Optimal Learning Rates for Localized SVMs

Mona Eberts
Ingo Steinwart

Institute for Stochastics and Applications
University of Stuttgart
70569 Stuttgart, Germany

Abstract

One of the limiting factors of using support vector machines (SVMs) in large scale applications are their super-linear computational requirements in terms of the number of training samples. To address this issue, several approaches that train SVMs on many small chunks of large data sets separately have been proposed in the literature. So far, however, almost all these approaches have only been empirically investigated. In addition, their motivation was always based on computational requirements. In this work, we consider a localized SVM approach based upon a partition of the input space. For this local SVM, we derive a general oracle inequality. Then we apply this oracle inequality to least squares regression using Gaussian kernels and deduce local learning rates that are essentially minimax optimal under some standard smoothness assumptions on the regression function. This gives the first motivation for using local SVMs that is not based on computational requirements but on theoretical predictions on the generalization performance. We further introduce a data-dependent parameter selection method for our local SVM approach and show that this method achieves the same learning rates as before. Finally, we present some larger scale experiments for our localized SVM showing that it achieves essentially the same test performance as a global SVM for a fraction of the computational requirements. In addition, it turns out that the computational requirements for the local SVMs are similar to those of a vanilla random chunk approach, while the achieved test errors are significantly better.

Keywords: least squares regression, support vector machines, localization

1. Introduction

Based on a training set \( D := \((x_1, y_1), \ldots, (x_n, y_n)\) of i.i.d. input/output observations drawn from an unknown distribution \( P \) on \( X \times Y \), where \( X \subset \mathbb{R}^d \) and \( Y \subset \mathbb{R} \), the goal of non-parametric regression is to find a function \( f_D : X \rightarrow \mathbb{R} \) such that important characteristics of the conditional distribution \( P(Y|x), x \in X \), can be recovered. For instance, an \( f_D \) approximating the conditional mean \( \mathbb{E}(Y|x), x \in X \), is sought in the non-parametric least squares regression. This classical non-parametric regression problem has been extensively studied in the literature, where a general reference is the book (Győrfi et al., 2002), presenting plenty of results concerning the non-parametric least squares regression.

In the literature, there are many learning methods that solve the non-parametric regression problems, some of them are e.g. described in (Győrfi et al., 2002; Koenker, 2005; Simonoff, 1996). In this paper, we utilize some kernel-based regularized empirical risk minimizers, also known as support vector machines (SVMs), which solve the regularized
problem

\[ f_{D, \lambda} \in \arg \min_{f \in H} \lambda \| f \|_H^2 + \mathcal{R}_{L,D}(f). \]  

(1)

Here, \( \lambda > 0 \) is a fixed real number and \( H \) is a reproducing kernel Hilbert space (RKHS) over \( X \) with reproducing kernel \( k : X \times X \to \mathbb{R} \), see e.g. (Aronszajn, 1950; Berlinet and Thomas-Agnan, 2004; Steinwart and Christmann, 2008a). Besides, \( \mathcal{R}_{L,D}(f) \) denotes the empirical risk of a function \( f : X \to \mathbb{R} \), that is

\[ \mathcal{R}_{L,D}(f) = \frac{1}{n} \sum_{i=1}^{n} L(x_i, y_i, f(x_i)), \]

where \( D \) is the empirical measure associated to the data \( D \) defined by \( D := \frac{1}{n} \sum_{i=1}^{n} \delta_{(x_i, y_i)} \) with Dirac measure \( \delta_{(x, y)} \) at \( (x, y) \). Note that the empirical SVM solution \( f_{D, \lambda} \) exists and is unique (cf. Steinwart and Christmann [2008a], Theorem 5.5) whenever the loss \( L \) is convex in its last argument. Moreover, an SVM is \( L \)-risk consistent under a few assumptions on the RKHS \( H \) and the regularization parameter \( \lambda \), see (Steinwart and Christmann, 2008a, Section 6.4) for more details. Besides, it is worth mentioning that the ability to choose the RKHS \( H \) as well as the loss function \( L \) in (1) provides the possibility to flexibly apply SVMs to various learning problems. Namely, the learning target is modeled by the loss function, e.g. the least squares loss is used to estimate the conditional mean. Moreover, since RKHSs are defined on arbitrary \( X \), data types that are not \( \mathbb{R}^d \)-valued can be handled, too. Furthermore, SVMs are enjoying great popularity, since they can be implemented and applied in a relatively simple way and only have a few free parameters that can usually be determined by cross validation.

An essential theoretical task, which has attracted many considerations, is the investigation of learning rates for SVMs. For example, such rates for SVMs using the least squares loss and generic kernels can be found in (Cucker and Smale, 2002; De Vito et al., 2005; Smale and Zhou, 2007; Caponnetto and De Vito, 2007; Mendelson and Neeman, 2010; Steinwart et al., 2009) and the references therein, while similar rates for SVMs using the pinball loss can be found in (Steinwart and Christmann, 2008a, 2011). At this point, we do not want to take a closer look at these results, instead we relegate to (Eberts and Steinwart, 2013), where a detailed discussion can be found. More important for our purposes is the fact that Eberts and Steinwart (2011, 2013) establish (essentially) asymptotically optimal learning rates for least squares SVMs (LS-SVMs) using Gaussian RBF kernels. More precisely, for a domain \( X \subset B_{\mu \delta}, Y := [-M, M] \) with \( M > 0 \), a distribution \( P \) on \( X \times Y \) such that \( P_X \) has a bounded Lebesgue density on \( X \), and for \( f^* \) contained in the Sobolev space \( W^2_2(P_X), \alpha \in \mathbb{N}, \) or in the Besov space \( B^2_{2,\infty}(P_X), \alpha \geq 1, \) respectively, the LS-SVM using Gaussian kernels learns for all \( \xi > 0 \) with rate \( n^{-\frac{2\alpha}{2\alpha + d} + \xi} \) with a high probability. Although these rates are essentially asymptotically optimal, they depend on the order of smoothness of the regression function on the entire input space \( X \). That is, if the regression function \( f^* \) is on some area of \( X \) smoother than on another area, the learning rate is determined by the part of \( X \), where the regression function \( f^* \) is least smooth (cf. Figure 1).
In contrast to this, it would be desirable to achieve a learning rate on every region of $X$ that corresponds with the order of smoothness of $f^*$ on this region. Therefore, one of our goals of this paper is to modify the standard SVM approach such that we achieve local learning rates that are asymptotically optimal. Our technique to achieve such local learning rates is a special local SVM approach. Local SVMs have been extensively investigated in the literature to speed-up the training time, see for instance, the early works \cite{BottouVapnik1992, VapnikBottou1993}. The basic idea of many local approaches is to a) split the training data and just consider a few examples near a testing sample, b) train on this small subset of the training data, and c) use the solution for a prediction w.r.t. the test sample. Here, many up-to-date investigations use SVMs to train on the local data set but, yet there are different ways to split the whole training data set into smaller, local sets. For example, \cite{Chang2010, Wu1999, BennettBlue1998} use decision trees while in \cite{Hable2013, SegataBlanzieri2010, SegataBlanzieri2008, BlanzieriBryl2007a, Zhang2006} local subsets are built considering $k$ nearest neighbors. The latter approaches further vary, for example, \cite{Zhang2006, BlanzieriBryl2007a, Hable2013} consider different metrics w.r.t. the input space whereas \cite{SegataBlanzieri2008, BlanzieriMelgani2008, BlanzieriBryl2007a} consider metrics w.r.t. the feature space. Nonetheless, the basic idea of all these articles is that an SVM problem based on $k$ training samples is solved for each test sample. Another approach using $k$ nearest neighbors is investigated in \cite{SegataBlanzieri2010}. Here, $k$-neighborhoods consisting of training samples and collectively covering the training data set are constructed and an SVM is calculated on each neighborhood. The prediction for a test sample is then made according to the nearest training sample that is a center of a $k$-neighborhood. As for the other nearest neighbor approaches, however, the results are mainly experimental. An exception to this rule is \cite{Hable2013}, where universal consistency for localized versions of SVMs, or more precisely, a large class of regularized kernel methods, is proven. Another article presenting theoretical results for localized versions of learning methods is \cite{ZakaiRitov2009}. Here, the authors show that a consistent learning method behaves locally, i.e. the prediction is essentially influenced by close by samples. However, this result is based on a localization technique considering only training samples contained in a neighborhood with a fixed radius and center $x$ when an estimate in $x$ is sought. Probably closest to our approach is the one examined in \cite{Cheng2010} and \cite{Cheng2007}, where the training data is splitted into clusters and then an SVM is trained on each cluster. However, the presented results are only of experimental character.

In this article, we partition the input space $X$ according to a cover of $X$ with radius $r_n$ and build an SVM model for each partition cell. The following section is dedicated to the detailed description of this method. Section 3 then presents some theoretical results that
enable the analysis of this new method. For example, we examine extensions and direct sums of RKHSs. At the end of Section 3, we finally present a first oracle inequality for the localized SVM. In Section 4, we focus on RKHSs using Gaussian RBF kernels and, in conjunction with that, we study some entropy estimates. After that, Section 5 concentrates on the least squares loss and introduces an oracle inequality and learning rates for our localized SVM method using Gaussian kernels. Moreover, a data-dependent parameter selection method is studied that induces the same rates. Section 6 then presents some experimental results w.r.t. the localized SVM technique. All proofs can be found in Section 7, and the appendix contains various tables displaying detailed results of our experiments.

2. Description of the Localized SVM Approach

In this section, we introduce some general notations and assumptions. Based on the latter we modify the standard SVM approach. Let us start with the probability measure $P$ on $X \times Y$, where $X \subset \mathbb{R}^d$ is non-empty and $Y := [-M, M]$ for some $M > 0$. Depending on the learning target one chooses a loss function $L$, i.e. a function $L : X \times Y \times \mathbb{R} \to [0, \infty)$ that is measurable. Then, for a measurable function $f : X \to \mathbb{R}$, the $L$-risk is defined by

$$R_{L, P}(f) = \int_{X \times Y} L(x, y, f(x)) \, dP(x, y)$$

and the optimal $L$-risk, called the Bayes risk with respect to $P$ and $L$, is given by

$$R_{\ast L, P} := \inf \left\{ R_{L, P}(f) \mid f : X \to \mathbb{R} \text{ measurable} \right\}.$$ 

A measurable function $f_{\ast L, P} : X \to \mathbb{R}$ with $R_{L, P}(f_{\ast L, P}) = R_{\ast L, P}$ is called a Bayes decision function. For the commonly used losses such as the least squares loss treated in Section 5, the Bayes decision function $f_{\ast L, P}$ is P$X$-almost surely $[-M, M]$-valued, since $Y = [-M, M]$. In this case, it seems obvious to consider estimators with values in $[-M, M]$ on $X$. To this end, we now introduce the concept of clipping the decision function. Let $\hat{t}$ be the clipped value of some $t \in \mathbb{R}$ at $\pm M$ defined by

$$\hat{t} := \begin{cases} -M & \text{if } t < -M \\ t & \text{if } t \in [-M, M] \\ M & \text{if } t > M. \end{cases}$$

Then a loss is called clippable at $M > 0$ if, for all $(x, y, t) \in X \times Y \times \mathbb{R}$, we have

$$L(x, y, \hat{t}) \leq L(x, y, t).$$

Obviously, the latter implies

$$R_{L, P}(\hat{f}) \leq R_{L, P}(f)$$

for all $f : X \to \mathbb{R}$. In other words, restricting the decision function to the interval $[-M, M]$ containing our labels cannot worsen the risk, in fact, clipping this function typically reduces the risk. Hence, we consider the clipped version $\hat{f}_D$ of the decision function as well as the
risk $R_{L,P}(\hat{f}_D)$ instead of the risk $R_{L,P}(f_D)$ of the unclipped decision function. Note, this clipping idea does not change the learning method since it is performed after the training phase.

To modify the standard SVM approach (1), we assume that $(A_j)_{j=1,...,m}$ is a partition of $X$ such that $A_j \neq \emptyset$ for every $j \in \{1, \ldots, m\}$. Obviously, this implies $A_{j_1} \cap A_{j_2} = \emptyset$ for all $j_1, j_2 \in \{1, \ldots, m\}$ with $j_1 \neq j_2$ and

$$X = \bigcup_{j=1}^m A_j.$$ 

Now, the basic idea of the approach developed in this paper is to consider for each set of the partition $(A_j)_{j=1,...,m}$ an individual SVM. To describe this approach in a mathematically rigorous way, we have to introduce some more definitions and notations. Let us begin with the index set

$$I_j := \{i \in \{1, \ldots, n\} : x_i \in A_j\}, \quad j = 1, \ldots, m,$$

indicating the samples of $D$ contained in $A_j$, as well as the corresponding data set

$$D_j := \{(x_i, y_i) \in D : i \in I_j\}, \quad j = 1, \ldots, m.$$ 

Moreover, for every $j \in \{1, \ldots, m\}$, we define a (local) loss function $L_j: X \times Y \times \mathbb{R} \to [0, \infty)$ by

$$L_j(x, y, t) := \mathbb{1}_{A_j}(x)L(x, y, t), \quad (2)$$

where $L: X \times Y \times \mathbb{R} \to [0, \infty)$ is the loss that corresponds to our learning problem at hand. We further assume that $H_j$ is an RKHS over $A_j$ with kernel $k_j: A_j \times A_j \to \mathbb{R}$. Note that every function $f \in H_j$ is only defined on $A_j$ even though a function $f_D: X \to \mathbb{R}$ is finally sought. To this end, for $f \in H_j$, we define a function $\hat{f}: X \to \mathbb{R}$ by

$$\hat{f}(x) := \begin{cases} f(x), & x \in A_j, \\ 0, & x \notin A_j. \end{cases}$$

Then the space $\hat{H}_j := \{\hat{f} : f \in H_j\}$ equipped with the norm

$$\|\hat{f}\|_{\hat{H}_j} := \|f\|_{H_j}, \quad \hat{f} \in \hat{H}_j,$$

is an RKHS on $X$ (cf. Lemma [2]). That is, $\hat{H}_j$ is an isometrically isomorphic extension of the RKHS $H_j$ on $A_j$ to an RKHS on $X$. After all, we are now able to formulate a modified SVM approach. To this end, for every $j \in \{1, \ldots, m\}$, consider the local SVM optimization problem

$$f_{D_j, \lambda_j} = \arg \min_{\hat{f} \in \hat{H}_j} \lambda_j\|\hat{f}\|_{\hat{H}_j}^2 + \frac{1}{n} \sum_{i=1}^n L_j(x_i, y_i, \hat{f}(x_i)), \quad (3)$$
where $\lambda_j > 0$ for every $j \in \{1, \ldots, m\}$. Based on these empirical SVM solutions, we then define the decision function $f_{D, \lambda} : X \to \mathbb{R}$ by

$$f_{D, \lambda}(x) := \sum_{j=1}^{m} f_{D_j, \lambda_j}(x) = \sum_{j=1}^{m} \mathbbm{1}_{A_j}(x) f_{D_j, \lambda_j}(x), \quad (4)$$

where $\lambda := (\lambda_1, \ldots, \lambda_m)$. Here, clipping $f_{D, \lambda}$ at $M$ yields

$$f_{D, \lambda}(x) = \sum_{j=1}^{m} \mathbbm{1}_{A_j}(x) \hat{f}_{D_j, \lambda_j}(x)$$

for every $x \in X$. Note that the empirical SVM solutions $f_{D_j, \lambda_j}$ in (4) exist and are unique by [Steinwart and Christmann, 2008a, Theorem 5.5] and that, for arbitrary $j \in \{1, \ldots, m\}$, $f_{D_j, \lambda_j} = 0$ if $x_i \notin A_j$ for all $i \in \{1, \ldots, n\}$. In addition, the SVM optimization problem (3) equals the SVM optimization problem (1) using $H_j$, $D_j$, and the regularization parameter $\lambda_j := \frac{n}{|I_j|} \lambda_j$, since, for $\hat{f} \in \hat{H}_j$ and $f := f|_{A_j}$, we have

$$\lambda_j \|\hat{f}\|^2_{H_j} + \frac{1}{n} \sum_{i=1}^{n} L_j(x_i, y_i, \hat{f}(x_i)) = \lambda_j \|f\|^2_{H_j} + \frac{1}{n} \sum_{i \in I_j} L(x_i, y_i, f(x_i)) = \frac{|I_j|}{n} \left( \lambda_j \|f\|^2_{H_j} + \mathcal{R}_{L,D_j}(f) \right).$$

That is, $f_{D_j, \lambda_j}$ as in (3) and $h_{D_j, \lambda_j} := \arg\min_{f \in H_j} \lambda_j \|f\|^2_{H_j} + \mathcal{R}_{L,D_j}(f)$ satisfy

$$h_{D_j, \lambda_j} = f_{D_j, \lambda_j}|_{A_j}.$$

For the sake of completeness, we briefly examine the Bayes risks w.r.t. $P$ and $L_j$. To this end, let $X \subset \mathbb{R}^d$, $Y \subset \mathbb{R}$, $L : X \times Y \times \mathbb{R} \to [0, \infty)$ be a loss function and $P$ be a distribution on $X \times Y$ such that a Bayes decision function $f^*_{L,P} : X \to \mathbb{R}$ exists. Then, for all $j \in \{1, \ldots, m\}$ and losses $L_j$ defined by (2), it is easy to show

$$\mathcal{R}_{L_j,P}(f^*_{L,P}) = \mathcal{R}^*_j,$$

whenever $f^*_{L,P}$ exists. In other words, a Bayes decision function $f^*_{L,P}$ w.r.t. $P$ and $L$ additionally is a Bayes decision function w.r.t. $P$ and $L_j$. Moreover, for function spaces $\mathcal{F}_1, \ldots, \mathcal{F}_m$ over $X$, we have

$$\sum_{j=1}^{m} \min_{f_j \in F_j} \mathcal{R}_{L_j,D}(f_j) = \min_{f_1 \in \mathcal{F}_1, \ldots, f_m \in \mathcal{F}_m} \sum_{j=1}^{m} \mathcal{R}_{L_j,D}(f_j) \quad (5)$$

by the construction of the loss $L_j$.

