Feature Selection for Density Level-Sets

Marius Kloft[1], Shinichi Nakajima[2], and Ulf Brefeld[1]

[1] Machine Learning Group, Technische Universität Berlin, Berlin, Germany
{mkloft,brefeld}@cs.tu-berlin.de
[2] Optical Research Laboratory, Nikon Corporation, Tokyo, Japan
nakajima.s@nikon.co.jp

Abstract. A frequent problem in density level-set estimation is the choice of the right features that give rise to compact and concise representations of the observed data. We present an efficient feature selection method for density level-set estimation where optimal kernel mixing coefficients *and* model parameters are determined simultaneously. Our approach generalizes one-class support vector machines and can be equivalently expressed as a semi-infinite linear program that can be solved with interleaved cutting plane algorithms. The experimental evaluation of the new method on network intrusion detection and object recognition tasks demonstrate that our approach not only attains competitive performance but also spares practitioners from *a priori* decisions on feature sets to be used.

1 Introduction

The set of points on which a function f exceeds a certain value ρ, e.g., $D_\rho = \{\boldsymbol{x} : f(\boldsymbol{x}) \geq \rho\}$, is called a level-set D_ρ. Boundaries of such sets typically constitute submanifolds in feature space whereas level-set approaches are frequently used for function estimation and denoising.

For anomaly and outlier detection tasks, level-set methods are often observed to outperform probability density estimators which have to be thresholded accordingly to act as detectors for unlikely and rare events. Statistical approaches frequently focus on *high density* regions to capture the underlying probability distribution. By contrast, density level-set estimators are specially tailored to work well in *low density* regions which is a crucial property for detecting anomalous events.

In this paper, we focus on level-set estimation for anomaly and outlier detection [9,4], where a model of normality is devised from available observations. Anomality of new objects is measured by their distance (in some metric space) from the learned model of normality. Apart from theoretical observations, in practice the effectiveness of density level-set estimation crucially depends on the representation of the observations and thus on the choice of features.

However, characteristic traits of particular learning problems are often spread across multiple features that capture various properties of data, giving rise to a set of kernel matrices $K_1, \ldots, K_m$ that have to be combined appropriately. As

W. Buntine et al. (Eds.): ECML PKDD 2009, Part I, LNAI 5781, pp. 692–704, 2009.

a motivating example, consider network intrusion detection where various sets of features have been deployed, including raw values of IP and TCP protocol headers [15,16], time and connection windows [13], byte histograms and n-grams [29,28], and "bag-of-tokens" language models [21,22]. While packet header based features have been shown to be effective against probes and scans, other kinds of attacks, e.g. remote buffer overflows, require more advanced payload processing techniques. The right kind of features for a particular application has always been considered as the matter of a judicious choice (or trial and error).

But what if this decision is really difficult to make? Given the choice of several kinds of features, a poor a priori decision would lead to an inappropriate model of normality being learned. A better strategy is to have a learning algorithm itself decide which set of features is the best. The reason for that is that learning algorithms find models with optimal generalization properties, i.e. the ones that are valid not only for observed data but also for the data to be dealt with in the future. The a priori choice of features may bias the learning process and lead to worse detection performance. By leaving this choice to the learning algorithm, the possibility of such bias is eliminated.

A natural way to address the kernel fusion problem is to learn a linear combination $K = \sum_{j=1}^{m} \theta_j K_j$ with mixing coefficients $\boldsymbol{\theta}$ together with model parameters, so as to maximize the generalization ability. To promote sparse solutions in terms of the linear kernel mixture, one frequently employs 1-norm simplex constraints on the mixing coefficients. This framework, known as multiple kernel learning (MKL), was first introduced for binary classification by [12]. Recently, efficient optimization strategies have been proposed for semi-infinite linear programming [25], second order approaches [3], and gradient-based optimization [20]. Other variants of two-class MKL have been proposed in subsequent work addressing practical algorithms for multi-class [19,32] and multi-label [8] problems.

We translate the multiple kernel learning framework to density level-set estimation to find a linear combination of features that realizes a minimal-volume description of the data. Furthermore, we generalize the MKL simplex constraint on the mixing coefficients to allow for arbitrary p-norms regularizations, where $p \geq 1$, hence leading to non-sparse kernel mixtures. Our approach also generalizes the one-class support vector machine [23] that is obtained as a special case for learning with only a single kernel. The optimization problem of our new method is efficiently solved by interleaved column generation and semi-infinite programming. Empirically, we evaluate our approach on network intrusion detection and object recognition tasks and compare its performance for different norms with unweighted-sum kernel mixtures. We observe our approach to attain higher predictive performances than baseline approaches.

The remainder of this paper is structured as follows. Section 2 briefly reviews the one-class support vector machine and presents our main contribution to density level-set estimation with multiple kernels. Section 3 reports on empirical results and Section 4 concludes.

