Weston et al. [6] who

$$\text{minimise} \quad \sum\nolimits_{j=1}^{d} \ln\left(\epsilon + |w_j|\right) \quad \text{subject to} \quad y_i \boldsymbol{x}_i^{\mathrm{T}} \boldsymbol{w} \geq 1 \tag{3}$$

with $0 < \epsilon \ll 1$. A local minimum of (3) is found using an iterative scheme based on linear programming. However, the following approach was found to identify relevant features more effectively. Instead of modifying the SVM setting as in [6], we slightly change (2) such that we

$$\text{minimise} \quad \|\boldsymbol{w}\|_0^0 \quad \text{subject to} \quad y_i \boldsymbol{x}_i^{\mathrm{T}} \boldsymbol{w} \geq 0 \quad \text{and} \quad \left(\frac{1}{n}\sum_{i=1}^{n} y_i \boldsymbol{x}_i\right)^{\mathrm{T}} \boldsymbol{w} = 1\,. \tag{4}$$

The second constraint excludes $\boldsymbol{w} = \boldsymbol{0}$ and solving (4) yields a solution to the ultimate problem (2). Since we have linear constraints, for solving (4) we can employ the same framework Weston et al. [6] used for solving their problem. However, our experiments show that by

$$\text{minimising} \sum_{j=1}^{d} \ln\left(\epsilon + |w_j|\right) \quad \text{subject to } y_i \boldsymbol{x}_i^{\mathrm{T}} \boldsymbol{w} \geq 0 \text{ and } \left(\frac{1}{n}\sum_{i=1}^{n} y_i \boldsymbol{x}_i\right)^{\mathrm{T}} \boldsymbol{w} = 1$$

we obtain significantly better solutions to the ultimate problem than by solving (3). It seems that the new cost function is much less prone to local minima. For solving the above problem, we apply a constrained gradient descent technique based on Frank and Wolfe's method [1]:

1. Set $\boldsymbol{v} = (1, \ldots, 1)$.
2. Minimise $|\boldsymbol{w}|$ such that $y_i(\boldsymbol{x}_i * \boldsymbol{v})^{\mathrm{T}}\boldsymbol{w} \geq 0$ and $\left(\frac{1}{n}\sum_{i=1}^{n} y_i(\boldsymbol{x}_i * \boldsymbol{v})\right)^{\mathrm{T}}\boldsymbol{w} = 1$
3. Set $\boldsymbol{v} = \boldsymbol{v} * \boldsymbol{w}$.
4. Repeat until convergence.

Here, $\boldsymbol{v}$ is the iteratively adapted scaling vector and the operator $*$ denotes the element-wise multiplication. The solution is optimal with respect to feature selection if a solution to (4) is found, i.e. if both solutions coincide.

3 Experiments

We compared the performance of both approaches with respect to k and n. For that purpose, we constructed artificial scenarios with balanced classes. The first k dimensions $x_i, \ldots, x_k$ were drawn as $x_i = \mathcal{N}(c \cdot y, \sigma^2)$. The parameter c controls the distance between both classes. The remaining features $x_{k+1}, \ldots, x_d$ were noise drawn as $x_i = \mathcal{N}(0, \sigma^2)$. Additionally, we ensured that both classes were linearly separable. However, it was possible that both classes were separable with less than k features.

The results are shown in Fig. 1. Obviously, the SFM returns both a lower total number of features and a higher percentage of correct features. So, Weston's method returns more irrelevant features than the SFM. Besides, increasing the number of features (see Fig. 1, bottom) has a different impact on both methods. If we increase the number of data points (in this case to 100), the SFM will identify all relevant features correctly. The SVM-based method fails to converge to the correct number of features even if the number of data points is further increased (e.g. to 1000). So, in this setting the SFM converges for large n to the correct set of features, while the SVM-based approach gets stuck in a local minimum even for large datasets. It is also obvious, that the SFM solution in the first iteration is already very close to the final solution, while the SVM-based method needs more iterations. In scenarios with a large number of data points the SFM converges already after one iteration (see Fig. 1, bottom left).