Let us now present an advantageous characteristic of our modified SVM, namely the required computing time. Solving an usual SVM problem has a computational cost of $O(n^q)$ where $q \in [2, 3]$ and $n$ is the sample size. For the new approach we consider $m$ working sets of size $n_1, \ldots, n_m$ where $n_i \approx n_j$ for all $i, j \in \{1, \ldots, m\}$, i.e. $n_i \approx \frac{n}{m}$. Then for each
working set an usual SVM problem has to be solved such that, altogether, the modified SVM induces a computational cost of $O(m \left( \frac{n}{m} \right)^q)$. That is, for some $\beta > 0$ and $m \approx n^\beta$ our approach is computationally cheaper than a traditional SVM. Note that our strategy using a partition of the input space is a typical way to speed-up algorithms and handle large data sets. Other techniques that possess similar properties are e.g. applied in the articles cited in the introduction. Besides, we refer to (Tsang et al., 2007) and (Tsang et al., 2005) using enclosing ball problems to solve an SVM, to (Graf et al., 2005) presenting a model of multiple filtering SVMs and to (Collobert et al., 2001) investigating a mixture of SVMs based on several subsets of the training set.

To describe the above SVM approach $(A_j)_{j=1,\ldots,m}$ only has to be some partition of $X$. However, for the theoretical investigations concerning learning rates of our new approach, we have to further specify the partition. To this end, we denote by $B_{\ell^2_d}$ the closed unit ball in the $d$-dimensional Euclidean space $\ell^2_d$ and we define balls $B_1, \ldots, B_m$ with radius $r > 0$ and mutually distinct centers $z_1, \ldots, z_m \in X$ by

$$B_j := B_r(z_j) := \{x \in X : \|x - z_j\|_2 \leq r\}, \quad j \in \{1, \ldots, m\},$$

where $\| \cdot \|_2$ is the Euclidean norm in $\mathbb{R}^d$. Moreover, choose $r$ and $z_1, \ldots, z_m$ such that

$$\bigcup_{j=1}^m B_j = X,$$

i.e. such that the balls $B_1, \ldots, B_m$ cover $X$ (cf. Figure 2). The following well-known lemma relates the radius of such a cover with the number of centers.

**Lemma 1** Let $X \subset \mathbb{R}^d$ be a bounded subset, i.e. $X \subset cB_{\ell^2_d}$ for some constant $c > 0$. Then there exist balls $(B_j)_{j=1,\ldots,m}$ with radius $r > 0$ covering $X$ such that

$$r \leq 8cm^{-\frac{1}{d}}.$$

For simplicity of notation, we assume in the following that $X \subset B_{\ell^2_d}$, i.e. according to Lemma 1 there exists a cover $(B_j)_{j=1,\ldots,m}$ with

$$r \leq 8m^{-\frac{1}{d}}.$$  \hspace{1cm} (7)

Finally, we can specify the partition $(A_j)_{j=1,\ldots,m}$ of $X$ by the following assumption.

**A** Let $(A_j)_{j=1,\ldots,m}$ be a partition of $X \subset B_{\ell^2_d}$ such that $A_j \neq \emptyset$ for every $j \in \{1, \ldots, m\}$ and such that there exist mutually distinct $z_1, \ldots, z_m \in X$ with $A_j \subset B_r(z_j) =: B_j$, where $(B_j)_{j=1,\ldots,m}$ is a cover of $X$ satisfying (7).

In the remaining sections we will frequently refer to Assumption (A). However, the results hold as well if we merely assume $z_1, \ldots, z_m \in B_{\ell^2_d}$ instead of $z_1, \ldots, z_m \in X \subset B_{\ell^2_d}$ in (A). The following example illustrates that (A) is indeed a natural assumption.
Figure 2: Cover \( (B_j)_{j=1,...,m} \) of \( X \), where \( B_1, \ldots, B_m \) are balls with radius \( r \) and centers \( z_j \) (\( j = 1, \ldots, m \)).

Figure 3: Voronoi partition \( (A_j)_{j=1,...,m} \) of \( X \) defined by \( \delta \), where \( A_j \subset B_j \) for every \( j \in \{1, \ldots, m\} \).

**Example 1** For some \( r > 0 \), let us consider an \( r \)-net \( z_1, \ldots, z_m \) of \( X \), where \( z_1, \ldots, z_m \) are mutually distinct. Based on these \( z_1, \ldots, z_m \), a Voronoi partition \( (A_j)_{j=1,...,m} \) of \( X \) is defined by

\[
A_j := \left\{ x \in X : j = \min \left\{ \arg \min_{k \in \{1,...,m\}} \| x - z_k \|_2 \right\} \right\}, \tag{8}
\]

cf. Figure 3. That is, \( A_j \) contains all \( x \in X \) such that the center \( z_j \) is the nearest center to \( x \), and if there exist \( j_1 \) and \( j_2 \) with \( j_1 < j_2 \) and

\[
\| x - z_{j_1} \|_2 = \| x - z_{j_2} \|_2 < \| x - z_k \|_2
\]

for all \( k \in \{1,...,m\}\setminus\{j_1, j_2\} \), then \( x \in A_{j_1} \) since \( j_1 < j_2 \). In other words, they are resolved in favor of the smallest index of the involved centers. Moreover, it is obvious that \( A_j \neq \emptyset \), \( A_j \subset B_r(z_j) \) for all \( j \in \{1,...,m\} \), \( A_{j_1} \cap A_{j_2} = \emptyset \) for all \( j_1, j_2 \in \{1,...,m\} \) with \( j_1 \neq j_2 \), and \( X = \bigcup_{j=1}^{m} A_j \). In other words, a Voronoi partition based on an \( r \)-net \( z_1, \ldots, z_m \) of \( X \) satisfies condition (A), if \( r \) and \( m \) fulfill (7).

Following Example 1, we call the learning method producing \( f_{D, \lambda} \) given by (4) a Voronoi partition support vector machine, in short VP-SVM. Nevertheless, we just take a partition \( (A_j)_{j=1,...,m} \) satisfying (A) as basis here instead of requesting \( (A_j)_{j=1,...,m} \) to be a Voronoi partition.

Recall that our goal is to derive not only global but also local learning rates for this VP-SVM approach. To this end, we additionally consider an arbitrary measurable set \( T \subset X \) such that \( P_X(T) > 0 \). Then we examine the learning rate of the VP-SVM on this subset \( T \) of \( X \). To formalize this, it is necessary to introduce some basic notations related to \( T \). Let us define the index set \( J_T \) by

\[
J_T := \{ j \in \{1,...,m\} : A_j \cap T \neq \emptyset \} \tag{9}
\]

specifying every set \( A_j \) that has at least one common point with \( T \). Note that, for every non-empty set \( T \subset X \), the index set \( J_T \) is non-empty, too, i.e. \( |J_T| \geq 1 \). Besides, deriving
local rates on $T$ requires us to investigate the excess risk of the VP-SVM with respect to the distribution $P$ and the loss $L_T : X \times Y \times \mathbb{R} \to [0, \infty)$ defined by

$$L_T(x, y, t) := 1_T(x)L(x, y, t).$$

However, to manage the analysis we additionally need the loss $L_{J_T} : X \times Y \times \mathbb{R} \to [0, \infty)$ given by

$$L_{J_T}(x, y, t) := 1_{\bigcup_{j \in J_T} A_j}(x)L(x, y, t)$$

which may only be nonzero, if $x$ is contained in some set $A_j$ with $j \in J_T$. Note that the risks $R_{L_T, P}(f)$ and $R_{L_{J_T}, P}(f)$ quantify the quality of some function $f$ just on $T$ and

$$A_T := \bigcup_{j \in J_T} A_j \supset T,$$

respectively. Hence, examining the excess risks $R_{L_T, P}(\hat{f}_{D, \lambda}) - R_{L_T, P}^* \leq R_{L_{J_T}, P}(\hat{f}_{D, \lambda}) - R_{L_{J_T}, P}^*$ leads to learning rates on $A_T$ and implicitly on $T$. Recapitulatory, let us declare a second set of assumptions.

**T** For $T \subset X$, we define an index set $J_T$ by (9), loss functions $L_T, L_{J_T} : X \times Y \times \mathbb{R} \to [0, \infty)$ by (10) and (11), and the set $A_T := \bigcup_{j \in J_T} A_j$.

### 3. An Oracle Inequality for VP-SVMs

In this section, we first focus on RKHSs and direct sums of RKHSs. Then we present a lemma that relates the risk of a function w.r.t. the general loss $L$ to the risks w.r.t. the losses $L_j$. Finally, we establish a first oracle inequality for VP-SVMs.

Let us begin with some basic notations. For $q \in [1, \infty]$ and a measure $\nu$, we denote by $L_q(\nu)$ the Lebesgue spaces of order $q$ w.r.t. $\nu$ and for the Lebesgue measure $\mu$ on $X \subset \mathbb{R}^d$ we write $L_q(X) := L_q(\mu)$. In addition, for a measurable space $X$, the set of all real-valued measurable functions on $X$ is given by $L_0(X) := \{f : X \to \mathbb{R} \mid f \text{ measurable}\}$. Moreover, for a measure $\nu$ on $X$ and measurable $\tilde{X} \subset X$, we define the trace measure $\nu|_{\tilde{X}}$ of $\nu$ in $\tilde{X}$ by $\nu|_{\tilde{X}}(A) = \nu(A \cap \tilde{X})$ for every $A \subset X$.

Our first goal is to show that $f_{D, \lambda}$ in (4) is actually an ordinary SVM solution. To this end, we consider an RKHS on some $A \subset X$ and extend it to an RKHS on $X$ by the following lemma, where we omit the obvious proof.

**Lemma 2** Let $A \subset X$ and $H_A$ be an RKHS on $A$ with corresponding kernel $k_A$. Denote by $\hat{f}$ the extension of $f \in H_A$ to $X$ defined by

$$\hat{f}(x) := \begin{cases} f(x), & \text{for } x \in A, \\ 0, & \text{for } x \in X \setminus A. \end{cases}$$
Then the space $\hat{H}_A := \{ \hat{f} : f \in H_A \}$ equipped with the norm

$$\| \hat{f} \|_{\hat{H}_A} := \| f \|_{H_A}$$

is an RKHS on $X$ and its reproducing kernel is given by

$$\hat{k}_A(x, x') := \begin{cases} k_A(x, x'), & \text{if } x, x' \in A, \\ 0, & \text{else.} \end{cases} \tag{12}$$

Based on this lemma, we are now able to construct an RKHS by a direct sum of RKHSs $\hat{H}_A$ and $\hat{H}_B$ with $A, B \subset X$ and $A \cap B = \emptyset$. Here, we skip the proof once more, since the assertion follows immediately using, for example, orthonormal bases of $\hat{H}_A$ and $\hat{H}_B$.

**Lemma 3** For $A, B \subset X$ such that $A \cap B = \emptyset$ and $A \cup B \subset X$, let $H_A$ and $H_B$ be RKHSs of $k_A$ and $k_B$ over $A$ and $B$, respectively. Furthermore, let $\hat{H}_A$ and $\hat{H}_B$ be the RKHSs of all functions of $H_A$ and $H_B$ extended to $X$ in the sense of Lemma 2 and let $\hat{k}_A$ and $\hat{k}_B$ given by (12) be the associated reproducing kernels. Then $\hat{H}_A \cap \hat{H}_B = \{ 0 \}$ and hence the direct sum

$$H := \hat{H}_A \oplus \hat{H}_B \tag{13}$$

exists. For $\lambda_A, \lambda_B > 0$ and $f \in H$, let $\hat{f}_A \in \hat{H}_A$ and $\hat{f}_B \in \hat{H}_B$ be the unique functions such that $f = \hat{f}_A + \hat{f}_B$. Then we define the norm $\| \cdot \|_H$ by

$$\| f \|_H^2 := \lambda_A \| \hat{f}_A \|_{\hat{H}_A}^2 + \lambda_B \| \hat{f}_B \|_{\hat{H}_B}^2 \tag{14}$$

and $H$ equipped with the norm $\| \cdot \|_H$ is again an RKHS for which

$$k(x, x') := \lambda_A^{-1} \hat{k}_A(x, x') + \lambda_B^{-1} \hat{k}_B(x, x') , \quad x, x' \in X ,$$

is the reproducing kernel.

To relate Lemmas 2 and 3 with (4), we have to introduce some more notations. For pairwise disjoint sets $A_1, \ldots, A_m \subset X$, let $H_j$ be an RKHS on $A_j$ for every $j \in \{ 1, \ldots, m \}$. Then, based on RKHSs $\hat{H}_1, \ldots, \hat{H}_m$ on $X$ defined by Lemma 2 the joined RKHSs can be designed analogously to Lemma 3. That is, for an arbitrary index set $J \subset \{ 1, \ldots, m \}$ and a vector $\lambda = (\lambda_j)_{j \in J} \in (0, \infty)^{|J|}$, the direct sum

$$H_J := \bigoplus_{j \in J} \hat{H}_j = \left\{ f = \sum_{j \in J} f_j : f_j \in \hat{H}_j \text{ for all } j \in J \right\}$$

is again an RKHS equipped with the norm

$$\| f \|_{H_J}^2 = \sum_{j \in J} \lambda_j \| f_j \|_{\hat{H}_j}^2 . \tag{15}$$

If $J = \{ 1, \ldots, m \}$ we simply write

$$H := H_J \tag{16}$$

Note that $H$ contains inter alia $f_{D, \lambda}$ given by (4). Summarizing, we can define another assumption set.
Lemma 4 expressed by the following lemma. For $A, B \subset X$ as well as $f$ and $\lambda$ have

Based on this generalization and the whole index set $J \subset \{1, \ldots, m\}$, let us consider an arbitrary index set $J \subset \{1, \ldots, m\}$ and define the corresponding loss function $L_J : X \times Y \times \mathbb{R} \to [0, \infty)$ by

$$L_J(x, y, t) := \mathbb{1}_{J \subset A}(x)L(x, y, t).$$

Now, it is straightforward to show

$$\mathcal{R}_{L,J,P}(f) = \sum_{j \in J} \mathcal{R}_{L,j,P}(f)$$

for every function $f : X \to \mathbb{R}$. Based on this generalization and the whole index set $J = \{1, \ldots, m\}$, let us briefly consider Lemma 4 for the empirical measure $D$ and for $\hat{f}_{D,\lambda} = \sum_{j=1}^m \mathbb{1}_{A_j} f_{D,\lambda_j}$, where $f_{D,\lambda_j}, j = 1, \ldots, m$, are defined by (3). Then, for an arbitrary $f \in H$, it immediately follows

$$\|f_{D,\lambda}\|_H^2 + \mathcal{R}_{L,D}(\hat{f}_{D,\lambda}) = \sum_{j=1}^m \left( \lambda_j \|f_{D,j,\lambda_j}\|_{\hat{H}_j}^2 + \mathcal{R}_{L,j,D}(\hat{f}_{D,\lambda}) \right)$$

$$\leq \sum_{j=1}^m \left( \lambda_j \|\mathbb{1}_{A_j} f\|_{\hat{H}_j}^2 + \mathcal{R}_{L,j,D}(f) \right)$$

$$= \|f\|_H^2 + \mathcal{R}_{L,D}(f). \quad (17)$$
That is, $f_{D,A}$ is the decision function of an SVM using $H$ and $L$ as well as the regularization parameter $\lambda = 1$. In other words, the latter SVM equals the VP-SVM given by (1). This will be a key insight used in our analysis.

