JMLR: Workshop and Conference Proceedings 29:62–71, 2013 ACML 2013

Polynomial Runtime Bounds for Fixed-Rank Unsupervised Least-Squares Classification

Fabian Gieseke FABIAN.GIESEKE@DIKU.DK
Department of Computer Science, University of Copenhagen, Denmark

Tapio Pahikkala TAPIO.PAHIKKALA@UTU.FI
Department of Information Technology and Turku Centre for Computer Science, University of Turku, Finland

Christian Igel IGEL@DIKU.DK
Department of Computer Science, University of Copenhagen, Denmark

Editor: Cheng Soon Ong and Tu Bao Ho

Abstract

Maximum margin clustering can be regarded as the direct extension of support vector machines to unsupervised learning scenarios. The goal is to partition unlabeled data into two classes such that a subsequent application of a support vector machine would yield the overall best result (with respect to the optimization problem associated with support vector machines). While being very appealing from a conceptual point of view, the combinatorial nature of the induced optimization problem renders a direct application of this concept difficult. In order to obtain efficient optimization schemes, various surrogates of the original problem definition have been proposed in the literature. In this work, we consider one of these variants, called unsupervised regularized least-squares classification, which is based on the square loss, and develop polynomial upper runtime bounds for the induced combinatorial optimization task. In particular, we show that for n patterns and kernel matrix of fixed rank r (with given eigendecomposition), one can obtain an optimal solution in $\mathcal{O}(n^r)$ time for $r \leq 2$ and in $\mathcal{O}(n^{r-1})$ time for $r \geq 3$. The algorithmic framework is based on an interesting connection to the field of quadratic zero-one programming and permits the computation of exact solutions for the more general case of non-linear kernel functions in polynomial time.

Keywords: Maximum Margin Clustering, Combinatorial Optimization, Unsupervised Learning

1. Introduction

Maximum margin clustering (Xu et al., 2005) extends *support vector machines* (SVMs) (Boser et al., 1992; Cortes and Vapnik, 1995) to unsupervised learning: Instead of assuming the labels of the training patterns to be given, one aims at finding a partition of the data into two classes such that a standard supervised SVM yields an optimal value of the SVM optimization problem given the same hyperparameters. While SVMs induce convex optimization tasks that can be solved efficiently in polynomial time, the unsupervised extension yields a mixed-integer programming task, and these problems are known to be NP-hard in general (Vavasis, 1991).

While being very interesting from an application point of view, the combinatorial nature of the maximum margin clustering problem renders the search for an optimal (or, at least, a good) solution extremely difficult. A direct approach to solving the induced task is based on applying standard solvers for mixed-integer programming problems. For example, Bennett and Demiriz (1999) follow this approach for the related task of training *semi-supervised SVMs* (Vapnik and

Sterin, 1977). However, the running time of such schemes is exponential in the worst case, similar to the brute-force approach that checks every possible partition of the patterns into two classes. A variety of heuristic optimization approaches have been proposed that can generate reasonable candidate solutions efficiently. Ways to deal with the task include relaxing the original problem definition to obtain "easier" surrogates that are more amenable to efficient optimization strategies (Li et al., 2009; Valizadegan and Jin, 2007; Xu et al., 2005) or considering special cases that are, for instance, induced by linear kernel functions (Wang et al., 2010; Zhao et al., 2008).

Most work conducted so far has focused on the development of practical optimization schemes that result in valuable (but possibly suboptimal) candidate solutions. Surprisingly, the theoretical analysis of the underlying optimization task has gained little attention up to now. Among the few papers devoted to this topic is the one of Karnin et al. (2012), who propose several upper runtime bounds and hardness results for the maximum margin clustering problem induced by linear hard-margin SVMs (more precisely, for the so-called *furthest hyperplane problem*). Another approach is given by Peng et al. (2012), who combine an enumeration approach with a feature selection scheme. As mentioned above, applying standard solvers or branch-and-bound strategies yield optimal solutions (Chapelle et al., 2007), but no polynomial upper runtime bounds can be obtained this way.

Contribution: We consider *unsupervised regularized least-squares classification*, a prominent maximum margin clustering variant that is induced by the concept of regularized least-squares classification (Rifkin et al., 2003; Suykens and Vandewalle, 1999). As shown by several authors (Bach and Harchaoui, 2007; Gieseke et al., 2009; Zhang et al., 2007), this variant can yield a superior clustering performance for real-world data compared to the original problem definition (which is based on the hinge loss). Still, exactly as for the original task, the combinatorial nature usually requires heuristic approaches to efficiently generate valuable candidate solutions. We derive a polynomial time algorithm for solving the induced optimization task *exactly* in case the underlying kernel matrix (with given eigendecomposition) is of fixed rank r. Our bound is based on an interesting connection to the field of quadratic zero-one programming (Allemand et al., 2001; Ferrez et al., 2005), which yields a polynomial time enumeration framework that can be used to compute optimal solutions for n patterns in $\mathcal{O}(n^r)$ time for $r \leq 2$ and in $\mathcal{O}(n^{r-1})$ time for $r \geq 3$. In contrast to previous results, our approach is not restricted to the linear case only, but also applies to arbitrary kernel functions.

2. Mathematical Background

We start by briefly reviewing the concept of unsupervised regularized least-squares classification and sketch results from the field of zero-one programming that will be of relevance for this work.