4 Optimality of the Support Feature Machine

In general, it is not possible to decide whether the zero-norm and one-norm solution coincide. However, one may give some plausibility considerations to show why in most cases the SFM is closer to the ultimate zero-norm solution than the SVM-based approach.

This section is organised as follows. First, we introduce notations to improve the readability of the admittedly complex plausibility considerations. Then, we derive a condition for zero- and one-norm minimising solutions to coincide. This condition holds both for the SFM and the SVM-based approach. Finally, we demonstrate that in certain scenarios it is beneficial to use the SFM.

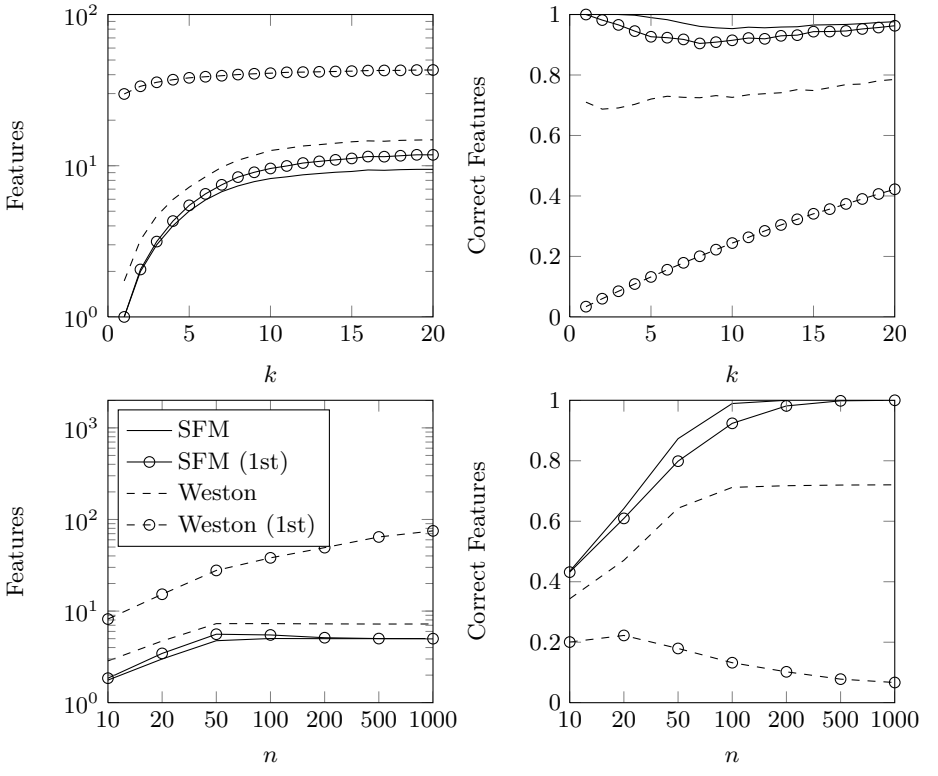

Fig. 1. Feature selection performance depending on k and n. The top row shows the mean number of features and the mean percentage of correctly identified features after the first and after the last iteration depending on k ($k = 1, \ldots, 20, n = 100, \sigma = 1, d = 100, c = 0.3$, 1000 repetitions), while the bottom row shows the same aspects for different values of n ($\sigma = 1, d = 100, k = 5, c = 0.3$, 1000 repetitions).