To derive an oracle inequality, i.e. an appropriate upper bound for the excess risk $R_{L,f,P}(\hat{f}_{D,A}) - R^*_{L,f,P}$ for some index set $J \subset \{1, \ldots, m\}$, we have to introduce a few more notations. Let $P$ be a distribution on $X \times Y$ such that a Bayes decision function $f^*_{L,P} : X \to [-M, M]$ exists, for some constant $M > 0$ at which $L$ can be clipped. Moreover, we denote by $L \circ f$ the function $(x, y) \mapsto L(x, y, f(x))$. If there exist constants $B > 0$, $\vartheta \in [0, 1]$, and $V \geq B^{2-\vartheta}$ such that we have

$$L(x, y, t) \leq B,$$

$$\mathbb{E}_P \left( L \circ f - L \circ f^*_{L,P} \right)^2 \leq V \cdot \left( \mathbb{E}_P \left( L \circ f - L \circ f^*_{L,P} \right) \right)^{\vartheta},$$

for all $(x, y) \in X \times Y$, $t \in [-M, M]$, and $f : X \to [-M, M]$, we say that the supremum bound (18) and the variance bound (19), respectively, is fulfilled. Actually, (18) immediately yields

$$L_J(x, y, t) = \mathbb{I}_{\cup_{j \in J} A_j}(x)L(x, y, t) \leq L(x, y, t) \leq B$$

for all $(x, y) \in X \times Y$ and $t \in [-M, M]$, i.e. the supremum bound is also satisfied for $L_J$. Moreover, if (19) holds for all $f : X \to [-M, M]$, the variance bound using the loss $L_J$ is satisfied, too. Indeed, by the use of $\tilde{f}(x) := \mathbb{I}_{\cup_{j \in J} A_j}(x)f(x) + \mathbb{I}_{X \setminus (\cup_{j \in J} A_j)}(x)f^*_{L,P}(x)$ for all $x \in X$, we have

$$\mathbb{E}_P \left( L_J \circ f - L_J \circ f^*_{L,P} \right)^2 = \mathbb{E}_P \left( L_J \circ \tilde{f} - L_J \circ f^*_{L,P} \right)^2$$

$$= \mathbb{E}_P \left( L \circ \tilde{f} - L \circ f^*_{L,P} \right)^2$$

$$\leq V \cdot \left( \mathbb{E}_P \left( L \circ \tilde{f} - L \circ f^*_{L,P} \right) \right)^{\vartheta}$$

$$\leq V \cdot \left( \mathbb{E}_P \left( L \circ f - L \circ f^*_{L,P} \right) \right)^{\vartheta}$$

for all $f : X \to [-M, M]$. Let us quickly define a third assumption set.

**P** Let $P$ be a distribution on $X \times Y$ such that the variance bound (19) is satisfied for constants $\vartheta \in [0, 1]$, $V \geq B^{2-\vartheta}$, and all functions $f : X \to [-M, M]$.

Up to now, there is still missing a classical tool that is used to derive learning rates, namely entropy numbers, see (Carl and Stephan, 1990) or (Steinwart and Christmann, 2008a, Definition A.5.26). Recall that, for normed spaces $(E, \| \cdot \|_E)$ and $(F, \| \cdot \|_F)$ as well as an integer $i \geq 1$, the $i$-th (dyadic) entropy number of a bounded, linear operator $S : E \to F$ is defined by

$$e_i(S : E \to F) := e_i(SB_E, \| \cdot \|_F)$$

$$:= \inf \left\{ \varepsilon > 0 : \exists s_1, \ldots, s_{2^{i-1}} \in SB_E \text{ such that } SB_E \subset \bigcup_{j=1}^{2^{i-1}} (s_j + \varepsilon B_F) \right\},$$
where we use the convention \( \inf \emptyset := \infty \), and \( B_F \) as well as \( B_F \) denote the closed unit balls in \( E \) and \( F \), respectively. Finally, we present a first oracle inequality involving an upper bound for the excess risk \( R_{L,J,P}(f_{D,\lambda}) - R_{L,J,P}^* \), where \( J \subset \{1, \ldots, m\} \) is an arbitrary index set.

**Theorem 5** Let \( L : X \times Y \times \mathbb{R} \to [0, \infty) \) be a locally Lipschitz continuous loss that can be clipped at \( M > 0 \) and that satisfies the supremum bound \( (15) \) for some \( B > 0 \). Based on a partition \( (A_j)_{j=1,\ldots,m} \) of \( X \), where \( A_j \neq \emptyset \) for every \( j \in \{1, \ldots, m\} \), we assume \((H)\). Furthermore, for an arbitrary index set \( J \subset \{1, \ldots, m\} \), we suppose \((P)\). Assume that, for fixed \( n \geq 1 \), there exist constants \( p \in (0,1) \) and \( a_1, \ldots, a_m > 0 \) such that for all \( j \in \{1, \ldots, m\} \)

\[
    e_i(\text{id} : H_j \to L_2(\mathbb{P}_{X|A_j})) \leq a_j \frac{1}{\tau^p}, \quad i \geq 1.
\]

Finally, fix an \( f_0 \in H \) and a constant \( B_0 \geq B \) such that \( \|L_J \circ f_0\|_\infty \leq B_0 \). Then, for all fixed \( \tau > 0 \), \( \lambda = (\lambda_1, \ldots, \lambda_m) > 0 \), and

\[
a := \max \left\{ c_p \sqrt{m} \left( \sum_{j=1}^{m} \lambda_j^{-p} a_j^{2p} \right)^{\frac{1}{2p}}, B \right\},
\]

the VP-SVM given by \((4)\) using \( \hat{H}_1, \ldots, \hat{H}_m \) and \( L_J \) satisfies

\[
\begin{align*}
\sum_{j=1}^{m} \lambda_j \|f_{D,\lambda_j}\|_{H_j}^2 + R_{L,J,P}(\hat{f}_{D,\lambda}) - R_{L,J,P}^* & \leq 9 \left( \sum_{j=1}^{m} \lambda_j \|f_0\|_{H_j}^2 + R_{L,J,P}(f_0) - R_{L,J,P}^* \right) + C \left( a_1^{2p} n^{-1} \right)^{\frac{1}{2p} - \frac{1}{2p} + \frac{1}{p} + 3 \left( \frac{72V \tau}{n} \right)^{\frac{1}{2p} - \frac{1}{p} + \frac{1}{p}} + 15B_0 \tau \frac{1}{n} \right)
\end{align*}
\]

with probability \( \mathbb{P}^n \) not less than \( 1 - 3e^{-\tau} \), where \( C > 0 \) is a constant only depending on \( p \), \( M, V, \vartheta, \) and \( B \).

The above theorem deals with the case of a partition with quite a few sets \( A_j, j \in \{1, \ldots, m\} \). However, if we consider a partition consisting of just one set \( A_1 := X \), Theorem [5] is supposed to provide an oracle inequality that is comparable to the already known ones. To make that sure, let us briefly consider the case \( m = 1 \) and hence \( A_1 := X \), \( \lambda_1 := \lambda \) as well as RKHSs \( H_1 = \hat{H}_1 = H \) over \( X \) with \( \| \cdot \|_H^2 = \| \cdot \|_{H_1}^2 \). Note that in this case we have \( f_{D,\lambda} = f_{D,\lambda_1} \). If \((20)\) holds for \( H_1 \), Theorem [5] yields that an SVM using \( H \) and \( L_J = L \) satisfies

\[
\begin{align*}
\lambda \|f_{D,\lambda}\|_{H_1}^2 + R_{L,P}(\hat{f}_{D,\lambda}) - R_{L,P}^* & \leq 9 \left( \lambda \|f_0\|_{H_1}^2 + R_{L,P}(f_0) - R_{L,P}^* \right) + C \left( \hat{c}_p a_1^{2p} \lambda^{\frac{1}{2p} - \frac{1}{2p} + \frac{1}{p}} \right) + 3 \left( \frac{72V \tau}{n} \right)^{\frac{1}{2p} - \frac{1}{p} + \frac{1}{p}} + 15B_0 \tau \frac{1}{n} \right)
\end{align*}
\]
with probability $P^n$ not less than $1 - 3e^{-\tau}$ for fixed $\tau > 0$. Note that this oracle inequality indeed matches with the one stated in (Steinwart and Christmann 2008), Theorem 7.23 apart from the constant $\tilde{c}_p$, which is, however, only depending on $p$, $\vartheta$, and $B$.

In the following section, we focus on RKHSs using Gaussian RBF kernels and examine the associated entropy numbers to specify (20). Subsequently in Section 5, we additionally consider the least squares loss and adapt the oracle inequality of Theorem 5.

4. Entropy Estimates for Local Gaussian RKHSs

In this section, we refine assumption (20). More precisely, in the subsequent theorem we determine an upper bound for the entropy numbers of the operator $\text{id} : H_\gamma(A) \to L_2(P_X|A)$, where $H_\gamma(A)$ is the RKHS over $A$ of the Gaussian RBF kernel $k_\gamma$ on $A \subset \mathbb{R}^d$ defined by

$$k_\gamma(x, x') := \exp \left(-\gamma^{-2} \|x - x'\|^2_2\right), \quad x, x' \in A,$$

for some width $\gamma > 0$.

**Theorem 6** Let $X \subset \mathbb{R}^d$, $P_X$ be a distribution on $X$ and $A \subset X$ be such that $\hat{A} \neq \emptyset$ and such that there exists an Euclidean ball $B \subset \mathbb{R}^d$ with radius $r > 0$ containing $A$, i.e. $A \subset B$. Moreover, for $0 < \gamma \leq r$, let $H_\gamma(A)$ be the RKHS of the Gaussian RBF kernel $k_\gamma$ over $A$. Then, for all $p \in (0, 1)$, there exists a constant $c_p > 0$ such that

$$e_i(\text{id} : H_\gamma(A) \to L_2(P_X|A)) \leq c_p \sqrt{P_X(A)} r^{d+2p} \gamma^{-d+2p} i^{-\frac{1}{2d}}, \quad i \geq 1.$$

Obviously, this theorem specifies assumption (20). Now, for the Gaussian case we elaborate assumption (H) and introduce the following additional set of assumptions.

**G** Let $A_1, \ldots, A_m$ be pairwise disjoint subsets of $X$ with non-empty interior such that, for some fixed $r > 0$ and every $j \in \{1, \ldots, m\}$, $\sup_{x, x' \in A_j} \|x - x'\|_2 \leq 2r$ is satisfied. Furthermore, for every $j \in \{1, \ldots, m\}$, let $H_j := H_{\gamma_j}(A_j)$ be the RKHS of the Gaussian kernel $k_{\gamma_j}$ with width $\gamma_j \in (0, r]$ over $A_j$. Consequently, for $\lambda := (\lambda_1, \ldots, \lambda_m) \in (0, \infty)^m$, we define the joined RKHS $H := \bigoplus_{j=1}^m H_{\gamma_j}(A_j)$ by (16) equipped with the norm (15).

Since we do not consider SVMs with a fixed kernel, we use a more detailed notation than (3) and (4) in the following specifying the kernel width $\gamma_j$ of the RKHS $H_{\gamma_j}(A_j)$ at hand. For all $j \in \{1, \ldots, m\}$ and $\gamma := (\gamma_1, \ldots, \gamma_m)$, we thus write

$$f_{D, \lambda, \gamma_j} = \arg \min_{f \in H_{\gamma_j}(A_j)} \lambda_j \|f\|^2_{H_{\gamma_j}(A_j)} + \frac{1}{n} \sum_{i=1}^n L_j(x_i, y_i, f(x_i)),$$

and

$$f_{D, \lambda, \gamma} := \sum_{j=1}^m f_{D_j, \lambda, \gamma_j}$$

instead of $f_{D, \lambda}$ and $f_{D, \lambda}$ in the remainder of this work.

In the subsequent section, we consider the least squares loss which, together with Assumption (G) and Theorem 6, allows us to elaborate the oracle inequality stated in Theorem 5 so that we finally obtain learning rates.
5. Learning Rates for Least Squares VP-SVMs

In this section, the non-parametric least squares regression problem is considered using the least squares loss $L : Y \times \mathbb{R} \to [0, \infty)$ defined by $L(y, t) := (y - t)^2$. It is well known that, in this case, the Bayes decision function $f_{L,P}^* : \mathbb{R}^d \to \mathbb{R}$ is given by $f_{L,P}^*(x) = \mathbb{E}_p(Y|x)$ for $P_X$-almost all $x \in \mathbb{R}^d$. Moreover, this function is unique up to zero-sets. Besides, for the least squares loss the equality

$$R_{L,P}(f) - R_{L,P}^* = \|f - f_{L,P}^*\|_{L_2(P_X)}^2$$

can be shown by some simple, well-known transformations. Recall that $T$ is a non-empty subset of $X$, where the index set $J_T$ defined by $\square$ indicates every set $A_j$ of the partition $(A_j)_{j=1,\ldots,m}$ of $X$ that shares at least one point with $T$. The associated loss function $L_{J_T} : X \times Y \times \mathbb{R} \to [0, \infty)$ is defined by $\square$.