2.1. Unsupervised Regularized Least-Squares Classification

Let $T = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\} \subset \mathbb{R}^d$ be a set of unlabeled patterns. The maximum margin clustering problem aims at partitioning these patterns into two classes such that a standard application of a SVM yields the best overall result. From a mathematical point of view, this yields (Xu et al., 2005):

$$\begin{aligned} \underset{\substack{\mathbf{y}\in\{-1,+1\}^n,\\ \mathbf{w}\in\mathbb{R}^d,\, b\in\mathbb{R},\, \boldsymbol{\xi}\in\mathbb{R}^n}}{\text{minimize}} \quad & \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n}\xi_i \\ \text{s.t.} \quad & y_i(\langle \mathbf{w}, \mathbf{x}_i\rangle + b) \geq 1 - \xi_i,\ \xi_i \geq 0, \end{aligned} \tag{1}$$

where $C > 0$. In addition, some kind of balancing constraint is needed to prevent undesired solutions such as assigning all patterns to one class. Typical candidates are constraints of the form $-l \leq \sum_{i=1}^{n} y_i \leq l$ with user-defined parameter $l \in \mathbb{N}$ or $\frac{1}{n}\sum_{i=1}^{n}\langle \mathbf{w}, \mathbf{x}_i\rangle + b \approx 0$ (Chapelle and Zien, 2005; Joachims, 1999; Xu et al., 2005).

Obviously, the difficulty of the optimization task consists in finding the correct assignment for the partition vector $\mathbf{y}$. Since we have both real-valued and integer optimization variables, we are dealing with a mixed-integer programming problem, and this class of optimization tasks is generally NP-hard (Vavasis, 1991). Exactly as for the concept of SVMs (Cortes and Vapnik, 1995), one can obtain more flexible models via the use of *kernel functions* $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ defined on an arbitrary input space $\mathcal{X}$ (Aronszajn, 1950). This leads to optimization problems of the more general form

$$\underset{\mathbf{y}\in\{-1,+1\}^n,\, f\in\mathcal{H}_k}{\text{minimize}} \; \sum_{i=1}^{n} L\big(y_i, f(\mathbf{x}_i)\big) + \lambda \|f\|_{\mathcal{H}_k}^2, \tag{2}$$

where $\lambda \in \mathbb{R}^+$ is a *regularization parameter*, $\mathcal{H}_k$ a *reproducing kernel Hilbert space* induced by the considered kernel function, and $L : \mathbb{R} \times \mathbb{R} \to [0, \infty)$ a *loss function*. Note that a balancing constraint of the form $\frac{1}{n}\sum_{i=1}^{n}\langle \mathbf{w}, \mathbf{x}_i\rangle + b = 0$ is implicitly enforced in the above formulation by setting $b = 0$ and by assuming that the data patterns are centered in the feature space (Chapelle and Zien, 2005). That is, we in general assume $\sum_{i=1}^{n} \Phi(\mathbf{x}_i) = \mathbf{0}$, where $\Phi(\mathbf{x}) = k(\cdot, \mathbf{x})$ is the mapping induced by the kernel function k.[1]

As pointed out above, several variants of the original maximum margin clustering problem have been proposed in the literature that stem from replacing the hinge loss $L(y, t) = \max(0, 1 - yt)$ by other loss functions. Among these modifications is the unsupervised regularized least-squares classification variant that is induced by the square loss (Bach and Harchaoui, 2007; Gieseke et al., 2009; Zhang et al., 2007):

$$\underset{\mathbf{y}\in\{-1,+1\}^n,\, f\in\mathcal{H}_k}{\text{minimize}} \; \sum_{i=1}^{n} \big(y_i - f(\mathbf{x}_i)\big)^2 + \lambda \|f\|_{\mathcal{H}_k}^2 \tag{3}$$

Due to the *representer theorem* (Schölkopf and Smola, 2001) given a fixed partition vector $\mathbf{y} \in \{-1, +1\}^n$, any optimal $f^* \in \mathcal{H}_k$ is of the form $f^*(\cdot) = \sum_{i=1}^{n} c_i k(\cdot, \mathbf{x}_i)$ with $c_1, \ldots, c_n \in \mathbb{R}$. Hence, using $\|f^*\|_{\mathcal{H}_k}^2 = \mathbf{c}^{\mathrm{T}}\mathbf{K}\mathbf{c}$ (Schölkopf and Smola, 2001), one can rewrite the above task as

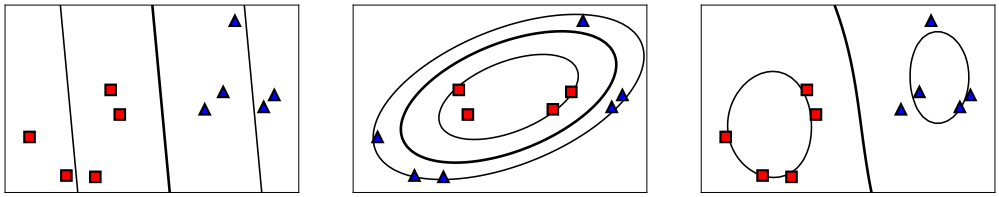
$$\underset{\mathbf{y}\in\{-1,+1\}^n,\, \mathbf{c}\in\mathbb{R}^n}{\text{minimize}} \; (\mathbf{y} - \mathbf{K}\mathbf{c})^{\mathrm{T}}(\mathbf{y} - \mathbf{K}\mathbf{c}) + \lambda \mathbf{c}^{\mathrm{T}}\mathbf{K}\mathbf{c}. \tag{4}$$

Note that solving this task is still very challenging due to the vector $\mathbf{y}$. In Figure 1, three optimal partitions are shown for a two-dimensional point set that stem from different kernel functions.