4.1 Preliminaries

For simplicity, we define $\boldsymbol{z}_i = y_i \boldsymbol{x}_i$ and $\boldsymbol{Z} = (\boldsymbol{z}_1, \ldots, \boldsymbol{z}_n)$ and $\bar{\boldsymbol{z}} = \frac{1}{n} \sum_{i=1}^{n} \boldsymbol{z}_i$. Additionally, $\mathbf{0}$ and $\mathbf{1}$ are vectors that consist of zeros and ones, respectively. For reasons of readability, we omit the length of these vectors where possible. Using this notation, Weston et al. aim to

$$\text{minimise} \quad ||\boldsymbol{w}||_0 \quad \text{subject to} \quad \boldsymbol{Z}^{\mathrm{T}} \boldsymbol{w} \geq \mathbf{1}\,, \tag{5}$$

while in the SFM setting we aim to

$$\text{minimise} \quad ||\boldsymbol{w}||_0 \quad \text{subject to} \quad \boldsymbol{Z}^{\mathrm{T}} \boldsymbol{w} \geq \mathbf{0} \quad \text{and} \quad \bar{\boldsymbol{z}}^{\mathrm{T}} \boldsymbol{w} = 1\,. \tag{6}$$

First, we focus on (5) — minor changes will lead us to (6). To simplify a comparison, we consider only results after the very first iteration of the overall optimisation

procedure. We denote the solution space with Ω and define the following two weight vectors:

$$\boldsymbol{w}_0 = \underset{\boldsymbol{w}\in\Omega}{\arg\min} \quad ||\boldsymbol{w}||_1 \quad \text{subject to} \quad \boldsymbol{Z}^{\mathrm{T}}\boldsymbol{w} \geq \mathbf{1}$$
$$\boldsymbol{w}_1 = \underset{\boldsymbol{w}\in\mathbb{R}^d}{\arg\min} \quad ||\boldsymbol{w}||_1 \quad \text{subject to} \quad \boldsymbol{Z}^{\mathrm{T}}\boldsymbol{w} \geq \mathbf{1}$$

So, among all solutions of (5), $\boldsymbol{w}_0$ is the solution with lowest one-norm. Note that if $\boldsymbol{w}_1$ is in Ω then $\boldsymbol{w}_1 = \boldsymbol{w}_0$. As in practise (5) cannot be solved directly, Ω is in general unknown as well as $\boldsymbol{w}_0$. However, both are well-defined. In contrast, $\boldsymbol{w}_1$ is the solution on the entire $\mathbb{R}^d$ and can efficiently be solved by linear programming. If $\boldsymbol{w}_0 = \boldsymbol{w}_1$ for a specific dataset, then the optimal feature set would be found by optimising for the one-norm.

In the following, we assume $\Omega \neq \emptyset$ and $\boldsymbol{w}_1$ to be unique. This is only a minor restriction as non-uniqueness of $\boldsymbol{w}_1$ will occur only in degenerate cases. Since $\boldsymbol{Z}$ is drawn from a probability distribution, the probability of these cases is of measure zero. The probabilistic nature of the input data also ensures that all quadratic submatrices of $\boldsymbol{Z}$ have full rank.

Without loss of generality, for the following considerations we assume:

1. All entries of the weight vector are positive, i.e. $w_{0,i} \geq 0$. Otherwise, invert the corresponding input dimension.
2. The training data is ordered such that $\boldsymbol{Z} = \left(\hat{\boldsymbol{Z}}\ \check{\boldsymbol{Z}}\right)$ with $\hat{\boldsymbol{Z}}^{\mathrm{T}}\boldsymbol{w}_0 = \mathbf{1}$ and $\check{\boldsymbol{Z}}^{\mathrm{T}}\boldsymbol{w}_0 > \mathbf{1}$.
3. The dimensions of $\mathcal{D}$ are sorted, such that exactly the first k dimensions of $\boldsymbol{w}_0$ are non-zero, i.e.

$$w_{0,i} \begin{cases} > 0 & i = 1, \ldots, k \\ = 0 & \text{otherwise} \end{cases} \quad \text{such that} \quad \boldsymbol{w}_0 = \begin{pmatrix} \hat{\boldsymbol{w}}_0 \\ \mathbf{0} \end{pmatrix}$$