5.1 Basic Oracle Inequalities for LS-VP-SVMs

To formulate oracle inequalities and derive rates for VP-SVMs using the least squares loss, the target function $f_{L,P}^*$ is assumed to satisfy certain smoothness conditions. To this end, we initially recall the modulus of smoothness, a device to measure the smoothness of functions (see e.g. (DeVore and Lorentz, 1993, p. 44), (DeVore and Popov, 1988, p. 398), and (Beres and DeVore, 1978, p. 360)). Denote by $\| \cdot \|_2$ the Euclidean norm and let $X \subset \mathbb{R}^d$ be a subset with non-empty interior, $\nu$ be an arbitrary measure on $X$, $p \in (0, \infty]$, and $f : X \to \mathbb{R}$ be contained in $L_p(\nu)$. Then, for $s \in \mathbb{N}$, the $s$-th modulus of smoothness of $f$ is defined by

$$\omega_{s,L_p(\nu)}(f, t) := \sup_{\|h\|_2 \leq t} \|\Delta_h^s(f, \cdot)\|_{L_p(\nu)},$$

where $\Delta_h^s(f, \cdot)$ denotes the $s$-th difference of $f$ given by

$$\Delta_h^s(f, x) = \begin{cases} \sum_{j=0}^{s} \binom{s}{j} (-1)^{s-j} f(x + jh) & \text{if } x \in X_{s,h} \\ 0 & \text{if } x \notin X_{s,h} \end{cases}$$

for $h = (h_1, \ldots, h_d) \in [0, \infty)^d$ and $X_{s,h} := \{x \in X : x + th \in X \text{ f.a. } t \in [0, s]\}$. Based on the modulus of smoothness, we introduce Besov spaces, i.e. function spaces that provide a finer scale of smoothness than the commonly used Sobolev spaces and that will thus be assumed to contain the target function later on. To this end, let $1 \leq p, q \leq \infty$, $\alpha > 0$, $s := \lfloor \alpha \rfloor + 1$, and $\nu$ be an arbitrary measure. Then the Besov space $B^{\alpha}_{p,q}(\nu)$ is defined by

$$B^{\alpha}_{p,q}(\nu) := \left\{ f \in L_p(\nu) : |f|_{B^{\alpha}_{p,q}(\nu)} < \infty \right\},$$

where the seminorm $| \cdot |_{B^{\alpha}_{p,q}(\nu)}$ is given by

$$|f|_{B^{\alpha}_{p,q}(\nu)} := \left( \int_0^\infty \left( t^{-\alpha} \omega_{s,L_p(\nu)}(f, t) \right)^q \frac{dt}{t} \right)^{\frac{1}{q}},$$

or

$$|f|_{B^{\alpha}_{p,\infty}(\nu)} := \sup_{t > 0} \left( t^{-\alpha} \omega_{s,L_p(\nu)}(f, t) \right),$$

15
see e.g. (Adams and Fournier, 2003, Section 7) and (Triebel, 2010, Sections 2 and 3). Note that \(\|f\|_{B_p^q(\nu)} := \|f\|_{L_p(\nu)} + \|f\|_{B_p^q(\nu)}\) actually describes a norm of \(B_p^q(\nu)\) for all \(q \in [1, \infty]\), see e.g. (DeVore and Lorentz, 1993, pp. 54/55) and (DeVore and Popov, 1988, p. 398). Again, if \(\nu\) is the Lebesgue measure on \(X\), we write \(B_p^q(X) := B_p^q(\nu)\). For the sake of completeness, recall from e.g. (Adams and Fournier, 2003, Section 3) and (Triebel, 2010, Sections 2 and 3) the scale of Sobolev spaces \(W_p^\alpha(\nu)\) defined by

\[
W_p^\alpha(\nu) := \left\{ f \in L_p(\nu): \partial^{(\beta)} f \in L_p(\nu) \text{ exists for all } \beta \in \mathbb{N}_0^d \text{ with } |\beta| \leq \alpha \right\},
\]

where \(\alpha \in \mathbb{N}_0, 1 \leq p \leq \infty, \nu\) is an arbitrary measure, and \(\partial^{(\beta)}\) is the \(\beta\)-th weak derivative for a multi-index \(\beta = (\beta_1, \ldots, \beta_d) \in \mathbb{N}_0^d\) with \(|\beta| = \sum_{i=1}^d \beta_i\). That is, \(W_p^\alpha(\nu)\) is the space of all functions in \(L_p(\nu)\), whose weak derivatives up to order \(\alpha\) exist and are contained in \(L_p(\nu)\). Moreover, the Sobolev space is equipped with the Sobolev norm

\[
\|f\|_{W_p^\alpha(\nu)}^p := \sum_{|\beta| \leq \alpha} \left\| \partial^{(\beta)} f \right\|_{L_p(\nu)}^p,
\]

(cf. Adams and Fournier, 2003, page 60). We write \(W_p^0(\nu) = L_p(\nu)\) and, for the Lebesgue measure \(\mu\) on \(X \subset \mathbb{R}^d\), we define \(W_p^\alpha(X) := W_p^\alpha(\mu)\). It is well-known, see e.g. (Edmunds and Triebel, 1996, p. 25 and p. 44), that the Sobolev spaces \(W_p^\alpha(\mathbb{R}^d)\) fall into the scale of Besov spaces, namely

\[
W_p^\alpha(\mathbb{R}^d) \subset B_p^{\alpha,q}(\mathbb{R}^d)
\]

for \(\alpha \in \mathbb{N}, p \in (1, \infty),\) and \(\max\{p,2\} \leq q \leq \infty\). Moreover, for \(p = q = 2\) we actually have equality, that is \(W_2^\alpha(\mathbb{R}^d) = B_2^{\alpha,2}(\mathbb{R}^d)\) with equivalent norms.

Based on the least squares loss and RKHSs using Gaussian kernels over the partition sets \(A_j\), the subsequent theorem refines the oracle inequality stated in Theorem 5.

**Theorem 7** Let \(Y := [-M, M]\) for \(M > 0, L : Y \times \mathbb{R} \to [0, \infty)\) be the least squares loss and \(P\) be a distribution on \(\mathbb{R}^d \times Y\). We write \(X := \text{supp} \, P_X\). Furthermore, let (A) and (G) be satisfied. In addition, for an arbitrary subset \(T \subset X\), we assume (T). Moreover, let \(f_{L,P}^* : \mathbb{R}^d \to \mathbb{R}\) be a Bayes decision function such that \(f_{L,P}^* \in L_2(\mathbb{R}^d) \cap L_\infty(\mathbb{R}^d)\) as well as \(f_{L,P}^* \in B_2^{\alpha,q}(P_{X|A_T})\) for some \(\alpha \geq 1\). Then, for all \(p \in (0, 1), n \geq 1, \tau \geq 1, \gamma = (\gamma_1, \ldots, \gamma_m) \in (0, r]^m, \) and \(A = (A_1, \ldots, A_m) > 0,\) the VP-SVM given by (4) using \(\hat{H}_{\gamma_1}(A_1), \ldots, \hat{H}_{\gamma_m}(A_m)\), and the loss \(L_{JT}\) satisfies

\[
\sum_{j=1}^m \lambda_j \|f_{D,j,\lambda_j,\gamma_j}\|_{H_{\gamma_j}(A_j)}^2 + R_{L_{J,T},P}(\hat{f}_{D,\lambda,\gamma}) - R_{L_{J,T},P}^*
\leq C_{M,\alpha,P} \left( \sum_{j \in J_T} \lambda_j \gamma_j^{-d} + \frac{(\max_{j \in J_T} \gamma_j \gamma_j^d)}{(\min_{j \in J_T} \gamma_j \gamma_j^{d})} \max_{j \in J_T} \gamma_j^{2d} + r^{2p} \left( \sum_{j=1}^m \lambda_j^{-1} \gamma_j^{-d+2p} P_X(A_j) \right)^{p} \right)^{n^{-1} + \tau n^{-1}},
\]

with probability \(P_n\) not less than \(1 - e^{-\tau\gamma}\), where \(C_{M,\alpha,P} > 0\) is a constant only depending on \(M, \alpha, p, d, \|f_{L,P}^*\|_{L_2(\mathbb{R}^d)}, \|f_{L,P}^*\|_{L_\infty(\mathbb{R}^d)}, \) and \(\|f_{L,P}^*\|_{B_2^{\alpha,q}(P_{X|A_T})}\).
Using this oracle inequality, we derive learning rates w.r.t. the loss \( L_{J_T} \) for the learning method described by (3) and (4) in the following theorem.

**Theorem 8** Let \( \tau \geq 1 \) be fixed and \( \beta \geq \frac{2\alpha}{d} + 1 \). Under the assumptions of Theorem 7 and with

\[
\begin{align*}
  r_n &= c_1 n^{-\frac{1}{2d}}, \\
  \lambda_{n,j} &= c_2 r^d n^{-1}, \\
  \gamma_{n,j} &= c_3 n^{-\frac{1}{2d+1}},
\end{align*}
\]

for every \( j \in \{1, \ldots, m_n\} \), we have, for all \( n \geq 1 \) and \( \xi > 0 \),

\[
\mathcal{R}_{L_{J_T}^*,P}(f_D, \lambda, \tau_n) - \mathcal{R}_{L_{J_T}^*,P}^* \leq C \tau n^{-\frac{2\alpha}{2d+\alpha} + \xi}
\]

with probability \( P^n \) not less than \( 1 - e^{-\tau} \), where \( \lambda_n := (\lambda_{n,1}, \ldots, \lambda_{n,m_n}) \) as well as \( \gamma_n := (\gamma_{n,1}, \ldots, \gamma_{n,m_n}) \) and \( C, c_1, c_2, c_3 \) are positive constants with \( c_3 \leq c_1 \).

In the latter theorem the condition \( \beta \geq \frac{2\alpha}{d} + 1 \) is required to ensure \( \gamma_{n,j} \leq r_n, j = 1, \ldots, m_n \), which in turn is a prerequisite arising from Theorem 6 and the used entropy estimate. Let us briefly examine the extreme case \( \beta = \frac{2\alpha}{d} + 1 \). Using \( r_n \approx n^{-\frac{1}{2d}} \) and (7) leads to covering numbers of the form \( m_n \approx n^{\frac{d}{2d+\alpha}} \) and computational costs of \( \mathcal{O}(m_n^{\frac{d}{2d+\alpha}}) = \mathcal{O}(n^{\frac{d}{2d+\alpha}}) \) which is actually less than the computational cost of order \( n^{q}, q \in [2, 3] \), of an usual SVM. Note that for increasing \( \beta \) the computational cost of an VP-SVM is increasing as well. However, for \( \beta > \frac{2\alpha}{d} + 1, r_n \approx n^{-\frac{1}{2d}}, \) and \( m_n \approx n^{\frac{1}{\beta}} \), a VP-SVM has costs of \( \mathcal{O}(n^{\frac{1}{1+(\beta-1)q}}) \) which still is less that \( \mathcal{O}(n^q) \).

Let us finally take a closer look at the VP-SVM given by (4) and the considerations related to (17), where \( f_{D, \lambda} \in H = \bigoplus_{j=1}^m \hat{H}_j \) solves the minimization problem

\[
f_{D, \lambda} = \arg\min_{f_j \in \hat{H}_1, \ldots, f_m \in \hat{H}_m} \sum_{j=1}^m \lambda_j \| f_j \|_{\hat{H}_j}^2 + \mathcal{R}_{L,D}(\sum_{j=1}^m f_j).
\]

Choosing \( \lambda_1 = \ldots = \lambda_m \), the VP-SVM problem can be understood as \( \ell_2 \)-multiple kernel learning (MKL) problem using the RKHSs \( \hat{H}_1, \ldots, \hat{H}_m \). Learning rates for MKL have been treated, for example, in (Suzuki, 2011) and (Kloft and Blanchard, 2012). Assuming \( f^*_L,H \in H \), the learning rate achieved in (Suzuki, 2011) is \( mn^{-\frac{1}{1+s}} \) for dense settings, where \( s \) is the so-called spectral decay coefficient. In addition, Kloft and Blanchard (2012) obtain essentially the same rates under these assumptions. Let us therefore briefly investigate the above rate of (Suzuki, 2011). For RKHSs that are continuously embedded in a Sobolev space \( W^q_2(X) \), we have \( s = \frac{d}{2q} \) such that the learning rate reduces to \( mn^{-\frac{2\alpha}{2d+\alpha}} \). Note that this learning rate is \( m \) times the optimal learning rate \( n^{-\frac{2\alpha}{2d+\alpha}} \), where the number \( m = m_n \) of kernels may increase with the sample size \( n \). In particular, if \( m_n \to \infty \) polynomially, then the rates obtained in (Suzuki, 2011) become substantially worse than the optimal rate. In contrast, due to the special choice of the RKHSs, this is not the case for our VP-SVM problem, provided that \( m_n \) does not grow faster than \( n^{1/\beta} \).
Figure 5: An input space $X$ with the corresponding Voronoi partition as well as a subset $T \subset X$ enlarged by a \( \delta \)-tube to $T + \delta$.

Note that the oracle inequalities and learning rates achieved in Theorems 7 and 8 require $f_{L,P}^* \in B_{2,\infty}^\alpha(P_X | \bigcup_{j \in J_T} A_j)$. However, for an increasing sample size $n$, the sets $A_j$ shrink and the index set $J_T$, indicating every set $A_j$ such that $A_j \cap T \neq \emptyset$ and $T \subset \bigcup_{j \in J_T} A_j$, increases. In particular, this also involves that the set $\bigcup_{j \in J_T} A_j$ covering $T$ changes in tandem with $n$. Since this is very inconvenient and since it would be desirable to assume a certain level of smoothness of the target function on a fixed region for all $n \in \mathbb{N}$, we consider the set $T$ enlarged by an \( \delta \)-tube. To this end, for $\delta > 0$, we define $T + \delta$ by

\[
T + \delta := \{ x \in X : \exists t \in T \text{ such that } \|x - t\|_2 \leq \delta \},
\]

which implies $T \subset T + \delta \subset X$, cf. Figure 5. Note that, for every $\delta > 0$, there exists an $n_\delta \in \mathbb{N}$ such that for every $n \geq n_\delta$ the union of all partition sets $A_j$, having at least one common point with $T$, is contained in $T + \delta$, i.e.

\[
\forall \delta > 0 \ \exists n_\delta \in \mathbb{N} \ \forall n \geq n_\delta : \bigcup_{j \in J_T} A_j \subset T + \delta,
\]

where $J_T := \{ j \in \{1, \ldots, m_n\} : A_j \cap T \neq \emptyset \}$. Collectively, this implies

\[
T \subset \bigcup_{j \in J_T} A_j \subset T + \delta
\]

for all $n \geq n_\delta$. Furthermore, since every set $A_j$ is contained in a ball with radius $r_n = cn^{-\frac{1}{d}}$, the lowest sample size $n_\delta$ in (25) can be determined by choosing the smallest $n_\delta \in \mathbb{N}$ such that $\delta \geq 2r_{n_\delta}$ with $r_{n_\delta}$ as in (7), that is

\[
n_\delta = \left\lceil \left( \frac{2c}{\delta} \right)^{\frac{1}{d}} \right\rceil.
\]

This leads to the following corollary where we present an oracle inequality and learning rates assuming the smoothness level $\alpha$ of the target function on a fixed region.

**Corollary 9** Let $Y := [-M, M]$ for $M > 0$, $L : Y \times \mathbb{R} \to [0, \infty)$ be the least squares loss, and $P$ be a distribution on $\mathbb{R}^d \times Y$. We write $X := \text{supp} P_X$. Furthermore, let (A) and (G) be satisfied. In addition, for an arbitrary subset $T \subset X$, we assume (T). Moreover, let $f_{L,P}^* : \mathbb{R}^d \to \mathbb{R}$ be a Bayes decision function such that $f_{L,P}^* \in L_2(\mathbb{R}^d) \cap L_\infty(\mathbb{R}^d)$ as well as

\[
f_{L,P}^* \in B_{2,\infty}^\alpha(P_X | T + \delta)
\]
for \( \alpha \geq 1 \) and some \( \delta > 0 \). Then, for all \( p \in (0,1), n \geq n_\delta, \tau \geq 1, \gamma = (\gamma_1, \ldots, \gamma_m) \in (0,r]^m \), and \( \lambda = (\lambda_1, \ldots, \lambda_m) > 0 \), the VP-SVM given by (1) using \( \hat{H}_{\gamma_1}(A_1), \ldots, \hat{H}_{\gamma_m}(A_m) \), and the loss \( L_T \) satisfies

\[
\sum_{j=1}^{m} \lambda_j \|f_{D_j,\lambda_j,\gamma_j}\|^2_{H_{\gamma_j}(A_j)} + R_{LT,P}(\hat{f}_{D,\lambda,\gamma}) - R^*_L \leq C_{M,\alpha,p} \left( \sum_{j \in J_T} \lambda_j \gamma_j^{-d} + \left( \max_{j \in J_T} \frac{\gamma_j}{\min_{j \in J_T} \gamma_j} \right)^d \max_{j \in J_T} \gamma_j^{2\alpha} + \tau^2 p \left( \sum_{j=1}^{m} \lambda_j^{-1} \gamma_j^{d+2p} P_X(A_j) \right)^p \right)^{n^{-1} + \tau n^{-1}}
\]

with probability \( P^n \) not less than \( 1 - e^{-\tau^2} \), where \( C_{M,\alpha,p} > 0 \) is the same constant as in Theorem 7.

Additionally, let \( \beta \geq \frac{2\alpha}{d} + 1 \) as well as, for every \( j \in \{1, \ldots, m\} \), \( r_n, \lambda_n, j \), and \( \gamma_n,j \) be as in (21), (22), and (23), respectively, where \( c_1, c_2, c_3 \) are user-specified positive constants with \( c_3 \leq c_1 \). Then, for all \( n \geq n_\delta \) \( = \left( \frac{2\alpha}{d} \right)^{\beta} \) and \( \xi > 0 \), we have

\[
R_{LT,P}(f_{D,\lambda_n,\gamma_n}) - R^*_L \leq C \tau n^{-\frac{2\alpha}{d+2\alpha} + \xi}
\]

with probability \( P^n \) not less than \( 1 - e^{-\tau} \), where \( \lambda_n := (\lambda_{n,1}, \ldots, \lambda_{n,m_n}) \), \( \gamma_n := (\gamma_{n,1}, \ldots, \gamma_{n,m_n}) \), and \( C \) is a positive constant.