2 Multiple Kernel Learning for Density Level-Sets

2.1 Density Level-Sets

In this paper, we focus on one-class classification problems. That is, we are given n data points $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$, where $\boldsymbol{x}_i$ lies in some input space $\mathcal{X}$. The goal is to find a model $f : \mathcal{X} \to \mathbb{R}$ and a density level-set $D_\rho = \{\boldsymbol{x} : f(\boldsymbol{x}) \geq \rho\}$ that generalizes well on new and unseen data such that the level-set encloses the normal data, i.e., $\boldsymbol{x} \in D_\rho$, while for outliers $\boldsymbol{x}' \notin D_\rho$ holds. A common approach is to employ linear models of the form

$$f(\boldsymbol{x}) = \boldsymbol{w}'\psi(\boldsymbol{x}) \tag{1}$$

together with a (possibly non-linear) feature mapping $\psi : \mathcal{X} \to \mathcal{H}$. A max-margin approach leads to the (primal) one-class SVM optimization problem [23] for $\nu \in]0, 1]$,

$$\begin{aligned} \min_{\boldsymbol{w},\rho,\boldsymbol{\xi}} \quad & \frac{1}{2}\boldsymbol{w}'\boldsymbol{w} + \frac{1}{\nu n}\|\boldsymbol{\xi}\|_1 - \rho \\ \text{s.t.} \quad & \forall i : \boldsymbol{w}'\psi(\boldsymbol{x}_i) \geq \rho - \xi_i, \quad \forall i : \xi_i \geq 0. \end{aligned} \tag{2}$$

Once optimal parameters $\boldsymbol{w}^*$ and ρ^* are found, these are plugged into Equation (1), and new instances $\tilde{\boldsymbol{x}}$ are classified according to $\text{sign}(f(\tilde{\boldsymbol{x}}) - \rho^*)$.

2.2 Density Level-Set Estimation with Multiple Kernels

When learning with multiple kernels, we are given m different feature mappings $\psi_1, \ldots, \psi_m$ in addition to the data points $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$. Every mapping $\psi_j : \mathcal{X} \to \mathcal{H}_j$ gives rise to a reproducing kernel k_j of $\mathcal{H}_j$ such that

$$k_j(\boldsymbol{x}, \tilde{\boldsymbol{x}}) = \langle \psi_j(\boldsymbol{x}), \psi_j(\tilde{\boldsymbol{x}}) \rangle_{\mathcal{H}_j}.$$

The goal of one-class multiple kernel learning is to find a linear combination $\sum_{j=1}^m \theta_j K_j$ of kernels and parameters $\boldsymbol{w}$, $\boldsymbol{\xi}$, and ρ simultaneously, such that the resulting hypothesis f leads to a minimum-volume description of the normal data. We incorporate the kernel mixture into the model in Equation (1) and arrive at

$$f(\boldsymbol{x}) \;=\; \sum_{k=1}^{m} \theta_j \, \boldsymbol{w}_j'\psi_j(\boldsymbol{x}) \;=\; \boldsymbol{w}_{\boldsymbol{\theta}}'\psi_{\boldsymbol{\theta}}(\boldsymbol{x}),$$

where the weight vector and the feature mapping have a block structure

$$\boldsymbol{w}_{\boldsymbol{\theta}} = (\sqrt{\theta_j}\boldsymbol{w}_j)_{j=1,\ldots,m}, \quad \psi_{\boldsymbol{\theta}}(\boldsymbol{x}_i) = (\sqrt{\theta_j}\psi_j(\boldsymbol{x}_i))_{j=1,\ldots,m}, \tag{3}$$

with mixing coefficients $\theta_j \geq 0$.

Incorporating (3) into (2) and imposing a general p-norm constraint $\|\boldsymbol{\theta}\|_p = 1$ for $p \geq 1$ on the mixing coefficients leads to the following primal optimization problem for $\nu \in]0, 1]$, and $p \geq 1$.

$$\min_{\boldsymbol{\theta}, \boldsymbol{w}, \rho, \boldsymbol{\xi}} \quad \frac{1}{2}\boldsymbol{w}_{\boldsymbol{\theta}}'\boldsymbol{w}_{\boldsymbol{\theta}} + \frac{1}{\nu n}\|\boldsymbol{\xi}\|_1 - \rho \tag{3a}$$

$$\text{s.t.} \quad \forall i: \ \boldsymbol{w}_{\boldsymbol{\theta}}'\psi_{\boldsymbol{\theta}}(\boldsymbol{x}_i) \geq \rho - \xi_i; \quad \boldsymbol{\xi} \geq \mathbf{0}\ ; \quad \boldsymbol{\theta} \geq \mathbf{0}\ ; \quad \|\boldsymbol{\theta}\|_p = 1. \tag{3b}$$

The above optimization problem is non-convex because (i) the products $\theta_j \boldsymbol{w}_j$ are non-convex which, however, can be easily removed by a change of variables $\boldsymbol{v}_j := \theta_j \boldsymbol{w}_j$ (e.g. see [2]), and (ii) the set $\{\boldsymbol{\theta} : \|\boldsymbol{\theta}\|_p = 1\}$ is not convex. As a remedy to (ii), we relax the constraint on $\boldsymbol{\theta}$ to become an inequality constraint, i.e., $\|\boldsymbol{\theta}\|_p \leq 1$. Treating the above optimization problem as interleaved minimization – over $\boldsymbol{\theta}$ *and* $\boldsymbol{w}$, $\boldsymbol{\xi}$, and ρ – it is easily verified that the optimal $\boldsymbol{\theta}^*$ in the $\boldsymbol{\theta}$-step always fulfills $\|\boldsymbol{\theta}^*\|_p = 1$ for all $p \geq 1$; essentially, we solve $\min_{\boldsymbol{\theta}} \sum_j c_j/\theta_j \ s.t. \ \|\boldsymbol{\theta}\|_p \leq 1$ which induces solutions $\boldsymbol{\theta}^*$ at the border $\|\boldsymbol{\theta}^*\|_p = 1$. We thus arrive at the following equivalent optimization problem, which now is convex.