In total the input data matrix $\boldsymbol{Z}$ has the following structure:

$$\boldsymbol{Z} = \begin{pmatrix} \hat{\boldsymbol{Z}}_1 \check{\boldsymbol{Z}} \\ \hat{\boldsymbol{Z}}_2 \end{pmatrix} \quad \text{with} \quad \hat{\boldsymbol{Z}}_1 \in \mathbb{R}^{k\times k^*}, \hat{\boldsymbol{Z}}_2 \in \mathbb{R}^{d-k\times k^*}, \check{\boldsymbol{Z}} \in \mathbb{R}^{d\times n-k^*}$$

Lemma 1. *If $\boldsymbol{w}_0$ contains k non-zero entries, exactly k equations in $\boldsymbol{Z}^T\boldsymbol{w}_0 \geq \mathbf{1}$ are active, i.e. $k = k^*$.*

Proof. By definition, the problem is feasible and non-degenerate. Thus, as an optimal solution exists, also a basic optimal solution exist, which is known from linear programming theory. Due to $\hat{\boldsymbol{Z}}^{\mathrm{T}}\boldsymbol{w}_0 = \hat{\boldsymbol{Z}}_1^{\mathrm{T}}\hat{\boldsymbol{w}}_0 = \mathbf{1}$, the initial d-dimensional problem is reduced to a k-dimensional one. Thus, in a basic solution k constraints are active and $k^* = k$ follows. □

4.2 Optimality Condition

Theorem 1. *For $\boldsymbol{w}_1 = \boldsymbol{w}_0$, it is necessary that* $\left\| \hat{\boldsymbol{Z}}_2 \hat{\boldsymbol{Z}}_1^{\mathrm{T}} \left(\hat{\boldsymbol{Z}}_1 \hat{\boldsymbol{Z}}_1^{\mathrm{T}} \right)^{-1} \mathbf{1} \right\|_\infty < 1.$

Proof. If $\boldsymbol{w}_0 = \boldsymbol{w}_1$, for each infinitesimal $\boldsymbol{\Delta}$ with $\hat{\boldsymbol{Z}}^{\mathrm{T}}(\boldsymbol{w}_0+\boldsymbol{\Delta}) = \mathbf{1}$ and $\check{\boldsymbol{Z}}^{\mathrm{T}}(\boldsymbol{w}_0+\boldsymbol{\Delta}) > \mathbf{1}$ we have

$$
\begin{aligned}
&||\boldsymbol{w}_0 + \boldsymbol{\Delta}||_1 > ||\boldsymbol{w}_0||_1 \\
\Leftrightarrow \quad &\sum_{i=1}^{d} |w_{0,i} + \Delta_i| > \sum_{i=1}^{d} |w_{0,i}| = \sum_{i=1}^{d} w_{0,i} \\
\Leftrightarrow \quad &\sum_{i=1}^{k} |w_{0,i} + \Delta_i| + \sum_{i=k+1}^{d} |\underbrace{w_{0,i}}_{=0} + \Delta_i| > \sum_{i=1}^{k} w_{0,i} \\
\Leftrightarrow \quad &\sum_{i=1}^{k} (w_{0,i} + \Delta_i) + \sum_{i=k+1}^{d} |\Delta_i| > \sum_{i=1}^{k} w_{0,i} \qquad (7) \\
\Leftrightarrow \quad &\sum_{i=1}^{k} \Delta_i + \sum_{i=k+1}^{d} |\Delta_i| > 0\,. \qquad (8)
\end{aligned}
$$

Next, we apply the structure of the matrix $\hat{\boldsymbol{Z}}$ and split the disparity vector, i.e. $\Delta^{\mathrm{T}} = \left(\boldsymbol{\Delta}_1^{\mathrm{T}} \boldsymbol{\Delta}_2^{\mathrm{T}}\right)$ with $\boldsymbol{\Delta}_1 \in \mathbb{R}^k, \boldsymbol{\Delta}_2 \in \mathbb{R}^{d-k}$. After some rearrangements, we can derive a closed formulation for $\boldsymbol{\Delta}_1$:

$$
\begin{aligned}
\Leftrightarrow \quad &\hat{\boldsymbol{Z}}^{\mathrm{T}} \boldsymbol{\Delta} = \hat{\boldsymbol{Z}}_1^{\mathrm{T}} \boldsymbol{\Delta}_1 + \hat{\boldsymbol{Z}}_2^{\mathrm{T}} \boldsymbol{\Delta}_2 = \mathbf{0} \\
\Leftrightarrow \quad &\hat{\boldsymbol{Z}}_1^{\mathrm{T}} \boldsymbol{\Delta}_1 = -\hat{\boldsymbol{Z}}_2^{\mathrm{T}} \boldsymbol{\Delta}_2 \\
\Leftrightarrow \quad &\hat{\boldsymbol{Z}}_1 \hat{\boldsymbol{Z}}_1^{\mathrm{T}} \boldsymbol{\Delta}_1 = -\hat{\boldsymbol{Z}}_1 \hat{\boldsymbol{Z}}_2^{\mathrm{T}} \boldsymbol{\Delta}_2 \\
\Leftrightarrow \quad &\boldsymbol{\Delta}_1 = -\left(\hat{\boldsymbol{Z}}_1 \hat{\boldsymbol{Z}}_1^{\mathrm{T}}\right)^{-1} \hat{\boldsymbol{Z}}_1 \hat{\boldsymbol{Z}}_2^{\mathrm{T}} \boldsymbol{\Delta}_2 \qquad (9) \\
\Rightarrow \quad &\mathbf{1}^{\mathrm{T}} \boldsymbol{\Delta}_1 = -\underbrace{\mathbf{1}^{\mathrm{T}} \left(\hat{\boldsymbol{Z}}_1 \hat{\boldsymbol{Z}}_1^{\mathrm{T}}\right)^{-1} \hat{\boldsymbol{Z}}_1 \hat{\boldsymbol{Z}}_2^{\mathrm{T}}}_{:=\boldsymbol{\alpha}^{\mathrm{T}}} \boldsymbol{\Delta}_2
\end{aligned}
$$

Finally, (8) can be expressed using $\boldsymbol{\alpha}$ and $\boldsymbol{\Delta}_2$:

$$
\sum_{i=1}^{k} \Delta_i + \sum_{i=k+1}^{d} |\Delta_i| = -\boldsymbol{\alpha}^{\mathrm{T}} \boldsymbol{\Delta}_2 + ||\boldsymbol{\Delta}_2||_1 = \sum_{i=k+1}^{d} -\alpha_{i-k} \Delta_i + |\Delta_i| > 0 \quad (10)
$$

Equation (10) has to hold for any infinitesimal $\boldsymbol{\Delta}_2$. This is only the case if $|\alpha_i| < 1$ holds for all i, i.e. if

$$
||\boldsymbol{\alpha}||_\infty = \left\| \hat{\boldsymbol{Z}}_2 \hat{\boldsymbol{Z}}_1^{\mathrm{T}} \left(\hat{\boldsymbol{Z}}_1 \hat{\boldsymbol{Z}}_1^{\mathrm{T}}\right)^{-1} \mathbf{1} \right\|_\infty < 1. \qquad (11)
$$

(Note: $\boldsymbol{\Delta}_2 = \mathbf{0}$ and simultaneously $\boldsymbol{\Delta}_1 \neq \mathbf{0}$ is excluded according to (9)). □

So far, the above observations only apply for the optimisation problem (5). However, with the following minor changes the same condition is derived for (6):

1. The weight vectors $\boldsymbol{w}_0$ and $\boldsymbol{w}_1$ are defined analogously:

$$\boldsymbol{w}_0 = \underset{\boldsymbol{w}\in\Omega}{\arg\min} \quad ||\boldsymbol{w}||_1 \quad \text{subject to} \quad \boldsymbol{Z}^{\mathrm{T}}\boldsymbol{w} \geq \boldsymbol{0} \quad \text{and} \quad \bar{\boldsymbol{z}}^{\mathrm{T}}\boldsymbol{w} = 1$$
$$\boldsymbol{w}_1 = \underset{\boldsymbol{w}\in\mathbb{R}^d}{\arg\min} \quad ||\boldsymbol{w}||_1 \quad \text{subject to} \quad \boldsymbol{Z}^{\mathrm{T}}\boldsymbol{w} \geq \boldsymbol{0} \quad \text{and} \quad \bar{\boldsymbol{z}}^{\mathrm{T}}\boldsymbol{w} = 1$$