2.2. Unconstrained Zero-One Programming

As we will show below, one can eliminate the real-valued part of the optimization variables for the case of the square loss, which yields a pure integer programming problem. Such problems have gained a considerable attention in the field of mathematical optimization during the last decades,

1. Let $\mathbf{K} \in \mathbb{R}^{n\times n}$ be the positive semidefinite kernel (Gram) matrix induced by the sequence $\mathbf{x}_1, \ldots, \mathbf{x}_n$. Then centering can be easily achieved by considering $\tilde{\mathbf{K}} = \mathbf{K} - \frac{1}{n}\mathbf{1}_{nn}\mathbf{K} - \frac{1}{n}\mathbf{K}\mathbf{1}_{nn} + \frac{1}{n^2}\mathbf{1}_{nn}\mathbf{K}\mathbf{1}_{nn} = (\mathbf{I} - \frac{\mathbf{1}_{nn}}{n})\mathbf{K}(\mathbf{I} - \frac{\mathbf{1}_{nn}}{n})$ instead of $\mathbf{K}$, where $\mathbf{1}_{nn} \in \mathbb{R}^{n\times n}$ denotes the matrix full of ones (Schölkopf et al., 1998).



(*a*) (*b*) (*c*)

Figure 1: Three optimal partitions with respect to the unsupervised regularized least-squares classification task (4) given (a) a linear, (b) a polynomial, and (c) an RBF kernel.

both from a practical and a theoretical perspective. A special case of such problem instances are so-called *unconstrained quadratic maximization problems in zero-one variables* (01QPs) (Allemand et al., 2001; Ferrez et al., 2005) that are of the form

$$\underset{\mathbf{z}\in\{0,1\}^n}{\text{maximize}}\, h(\mathbf{z}) = \mathbf{z}^{\mathrm{T}}\mathbf{Q}\mathbf{z} \tag{5}$$

with an appropriate matrix $\mathbf{Q} \in \mathbb{R}^{n\times n}$. As pointed out by Allemand et al. (2001), problems of this type are still NP-hard in general (and this even holds in case $\mathbf{Q}$ is positive definite). However, there exists a small number of polynomial cases (e.g., if the matrix has rank one). Recently, Allemand et al. (2001) discovered one of them that is given when the matrix $\mathbf{Q}$ is positive semidefinite *and* of fixed rank r:

Theorem 1 (Allemand et al. (2001)) *Let $\mathbf{Q}$ be a positive semidefinite matrix with fixed rank $r = \operatorname{rank}(\mathbf{Q})$. In case a decomposition of the form $\mathbf{Q} = \mathbf{C}^{\mathrm{T}}\mathbf{C}$ with $\mathbf{C} \in \mathbb{R}^{r\times n}$ is explicitly given (i.e., precomputed and available in memory), one can compute an optimal solution for task (5) in $\mathcal{O}(n^r)$ time for $r \leq 2$ and in $\mathcal{O}(n^{r-1})$ time for $r \geq 3$.*

Proof sketch We briefly sketch the key ideas of the proof since our approach makes use of this algorithmic building block, see Allemand et al. (2001) for details. The linear map $g : \mathbb{R}^n \to \mathbb{R}^r$ with $g(\mathbf{z}) := \mathbf{C}\mathbf{z}$ maps the hypercube $[0,1]^n$ to a special convex polytope P in $\mathbb{R}^r$, called *zonotope* (Edelsbrunner, 1987). Furthermore, for the optimal $h^* = \operatorname{maximize}_{\mathbf{z}\in\{0,1\}^n} \mathbf{z}^{\mathrm{T}}\mathbf{Q}\mathbf{z}$, we have

$$h^* = \underset{\mathbf{z}\in\{0,1\}^n}{\text{maximize}}\, \mathbf{z}^{\mathrm{T}}\mathbf{C}^{\mathrm{T}}\mathbf{C}\mathbf{z} = \underset{\mathbf{z}\in[0,1]^n}{\text{maximize}}\, \mathbf{z}^{\mathrm{T}}\mathbf{C}^{\mathrm{T}}\mathbf{C}\mathbf{z} = \underset{\mathbf{p}\in P}{\text{maximize}}\, \|\mathbf{p}\|^2, \tag{6}$$

where the second equality follows from the convexity of the objective. As pointed out by Allemand et al. (2001), the last term depicts the maximization of a convex function over the convex set P. Thus, the maximum is attained at one of the extreme points of P. Furthermore, for each extreme point $\hat{\mathbf{p}} \in P$, there exists an extreme point $\hat{\mathbf{z}}$ of $[0,1]^n$ with $\hat{\mathbf{p}} = \mathbf{C}\hat{\mathbf{z}}$ (Allemand et al., 2001).

The construction of zonotopes is a well-known task in discrete geometry. In the above setting, P is the *Minkowski sum* of the n line segments $[\mathbf{0}, \mathbf{c}_i]$, where the so-called *generator* $\mathbf{c}_i$ denotes the ith column of $\mathbf{C}$ (Allemand et al., 2001). There exists a classical result in the field of discrete geometry that provides an upper bound on the number $h_0(P)$ of extreme points of P:

Theorem 2 (Edelsbrunner (1987)) *The number $h_0(P)$ of extreme points of P is in $\mathcal{O}(n^{r-1})$.*

As shown by Allemand et al. (2001), there exists an efficient scheme that enumerates all extreme points $\hat{\mathbf{p}}$ of P (along with an associated vector $\hat{\mathbf{z}} \in \{0,1\}^n$ with $\hat{\mathbf{p}} = \mathbf{C}\hat{\mathbf{z}}$) in $\mathcal{O}(n^r)$ time for $r \leq 2$ and in $\mathcal{O}(n^{r-1})$ time for $r \geq 3$. For each extreme point, one can then evaluate the right-hand side of (6) spending $\mathcal{O}(r) = \mathcal{O}(1)$ time to obtain the overall best objective. ■

Note that the above result assumes that elementary operations for real numbers can be performed in constant time (as it is assumed in, e.g., the *real-random access memory model* (Edelsbrunner, 1987; Preparata and Shamos, 1985)).

3. Unsupervised Least-Squares via Convex Zero-One Programming

In this section, we will show that the concept of unsupervised regularized least-squares classification is, indeed, of the special form depicted above. Note that the induced optimization task (4) is, in its original form, still very challenging to address due to the partition vector $\mathbf{y} \in \{-1,+1\}^n$.