$$\min_{\boldsymbol{\theta}, \boldsymbol{v}, \boldsymbol{\xi}, \rho} \quad \frac{1}{2}\sum_{j=1}^{m} \frac{\boldsymbol{v}_j'\boldsymbol{v}_j}{\theta_j} + \frac{1}{\nu n}\|\boldsymbol{\xi}\|_1 - \rho \tag{4a}$$

$$\text{s.t.} \quad \forall i: \ \sum_{j=1}^{m} \boldsymbol{v}_j'\psi_j(\boldsymbol{x}_i) \geq \rho - \xi_i; \quad \boldsymbol{\xi} \geq \mathbf{0}\ ; \quad \boldsymbol{\theta} \geq \mathbf{0}\ ; \quad \|\boldsymbol{\theta}\|_p \leq 1. \tag{4b}$$

Several previous algorithms for two-class multiple kernel learning utilized a two-step structure by alternating full SVM steps with $\boldsymbol{\theta}$ steps of different flavor [32,20,30]. In contrast, we follow [25] and propose to alternate $\boldsymbol{\theta}$ steps with *minor iterations* of SVM optimizers *without running them to completion.* We chose SVM$^{\text{light}}$ [10] as a basic solver, since its underlying chunking idea employs efficient $\boldsymbol{\alpha}$ minimization steps, making it well-suited for an interleaved $\boldsymbol{\alpha}, \boldsymbol{\theta}$ minimization. To solve the p-norm one-class MKL problem, we now devise a semi-infinite programming (SIP) approach similar to [25].

The underlying idea is to interleave the optimization of the upper bound on the objective of the SVM step and the $\boldsymbol{\theta}$ step. Fixing $\boldsymbol{\theta} \in \Theta$, where $\Theta = \{\boldsymbol{\theta} \in \mathbb{R}^n \mid \boldsymbol{\theta} \geq \mathbf{0}\ , \ \ \|\boldsymbol{\theta}\|_p \leq 1\}$, we build the partial Lagrangian with respect to $\boldsymbol{v}$, $\boldsymbol{\xi}$, and ρ by introducing componentwise non-negative Lagrange multipliers $\boldsymbol{\alpha}, \boldsymbol{\gamma} \in \mathbb{R}^n, \delta \in \mathbb{R}$. The partial Lagrangian is given by

$$L = \frac{1}{2}\sum_{j=1}^{m} \frac{\boldsymbol{v}_j'\boldsymbol{v}_j}{\theta_j} + \frac{1}{\nu n}\sum_{i=1}^{n} \xi_i - \sum_{i=1}^{n} \gamma_i\xi_i - \sum_{i=1}^{n} \alpha_i\Big(\sum_{j=1}^{m} \boldsymbol{v}_j'\psi_j(\boldsymbol{x}_i) - \rho + \xi_i\Big) - \delta\rho.$$

Setting the partial derivatives with respect to the primal variables to zero yields the relations $0 \leq \alpha_i \leq \frac{1}{\nu n}$, $\sum_i \alpha_i = 1$, and $\boldsymbol{v}_j = \sum_i \alpha_i\theta_j\psi_j(\boldsymbol{x}_i)$ for $1 \leq i \leq n$ and $1 \leq j \leq p$. The KKT conditions trivially hold and re-substitution into the Lagrangian gives rise to the min-max formulation for $\nu \in]0, 1]$ and $p \geq 1$,

$$\min_{\boldsymbol{\theta}} \max_{\boldsymbol{\alpha}} \quad -\frac{1}{2}\sum_{i,l=1}^{n}\alpha_i\alpha_l\sum_{j=1}^{m}\theta_j k_j(\boldsymbol{x}_i,\boldsymbol{x}_l) \tag{5a}$$

$$\text{s.t.} \quad \mathbf{0}\le\boldsymbol{\alpha}\le\frac{1}{\nu n}\mathbf{1}; \quad \mathbf{1}'\boldsymbol{\alpha}=1; \quad \boldsymbol{\theta}\ge\mathbf{0}; \quad \|\boldsymbol{\theta}\|_p\le 1. \tag{5b}$$

The above optimization problem can be solved directly by gradient-based techniques exploiting the smoothness of the objective [1]. Alternatively, we can translate it into an equivalent semi-infinite program (SIP) as follows. Suppose $\boldsymbol{\alpha}^*$ is optimal, then denoting the value of the target function by $t(\boldsymbol{\alpha},\boldsymbol{\theta})$, we have $t(\boldsymbol{\alpha}^*,\boldsymbol{\theta})\ge t(\boldsymbol{\alpha},\boldsymbol{\theta})$ for all $\boldsymbol{\alpha}$ and $\boldsymbol{\theta}$. Hence we can equivalently minimize an upper bound λ on the optimal value. We thus arrive at the following optimization problem,

$$\min_{\lambda,\boldsymbol{\theta}} \quad \lambda \quad \text{s.t.} \quad \lambda\ge-\frac{1}{2}\boldsymbol{\alpha}'\sum_{j=1}^{m}\theta_j K_j\boldsymbol{\alpha} \tag{6}$$

for all $\boldsymbol{\alpha}\in\mathbb{R}^n$ with $\mathbf{0}\le\boldsymbol{\alpha}\le\frac{1}{\nu n}\mathbf{1}$, $\mathbf{1}'\boldsymbol{\alpha}=1$, and $\boldsymbol{\alpha}\ge\mathbf{0}$, as well as $\|\boldsymbol{\theta}\|_p\le 1$ and $\boldsymbol{\theta}\ge\mathbf{0}$. The optimization problem in Equation (6) generalizes the idea of [25] to the case $p\ge 1$. Analogously, it can be optimized with interleaving cutting plane algorithms, that is, the solution of a quadratic program (here a one-class SVM) generates the most strongly violated constraint for the actual mixture $\boldsymbol{\theta}$. The optimal $(\boldsymbol{\theta}^*,\lambda)$ however depends on the value of p. We differentiate between two cases, $p=1$ and $p>1$.