2. If $\boldsymbol{w}_0$ contains k non-zero entries, exactly k equations are active. The last of these constraints is the equality constraint $\bar{\boldsymbol{z}}^{\mathrm{T}}\boldsymbol{w} = 1$. To allow a compact notation, we include this constraint into the matrix $\boldsymbol{Z}$, i.e. we append the vector $\bar{\boldsymbol{z}}$.
3. The proof of Theorem 1 works analogously and leads to the same condition.

So, both approaches are very closely connected. However, they are not identical as the matrices $\boldsymbol{Z}$ are not the same.

4.3 Arguments for the Superior Results of the SFM

Due to the complexity of both approaches, it is not possible to give a rigorous mathematical proof for the superior performance of the SFM (6) compared to Weston's approach (5). However, within a simplified scenario and with approximate arguments we can use the result of the above theorem to make the superior performance plausible.

We consider the same scenario as in our experiments and assume the rows of $\boldsymbol{Z}$ to be drawn as $\mathcal{Z}_i$. The first k features are relevant — all others are irrelevant, i.e. the expected value of the first k features differs from zero, all others are exactly zero: $E(\mathcal{Z}_i) = c$ for $i = 1, \ldots, k$ and $E(\mathcal{Z}_i) = 0$ otherwise. For Weston's approach (5) we have $\hat{\boldsymbol{Z}}_1^{\mathrm{T}}\hat{\boldsymbol{w}}_0 = 1$ and obtain

$$\hat{\boldsymbol{Z}}_1\hat{\boldsymbol{Z}}_1^{\mathrm{T}}\hat{\boldsymbol{w}}_0 = \hat{\boldsymbol{Z}}_1\boldsymbol{1} \approx k \cdot c \cdot \boldsymbol{1} \qquad \Leftrightarrow \qquad \hat{\boldsymbol{w}}_0 \approx k \cdot c \cdot \left(\hat{\boldsymbol{Z}}_1\hat{\boldsymbol{Z}}_1^{\mathrm{T}}\right)^{-1}\boldsymbol{1}$$

such that

$$||\boldsymbol{\alpha}||_\infty \approx \left\| \frac{\hat{\boldsymbol{Z}}_2\hat{\boldsymbol{Z}}_1^{\mathrm{T}}\hat{\boldsymbol{w}}_0}{k \cdot c} \right\|_\infty = \left\| \frac{\hat{\boldsymbol{Z}}_2\boldsymbol{1}}{k \cdot c} \right\|_\infty = \left\| \frac{\boldsymbol{\epsilon}_k}{c} \right\|_\infty \quad \text{with} \quad \boldsymbol{\epsilon}_k := \frac{\hat{\boldsymbol{Z}}_2\boldsymbol{1}}{k} \in \mathbb{R}^{d-k}\,.$$

Here, the entries of the vector $\boldsymbol{\epsilon}_k$ are distributed as $\mathcal{N}(0, {}^{\sigma^2}/_k)$. In contrast, for the SFM (6), where the last column of $\hat{\boldsymbol{Z}}$ is the mean of all $\boldsymbol{z}_i$, we have $\hat{\boldsymbol{Z}}_1^{\mathrm{T}}\hat{\boldsymbol{w}}_0 = \binom{\boldsymbol{0}}{1}$ and obtain