3.1. Convex Quadratic Objective

For a fixed partition vector, the task (4) gives rise to a convex optimization problem: The gradient and the Hessian of the objective are given by $\nabla_{\mathbf{c}} J(\mathbf{y},\mathbf{c}) = -2(\mathbf{K})^{\mathrm{T}}(\mathbf{y} - \mathbf{K}\mathbf{c}) + 2\lambda\mathbf{K}\mathbf{c}$ and $\nabla^2_{\mathbf{c}} J(\mathbf{y},\mathbf{c}) = 2(\mathbf{K})^{\mathrm{T}}\mathbf{K} + 2\lambda\mathbf{K}$, respectively. Due to the kernel matrix being positive semidefinite, the Hessian is positive semidefinite as well. Further, an optimal solution can be obtained via $\mathbf{c}^* = \mathbf{G}\mathbf{y}$ with $\mathbf{G} = (\mathbf{K} + \lambda\mathbf{I})^{-1}$. Plugging in these intermediate solutions into the objective (4) leads to:

$$\begin{aligned} F(\mathbf{y}) &= (\mathbf{y} - \mathbf{K}\mathbf{c}^*)^{\mathrm{T}}(\mathbf{y} - \mathbf{K}\mathbf{c}^*) + \lambda(\mathbf{c}^*)^{\mathrm{T}}\mathbf{K}\mathbf{c}^* \\ &= \mathbf{y}^{\mathrm{T}}\left(\mathbf{I} - \mathbf{K}\mathbf{G} - \mathbf{G}\mathbf{K} + \mathbf{G}\mathbf{K}\mathbf{K}\mathbf{G} + \lambda\mathbf{G}\mathbf{K}\mathbf{G}\right)\mathbf{y}, \end{aligned} \tag{7}$$

where we used $\mathbf{K} = \mathbf{K}^{\mathrm{T}}$ and $\mathbf{G}^{\mathrm{T}} = \mathbf{G}$. The next lemma shows that one can obtain the following equivalent zero-one programming task:

Lemma 3 *The optimization task (4) is equivalent to*

$$\underset{\mathbf{z}\in\{0,1\}^n}{\text{maximize}}\ \mathbf{z}^{\mathrm{T}}\mathbf{V}\mathbf{D}\mathbf{D}_\lambda\mathbf{V}^{\mathrm{T}}\mathbf{z}, \tag{8}$$

where $\mathbf{K} = \mathbf{V}\mathbf{D}\mathbf{V}^{\mathrm{T}}$ *is the eigendecomposition of* $\mathbf{K}$ *and* $\mathbf{D}_\lambda := (\mathbf{D} + \lambda\mathbf{I})^{-1}$.

Proof Let $\mathbf{K} = \mathbf{V}\mathbf{D}\mathbf{V}^{\mathrm{T}}$ be the eigendecomposition (Golub and Van Loan, 1996) of the positive semidefinite kernel matrix $\mathbf{K}$ with orthogonal matrix $\mathbf{V} \in \mathbb{R}^{n\times n}$ and diagonal matrix $\mathbf{D} \in \mathbb{R}^{n\times n}$. Since the matrix $\mathbf{V} \in \mathbb{R}^{n\times n}$ is orthogonal, we have

$$\mathbf{G} = (\mathbf{K} + \lambda\mathbf{I})^{-1} = (\mathbf{V}\mathbf{D}\mathbf{V}^{\mathrm{T}} + \lambda\mathbf{V}\mathbf{V}^{\mathrm{T}})^{-1} = (\mathbf{V}(\mathbf{D} + \lambda\mathbf{I})\mathbf{V}^{\mathrm{T}})^{-1} = \mathbf{V}\mathbf{D}_\lambda\mathbf{V}^{\mathrm{T}}, \tag{9}$$

which in turn implies the following four equations:

$$\begin{aligned} \mathbf{K}\mathbf{G} &= \mathbf{V}\mathbf{D}\mathbf{V}^{\mathrm{T}}\mathbf{V}\mathbf{D}_\lambda\mathbf{V}^{\mathrm{T}} = \mathbf{V}\mathbf{D}\mathbf{D}_\lambda\mathbf{V}^{\mathrm{T}} \\ \mathbf{G}\mathbf{K} &= \mathbf{V}\mathbf{D}_\lambda\mathbf{V}^{\mathrm{T}}\mathbf{V}\mathbf{D}\mathbf{V}^{\mathrm{T}} = \mathbf{V}\mathbf{D}_\lambda\mathbf{D}\mathbf{V}^{\mathrm{T}} \\ \mathbf{G}\mathbf{K}\mathbf{K}\mathbf{G} &= \mathbf{V}\mathbf{D}_\lambda\mathbf{V}^{\mathrm{T}}\mathbf{V}\mathbf{D}\mathbf{V}^{\mathrm{T}}\mathbf{V}\mathbf{D}\mathbf{V}^{\mathrm{T}}\mathbf{V}\mathbf{D}_\lambda\mathbf{V}^{\mathrm{T}} = \mathbf{V}\mathbf{D}_\lambda\mathbf{D}\mathbf{D}\mathbf{D}_\lambda\mathbf{V}^{\mathrm{T}} \\ \mathbf{G}\mathbf{K}\mathbf{G} &= \mathbf{V}\mathbf{D}_\lambda\mathbf{V}^{\mathrm{T}}\mathbf{V}\mathbf{D}\mathbf{V}^{\mathrm{T}}\mathbf{V}\mathbf{D}_\lambda\mathbf{V}^{\mathrm{T}} = \mathbf{V}\mathbf{D}_\lambda\mathbf{D}\mathbf{D}_\lambda\mathbf{V}^{\mathrm{T}} \end{aligned}$$