Optimizing $\boldsymbol{\theta}$ for $\boldsymbol{p=1}$: for $p=1$ is then identified by solving a linear program with respect to set of active constraints.

Optimizing $\boldsymbol{\theta}$ for $\boldsymbol{p>1}$: For the general case $p>1$, a non-linearity is introduced by requiring $\|\boldsymbol{\theta}\|_p\le 1$. Such constraint is rather uncommon in standard optimization toolboxes that often handle only linear and quadratic constraints. As a remedy we propose to solve a sequence of quadratically constrained subproblems. To this end, we substitute the p-norm constraint by sequential second-order Taylor approximations of the form

$$\begin{aligned}||\boldsymbol{\theta}||_p^p &\approx 1+p(\boldsymbol{\theta}_k^{p-1})'(\boldsymbol{\theta}-\boldsymbol{\theta}^{old})\\ &\quad+\frac{p(p-1)}{2}(\boldsymbol{\theta}-\boldsymbol{\theta}^{old})'\mathrm{diag}((\boldsymbol{\theta}^{old})^{p-2})(\boldsymbol{\theta}-\boldsymbol{\theta}^{old})\\ &=1-\frac{p(3-p)}{2}-\sum_j p(p-2)(\theta_j^{old})^{p-1}\,\theta_j\\ &\quad+\frac{p(p-1)}{2}\sum_j(\theta_j^{old})^{p-2}\,\theta_j^2,\end{aligned}$$

where $\boldsymbol{\theta}^p$ is defined element-wise, that is $\boldsymbol{\theta}^p:=(\theta_1^p,...,\theta_m^p)$. We use $\boldsymbol{\theta}^{old}=\sqrt[p]{\frac{1}{m}}\mathbf{1}$ as a starting point. Note that the quadratic term in the approximation is diagonal. As a result the quadratically constrained problem can be solved very efficiently. For

Algorithm 1. p-Norm MKL chunking-based training algorithm. It simultaneously optimizes $\boldsymbol{\alpha}$ and the kernel weighting $\boldsymbol{\theta}$. The accuracy parameter ϵ and the subproblem size Q are assumed to be given to the algorithm. For simplicity, a few speed-up tricks are not shown: the removal of inactive constraints and hot-starts.

1: $g_{j,i} = 0$, $\hat{g}_i = 0$, $\alpha_i = 0$, $\theta_j = \sqrt[p]{1/m}$ for $j = 1, \ldots, m$ and $i = 1, \ldots, n$
2: **for** $t = 1, 2, \ldots$ and while SVM and MKL optimality conditions are not satisfied **do**
3: Select Q suboptimal variables $\alpha_{i_1}, \ldots, \alpha_{i_Q}$ based on the gradient $\hat{\mathbf{g}}$ and $\boldsymbol{\alpha}$; store $\boldsymbol{\alpha}^{old} = \boldsymbol{\alpha}$
4: Solve SVM dual with respect to the selected variables and update $\boldsymbol{\alpha}$
5: Update gradient $g_{j,i} \leftarrow g_{j,i} + \sum_{q=1}^{Q} (\alpha_{i_q} - \alpha_{i_q}^{old}) k_j(\boldsymbol{x}_{i_q}, \boldsymbol{x}_i)$ for all $j = 1, \ldots, m$ and $i = 1, \ldots, n$
6: **for** $j = 1, \ldots, m$ **do**
7: $S_j^t = \frac{1}{2} \sum_i g_{j,i} \alpha_i$
8: **end for**
9: $S^t = \sum_j \theta_j S_j^t$
10: **if** $|1 - \frac{S^t}{\lambda}| \geq \epsilon$
11: **for** $k = 1, 2, \ldots$ and while MKL optimality conditions are not satisfied **do**
12: $\boldsymbol{\theta}^{old} = \boldsymbol{\theta}$
13: $(\boldsymbol{\theta}, \lambda) \leftarrow$ argmax λ
14: w.r.t. $\boldsymbol{\theta} \in \mathbb{R}^m, \lambda \in \mathbb{R}$
15: s.t. $\mathbf{0} \leq \boldsymbol{\theta} \leq \mathbf{1}$, $\sum_j \theta_j S_j^r \geq \lambda$ for $r = 1, \ldots, t$
16: $\frac{p(p-1)}{2} \sum_j (\theta_j^{old})^{p-2} \theta_j^2 - \sum_j p(p-2)(\theta_j^{old})^{p-1} \theta_j \leq \frac{p(3-p)}{2}$
17: $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} / ||\boldsymbol{\theta}||_p$
18: **end for**
19: **end if**
20: $\hat{g}_i = \sum_j \theta_j g_{j,i}$ for all $i = 1, \ldots, n$
21: **end for**

the special case $p = 2$, the Taylor approximation is tight and hence the sequence of quadratically constrained sub-problems converges after one iteration.