$$\hat{\boldsymbol{Z}}_1\hat{\boldsymbol{Z}}_1^{\mathrm{T}}\hat{\boldsymbol{w}}_0 = \hat{\boldsymbol{Z}}_1\begin{pmatrix}\boldsymbol{0}\\1\end{pmatrix} \approx c \cdot \boldsymbol{1} \qquad \Leftrightarrow \qquad \hat{\boldsymbol{w}}_0 \approx c \cdot \left(\hat{\boldsymbol{Z}}_1\hat{\boldsymbol{Z}}_1^{\mathrm{T}}\right)^{-1}\boldsymbol{1}$$

and

$$||\boldsymbol{\alpha}||_\infty \approx \left\| \frac{\hat{\boldsymbol{Z}}_2\hat{\boldsymbol{Z}}_1^{\mathrm{T}}\hat{\boldsymbol{w}}_0}{c} \right\|_\infty = \left\| \frac{\hat{\boldsymbol{Z}}_2}{c}\begin{pmatrix}\boldsymbol{0}\\1\end{pmatrix} \right\|_\infty = \left\| \frac{\boldsymbol{\epsilon}_n}{c} \right\|_\infty \text{ with } \boldsymbol{\epsilon}_n := \hat{\boldsymbol{Z}}_2\binom{\boldsymbol{0}}{1} \in \mathbb{R}^{d-k+1}\,.$$

Obviously, for $k \ll n$, the probability that all elements of $\boldsymbol{\alpha}$ stay below 1 and, hence, that the condition in Theorem 1 to successfully find $\boldsymbol{w}_0$ is fulfilled, is much larger for the SFM. As expected, the larger c, the easier it is for both approaches to be successful. Note, that we assumed that the elements of $\hat{\boldsymbol{Z}}_1$ and $\hat{\boldsymbol{Z}}_2$ are independent stochastic variables. Of course, since $\hat{\boldsymbol{Z}}_1$ and $\hat{\boldsymbol{Z}}_2$ are selected by the respective algorithm according to certain criteria, this is not really the case.

5 Conclusions

The recently proposed SFM approach for feature selection identifies relevant features very effectively and may improve the generalisation performance significantly. It is based on the approximation of the weight vector's zero-norm by its one-norm. Here, we derived a condition under which both measures coincide. Unfortunately, in practise it is not possible to decide whether the condition is fulfilled for a specific dataset or not. However, one can compare the SFM approach with other zero-norm approximating methods such as Weston's method.

We found that the coincidence constraint in the SFM approach relies on averaging over n values, while in Weston's approach it relies on averaging over k values. According to this finding, it is beneficial to use the more stable SFM approach in scenarios with $n > k$. In toy experiments, we found that in almost all cases the SFM returns a lower number of features and a higher percentage of truly relevant features than Weston's method.

Further work will include a comparison of the SFM to other zero-norm approximating methods and the derivation of more strict constraints that could possibly be used to judge whether the solution to a specific dataset is close to the optimal one or not.

References

1. Frank, M., Wolfe, P.: An algorithm for quadratic programming. Naval Research Logistics Quarterly 3, 95–110 (1956)
2. Golub, T.R., Slonim, D.K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J.P., Coller, H., Loh, M.L., Downing, J.R., Caligiuri, M.A., Bloomfield, C.D.: Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. Science 286, 531–537 (1999)
3. Haynes, J.-D., Rees, G.: Decoding mental states from brain activity in humans. Nature Reviews Neuroscience 7, 523–534 (2006)
4. Klement, S., Martinetz, T.: A new approach to classification with the least number of features. In: ICMLA 2010, December 12-14, pp. 141–146. IEEE Computer Society, Washington, D.C, USA (2010)
5. Klement, S., Martinetz, T.: The support feature machine for classifying with the least number of features. In: Diamantaras, K., Duch, W., Iliadis, L.S. (eds.) ICANN 2010. LNCS, vol. 6353, pp. 88–93. Springer, Heidelberg (2010)
6. Weston, J., Elisseeff, A., Schölkopf, B., Tipping, M.: Use of the Zero-Norm with Linear Models and Kernel Methods. Journal of Machine Learning Research 3, 1439–1461 (2003)