Using these equations, we can follow Pahikkala et al. (2012) and rewrite the objective (7) as:

$$\begin{aligned} F(\mathbf{y}) &= \mathbf{y}^{\mathrm{T}} \left(\mathbf{I} - \mathbf{KG} - \mathbf{GK} + \mathbf{GKKG} + \lambda \mathbf{GKG}\right) \mathbf{y} \\ &= \mathbf{y}^{\mathrm{T}} \mathbf{V} \left(\mathbf{I} - 2\mathbf{D}\mathbf{D}_\lambda + \mathbf{D}^2\mathbf{D}_\lambda^2 + \lambda \mathbf{D}\mathbf{D}_\lambda^2\right) \mathbf{V}^{\mathrm{T}} \mathbf{y} \\ &= \mathbf{y}^{\mathrm{T}} \mathbf{V} \left(\mathbf{I} + (-2\mathbf{I} + \mathbf{D}_\lambda \mathbf{D} + \lambda \mathbf{D}_\lambda)\mathbf{D}\mathbf{D}_\lambda\right) \mathbf{V}^{\mathrm{T}} \mathbf{y} \\ &= \mathbf{y}^{\mathrm{T}} \mathbf{V} \left(\mathbf{I} + (-2\mathbf{I} + \mathbf{D}_\lambda (\mathbf{D} + \lambda \mathbf{I}))\mathbf{D}\mathbf{D}_\lambda\right) \mathbf{V}^{\mathrm{T}} \mathbf{y} \\ &= \mathbf{y}^{\mathrm{T}} \mathbf{V} \left(\mathbf{I} - \mathbf{D}\mathbf{D}_\lambda\right) \mathbf{V}^{\mathrm{T}} \mathbf{y} \\ &= n - \mathbf{y}^{\mathrm{T}} \mathbf{V}\mathbf{D}\mathbf{D}_\lambda \mathbf{V}^{\mathrm{T}} \mathbf{y} \end{aligned} \tag{10}$$

To obtain the desired zero-one programming task, we now make use of the bijective mapping $\phi : \{-1, +1\}^n \to \{0, 1\}^n$ with $\phi(\mathbf{y}) = \frac{\mathbf{y}+1}{2}$ and $\phi^{-1}(\mathbf{z}) = 2\mathbf{z} - 1$. This leads to an equivalent task for (4) having the form

$$\underset{\mathbf{z} \in \{0,1\}^n}{\text{minimize}} \; n - 4\mathbf{z}^{\mathrm{T}} \mathbf{V}\mathbf{D}\mathbf{D}_\lambda \mathbf{V}^{\mathrm{T}} \mathbf{z} + \mathbf{1}^{\mathrm{T}} \mathbf{V}\mathbf{D}\mathbf{D}_\lambda \mathbf{V}^{\mathrm{T}} \mathbf{1}, \tag{11}$$

which concludes the proof since the constant terms and the scaling factors can be ignored. ■

Thus, given an optimal $\mathbf{z}^* \in \{0, 1\}^n$ for task (8), one obtains an optimal partition vector $\mathbf{y}^* \in \{-1, +1\}^n$ for task (4) via $\mathbf{y}^* = \phi^{-1}(\mathbf{z}^*) = 2\mathbf{z}^* - 1$. An important property of the task (8) is the fact that the associated matrix is positive semidefinite:

Lemma 4 *The matrix $\mathbf{Q} := \mathbf{V}\mathbf{D}\mathbf{D}_\lambda \mathbf{V}^{\mathrm{T}}$ in optimization task (8) is positive semidefinite.*

Proof The diagonal matrix $\mathbf{D}\mathbf{D}_\lambda$ contains the eigenvalues of $\mathbf{Q}$ and has only non-negative entries since $\mathbf{K}$ is positive semidefinite. ■

3.2. Upper Runtime Bounds

The above derivations show that one can address the equivalent task (8) that is of the form

$$\underset{\mathbf{z} \in \{0,1\}^n}{\text{maximize}} \; \mathbf{z}^{\mathrm{T}} \underbrace{\mathbf{V}\mathbf{D}\mathbf{D}_\lambda \mathbf{V}^{\mathrm{T}}}_{=\mathbf{Q} \in \mathbb{R}^{n \times n}} \mathbf{z} \tag{12}$$

with positive semidefinite matrix $\mathbf{Q}$. Thus, in contrast to the original task (4), we are now dealing with a *convex* zero-one maximization task.

In the following, we will assume that the kernel matrix $\mathbf{K}$ has fixed rank $r < n$ and that its eigendecomposition $\mathbf{K} = \mathbf{V}\mathbf{D}\mathbf{V}^{\mathrm{T}}$ is given.[2] Under these conditions, the unsupervised regularized least-squares classification task can be solved in polynomial time:

Theorem 5 *Given a kernel matrix $\mathbf{K}$ of fixed rank $r < n$ and associated eigendecomposition $\mathbf{K} = \mathbf{V}\mathbf{D}\mathbf{V}^{\mathrm{T}}$ (precomputed and available in memory), one can solve the optimization task (4) in $\mathcal{O}(n^r)$ time for $r \leq 2$ and in $\mathcal{O}(n^{r-1})$ time for $r \geq 3$, respectively.*

2. One can obtain the eigendecomposition $\mathbf{K} = \mathbf{V}\mathbf{D}\mathbf{V}^{\mathrm{T}}$ of the kernel matrix in $\mathcal{O}(n^2 r)$ time (up to machine precision and in practice) (Golub and Van Loan, 1996; Halko et al., 2011). Similar to Allemand et al. (2001), we do not dwell on the complexity for computing such a decomposition precisely from a theoretical perspective and will simply assume it to be given. As pointed out by Allemand et al. (2001), "eliminating this assumption is a sufficiently interesting problem on its own".