Optimization Algorithm. Algorithm 1 outlines the interleaved $\boldsymbol{\alpha}, \boldsymbol{\theta}$ MKL training algorithm. Lines 3-5 are standard in chunking based SVM solvers and carried out by SVM$^{\text{light}}$. Lines 6-9 compute (parts of) SVM-objective values for each kernel independently. Finally lines 11 to 18 solve a sequence of semi-infinite programs with the p-norm constraint being approximated as a sequence of second-order constraints. The algorithm terminates if the maximum KKT violation (see [10]) falls below a predetermined precision ε_{svm} and for MKL if the normalized maximal constraint violation $|1 - \frac{S^t}{\lambda}| < \varepsilon_{mkl}$.

3 Empirical Results

In this section we study p-norm multiple kernel learning for density level-sets in terms of efficiency and accuracy. We experiment on network intrusion detection

and object recognition tasks and compare our approach to baseline one-class SVMs with unweighted-sum kernels $K = \sum_{j=1}^{m} K_j$ wich we refer to as ∞-norm MKL. We choose this baseline because for two-class multiple kernel learning approaches, unweighted-sum kernel mixtures have frequently been observed to outperform sparse kernel mixtures in practical applications.

3.1 Network Intrusion Detection

For the intrusion detection experiments we use HTTP traffic recorded at Fraunhofer Institute FIRST Berlin. The unsanitized data contains 2500 normal HTTP requests drawn randomly from incoming traffic recorded over two months. Malicious traffic is generated using the Metasploit framework [18]. We generate 30 instances of 10 real attack classes from recent exploits, including buffer overflows and PHP vulnerabilities. Every attack is recorded in different variants using virtual network environments and decoy HTTP servers.

The malicious data are normalized to match frequent attributes of the normal HTTP requests such that the payload provides the only indicator for separating normal from attack data. We deploy 10 spectrum kernels [14,24] for $1, 2, \ldots, 10$-gram feature representations. All kernels are normalized according to Equation (7) to avoid dependencies on the HTTP request length.

$$K(\boldsymbol{x}, \tilde{\boldsymbol{x}}) \longmapsto \frac{K(\boldsymbol{x}, \tilde{\boldsymbol{x}})}{\sqrt{K(\boldsymbol{x}, \boldsymbol{x}) K(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{x}})}}, \tag{7}$$

We randomly split the normal data into 1000 training, 500 validation and 1000 test examples. The training partition is used as it is since centroid-based learners assume uncorrupted training data. The validation and test partitions are mixed with 15 attack instances that are randomly chosen from the malicious pool. We make sure that attacks of the same class occur either in the holdout or in the test data but not in both, hence reflecting the goal of anomaly detection to recognize *previously unknown* attacks. We report on average areas under the ROC curve in the false-positive interval $[0, 0.01]$ ($\text{AUC}_{[0,0.01]}$) over 100 repetitions with distinct training, holdout, and test sets.

Table 1 shows the results for one-class multiple kernel learning with $p \in \{\infty, 1, \frac{4}{3}, 2, 4\}$. Depending on the actual value of p, the performances are quite different. The unweighted-sum kernel (∞-norm MKL) outperforms most of the one-class MKL approaches. However, employing a 2-norm constraint on the mixing coefficients leads to better results than the ∞-norm mixture. Notice that the 2-norm mixture is about 10% better than its sparse 1-norm counterpart.

Figure 1 reports on the optimal kernel mixture coefficients $\boldsymbol{\theta}$ for $p \in \{1, \frac{4}{3}, 2, 4\}$-norm MKL and the unweighted-sum kernel. The sparse 1-norm solution places all the weight into 1-grams that – although leading to concise representations because of the low dimensional feature space – result in inappropriate performances (see Table 1). The higher the value of p, the less weight is placed on the 1-gram kernel but spread across higher n-gram kernels. The 4-norm mixture is similar to the trivial ∞-norm solution. The best solution (2-norm) still places weight to 1-grams but incorporates all other n-gram kernels to some extend.

Table 1. Results for intrusion detection

MKL	$\text{AUC}_{0.01}$
∞-norm	89.4 ± 0.7
1-norm	79.4 ± 0.9
$\frac{4}{3}$-norm	85.7 ± 0.8
2-norm	90.7 ± 0.8
4-norm	88.9 ± 0.9