Note that the assumption \( f^*_{L,P} \in B^{\alpha}_2,\infty(P_X|T+\delta) \) made in Corollary 9 is satisfied if, for example, \( f^*_{L,P} \in B^{\alpha}_2,\infty(T+2\delta) \) and \( P_X \) has a bounded Lebesgue density on \( T+\delta \). Moreover, if this density is even bounded away from 0, it is well-known that the minmax rate is \( n^{-\frac{2\alpha}{d+2\alpha}} \) for \( \alpha > d/2 \) and target functions \( f^*_{L,P} \in W^\alpha_2(T) \). Modulo \( \xi \), our rate is therefore asymptotically optimal in a minmax sense on \( T \). In addition, for \( \alpha > d \), the learning rates obtained for \( f^*_{L,P} \in B^{\alpha}_2,\infty(T) \) are again asymptotically optimal modulo \( \xi \) on \( T \).

### 5.2 Data-Dependent Parameter Selection for VP-SVMs

Note that in the previous theorems the choice of the regularization parameters \( \lambda_{n,1}, \ldots, \lambda_{n,m_n} \) and the kernel widths \( \gamma_{n,1}, \ldots, \gamma_{n,m_n} \) requires us to know the smoothness parameter \( \alpha \). Unfortunately, in practice, we usually do know neither this value nor its existence. In this subsection, we thus show that a training/validation approach similar to the one examined in (Steinwart and Christmann, 2008a, Chapters 6.5, 7.4, 8.2) and (Eberts and Steinwart, 2013) achieves the same rates adaptively, i.e. without knowing \( \alpha \). For this purpose, let \( \Lambda := (\Lambda_n) \) and \( \Gamma := (\Gamma_n) \) be sequences of finite subsets \( \Lambda_n \subset (0,r_n^d] \) and \( \Gamma_n \subset (0,r_n] \). For a data set \( D := ((x_1, y_1), \ldots, (x_n, y_n)) \), we define

\[
D_1 := ((x_1, y_1), \ldots, (x_l, y_l)),
\]

\[
D_2 := ((x_{l+1}, y_{l+1}), \ldots, (x_n, y_n)),
\]

where \( l := \left\lfloor \frac{n}{2} \right\rfloor + 1 \) and \( n \geq 4 \). We further split these sets in data sets

\[
D^{(1)}_j := \{ (x_i, y_i) \in D_1 : x_i \in A_j \}, \quad j \in \{1, \ldots, m_n\},
\]

and the loss \( \text{L}_T \) satisfies
\[ D_j^{(2)} := \{ (x_i, y_i) \in D_2 : x_i \in A_j \} , \quad j \in \{1, \ldots, m_n\}, \]

and define \( l_j := |D_j^{(1)}| \) for all \( j \in \{1, \ldots, m_n\} \) such that \( \sum_{j=1}^{m_n} l_j = l \). For every \( j \in \{1, \ldots, m_n\} \), we basically use \( D_j^{(1)} \) as a training set, i.e. based on \( D_1 \) in combination with the loss function \( L_j := 1_{A_j} L \) we compute SVM decision functions

\[ f_{D_j^{(1)}, \lambda_j, \gamma_j} := \arg \min_{f \in H_{\lambda_j}(A_j)} \lambda_j \| f \|_{H_{\lambda_j}(A_j)}^2 + \mathcal{R}_{L_j, D_1}(f) , \quad (\lambda_j, \gamma_j) \in \Lambda_n \times \Gamma_n. \]

Again, note that \( f_{D_j^{(1)}, \lambda_j, \gamma_j} = 0 \) if \( D_j^{(1)} = \emptyset \). Next, for each \( j \), we use \( D_2 \) in tandem with \( L_j \) (or essentially \( D_j^{(2)} \)) to determine a pair \( (\lambda_{D_2,j}, \gamma_{D_2,j}) \in \Lambda_n \times \Gamma_n \) such that

\[ \mathcal{R}_{L_j, D_2}(\widehat{f}_{D_j^{(1)}, \lambda_{D_2,j}, \gamma_{D_2,j}}) = \min_{(\lambda_j, \gamma_j) \in \Lambda_n \times \Gamma_n} \mathcal{R}_{L_j, D_2}(f_{D_j^{(1)}, \lambda_j, \gamma_j}). \]

Finally, combining the decision functions \( f_{D_j^{(1)}, \lambda_{D_2,j}, \gamma_{D_2,j}} \) for all \( j \in \{1, \ldots, m_n\} \), and defining \( \lambda_{D_2} := (\lambda_{D_2,1}, \ldots, \lambda_{D_2,m_n}) \) and \( \gamma_{D_2} := (\gamma_{D_2,1}, \ldots, \gamma_{D_2,m_n}) \), we obtain a function

\[ f_{D_1, \lambda_{D_2}, \gamma_{D_2}} := \sum_{j=1}^{m_n} f_{D_j^{(1)}, \lambda_{D_2,j}, \gamma_{D_2,j}} = \sum_{j=1}^{m_n} 1_{A_j} f_{D_j^{(1)}, \lambda_{D_2,j}, \gamma_{D_2,j}} , \]

and we call every learning method that produces these resulting decision functions \( f_{D_1, \lambda_{D_2}, \gamma_{D_2}} \) a training validation Voronoi partition support vector machine (TV-VP-SVM) w.r.t. \( \Lambda \times \Gamma \). Moreover, using (5) we have, for \( \lambda := (\lambda_1, \ldots, \lambda_m) \) and \( \gamma := (\gamma_1, \ldots, \gamma_m) \),

\[ \mathcal{R}_{L, D_2}(\widehat{f}_{D_1, \lambda_{D_2}, \gamma_{D_2}}) = \sum_{j=1}^{m_n} \mathcal{R}_{L_j, D_2}(\widehat{f}_{D_j^{(1)}, \lambda_{D_2,j}, \gamma_{D_2,j}}) \]

\[ = \sum_{j=1}^{m_n} \min_{(\lambda_j, \gamma_j) \in \Lambda_n \times \Gamma_n} \mathcal{R}_{L_j, D_2}(\widehat{f}_{D_j^{(1)}, \lambda_j, \gamma_j}) \]

\[ = \min_{(\lambda, \gamma) \in \Lambda_n \times \Gamma_n} \sum_{j=1}^{m_n} \mathcal{R}_{L_j, D_2}(\widehat{f}_{D_j^{(1)}, \lambda_j, \gamma_j}) \]

\[ = \min_{(\lambda, \gamma) \in \Lambda_n \times \Gamma_n} \mathcal{R}_{L, D_2}(\widehat{f}_{D_1, \lambda, \gamma}) , \]

where \( f_{D_1, \lambda, \gamma} := \sum_{j=1}^{m_n} f_{D_j^{(1)}, \lambda_j, \gamma_j} \) with \( (\lambda_j, \gamma_j) \in \Lambda_n \times \Gamma_n \) for all \( j \in \{1, \ldots, m_n\} \). In other words, the function \( \widehat{f}_{D_1, \lambda_{D_2}, \gamma_{D_2}} \) really minimizes the empirical risk \( \mathcal{R}_{L, D_2} \) w.r.t. the validation data set \( D_2 \) and the loss \( L \), where the minimum is taken over all functions \( \widehat{f}_{D_1, \lambda, \gamma} \) with \( (\lambda, \gamma) \in \Lambda_n \times \Gamma_n \) and \( m_n \).

The following theorem presents learning rates for the above described TV-VP-SVM.

**Theorem 10** Let \( r_n := cn^{-\frac{1}{3d}} \) with constants \( c > 0 \) and \( \beta > 1 \). Under the assumptions of Theorem 7 we fix sequences \( \Lambda := (\Lambda_n) \) and \( \Gamma := (\Gamma_n) \) of finite subsets \( \Lambda_n \subset (0, r_n^d) \).
and $\Gamma_n \subset (0, r_n]$ such that $\Lambda_n$ is an $(r_n^n \varepsilon_n)$-net of $(0, r_n^d]$ and $\Gamma_n$ is a $\delta_n$-net of $(0, r_n]$ with $\varepsilon_n \leq n^{-1}$ and $\delta_n \leq n^{-\frac{d}{2\alpha}}$. Furthermore, assume that the cardinalities $|\Lambda_n|$ and $|\Gamma_n|$ grow polynomially in $n$. Then, for all $\xi > 0$, $\tau \geq 1$, and $\alpha < \beta - \frac{1}{2} d$, the TV-VP-SVM producing the decision functions $f_{D_1, \Lambda_{D_2}, \gamma_{D_2}}$ satisfies

$$P^n (\mathcal{R}_{L,P}(f^*_{D_1, \Lambda_{D_2}, \gamma_{D_2}}) - \mathcal{R}_{L,P}^*) \leq \frac{1}{c} \tau n^{-\frac{d}{2\alpha + d + \xi}} \geq 1 - e^{-\tau},$$

where $c > 0$ is a constant independent of $n$ and $\tau$.

Once more, we can replace the assumption $f^*_{L,P} \in B_{2,\infty}^\alpha(\mathcal{P}_X|A_T)$ by $f^*_{L,P} \in B_{2,\infty}^\alpha(\mathcal{P}_X|T+\delta)$ for some $\delta > 0$ and obtain the same learning rate as in Theorem $10$ for all $n \geq n_\delta$ although $T+\delta$ is fixed for all $n \in \mathbb{N}$. Note that, if $\mathcal{P}_X$ has a Lebesgue density that is bounded away from 0 and $\infty$ and either $f^*_{L,P} \in W_2^{\alpha}(T)$ for $\alpha > d/2$ or $f^*_{L,P} \in B_{2,\infty}^\alpha(T)$ for $\alpha > d$, these learning rates are again asymptotically optimal modulo $\xi$ on $T$ in a minimax sense. However, the condition $\alpha < \beta - \frac{1}{2} d$ restricts the set of $\alpha$-values where we obtain learning rates adaptively. To be more precise, there is a trade-off between $\alpha$ and $\beta$. On the one hand, for small values of $\beta$ only a small number of possible values for $\alpha$ is covered. On the other hand, for larger values of $\beta$ the set of $\alpha$-values where we achieve rates adaptively is increasing but the savings in terms of computing time is decreasing.

6. Experimental Results

In the previous sections we defined VP-SVMs and derived local learning rates that are essentially optimal. So far, it is, however, not clear if the theoretical results suggesting a generalization performance not worse than that of global SVMs can be empirically confirmed and if the predicted advantages of VP-SVMs in terms of computational costs are preserved in practice. Note that the latter is not as obvious as it may seem to be, since VP-SVMs create an overhead when generating the working sets, and the working sets themselves do not need to be as balanced as we assumed in our naive analysis. In this section, we thus investigate the performance of VP-SVMs empirically. Namely, we carry out some experiments using the least squares loss with the objective to answer the subsequent questions:

1. How do different radii affect the performance of VP-SVMs? In particular, what is the impact on the training time and the VP-SVM’s test error?

2. How do the VP-LS-SVMs perform compared to the usual LS-SVMs in terms of the test error? What is the speed-up?

3. How does the performance of VP-SVMs compare to vanilla data splitting approaches such as random chunking ($RC\text{-SVM}$), in which the data set is divided into a random partition with equally sized subsets, and the final decision function is the average of the SVMs computed on each subset?

4. How does the VP-LS-SVM behave compared to the global LS-SVM, if the regression function has interruptions of its smoothness on zero sets?
Figure 6: Unscaled basic functions used to generate the artificial data sets.
To address these questions we utilize two kinds of data sets. On the one hand, to answer questions (1), (2), and (3), we examine the three real data sets COVTYPE, IJCNN1, and COD-RNA, which we obtained from LIBSVM’s homepage, see [Chang and Lin, 2011]. Table 1 summarizes some characteristics of these data sets. On the other hand, we generated several artificial data sets to address the last question. In order to prepare the data sets for the experiments, we edited the data sets from LIBSVM in the following manner. If for a real-world data set type the raw data set was already split, we first merged these sets so that we obtained one data set for each data set type. In a next step, we scaled the data componentwise such that all samples including labels lie in $[-1,1]^{d+1}$, where $d$ is the dimension of the input data. Finally, for each data set type, we generated random subsets that were afterwards randomly splitted into a training and a test data set. In this manner, we obtained, for each of the three LIBSVM data set types, training sets consisting of $n = 1000, 2500, 5000, 10000, 25000, 50000, 100000$ samples. Additionally, for the data sets COVTYPE and COD-RNA, we created training sets of sizes 250000 and 500000, and of sizes 250000 and 400000, respectively. The test data sets associated to the various training sets consist of $n_{\text{test}} = 50000$ random samples, apart from the training sets with $n_{\text{train}} \leq 5000$, for which we took $n_{\text{test}} = 10000$ test samples.

For the artificial data, we proceeded in a slightly different way. To generate the data sets we took as fundament the five regression functions pictured in Figure 6 and as noise, the sum of two uniform distributions on $[-c(x), c(x)]$, where $c(x) = \frac{1}{4} (3 \sin \left( \frac{\pi}{2} |x| \right) + 1)$ for the one-dimensional data sets and $c(x) = \frac{1}{4} \left( \sin \left( \frac{\pi}{4} (|x_1| + |x_2|) \right) + 1 \right)$ for the two-dimensional data sets. Thus, we produced five different types of artificial data sets, where the various data set types are named according to their type numbers as in Figure 6. Initially, we created two sets, namely one training and one test data set, each consisting of 10000 random input samples contained in $[-1,1]$ and $[-1,1]^2$, respectively. Then, for each artificial data set type, we determined the labels belonging to the input data as sum of the corresponding functional value and the noise and, finally, scaled all 20000 labels to $[-1,1]$. In a last step, we randomly built subsets of the training sets of size $n = 1000, 2500, 5000$. In this way, we altogether obtained, for each type of artificial data, four training data sets of size $n = 1000, 2500, 5000, 10000$ and a corresponding test data set of size $n_{\text{test}} = 10000$. Based on the test data sets the Bayes risks can be determined, see Table 2 where the Bayes risks are summarized for the various artificial data set types.

| Type I | Type II | Type III | Type IV | Type V |
|--------|---------|----------|---------|--------|
| Bayes risk | 0.0254 | 0.0137 | 0.0529 | 0.0083 | 0.0634 |

Table 2: Bayes risks w.r.t. test data sets for the various artificial data set types.
To minimize random effects, we repeated the experiment for each setting several times. Since experiments using large data sets entail long run times, we ran every experiment using a training set of size $n \geq 50000$ only three times while for training sets of size $n = 10000$, 25000 we performed ten repetitions and for smaller training sets, namely of size $n = 1000$, 2500, 5000, even 100 runs. An exception are the experiments using artificial training sets of size $n = 10000$, where we realized 100 repetitions for the sake of uniformity.

To approach the above problems we used the least squares loss and Gaussian kernels for all experiments. We implemented an LS-SVM-solver in C++ similar to the one in (Steinwart et al., 2011). Around this solver, we then built the routines for the VP-SVM and the RC-SVM. The compilation of the three programmes was executed by LINUX’s gcc. To produce comparable results in terms of run time, all real-world data experiments were realized by the same professional compute server equipped with four INTEL XEON E7-4830 (2.13 GHz) 8-core processor, 256 GB RAM, and a 64 bit version of Debian GNU/Linux 6.0.7. In order that we can indeed compare their run time, we used eight cores to pre-compute the kernel matrix and to evaluate the final decision functions on the test set, and one core for the subsequent solver for every real data experiment. Since the artificial data sets consist of at most 10000 samples we performed the according experiments by a computer equipped with one INTEL CORE i7-3770K (3.50 GHz) quad core processor, 16 GB RAM, and a 64 bit version of Debian GNU/Linux 6.0.7. For all artificial data experiments we used four cores to pre-compute the kernel matrix and to evaluate the final decision functions on the test set, and again one core for the solver. Even with pre-computed kernel matrices, our experiments on the real-world data altogether required almost 810 hours (approximately 34 days) for training and additionally almost 4 days for testing. Moreover, the experiments on the artificial data took nearly 43 hours for training and 168 minutes for testing. Without pre-computing the kernel matrices, e.g. by applying a standard caching approach, preliminary experiments suggested a multiplication of the training time, which would have rendered the experiments infeasible. Besides, our experiments will show that the available amount of RAM does not restrict the size of the training sets used by an VP-SVMs as severely as the ones used by LS-SVMs.