Proof Let us assume that the eigenvalues (and the corresponding eigenvectors) are ordered such that $[\mathbf{D}]_{i,i} = 0$ for $i > r$ (this can be achieved spending $\mathcal{O}(n)$ time). In this case, we have $\mathbf{K} = \hat{\mathbf{V}}\hat{\mathbf{D}}\hat{\mathbf{V}}^{\mathrm{T}}$ with appropriate $\hat{\mathbf{V}} \in \mathbb{R}^{n \times r}$ and $\hat{\mathbf{D}} \in \mathbb{R}^{r \times r}$. Thus, one can explicitly compute $\mathbf{Q} = \hat{\mathbf{V}}\hat{\mathbf{D}}\hat{\mathbf{D}}_{\lambda}\hat{\mathbf{V}}^{\mathrm{T}} = \mathbf{C}^{\mathrm{T}}\mathbf{C}$ with $\mathbf{C} = \hat{\mathbf{B}}^{\mathrm{T}}\hat{\mathbf{V}}^{\mathrm{T}} \in \mathbb{R}^{r \times n}$ and appropriately chosen diagonal matrix $\hat{\mathbf{B}}$ with $\hat{\mathbf{B}}\hat{\mathbf{B}}^{\mathrm{T}} = \hat{\mathbf{D}}\hat{\mathbf{D}}_{\lambda}$.

Thus, we can formalize the equivalent optimization task (8) in linear time. Since $\mathbf{C}$ has rank at most r, $\mathbf{Q}$ has rank at most r as well. In addition, since $\mathbf{Q}$ is positive semidefinite by Lemma 4, we can apply Theorem 1 to obtain an optimal solution for task (8) in $\mathcal{O}(n^r)$ time for $r \leq 2$ and in $\mathcal{O}(n^{r-1})$ time for $r \geq 3$, respectively.

The optimal solution $\mathbf{z} \in \{0,1\}^n$ can then be mapped to an optimal partition vector $\mathbf{y}^* \in \{-1,+1\}^n$ via ϕ^{-1} spending linear time. Finally, the associated model vector can be obtained in $\mathcal{O}(nr) = \mathcal{O}(n)$ time via $\mathbf{c}^* = \mathbf{G}\mathbf{y}^* = \hat{\mathbf{V}}\hat{\mathbf{D}}_{\lambda}\hat{\mathbf{V}}^{\mathrm{T}}\mathbf{y}^*$. ■

The above theorem states that the unsupervised regularized least-squares classification task can be solved in polynomial time given a kernel matrix of rank r. As a direct application of this result, we may consider the rank r approximations of an arbitrary kernel matrix $\mathbf{K}$ (with possibly full rank), the best one (with respect to, e.g., the Frobenius and the 2-norm) being obtained from its truncated eigendecomposition $\mathbf{K} \approx \hat{\mathbf{V}}\hat{\mathbf{D}}\hat{\mathbf{V}}^{\mathrm{T}}$, that is, the matrix $\hat{\mathbf{D}} \in \mathbb{R}^{r \times r}$ containing the r largest eigenvalues and $\hat{\mathbf{V}} \in \mathbb{R}^{n \times r}$ the corresponding eigenvectors. Such an approximation can be computed in roughly $\mathcal{O}(n^2 r)$ time, for example, via the recently proposed randomized matrix decomposition algorithms (Halko et al., 2011).

Corollary 6 *Let $\bar{\mathbf{K}} = \hat{\mathbf{V}}\hat{\mathbf{D}}\hat{\mathbf{V}}^{\mathrm{T}}$ be a rank-r approximation (precomputed and available in memory) of an arbitrary kernel matrix $\mathbf{K}$. Then, one can solve the optimization task (4) for the approximated kernel matrix in $\mathcal{O}(n^r)$ time for $r \leq 2$ and in $\mathcal{O}(n^{r-1})$ time for $r \geq 3$, respectively.*

Another special case is given for linear kernel functions with $\mathbf{K} = \mathbf{X}\mathbf{X}^{\mathrm{T}}$, where $\mathbf{X} \in \mathbb{R}^{n \times d}$ contains the patterns as rows. Again, since we have $\mathrm{rank}(\mathbf{K}) \leq d$, it follows:

Corollary 7 *Let $\mathbf{K} = \mathbf{X}\mathbf{X}^{\mathrm{T}}$ with $\mathbf{X} \in \mathbb{R}^{n \times d}$ be the linear kernel matrix and $\mathbf{K} = \mathbf{V}\mathbf{D}\mathbf{V}^{\mathrm{T}}$ its eigendecomposition (precomputed and available in memory). Then, one can solve the optimization task (4) in $\mathcal{O}(n^d)$ time for $d \leq 2$ and in $\mathcal{O}(n^{d-1})$ time for $d \geq 3$, respectively.*

3.3. Algorithmic Framework

The overall framework for solving the original combinatorial task (4) is given in Algorithm 1: As shown in the proof of Theorem 5, we can rewrite the eigendecomposition of the kernel matrix $\mathbf{K} = \mathbf{V}\mathbf{D}\mathbf{V}^{\mathrm{T}}$ as $\mathbf{K} = \hat{\mathbf{V}}\hat{\mathbf{D}}\hat{\mathbf{V}}^{\mathrm{T}}$. This directly yields the decomposition $\mathbf{Q} = \mathbf{C}^{\mathrm{T}}\mathbf{C}$ with $\mathbf{C} \in \mathbb{R}^{r \times n}$ needed for initializing the equivalent task (8) in Step 2.