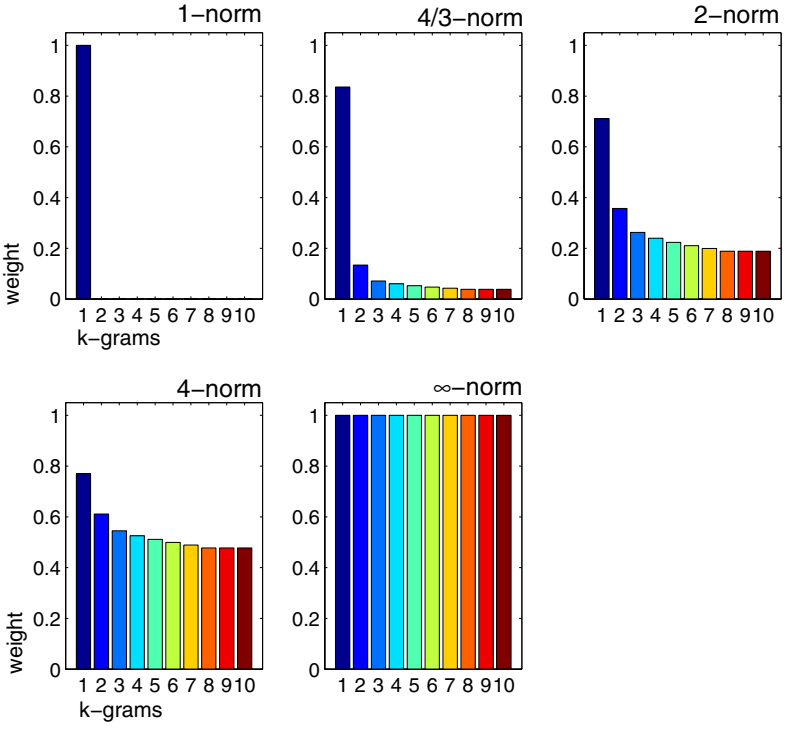

Fig. 1. Mixing coefficients for the intrusion detection task

3.2 Multi-label Image Categorization

Besides anomaly and outlier detection, one-class learning techniques are frequently applied to multi-class classification problems with temporally varying numbers of categories such as event detection and object recognition tasks. Their advantage lies in training a single model for every (new) category in contrast to maintaining expensive multi-class classifiers that have to be re-trained once a new category is included in the task.

To study one-class multiple kernel learning in this alternative scenario, we apply our approach to the multi-label classification task of the VOC 2008 challenge [7]. The data set contains 8780 images, divided into 2113 training, 2227 validation, and 4340 test images. Images are annotated with a subset of 20 class

labels such as *aeroplane*, *bicycle*, and *bird*. Since the ground-truth of the test set is not yet disclosed by the challenge organizers, we focus on the training and validation splits. From these two original sets, we draw 2111 training, 1111 validation, and 1110 test images at random and report on average precisions (AP) for all recall values over 10 runs with distinct training, holdout, and test sets.

We employ two sets of kernels inspired from the VOC 2007 winner (K12) [17] and the VOC 2008 winner (K30) [26]. For both approaches, all basic features are combined with the respective pyramid levels and translated into a χ^2 kernel [31], where the widths of the χ^2 kernels are chosen according to a heuristic [11]. The sets of kernels are obtained as follows.

K12. We extract 12 kernels based on four basic features: histograms of visual words [5] in the grey (HOW-G) and in the hue color channel (HOW-H), histogram of oriented gradient (HOG) [6], and histograms of the hue color channel (HOCOL) [17]. These representations are combined with a pyramidal representation of level 2 to capture spatial dependencies, i.e., each image is tiled into 1, 4, and 16 parts.

K30. We extract 30 kernels based on histograms of visual words with 2 different sampling methods (dense and interest points), 5 different sets of colors (grey, opponent color, normalized opponent color, normalized RG, and RGB) [27] and 3 different tilings (level-0 and level-1 of the pyramid, and 1×3 tiling) [26].

We compare the performance of the unweighted-sum kernel ∞, and 1- and 2-norm MKL with the optimal p-norm MKL that maximizes the average precision on the validation set for each class. For the latter approach, model selection is not only performed for trade-off parameter ν but extended to the MKL norm p. Table 2 shows the mean average precisions over 20 categories for the test data. Bold faces indicate significant results, that is, the best method and ones that are not comparably different from the best result according to a Wilcoxon signed-ranks test using a 5% confidence-level.

For the K12 set of kernels, 1-norm MKL outperforms both, the unweighted-sum kernel ∞-norm and a non-sparse 2-norm MKL, which perform equally well. However, model selection over p for each class leads to comparable results as 1-norm MKL. We do not display the optimal p^* values for all 20 classes, however, the respective mixtures are non-sparse (see also Figure 2) so that the sparse 1-norm approach denotes the best solution for K12 in terms of accuracy and interpretability.

For the K30 set of kernels, the outcome is different. Here, the 1-norm MKL performs significantly worse compared to its non-sparse counterparts. Although model selection over p leads to the highest average precisions, the results are not significantly different to 2-norm MKL and unweighted-sum kernel mixtures. Our experiments show that the right choice of the value p depends highly on the employed kernels. Vice versa, once a set of kernels is fixed, it is necessary to include the norm parameter p in the model selection to find the best kernel mixture.

Table 2. Results for the VOC 2008 data set

	1-norm	p^*-norm	2-norm	∞-norm
mean AP (K12)	**17.6±0.8**	**17.8±1.0**	17.1±0.8	17.0±0.6
mean AP (K30)	16.3±0.5	**17.1±0.9**	**17.1±0.6**	**17.0±0.7**