Let us quickly illustrate the routines of the VP- and the RC-SVM implemented around the LS-solver. For the VP-SVM, we first split the training set by Algorithm 1 in several working sets representing a Voronoi partition w.r.t. the user-specified radius. For this purpose, Algorithm 1 initially determines a cover of the input data applying the farthest first traversal algorithm, see (Dasgupta, 2008) and (Gonzalez, 1985) for more details. Note that this procedure induces working sets whose sizes may be considerably varying. In the case of an RC-SVM the working sets are created randomly, where their sizes are basically equal and the number of working sets is predefined by the user. Then, for the VP-SVM- as well as for the RC-SVM-algorithm the implemented LS-solver is applied on every working set. For each working set, we randomly split the respective training data set of size $n_{\text{train}}$ in five folds to apply 5-fold cross-validation in order to deal with the hyper-parameters $\lambda$ and $\gamma$ taken from an 10 by 10 grid geometrically generated in $[0.001 \cdot n_{\text{train}}^{-1}, 0.1] \times [0.5 \cdot n_{\text{train}}^{-1/d}, 10]$.

---

1. On this occasion, we would like to thank the Institute for Applied Analysis and Numerical Simulation of the University of Stuttgart, who placed the above mentioned compute server at our disposal and, thus, enabled us to realize our experiments on large real-world data sets. In consequence, the overall time available for our experiments was limited.
Finally, we obtain one decision function for each working set. To further process these decision functions the VP-SVM-algorithms picks exactly one decision function depending on the working set affiliation of the input value. On the contrary, the RC-SVM-algorithm simply takes the average of all the decision functions. Moreover, since we scaled the labels of all data sets to \([-1, 1]\), the computed decision functions are clipped at \(\pm 1\). Altogether, note that the usual LS-SVM-algorithm can be interpreted as special case of both the VP-SVM- and the RC-SVM-algorithm using one working set.

The experimental results for the three real data sets are summarized in Tables 3 to 6. These tables as well as Tables 7 to 11 containing the results for the experiments on the artificial data sets, can be found in the Appendix. In addition to the average run times of the training and test phases, these tables reflect inter alia the average test errors of the empirical SVM solutions. Additionally, the $L_2$-errors of the empirical SVM solutions, i.e. the value of

$$\sqrt{\frac{1}{n_{\text{test}}} \sum_{i=1}^{n_{\text{test}}} \left( \hat{f}_{D,\lambda,\gamma}(x_{\text{test}_i}) - f_{L,P}^*(x_{\text{test}_i}) \right)^2},$$

is determined for the artificial data sets. Moreover, note that some of the result tables are incomplete for very large real-world training data sets. In these cases, the kernel matrix, whose size depends on the training set size, did not fit into the RAM of the used computer and, thus, these experiments were left out.

### 6.1 Experiments on Real-World Data

In this subsection, we address questions (1), (2), and (3) by examining the results for the real-world data sets COVTYPE, COD-RNA, and IJCNN1, which are composed in Figures 7–9 and Tables 3–6.
6.1.1 Comparison of VP-SVMs Using Different Radii

In the following, we focus on the VP-SVMs using four different radii for the various real-world data sets, where the experimental results are summarized in Tables 3–6 as well as in Subfigures (d)–(f) of Figures 7–9. Examining the achieved training times for each data set type, we observe that, for increasing training set sizes, the radius that leads to the shortest training time typically decreases. More precisely, for the real data sets with sample size $n_{\text{train}} > 10,000$, the VP-SVMs using the smallest radius always train fastest, while for the data sets with $n_{\text{train}} \leq 10,000$, we can not make a uniform statement. Clearly, this finding is not surprising, since an SVM for a small data set trains considerably faster than an SVM for a large data set, such that splitting the large data set and running an SVM for each of the small data sets may altogether still be faster. Recall additionally the considerations in terms of the computational cost made in Section 2.

Let us now consider the VP-SVM results in terms of the realized test errors. As expected, for the real-world data sets, the test errors achieved by VP-SVMs with fixed radii decrease with increasing sample size of the used data sets except twice. In addition, for the real data sets COVTYPE and COD-RNA, the test errors decrease for increasing radii, cf. Figures 7(f) and 8(f). Here, however, the test errors achieved for the various radii get close to each other with increasing training sample size. The same behavior of the test errors appears for the IJCNN1 data set, though, for $n_{\text{train}} \geq 5,000$, both intermediate radii yield even smaller empirical risks than the largest radius, see Figure 9(f). In consequence, it is not straightforward to draw any conclusion on the relation between radius and test error. Nonetheless, we can say that VP-SVMs using small radii enjoy test errors that are never significantly larger and sometimes even smaller than those of VP-SVMs using the largest of the applied radii.

Besides, Tables 3 and 5 or Figures 7(e),(f) and 8(e),(f) contain an additional finding. For large data sets, namely for the COVTYPE data set of size 500,000 or for the COD-RNA data sets of size 250,000 and 400,000, the VP-SVMs with large radii did not yield any solution, since they failed due to the technical requirements caused by the used computer. More precisely, in these cases, there was at least one working set such that its kernel matrix did not fit into the RAM any more. Fortunately, the working sets of VP-SVMs using smaller radii were small enough such that we still received an outcome. What is more, these VP-SVMs yielded a better empirical risk in partially less training time compared to VP-SVMs with large radii and training sample sizes that still allowed a successful performance. That is, using a small radius for the VP-SVM and a training set that is oversized for VP-SVMs with a larger radius reduces the test error. More precisely, a large training set is crucial for a small empirical risk, where the possibly arising computational restrictions can be eluded by a VP-SVM with an appropriate radius.

All in all, localized SVMs using some small radius lead in substantially less training time to either negligible worse or even better test errors than VP-SVMs with large radii, if the training sample size is adequate, i.e. $n_{\text{train}} \geq 5,000$. In addition, the real data sets with a large sample size demonstrate that VP-SVMs with small radii are able to conquer the technical restrictions caused by the used computer and thus yield a better empirical risk than VP-SVMs with bigger radii can attain at all.
6.1.2 Comparing VP-SVMs with Global LS-SVMs

In the following, we compare the results of the VP-SVM using different radii to the standard LS-SVM. For the real-world data sets cod-rna and IJCNN1, the VP-SVMs, based on the largest of the applied radii, use only one working set. Thus, they coincide with the standard LS-SVM modulo different values generated by the random number generator. To verify this fact, we compare for the real data sets cod-rna and IJCNN1 the results of the VP-SVMs using one working set to the results of the standard LS-SVMs, see Tables 5 and 6. Here, we note that the LS-SVM test errors typically decrease with increasing training sample size. The same holds for the VP-SVMs using one working set. Moreover, the latter VP-SVM and the LS-SVMs perform equally well in terms of training time and empirical risk, however, for \( n_{\text{train}} \geq 25000 \) the VP-SVMs train slower.

In practice, a crucial problem is caused by the run time required by an algorithm. Hence, for each data set type, we compare hereafter the LS-SVM to the VP-SVM that trains fastest for the largest training data set. The required average training times and the average test errors of these SVMs are illustrated in Subfigures (g)–(i) of Figures 7–9. First, we notice that the selected VP-SVM uses the smallest of the applied radii for each data set type. Besides, the LS-SVM’s test errors are lower than those of the VP-SVMs. However, with increasing training set size the VP-SVM’s test errors get close to the ones of the LS-SVM. Moreover, for the IJCNN1 data set of size 100000, their empirical risks even coincide, cf. Figure 9(i). Besides, the VP-SVMs train considerably faster than the LS-SVMs. In particular, for \( n_{\text{train}} = 100000 \) the VP-SVMs require at most 8.5% of the LS-SVM’s training times, see Figures 7(h) and 8(h). Finally, recall that, for data sets of size \( n_{\text{train}} \geq 250000 \), the LS-SVM problem is infeasible with our computer, just like the VP-SVMs using the largest of the applied radii. In contrast, for \( n_{\text{train}} \geq 250000 \), VP-SVMs using small radii usually train considerably faster and achieve lower test errors than the LS-SVMs for \( n_{\text{train}} = 100000 \), cf. Figures 7(h)–(i) and 8(h)–(i).

Concluding, we have seen that the application of a VP-SVM using a small radius instead of the standard LS-SVM reduces the run time considerably entailing at most a negligible worsening or even an improvement of the test errors. Moreover, applying VP-SVMs with sufficiently small radii enables us to use large data sets and, thus, to elude the computational restrictions to sufficiently small data sets. As a result, handling really large data sets with the help of suitable VP-SVMs can lead to significantly improved test errors compared to an LS-SVM setting with memory constraints.

6.1.3 Comparison of VP-SVMs with RC-SVMs

First of all, let us investigate the RC-SVM results that are composed in Tables 4–6 as well as in Subfigures [a]–[c] of Figures 7–9. For the real data sets COVTYPE we considered ten, for the data sets COD-RNA nine, and for the data sets IJCNN1 eight different numbers of working sets. In each case, we started with an RC-SVM using one working set, i.e. with an RC-SVM that corresponds to the global LS-SVM modulo different values generated by the random generator, cf. Tables 5 and 6. Comparing for every data set the RC-SVMs using various numbers of working sets, we observe that the number of working sets, minimizing the RC-SVMs training time, increases in tandem with the sample size. Moreover, the RC-SVM using one working set never trains fastest compared to the other RC-SVMs using
more than one working set. Furthermore, the average test errors for the applied RC-SVMs usually decrease for a decreasing number of working sets and, hence, are minimized by the smallest possible number of working sets. Of course, all these findings are not surprising, since RC-SVMs are typically used to reduce the training time.

Let us now compare the results of VP- and RC-SVMs using roughly the same number of working sets, cf. Tables 3–6. Initially note that, even though we consider VP- and RC-SVM based on the same number of working sets, the RC-SVM working sets are about the same size whereas the VP-SVM working sets may have different sizes with a large range. That is, the VP-SVMs often deal with a few substantially larger working sets than the RC-SVMs. Consequently, the RC-SVMs often perform faster than the VP-SVMs, which require up to five times the RC-SVM’s training time for $n_{\text{train}} = 100\,000$. Contrarily, the average empirical risks achieved by the VP-SVMs are substantially lower than those of the RC-SVMs. Besides, in a few cases the VP-SVMs possess at least one working set which is oversized for the computer’s RAM, so that these VP-SVM problems are infeasible, whereas the comparable RC-SVMs avoid this conflict. Here, consider e.g. the RC-SVM using seven working sets and the VP-SVM with radius $r = 4$ for the covtype data set of size 500 000.

In Section 6.1.2 we compared for each data set type the LS-SVM with the VP-SVM that trains fastest for the largest training data set. Here, we additionally compare this VP-SVM to the RC-SVMs. To be able to draw a fair comparison in terms of the achieved test errors, we choose those RC-SVMs that train roughly as fast as the VP-SVM for the largest training set, i.e. the slowest RC-SVM training faster and the fastest RC-SVM training slower than the above VP-SVM. Subfigures (g)–(i) of Figures 7–9 illustrate the average training times and the average test errors of these RC-SVMs, the above VP-SVM, and the LS-SVM. Considering the RC-SVMs, the faster of the two requires for $n_{\text{train}} = 100\,000$ between 51% and 83% of the VP-SVMs training time and trains at most seven minutes faster than the VP-SVM. However, at least for $n_{\text{train}} \geq 5\,000$, both considered RC-SVMs induce substantially higher test errors than the VP- and LS-SVM. Finally, note that VP-SVMs for $n_{\text{train}} \geq 250\,000$ considerably outperform LS-SVMs for $n_{\text{train}} = 100\,000$, while RC-SVMs for $n_{\text{train}} \geq 250\,000$ lead to even worse test errors than the considered LS-SVMs.

Summarizing, we record that RC-SVMs using as few as possible working sets achieve the smallest RC-SVM test errors, however, those using more working sets perform faster. Furthermore, compared to VP-SVMs using roughly the same number of working sets as the RC-SVMs, the latter ones may learn faster though not as good as the VP-SVMs. Moreover, considering RC-SVMs that require roughly the same training time as the fastest VP-SVM, we saw that the RC-SVMs lead to much higher empirical risks. That is, if the required training time is a hard constraint, then the VP-SVM that satisfies this constraint achieves a better test error than a RC-SVM that also trains fast enough.

6.2 Experiments on Artificial Data

It remains to address the last question. To this end, we consider the results on the various artificial data sets, on the one hand, for the LS-SVM and, on the other hand, for the VP-SVM performing fastest for $n_{\text{train}} = 10\,000$. Moreover, for the sake of comparability, we again add to this selection the two RC-SVMs training roughly as fast as the VP-SVM for the artificial data sets of size $n_{\text{train}} = 10\,000$. However, for the artificial data sets of Type I,
Figure 7: Average training time and test error of LS-, VP-, and RC-SVMs for the real-world data covtype depending on the training set size $n_{\text{train}} = 1000, \ldots, 500\,000$. Subfigures (a)–(c) show the results for RC-SVMs using different numbers of working sets and Subfigures (d)–(f) illustrate the results for VP-SVMs using various radii. At the bottom, Subfigures (g)–(i) contain the average training times and the average test errors of the LS-SVM, one VP-SVM and two RC-SVMs. Here, the VP-SVM is the one which trains fastest for $n_{\text{train}} = 500\,000$ and the two RC-SVMs are those which achieve for $n_{\text{train}} = 500\,000$ roughly the same training time as the chosen VP-SVM. Here, note that, for $n_{\text{train}} = 10\,000$, the RC-SVM using one working set trains substantially slower than the LS-SVM, even though this RC-SVM is basically an LS-SVM. As a reason for this phenomenon, we conjecture that the used compute server was busy because of other influences.
Figure 8: Average training time and test error of LS-, VP-, and RC-SVMs for the real-world data cod-rna depending on the training set size $n_{\text{train}} = 1000, \ldots, 400\,000$. Subfigures (a)–(c) show the results for RC-SVMs using different numbers of working sets and Subfigures (d)–(f) illustrate the results for VP-SVMs using various radii. At the bottom, Subfigures (g)–(i) contain the average training times and the average test errors of the LS-SVM, one VP-SVM and two RC-SVMs. Here, the VP-SVM is the one which trains fastest for $n_{\text{train}} = 400\,000$ and the two RC-SVMs are those which achieve for $n_{\text{train}} = 400\,000$ roughly the same training time as the chosen VP-SVM.
Figure 9: Average training time and test error of LS-, VP-, and RC-SVMs for the real-world data IJCNN1 depending on the training set size $n_{\text{train}} = 1000, \ldots, 100,000$. Subfigures (a)-(c) show the results for RC-SVMs using different numbers of working sets and Subfigures (d)-(f) illustrate the results for VP-SVMs using various radii. At the bottom, Subfigures (g)-(i) contain the average training times and the average test errors of the LS-SVM, one VP-SVM and two RC-SVMs. Here, the VP-SVM is the one which trains fastest for $n_{\text{train}} = 100,000$ and the two RC-SVMs are those which achieve for $n_{\text{train}} = 100,000$ roughly the same training time as the chosen VP-SVM.
Figure 10: Average training time and test error of LS-, VP-, and RC-SVMs for the artificial data Type I depending on the training set size $n_{\text{train}} = 1000, \ldots, 10000$. Subfigures (a)-(c) show the results for RC-SVMs using different numbers of working sets and Subfigures (d)-(f) illustrate the results for VP-SVMs using various radii. At the bottom, Subfigures (g)-(i) contain the average training times and the average test errors of the LS-SVM, one VP-SVM and one RC-SVM. Here, the VP-SVM and the RC-SVM are those which train fastest for $n_{\text{train}} = 10000$. Note that in the case at hand none of the considered RC-SVMs performs faster than the fastest VP-SVM for $n_{\text{train}} = 10000$. 