One can then resort to the approach of Allemand et al. (2001) in Step 3 to enumerate all extreme points $\hat{\mathbf{p}}$ of the zonotope P that is generated by the n line segments $[\mathbf{0}, \mathbf{c}_i]$ (along with an associated vector $\hat{\mathbf{z}} \in \{0,1\}^n$ with $\hat{\mathbf{p}} = \mathbf{C}\hat{\mathbf{z}}$). This is illustrated in Figure 2, which shows the zonotope P that is induced by the point configuration depicted in Figure 1 (c).[3] Here, the red squares depict the extreme points $\hat{\mathbf{p}}$ of P and the black lines the generators $[\mathbf{0}, \mathbf{c}_1], \ldots, [\mathbf{0}, \mathbf{c}_n]$.

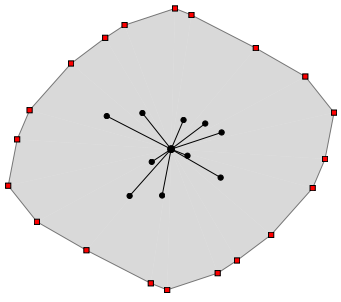


Figure 2: Zonotope P

3. Using an RBF kernel and a rank-2 approximation of $\mathbf{K}$.

Algorithm 1 Fixed-Rank Unsupervised Regularized Least-Squares Classification

Input: A kernel matrix $\mathbf{K}$ of fixed rank r with associated eigendecomposition $\mathbf{K} = \mathbf{V}\mathbf{D}\mathbf{V}^{\mathrm{T}}$.
Output: An optimal solution $(\mathbf{y}^*, \mathbf{c}^*) \in \{-1, +1\}^n \times \mathbb{R}^n$ for optimization task (4).
1: Compute $\mathbf{K} = \hat{\mathbf{V}}\hat{\mathbf{D}}\hat{\mathbf{V}}^{\mathrm{T}}$ with $\hat{\mathbf{V}} \in \mathbb{R}^{n \times r}$ and $\hat{\mathbf{D}} \in \mathbb{R}^{r \times r}$.
2: Set up the equivalent task (8) with $\mathbf{Q} = \mathbf{C}^{\mathrm{T}}\mathbf{C}$, where $\mathbf{C} = \hat{\mathbf{B}}^{\mathrm{T}}\hat{\mathbf{V}}^{\mathrm{T}} \in \mathbb{R}^{r \times n}$ and $\hat{\mathbf{B}}\hat{\mathbf{B}}^{\mathrm{T}} = \hat{\mathbf{D}}\hat{\mathbf{D}}_{\lambda}$.
3: Compute the extreme points of the zonotope P that is generated by the n line segments $[0, \mathbf{c}_i]$.
4: For each extreme point $\hat{\mathbf{p}}$, evaluate the right-hand side of (6) (and store the overall best result).
5: Let $\hat{\mathbf{p}}^*$ be the optimal extreme point and $\hat{\mathbf{z}}^* \in \{0, 1\}^n$ its associated vector with $\hat{\mathbf{p}}^* = \mathbf{C}\hat{\mathbf{z}}^*$.
6: Compute $\mathbf{y}^* = \phi^{-1}(\hat{\mathbf{z}}^*)$ and $\mathbf{c}^* = \mathbf{G}\mathbf{y}^* = \hat{\mathbf{V}}\hat{\mathbf{D}}_{\lambda}\hat{\mathbf{V}}^{\mathrm{T}}\mathbf{y}^*$.
7: **return** $(\mathbf{y}^*, \mathbf{c}^*)$

The overall best objective of the equivalent task (8) can then be obtained in Step 4 by evaluating the right-hand side of (6) for each extreme point. Both steps take $\mathcal{O}(n^r)$ for $r \leq 2$ and $\mathcal{O}(n^{r-1})$ for $r \geq 3$ and yield the optimal $\hat{\mathbf{z}}^*$ for task (8). Using ϕ^{-1}, one can finally obtain the optimal partition vector $\mathbf{y}^*$ for the original task (4) along with the model parameter $\mathbf{c}^*$ in $\mathcal{O}(rn) = \mathcal{O}(n)$ time.

4. Conclusions

We have shown that the optimization task induced by the concept of unsupervised regularized least-squares classification can be solved in $\mathcal{O}(n^r)$ time for $r \leq 2$ and in $\mathcal{O}(n^{r-1})$ time for $r \geq 3$, respectively, given an arbitrary kernel matrix $\mathbf{K}$ of fixed rank r and available eigendecomposition. While several theoretical results have been proposed for the linear case (for linear maximum margin clustering), our derivations also hold for the more general case covering non-linear kernel functions. Our framework makes use of a recently discovered polynomial case in the field of unconstrained zero-one programming, which in turn is based on non-trivial results and algorithms from discrete geometry.

While the results presented in this work are mostly of theoretical interest, they might also pave the way for efficient implementations from a practical point of view. For instance, the algorithmic framework outlined above yields an efficient optimization approach for low-dimensional feature spaces (e.g., fore- and background separation in image analysis).[4] Further, the special form of the equivalent optimization task (8) might be useful for developing efficient (parallel) branch-and-bound implementations. We plan to investigate these perspectives in near future.

Acknowledgements We would like to thank the anonymous reviewers for their detailed comments. Fabian Gieseke gratefully acknowledges support from *German Academic Exchange Service* (DAAD), Tapio Pahikkala from the Academy of Finland (grant 134020), and Christian Igel from *The Danish Council for Independent Research | Natural Sciences* through the project *Surveying the sky using machine learning* (SkyML).

4. For instance, intuitive clustering solutions for the well-known *Old Faithful Geyser* data set (see, e.g., Bishop (2007)) with $n = 272$ and $d = 2$ can be efficiently obtained spending less than half a second on a standard desktop machine, on which the brute-force approach cannot handle more than 20 points in a reasonable amount of time. The corresponding prototype implementation (written in `Python` and publicly available at the authors' websites) makes use of a one-to-one correspondence between the extreme points of the zonotope P and the r-dimensional cells of an associated *arrangement* $\mathcal{A}(P) \subset \mathbb{R}^r$ (Ferrez et al., 2005). The extreme points of P can therefore be obtained by computing appropriate position vectors in $\mathcal{A}(P)$, and for this task, parallel enumerations schemes (Ferrez et al., 2005) have been recently proposed that might be of practical relevance for upcoming implementations.