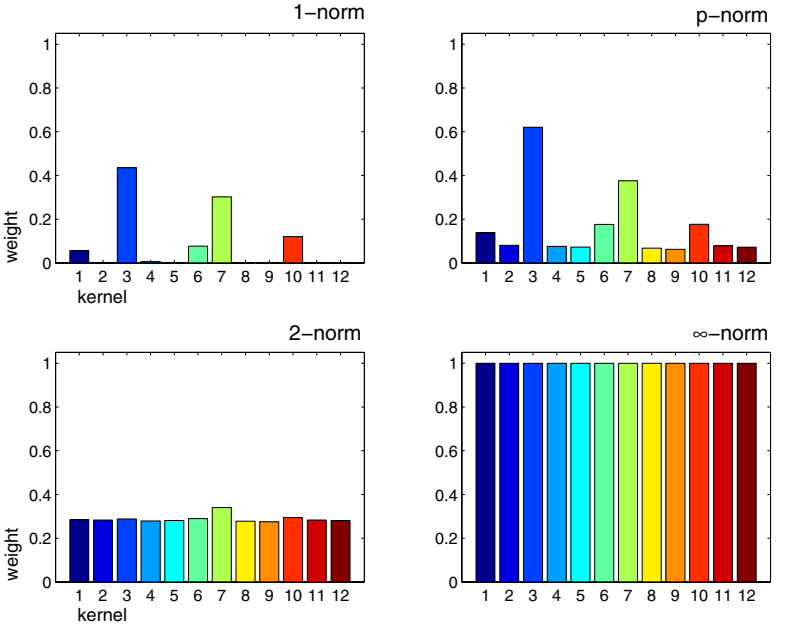

Fig. 2. Mixing coefficients for the multi-label image categorization experiment

Figure 2 shows the optimal mixing coefficients for the K12 task, averaged over 10 repetitions. The 1-norm solution picks a sparse combination resulting in a minimum volume description of the data. While a 2-norm solution distributes the weights almost uniformly on the 12 kernels, the p-norm solution lies in between and considers all kernels with non-zero mixing coefficients in the solution.

3.3 Execution Time

We show the efficiency of one-class MKL and compare the execution times for our approach with $p \in \{1, 1.333, 2, 3, 4, \infty\}$ to one-class SVMs using the unweighted sum-kernel as implemented in [10]. To show different aspects of our approach, we draw a sample of size n from a 10-dimensional Gaussian distribution for various values of n. Kernel matrices are computed using RBF-kernels with different bandwidth parameters. We optimize the duality gap for all methods up to a precision of 10^{-3}.

Figure 3 (left) displays the results for varying sample sizes in a log-log plot; errorbars indicate standard error over 5 repetitions. Unsurprisingly, the baseline one-class SVM using the sum-kernel is the fastest method. The execution time of

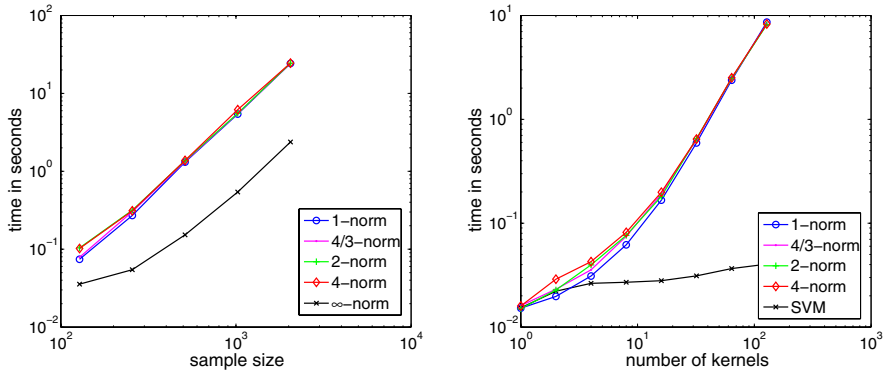

Fig. 3. Execution times for one-class MKL. Left: results for varying sample sizes. Right: execution times for varying numbers of kernels.

non-sparse MKL depends on the value p. We observe longer computation times for large values of p. However, all approaches scale similarly.

Figure 3 (right) shows execution times for varying numbers of kernels and fixed sample size $n = 100$. Again, the baseline one-class SVM with the unweighted-sum kernel is the fastest method. All one-class MKL approaches show reasonable run-times and converge quickly for 128 kernels.

4 Conclusion

We presented an efficient and accurate approach to multiple kernel learning for density level-set estimation. Our approach generalizes the standard setting of multiple kernel learning by allowing for arbitrary norms for the kernel mixture. This enabled us to study sparse and non-sparse kernel mixtures. Our method contains the one-class SVM as a special case for training with only a single kernel. Our optimization strategy is based on interleaved semi-infinite programming and chunking based SVM training. Empirical results proved the efficiency and accuracy of our methods compared to baseline approaches. We observed one-class MKL to be robust in situations where unweighted-sum kernels are prone to fail.

Acknowledgments

The authors wish to thank Sören Sonnenburg, Alexander Zien, and Pavel Laskov for fruitful discussions and helpful comments. Furthermore we thank Patrick Düssel and Christian Gehl for providing the network traffic and Alexander Binder, Christina Müller, Motoaki Kawanabe, and Wojciech Wojcikiewicz for sharing kernel matrices for the VOC data with us. This work was supported in

part by the German Bundesministerium für Bildung und Forschung (BMBF) under the project REMIND (FKZ 01-IS07007A) and by the FP7-ICT Programme of the European Community, under the PASCAL2 Network of Excellence, ICT-216886.