(a) Average training time of the various RC-SVMs
(b) Average empirical risk of the various RC-SVMs
(c) Average empirical $L_2$-error of the various RC-SVMs
(d) Average training time of the various VP-SVMs
(e) Average empirical risk of the various VP-SVMs
(f) Average empirical $L_2$-error of the various VP-SVMs
(g) Average training time of LS-, VP-, and RC-SVMs
(h) Average empirical risk of LS-, VP-, and RC-SVMs
(i) Average empirical $L_2$-error of LS-, VP-, and RC-SVMs
Figure 11: Average training time and test error of LS-, VP-, and RC-SVMs for the artificial data Type II depending on the training set size $n_{\text{train}} = 1000, \ldots, 10000$. Subfigures (a)–(c) show the results for RC-SVMs using different numbers of working sets and Subfigures (d)–(f) illustrate the results for VP-SVMs using various radii. At the bottom, Subfigures (g)–(i) contain the average training times and the average test errors of the LS-SVM, one VP-SVM and one RC-SVM. Here, the VP-SVM and the RC-SVM are those which train fastest for $n_{\text{train}} = 10000$. Note that in the case at hand none of the considered RC-SVMs performs faster than the fastest VP-SVM for $n_{\text{train}} = 10000$. 

33
Figure 12: Average training time and test error of LS-, VP-, and RC-SVMs for the artificial data Type III depending on the training set size $n_{\text{train}} = 1000, \ldots, 10000$. Subfigures (a)–(c) show the results for RC-SVMs using different numbers of working sets and Subfigures (d)–(f) illustrate the results for VP-SVMs using various radii. At the bottom, Subfigures (g)–(i) contain the average training times and the average test errors of the LS-SVM, one VP-SVM and one RC-SVM. Here, the VP-SVM and the RC-SVM are those which train fastest for $n_{\text{train}} = 10000$. Note that in the case at hand none of the considered RC-SVMs performs faster than the fastest VP-SVM for $n_{\text{train}} = 10000$. 

(a) Average training time of the various RC-SVMs

(b) Average empirical risk of the various RC-SVMs

(c) Average empirical $L_2$-error of the various RC-SVMs

(d) Average training time of the various VP-SVMs

(e) Average empirical risk of the various VP-SVMs

(f) Average empirical $L_2$-error of the various VP-SVMs

(g) Average training time of LS-, VP-, and RC-SVMs

(h) Average empirical risk of LS-, VP-, and RC-SVMs

(i) Average empirical $L_2$-error of LS-, VP-, and RC-SVMs
Figure 13: Average training time and test error of LS-, VP-, and RC-SVMs for the artificial data Type IV depending on the training set size $n_{\text{train}} = 1000, \ldots, 10000$. Subfigures (a)–(c) show the results for RC-SVMs using different numbers of working sets and Subfigures (d)–(f) illustrate the results for VP-SVMs using various radii. At the bottom, Subfigures (g)–(i) contain the average training times and the average test errors of the LS-SVM, one VP-SVM and two RC-SVMs. Here, the VP-SVM is the one which trains fastest for $n_{\text{train}} = 10000$ and the two RC-SVMs are those which achieve for $n_{\text{train}} = 10000$ roughly the same training time as the chosen VP-SVM.
Figure 14: Average training time and test error of LS-, VP-, and RC-SVMs for the artificial data Type V depending on the training set size $n_{\text{train}} = 1000, \ldots, 10000$. Subfigures (a)–(c) show the results for RC-SVMs using different numbers of working sets and Subfigures (d)–(f) illustrate the results for VP-SVMs using various radii. At the bottom, Subfigures (g)–(i) contain the average training times and the average test errors of the LS-SVM, one VP-SVM and two RC-SVMs. Here, the VP-SVM is the one which trains fastest for $n_{\text{train}} = 10000$ and the two RC-SVMs are those which achieve for $n_{\text{train}} = 10000$ roughly the same training time as the chosen VP-SVM.
Optimal Learning Rates for Localized SVMs

Figure 15: Predictions for the artificial data sets of Type I, drawn from the step function in Figure 6(a) with noise depending on $x$. The left graphic shows the predictions for the data set of size $n_{\text{train}} = 1000$ and the right graphic for the data set of size $n_{\text{train}} = 10000$. Here, note that the VP-SVM solutions are not necessarily continuous, nevertheless we continuously connected its predicted values in the above plots.

II, and III, none of the executed RC-SVMs trained faster for $n_{\text{train}} = 10000$ than the VP-SVM with the smallest radius, so that we only consider one RC-SVM in these cases. The required average training times and average test errors of the selected SVMs are illustrated in Subfigures (g)–(i) of Figures 10–14 and summarized in Tables 7–11. Here, we note that, for the artificial data of Type I, II, and III, the VP-SVM using the smallest of the applied radii trains fastest for $n_{\text{train}} = 10000$, while for the artificial data of Type IV and V it is the VP-SVM using the second smallest radius.

Expectedly, we detect an evident improvement of the various average empirical SVM solutions using 10000 training samples instead of 1000 samples. Besides, the considered VP-SVM trains substantially faster than the standard LS-SVM with less than 11% of the LS-SVM’s training time for $n_{\text{train}} = 10000$. Additionally, the VP-SVM’s test errors are usually considerably lower than the test errors of the LS-SVM. Regarding the test errors of the RC-SVMs, we note that, in the majority of cases, they are higher than the VP-SVM’s and the LS-SVM’s test errors.

So far, we examined the behavior of LS-, VP-, and RC-SVMs in terms of training time and test error. Let us finally compare the three different kinds of SVMs w.r.t. their optical appearance. To this end, the average empirical SVM solutions are plotted in Figures 15–20 for the different artificial data sets of size $n_{\text{train}} = 1000$ and 10000. Here, note that, for the artificial data of Type IV and V, we do not consider both RC-SVMs training roughly as fast as the selected VP-SVM but only the one of the both RC-SVMs with the lower test error.

The observation that, for the artificial data of Type I, II, and III, the VP-SVMs perform best, is reinforced by the average empirical VP-SVM solutions illustrated in Figures 15–18. More precisely, Figure 15 shows that only the VP-SVMs exhaust the widths of the steps
Figure 16: Predictions for the artificial data sets of Type II, drawn from the cracked function in Figure 6(b) with noise depending on \( x \). The left graphic shows the predictions for the data set of size \( n_{\text{train}} = 1000 \) and the right graphic for the data set of size \( n_{\text{train}} = 10000 \).

Figure 17: Predictions for the artificial data sets of Type II. The left graphic shows the predictions for \( x \in [-0.55, -0.4] \) and the data set of size \( n_{\text{train}} = 1000 \), while the graphic on its right-hand side pictures the predictions for the same interval for \( x \) and the data set of size \( n_{\text{train}} = 10000 \). The two graphics on the right-hand side illustrate the predictions for \( x \in [0.3, 0.6] \), the upper one for the data set of size \( n_{\text{train}} = 1000 \) and the lower one for the data set of size \( n_{\text{train}} = 10000 \).

of \( f^*_L, P \) almost completely. Moreover, in Figure 16 the smoothness interruptions of \( f^*_L, P \) are again best illustrated by the VP-SVMs, which becomes even more evident in Figure 17. Besides, Figure 18 illustrates that the peaks of \( f^*_L, P \) are best reproduced by the VP-SVMs.
Considering the LS- and the RC-SVMs, we can not draw an universally valid conclusion, which one performs worse. In particular, Figures 15 and 16 show that, for the artificial data sets of Type I and II, both, the average empirical LS- and RC-SVM solutions, are not very well suited to the Bayes decision function. Considering the data sets of Type III, the LS-SVMs dominate the RC-SVMs in terms of the better test errors, though both kinds of SVMs do not reproduce the peaks of $f^*_L, P$, especially for small values of $|x|$. It remains to optically analyze the results of the two-dimensional data sets in the following. For the artificial data sets of Type IV, the VP-SVM using 10 000 training samples achieves the best test error. Moreover, ensuing the optical impression, this VP-SVM is the only one of the considered SVM types that reflects the circular steps of the Bayes decision function as in Figure 6(d), cf. Figure 19. Finally, for the data sets of Type V, it is always the LS-SVM which performs best in terms of the test errors, cf. Table 11. This observation is also substantiated optically. To be more precise, for $n_{\text{train}} = 1 000$, the uneven average empirical decision function induced by the VP-SVM (cf. Figure 20) shows that the RC-SVM even performs better than the VP-SVM. However, for $n_{\text{train}} = 10 000$, the VP-SVM results are substantially improved such that the RC-SVM is now outperformed by the VP-SVM.

Recapulatory, we realize that the VP-SVMs possess the most distinctive ability to handle smoothness interruptions of the Bayes decision function in most of our artificial data cases, especially if $n_{\text{train}} = 10 000$. For the sake of completeness, we point out that the worst performance was induced by the RC-SVMs in almost all cases, in particular for a training sample size amounting to 10 000.

6.3 Conclusions

Finally, we summarize the essential findings of the previous subsections, where we considered standard LS-SVMs and two kinds of localized SVMs, namely VP-SVMs and RC-SVMs. As
Figure 19: Predictions for the artificial data sets of Type IV, drawn from the circular step function in Figure 6(d) with noise independent of $x$. 

(a) LS-SVM using 1000 training samples
(b) LS-SVM using 10000 training samples
(c) VP-SVM using radius $r = 0.5$ and 1000 training samples
(d) VP-SVM using radius $r = 0.5$ and 10000 training samples
(e) RC-SVM using 20 working sets and 1000 training samples
(f) RC-SVM using 20 working sets and 10000 training samples
Figure 20: Predictions for the artificial data sets of Type V, drawn from the 2-dimensional Euclidean norm in Figure 6(e) with noise independent of $x$.
just analyzed in Subsection 6.2, VP-SVMs have the evident advantage that they manage smoothness interruptions of the Bayes decision function better than LS- and RC-SVMs.

The real-world data sets demonstrated that the RC-SVMs perform considerably worse than the LS-SVMs and the VP-SVMs, while the performance of VP-SVMs using small radii is improved for increasing sample sizes. To be more precise, VP-SVMs outperform LS-SVMs or at most leads to a negligible worsening compared to LS-SVMs for a fraction of the training time and without memory constraints on the large data sets. For very small data sets, however, LS-SVMs actually train faster than VP-SVMs and, hence, are preferable. What is more, for data sets of size $n_{\text{train}} \leq 2500$ all LS-SVMs require less than 9s to train, so that there are probably no reasons to apply a VP-SVM. Besides, really small training sample sizes involve considerably smaller working sets for a VP-SVM using a small radius, so that it is hard to find a well suited prediction.

Furthermore, despite a faster training procedure, a VP-SVM using a sufficiently small radius induces considerably lower test errors for sample sizes $n_{\text{train}} > 100000$ than a LS-SVM for training data sets that still enable computational feasibility.

7. Proofs

This section is dedicated to prove the results of the previous sections. We begin with the proof of Lemma 1 relating the radius $r$ of a cover $B_r(z_1), \ldots, B_r(z_m)$ of $X$ defined by (6) with the number $m$ of centers $z_1, \ldots, z_m$.

**Proof** [Proof of Lemma 1] First of all, let us recall the $m$-th entropy number of $X$ defined by

$$
\varepsilon_m(X) := \inf \left\{ \varepsilon > 0 : \exists z_1, \ldots, z_m \in X \text{ such that } X \subset \bigcup_{j=1}^{m} (z_j + \varepsilon B_{\ell^2}) \right\}.
$$

Since $X \subset cB_{\ell^2}$, the $m$-th entropy number of $X$ can be upper bounded by

$$
\varepsilon_m(X) \leq 2\varepsilon_m(cB_{\ell^2}) \leq 2c\varepsilon_m(B_{\ell^2}).
$$

Additionally, we know by [Carl and Stephani, 1990, Section 1.1] that

$$
m^{-\frac{d}{4}} \leq \varepsilon_m(B_{\ell^2}) \leq 4m^{-\frac{d}{4}},
$$

so that we can find a cover $(B_j)_{j=1,\ldots,m}$ of $X \subset cB_{\ell^2}$ satisfying

$$
r \leq 8cm^{-\frac{d}{4}}.
$$

7.1 Proofs of Section 3

In Section 3 we presented a lemma that related the risk w.r.t. the loss $L$ to the risk w.r.t. the restricted loss $L_j$ and also transferred this result to the excess risk. Hereafter, the proof of this lemma can be found.
Proof [Proof of Lemma 4] Simple transformations using $A \cup B = X$ and $A \cap B = \emptyset$ show

$$\mathcal{R}_{L,P}(f) = \int_{X \times Y} L(x,y,1_A(x)f_A(x) + 1_B(x)f_B(x)) \, d\mathcal{P}(x,y)$$

$$= \int_{X \times Y} 1_A(x)L(x,y,f_A(x)) + 1_B(x)L(x,y,f_B(x)) \, d\mathcal{P}(x,y)$$

$$= \mathcal{R}_{L_A,P}(f_a) + \mathcal{R}_{L_B,P}(f_B).$$

The second assertion follows immediately.

To derive the new oracle inequality of Theorem 5 we first have to relate the entropy numbers of $H_j$, $j \in \{1, \ldots, m\}$, to those of $H$. To this end, we consider a similar concept to entropy numbers, namely covering numbers, cf. (Györfi et al., 2002, Definition 9.3) or (Steinwart and Christmann, 2008a, Definition 6.19).

Definition 11 Let $(T, d)$ be a metric space and $\varepsilon > 0$. A subset $S \subset T$ is called an $\varepsilon$-net of $T$ if for all $t \in T$ there exists an $s \in S$ such that $d(s,t) \leq \varepsilon$. Furthermore, we define the $\varepsilon$-covering number of $T$ by

$$\mathcal{N}(T,d,\varepsilon) := \inf \left\{ n \geq 1 : \exists s_1, \ldots, s_n \in T \text{ such that } T \subset \bigcup_{i=1}^{n} B_d(s_i,\varepsilon) \right\},$$

where $\inf \emptyset := \infty$ and $B_d(s,\varepsilon) := \{ t \in T : d(t, s) \leq \varepsilon \}$.

Note that an upper bound on entropy numbers involves a bound on covering numbers. To be more precise, for a metric space $(T, d)$ and constants $a > 0$ and $q > 0$, the implication

$$e_i(T,d) \leq a_i^{-1/q}, \quad i \geq 1 \quad \Rightarrow \quad \ln \mathcal{N}(T,d,\varepsilon) \leq \ln(4) \left( \frac{a}{\varepsilon} \right)^q, \quad \forall \varepsilon > 0$$

holds by (Steinwart and Christmann, 2008a, Lemma 6.21). Additionally, (Steinwart and Christmann, 2008a, Exercise 6.8) yields the opposite implication, namely

$$\ln \mathcal{N}(T,d,\varepsilon) < \left( \frac{a}{\varepsilon} \right)^q, \quad \varepsilon > 0 \quad \Rightarrow \quad e_i(T,d) \leq 3^{1/q} a_i^{-1/q}, \quad \forall i \geq 1.$$  

Recall that we pursue the target to estimate $e_i(\text{id} : H \to L_2(\mathcal{P}_X))$. In fact, the equivalence of entropy and covering numbers enables us to estimate the covering number $\mathcal{N}(B_H, \| \cdot \|_{L_2(\mathcal{P}_X)}, \varepsilon)$ of $H$ instead.

Lemma 12 Let $\nu$ be a distribution on $X$ and $A, B \subset X$ with $A \cap B = \emptyset$. Moreover, let $H_A$ and $H_B$ be RKHSs on $A$ and $B$ that are embedded into $L_2(\nu|_A)$ and $L_2(\nu|_B)$, respectively. Let the extended RKHSs $\hat{H}_A$ and $\hat{H}_B$ be defined as in Lemma 4 and denote their direct sum by $H$ as in (13), where the norm is given by (14) with $\lambda_A, \lambda_B > 0$. Then, for the $\varepsilon$-covering number of $H$ w.r.t. $\| \cdot \|_{L_2(\nu)}$, we have

$$\mathcal{N}(B_H, \| \cdot \|_{L_2(\nu)}, \varepsilon) \leq \mathcal{N}(\lambda_A^{-1/2} B_{\hat{H}_A}, \| \cdot \|_{L_2(\nu|_A)}, \varepsilon_A) \cdot \mathcal{N}(\lambda_B^{-1/2} B_{\hat{H}_B}, \| \cdot \|_{L_2(\nu|_B)}, \varepsilon_B),$$

where $\varepsilon_A, \varepsilon_B > 0$ and $\varepsilon := \sqrt{\varepsilon_A^2 + \varepsilon_B^2}$. 