References

Kim Allemand, Komei Fukuda, Thomas M. Liebling, and Erich Steiner. A polynomial case of unconstrained zero-one quadratic optimization. *Mathematical Programming*, 91(1):49–52, 2001.

Nachman Aronszajn. Theory of reproducing kernels. *Transactions of the American Mathematical Society*, 68(3):337–404, 1950.

Francis R. Bach and Zaïd Harchaoui. DIFFRAC: a discriminative and flexible framework for clustering. In *Advances in Neural Information Processing Systems 20*, pages 49–56. MIT Press, Cambridge, MA, 2007.

Kristin P. Bennett and Ayhan Demiriz. Semi-supervised support vector machines. In *Advances in Neural Information Processing Systems 11*, pages 368–374. MIT Press, 1999.

Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2007.

Bernhard E. Boser, Isabelle M. Guyon, and Vladimir N. Vapnik. A training algorithm for optimal margin classifiers. In *Proceedings of the 5th Annual Workshop on Computational Learning Theory*, pages 144–152, 1992.

Olivier Chapelle and Alexander Zien. Semi-supervised classification by low density separation. In *Proceedings of the 10th International Workshop on Artifical Intelligence and Statistics*, pages 57–64, 2005.

Olivier Chapelle, Vikas Sindhwani, and S. Sathiya Keerthi. Branch and bound for semi-supervised support vector machines. In *Advances in Neural Information Processing Systems 19*, pages 217–224. MIT Press, 2007.

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

Herbert Edelsbrunner. *Algorithms in Combinatorial Geometry*. Springer-Verlag, New York, NY, USA, 1987.

Jean Albert Ferrez, Komei Fukuda, and Thomas M. Liebling. Solving the fixed rank convex quadratic maximization in binary variables by a parallel zonotope construction algorithm. *European Journal of Operational Research*, 166(1):35–50, 2005.

Fabian Gieseke, Tapio Pahikkala, and Oliver Kramer. Fast evolutionary maximum margin clustering. In *Proc. of the International Conference on Machine Learning*, pages 361–368, 2009.

Gene H. Golub and Charles Van Loan. *Matrix Computations*. Johns Hopkins University Press, Baltimore and London, third edition, 1996.

Nathan Halko, Per-Gunnar Martinsson, and Joel A. Tropp. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM Review*, 53 (2):217–288, 2011.

Thorsten Joachims. Transductive inference for text classification using support vector machines. In *Proceedings of the International Conference on Machine Learning*, pages 200–209, 1999.

Zohar Karnin, Edo Liberty, Shachar Lovett, Roy Schwartz, and Omri Weinstein. Unsupervised svms: On the complexity of the furthest hyperplane problem. In *Proceedings of the 25th Annual Conference on Learning Theory (COLT)*. JMLR.org, 2012.

Yu-Feng Li, Ivor W. Tsang, James T. Kwok, and Zhi-Hua Zhou. Tighter and convex maximum margin clustering. In *Proceedings of the 12th International Conference on Artificial Intelligence and Statistics*, pages 344–351. JMLR: W&CP 5, 2009.

Tapio Pahikkala, Antti Airola, Fabian Gieseke, and Oliver Kramer. Unsupervised multi-class regularized least-squares classification. In *12th IEEE International Conference on Data Mining*, pages 585–594. IEEE Computer Society, 2012.

Jiming Peng, Lopamudra Mukherjee, Vikas Singh, Dale Schuurmans, and Linli Xu. An efficient algorithm for maximal margin clustering. *Journal of Global Optimization*, 52(1):123–137, 2012.

Franco P. Preparata and Michael Ian Shamos. *Computational Geometry – An Introduction*. Springer, 1985.

Ryan Rifkin, Gene Yeo, and Tomaso Poggio. Regularized least-squares classification. In *Advances in Learning Theory: Methods, Models and Applications*. IOS Press, 2003.

Bernhard Schölkopf and Alexander J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, Cambridge, MA, USA, 2001.

Bernhard Schölkopf, Alexander Smola, and Klaus-Robert Müller. Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, 10(5):1299–1319, July 1998.

Johan A. K. Suykens and Joos Vandewalle. Least squares support vector machine classifiers. *Neural Processing Letters*, 9(3):293–300, 1999.

Hamed Valizadegan and Rong Jin. Generalized maximum margin clustering and unsupervised kernel learning. In *Advances in Neural Information Processing Systems 19*, pages 1417–1424, 2007.

Vladimir N. Vapnik and Alexander M. Sterin. On structural risk minimization or overall risk in a problem of pattern recognition. *Automation and Remote Control*, 10(3):1495–1503, 1977.

Stephen A. Vavasis. *Nonlinear Optimization: Complexity Issues*. Oxford University Press, 1991.

Fei Wang, Bin Zhao, and Changshui Zhang. Linear time maximum margin clustering. *IEEE Transactions on Neural Networks*, 21(2):319–332, 2010.

Linli Xu, James Neufeld, Bryce Larson, and Dale Schuurmans. Maximum margin clustering. In *Advances in Neural Information Processing Systems 17*, pages 1537–1544, 2005.

Kai Zhang, Ivor W. Tsang, and James T. Kwok. Maximum margin clustering made practical. In *Proceedings of the International Conference on Machine Learning*, pages 1119–1126, 2007.

Bin Zhao, Fei Wang, and Changshui Zhang. Efficient maximum margin clustering via cutting plane algorithm. In *Proc. of the SIAM International Conference on Data Mining*, pages 751–762, 2008.