References

1. Bach, F.R., Lanckriet, G.R.G., Jordan, M.I.: Multiple kernel learning, conic duality, and the SMO algorithm. In: Proceedings of the Twenty-first International Conference on Machine Learning (2004)
2. Boyd, S., Vandenberghe, L.: Convex Optimization. Cambrigde University Press, Cambridge (2004)
3. Chapelle, O., Rakotomamonjy, A.: Second order optimization of kernel parameters. In: Proceedings of the NIPS Workshop on Kernel Learning: Automatic Selection of Optimal Kernels (2008)
4. Chhabra, P., Scott, C., Kolaczyk, E.D., Crovella, M.: Distributed spatial anomaly detection. In: Proceedings of the IEEE Infocom 2008 (2008)
5. Csurka, G., Bray, C., Dance, C., Fan, L.: Visual categorization with bags of keypoints. In: Workshop on Statistical Learning in Computer Vision, ECCV, Prague, Czech Republic, May 2004, pp. 1–22 (2004)
6. Dalal, N., Triggs, B.: Histograms of oriented gradientsfor human detection. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Diego, USA, June 2005, vol. 1, pp. 886–893 (2005)
7. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: Proceedings of the the PASCAL Visual Object Classes Challenge 2008, VOC 2008 (2008)
8. Ji, S., Sun, L., Jin, R., Ye, J.: Multi-label multiple kernel learning. In: Advances in Neural Information Processing Systems (2009)
9. Jiang, Z., Luosheng, W., Yong, F., Xiao, Y.C.: Intrusion detection based on density level sets estimation. In: NAS 2008: Proceedings of the 2008 International Conference on Networking, Architecture, and Storage (2008)
10. Joachims, T.: Making large–scale SVM learning practical. In: Schölkopf, B., Burges, C.J.C., Smola, A.J. (eds.) Advances in Kernel Methods — Support Vector Learning, pp. 169–184. MIT Press, Cambridge (1999)
11. Lampert, C.H., Blaschko, M.B.: A multiple kernel learning approach to joint multiclass object detection. In: Rigoll, G. (ed.) DAGM 2008. LNCS, vol. 5096, pp. 31–40. Springer, Heidelberg (2008)
12. Lanckriet, G., Cristianini, N., Ghaoui, L.E., Bartlett, P., Jordan, M.I.: Learning the kernel matrix with semi-definite programming. Journal of Machine Learning Research 5, 27–72 (2004)
13. Lee, W., Stolfo, S.J.: A framework for constructing features and models for intrusion detection systems. ACM Transactions on Information Systems Security 3, 227–261 (2000)
14. Leslie, C., Eskin, E., Noble, W.S.: The spectrum kernel: A string kernel for SVM protein classification. In: Proc. Pacific Symp. Biocomputing, pp. 564–575 (2002)
15. Mahoney, M.V., Chan, P.K.: Learning nonstationary models of normal network traffic for detecting novel attacks. In: Proc. of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), pp. 376–385 (2002)

16. Mahoney, M.V., Chan, P.K.: Learning rules for anomaly detection of hostile network traffic. In: Proc. of International Conference on Data Mining (ICDM) (2003)
17. Marszalek, M., Schmid, C.: Learning representations for visual object class recognition. In: Proceedings of the PASCAL Visual Object Classes Challenge 2007, VOC 2007 (2007)
18. Maynor, K., Mookhey, K., Cervini, J.F.R., Beaver, K.: Metasploit toolkit. Syngress (2007)
19. Rakotomamonjy, A., Bach, F., Canu, S., Grandvalet, Y.: More efficiency in multiple kernel learning. In: ICML, pp. 775–782 (2007)
20. Rakotomamonjy, A., Bach, F., Canu, S., Grandvalet, Y.: SimpleMKL. Journal of Machine Learning Research 9, 2491–2521 (2008)
21. Rieck, K., Laskov, P.: Detecting unknown network attacks using language models. In: Büschkes, R., Laskov, P. (eds.) DIMVA 2006. LNCS, vol. 4064, pp. 74–90. Springer, Heidelberg (2006)
22. Rieck, K., Laskov, P.: Language models for detection of unknown attacks in network traffic. Journal in Computer Virology 2(4), 243–256 (2007)
23. Schölkopf, B., Platt, J., Shawe-Taylor, J., Smola, A.J., Williamson, R.C.: Estimating the support of a high-dimensional distribution. Neural Computation 13(7), 1443–1471 (2001)
24. Shawe-Taylor, J., Cristianini, N.: Kernel methods for pattern analysis. Cambridge University Press, Cambridge (2004)
25. Sonnenburg, S., Rätsch, G., Schäfer, C., Schölkopf, B.: Large Scale Multiple Kernel Learning. Journal of Machine Learning Research 7, 1531–1565 (2006)
26. Tahir, M., van de Sande, K., Uijlings, J., Yan, F., Li, X., Mikolajczyk, K., Kittler, J., Gevers, T., Smeulders, A.: Surreyuva srkda method. In: Proceedings of the PASCAL Visual Object Classes Challenge 2008, VOC 2008 (2008)
27. van de Sande, K.E.A., Gevers, T., Snoek, C.G.M.: Evaluation of color descriptors for object and scene recognition. In: IEEE Conference on Computer Vision and Pattern Recognition (2008)
28. Wang, K., Parekh, J.J., Stolfo, S.J.: Anagram: A content anomaly detector resistant to mimicry attack. In: Zamboni, D., Krügel, C. (eds.) RAID 2006. LNCS, vol. 4219, pp. 226–248. Springer, Heidelberg (2006)
29. Wang, K., Stolfo, S.J.: Anomalous payload-based network intrusion detection. In: Jonsson, E., Valdes, A., Almgren, M. (eds.) RAID 2004. LNCS, vol. 3224, pp. 203–222. Springer, Heidelberg (2004)
30. Xu, Z., Jin, R., King, I., Lyu, M.R.: An extended level method for efficient multiple kernel learning. In: Advances in Neural Information Processing Systems (2009)
31. Zhang, J., Marszalek, M., Lazebnik, S., Schmid, C.: Local features and kernels for classification of texture and object categories: A comprehensive study. International Journal of Computer Vision 73(2), 213–238 (2007)
32. Zien, A., Ong, C.S.: Multiclass multiple kernel learning. In: Ghahramani, Z. (ed.) ICML. ACM International Conference Proceeding Series, vol. 227, pp. 1191–1198. ACM, New York (2007)