43
Proof First of all, we assume that there exist \(a, b \in \mathbb{N}\) and functions \(\hat{f}_1, \ldots, \hat{f}_a \in \lambda^{-\frac{1}{2}} B_{\hat{H}_A}\) and \(\hat{h}_1, \ldots, \hat{h}_b \in \lambda^{-\frac{1}{2}} B_{\hat{H}_B}\) such that \(\{\hat{f}_1, \ldots, \hat{f}_a\}\) is an \(\epsilon_A\)-cover of \(\lambda^{-\frac{1}{2}} B_{\hat{H}_A}\) w.r.t. \(\| \cdot \|_{L_2(\nu_A)}\), \(\{\hat{h}_1, \ldots, \hat{h}_b\}\) is an \(\epsilon_B\)-cover of \(\lambda^{-\frac{1}{2}} B_{\hat{H}_B}\) w.r.t. \(\| \cdot \|_{L_2(\nu_B)}\),

\[a = \mathcal{N}(\lambda^{-\frac{1}{2}} B_{\hat{H}_A}, \| \cdot \|_{L_2(\nu_A)}, \epsilon_A) \quad \text{and} \quad b = \mathcal{N}(\lambda^{-\frac{1}{2}} B_{\hat{H}_B}, \| \cdot \|_{L_2(\nu_B)}, \epsilon_B).\]

That is, for every function \(\hat{g}_A \in \lambda^{-\frac{1}{2}} B_{\hat{H}_A}\), there exists an \(i_A \in \{1, \ldots, a\}\) such that

\[
\left\| \hat{g}_A - \hat{f}_{i_A} \right\|_{L_2(\nu_A)} \leq \epsilon_A,
\]

and for every function \(\hat{g}_B \in \lambda^{-\frac{1}{2}} B_{\hat{H}_B}\), there exists an \(i_B \in \{1, \ldots, b\}\) such that

\[
\left\| \hat{g}_B - \hat{h}_{i_B} \right\|_{L_2(\nu_B)} \leq \epsilon_B.
\]

Let us now consider an arbitrary function \(g \in B_H\). Then there exists an \(\hat{g}_A \in \lambda^{-\frac{1}{2}} B_{\hat{H}_A}\) and an \(\hat{g}_B \in \lambda^{-\frac{1}{2}} B_{\hat{H}_B}\) such that \(g = \hat{g}_A + \hat{g}_B\). Together with (28) and (29), this implies

\[
\left\| g - (\hat{f}_{i_A} + \hat{h}_{i_B}) \right\|_{L_2(\nu)}^2 = \left\| (\hat{g}_A - \hat{f}_{i_A}) + (\hat{g}_B - \hat{h}_{i_B}) \right\|_{L_2(\nu)}^2 = \left\| \hat{g}_A - \hat{f}_{i_A} \right\|_{L_2(\nu_A)}^2 + \left\| \hat{g}_B - \hat{h}_{i_B} \right\|_{L_2(\nu_B)}^2 \leq \epsilon_A^2 + \epsilon_B^2 =: \epsilon^2.
\]

With this, we know that

\[
\left\{ \hat{f}_{i_A} + \hat{h}_{i_B} : \hat{f}_{i_A} \in \{\hat{f}_1, \ldots, \hat{f}_a\} \text{ and } \hat{h}_{i_B} \in \{\hat{h}_1, \ldots, \hat{h}_b\} \right\}
\]

is an \(\epsilon\)-net of \(H\) w.r.t. \(\| \cdot \|_{L_2(\nu)}\). Concerning the \(\epsilon\)-covering number of \(H\), this finally implies

\[
\mathcal{N}(B_H, \| \cdot \|_{L_2(\nu)}, \epsilon) \leq a \cdot b = \mathcal{N}(\lambda^{-\frac{1}{2}} B_{\hat{H}_A}, \| \cdot \|_{L_2(\nu_A)}, \epsilon_A) \cdot \mathcal{N}(\lambda^{-\frac{1}{2}} B_{\hat{H}_B}, \| \cdot \|_{L_2(\nu_B)}, \epsilon_B).
\]

\[
\Box
\]

Based on Lemma 12, the following theorem relates entropy numbers of \(H_A\) and \(H_B\) to those of \(H\).

**Theorem 13** Let \(P_X\) be a distribution on \(X\) and \(A_1, \ldots, A_m \subseteq X\) be pairwise disjoint. Moreover, we assume (II) with weights \(\lambda_1, \ldots, \lambda_m > 0\). In addition, assume that there exist constants \(p \in (0, 1)\) and \(a_j > 0, j \in \{1, \ldots, m\}\), such that for every \(j \in \{1, \ldots, m\}\)

\[
e_i(id : H_j \to L_2(P_{X|A_j})) \leq a_j i^{-\frac{j}{2p}}, \quad i \geq 1.
\]

(30)
Then we have
\[
e_i(\text{id} : H \rightarrow L_2(P_X)) \leq 2\sqrt{m} \left(3\ln(4) \sum_{j=1}^{m} \lambda_j^{-p} a_j^{2p} \right)^{\frac{1}{2p}} i^{-\frac{1}{2p}}, \quad i \geq 1,
\]
and, for the average entropy numbers,
\[
\mathbb{E}_{D_X \sim P_X} e_i(\text{id} : H \rightarrow L_2(D_X)) \leq c_p \sqrt{m} \left(\sum_{j=1}^{m} \lambda_j^{-p} a_j^{2p} \right)^{\frac{1}{2p}} i^{-\frac{1}{2p}}, \quad i, n \geq 1.
\]

**Proof** [Proof of Theorem 13] First of all, note that the restriction operator \( I : \hat{B}_H \rightarrow B_H \) with \( \hat{f} = f \) is an isometric isomorphism. Together with (Steinwart and Christmann, 2008a, (A.36)) and assumption (30), this yields
\[
e_i(\lambda_j^{-\frac{1}{2}} B_H, L_2(P_{X|A_j})) = 2\lambda_j^{-\frac{1}{2}} e_i(B_H, L_2(P_{X|A_j})) \leq 2\lambda_j^{-\frac{1}{2}} \|I : \hat{B}_H \rightarrow B_H\| e_i(B_H, L_2(P_{X|A_j})) \leq 2\lambda_j^{-\frac{1}{2}} a_j^{-\frac{1}{2p}}.
\]
Furthermore, we know by (26) that
\[
\ln N(\lambda_j^{-\frac{1}{2}} B_H, \|\cdot\|_{L_2(P_{X|A_j})}, \varepsilon) \leq \ln(4) \left(2\lambda_j^{-\frac{1}{2}} a_j\right)^{2p} \varepsilon^{-2p}
\]
holds for all \( \varepsilon > 0 \). With this and \( \varepsilon_j := \frac{\varepsilon}{\sqrt{m}} \) for every \( j \in \{1, \ldots, m\} \), Lemma 12 implies
\[
\ln N(B_H, \|\cdot\|_{L_2(P_X)}, \varepsilon) \leq \ln \left(\prod_{j=1}^{m} N(\lambda_j^{-\frac{1}{2}} B_H, \|\cdot\|_{L_2(P_{X|A_j})}, \varepsilon_j)\right)
\]
\[
= \sum_{j=1}^{m} \ln N(\lambda_j^{-\frac{1}{2}} B_H, \|\cdot\|_{L_2(P_{X|A_j})}, \frac{\varepsilon}{\sqrt{m}})
\]
\[
\leq \sum_{j=1}^{m} \ln(4) \left(2\lambda_j^{-\frac{1}{2}} a_j\right)^{2p} \left(\frac{\sqrt{m}}{\varepsilon}\right)^{2p}
\]
\[
= \left(2\ln(4) \frac{1}{2p} \sqrt{m} \left(\sum_{j=1}^{m} \lambda_j^{-p} a_j^{2p}\right)^{\frac{1}{2p}} i^{-\frac{1}{2p}}\right)^{2p} \varepsilon^{-2p}.
\]
Using (27), the latter bound for the covering number of \( B_H \) finally implies the following entropy estimate
\[
e_i(\text{id} : H \rightarrow L_2(P_X)) \leq 3\frac{1}{2p} \left(2\ln(4) \frac{1}{2p} \sqrt{m} \left(\sum_{j=1}^{m} \lambda_j^{-p} a_j^{2p}\right)^{\frac{1}{2p}}\right) i^{-\frac{1}{2p}}
\]
\[ \leq 2 \left( 3 \ln(4) \right)^{\frac{1}{2p}} \sqrt{m} \left( \sum_{j=1}^{m} \lambda_j^{-p} a_j^{2p} \right)^{\frac{1}{2p}} \cdot i^{-\frac{1}{2p}}. \]

The second assertion immediately follows by (Steinwart and Christmann, 2008a, Corollary 7.31).

Applying Theorem 13, we now prove Theorem 5 and thus an oracle inequality for VP-SVMs using an ordinary type of losses.

**Proof** [Proof of Theorem 5] Since \( H_1, \ldots, H_m \) are separable RKHS of measurable kernels \( k_1, \ldots, k_m \), \( H \) is a separable RKHS and its kernel \( k \) is measurable, too. Furthermore, Theorem 13 yields

\[ \mathbb{E}_{D_X \sim P_X} e_i(\text{id} : H \rightarrow L_2(D_X)) \leq c_p \sqrt{m} \left( \sum_{j=1}^{m} \lambda_j^{-p} a_j^{2p} \right)^{\frac{1}{2p}} i^{-\frac{1}{2p}}, \quad i, n \geq 1. \]

That is, we can apply (Steinwart and Christmann, 2008a, Theorem 7.23) for a regularization parameter \( \lambda = 1 \) and, for all fixed \( \tau > 0 \) and \( \lambda_j > 0, j \in \{1, \ldots, m\} \), we obtain

\[
\sum_{j=1}^{m} \lambda_j \| f_{D_j,\lambda_j} \|_{H_j}^2 + \mathcal{R}_{L_j,P}(\hat{f}_{D_j,\lambda}) - \mathcal{R}^*_{L_j,P} \\
= \| f_{D,\lambda} \|_{H}^2 + \mathcal{R}_{L_j,P}(\hat{f}_{D,\lambda}) - \mathcal{R}^*_{L_j,P} \\
\leq 9 \left( \| f_0 \|_{H}^2 + \mathcal{R}_{L_j,P}(f_0) - \mathcal{R}^*_{L_j,P} \right) + C \left( a^{2p} n^{-1} \right)^{\frac{1}{2-p-\theta+\phi p}} + 3 \left( \frac{72V \tau}{n} \right)^{\frac{1}{2-\phi}} + \frac{15B_0 \tau}{n} \\
\leq 9 \left( \sum_{j=1}^{m} \lambda_j \| f_{D_j,\lambda_j} \|_{H_j}^2 + \mathcal{R}_{L_j,P}(f_0) - \mathcal{R}^*_{L_j,P} \right) + C \left( a^{2p} n^{-1} \right)^{\frac{1}{2-p-\theta+\phi p}} + 3 \left( \frac{72V \tau}{n} \right)^{\frac{1}{2-\phi}} + \frac{15B_0 \tau}{n}
\]

with probability \( P^n \) not less than \( 1 - 3e^{-\tau} \), where \( C > 0 \) is the constant of (Steinwart and Christmann, 2008a, Theorem 7.23) only depending on \( p, M, V, \theta, \) and \( B \). Moreover,

\[ a := \max \left\{ c_p \sqrt{m} \left( \sum_{j=1}^{m} \lambda_j^{-p} a_j^{2p} \right)^{\frac{1}{2p}}, B \right\}, \]

where we need \( a \geq B \) since it is a condition of (Steinwart and Christmann, 2008a, Theorem 7.23).

### 7.2 Proofs Related to the Entropy Estimates of Section 4

In this subsection, just as in Section 4, we focus on Gaussian RBF kernels and the associated RKHSs. To be more precise, we derive a bound for the entropy numbers of \( H_\gamma(A) \), where \( \gamma > 0 \) and \( A \subset \mathbb{R}^d \) with \( \hat{A} \neq \emptyset \).
Proof [Proof of Theorem 6] First of all, we consider the commutative diagram

$$
\begin{array}{ccc}
H_\gamma(A) & \xrightarrow{\text{id}} & L_2(P_{X|A}) \\
\downarrow \mathcal{I}^{-1}_B \circ \mathcal{I}_A & & \downarrow \text{id} \\
H_\gamma(B) & \xrightarrow{\text{id}} & \ell_\infty(B)
\end{array}
$$

where the extension operator $\mathcal{I}_A : H_\gamma(A) \to H_\gamma(\mathbb{R}^d)$ and the restriction operator $\mathcal{I}^{-1}_B : H_\gamma(\mathbb{R}^d) \to H_\gamma(B)$ given by (Steinwart and Christmann, 2008a, Corollary 4.43) are isometric isomorphisms, so that $\|\mathcal{I}^{-1}_B \circ \mathcal{I}_A : H_\gamma(A) \to H_\gamma(B)\| = 1$. Furthermore, for $f \in \ell_\infty(B)$, we have

$$
\|f\|_{L_2(P_{X|A})} = \left( \int_X \mathbb{1}_A f(x)^2 dP_X(x) \right)^{1/2} \leq \|f\|_\infty \left( \int_X \mathbb{1}_A dP_X(x) \right)^{1/2} = \sqrt{P_X(A)} \|f\|_\infty,
$$

i.e. $\ell_\infty(B) \to L_2(P_{X|A}) \leq \sqrt{P_X(A)}$. Together with (Steinwart and Christmann, 2008a, (A.38) and (A.39)) as well as (Steinwart and Christmann, 2008a, Theorem 6.27), we obtain for all $i \geq 1$

$$
e_i(\text{id} : H_\gamma(A) \to L_2(P_{X|A})) \\
\leq \|\mathcal{I}^{-1}_B \circ \mathcal{I}_A : H_\gamma(A) \to H_\gamma(B)\| \cdot e_i(\text{id} : H_\gamma(B) \to \ell_\infty(B)) \cdot \|\text{id} : \ell_\infty(B) \to L_2(P_{X|A})\| \\
\leq \sqrt{P_X(A)} c_{m,d} r^m \gamma^{-m} i^{-d/p},
$$

where $m \geq 1$ is an arbitrary integer and $c_{m,d}$ a positive constant. For $p \in (0, 1)$, the choice $m = \left\lfloor \frac{d}{2p} \right\rfloor$ finally yields

$$
e_i(\text{id} : H_\gamma(A) \to L_2(P_{X|A})) \leq \sqrt{P_X(A)} c_{m,d} r^m \gamma^{-m} i^{-d/p} \leq c_p \sqrt{P_X(A)} r^{d+2p} \gamma^{-d+2p} i^{-1/p},
$$

7.3 Proofs Related to the Least Squares VP-SVMs

In this subsection, we prove the results that are linked with the least squares loss, i.e. the results of Section 5. Before we elaborate on the oracle inequality for VP-SVMs using the least squares loss as well as RKHSs of Gaussian kernels, we have to examine the excess risk

$$
\mathcal{R}_{L_{\mathcal{J},p}}(f_0) - \mathcal{R}^*_L, p = \|f_0 - f^*_L, p\|^2_{L_2(P_{X|A})}.
$$

Let us begin by writing for fixed $\gamma_j > 0$

$$
K_j : \mathbb{R}^d \to \mathbb{R}, \quad x \mapsto \sum_{\ell=1}^s \left( \frac{s}{\ell} \right) (-1)^{1-\ell} \left( \frac{2}{\ell^2 \gamma_j^2 \pi} \right)^{\ell/2} \exp \left( -\frac{2\|x\|_2^2}{\ell^2 \gamma_j^2} \right),
$$

47
and choosing \( f_0 := \sum_{j=1}^m 1_{A_j} \cdot (K_j \ast f_{L_p}) \). Then (31) can be estimated with the help of the following theorem, which is together with its proof basically a modification of (Eberts and Steinwart, 2013, Theorem 2.2). Indeed, the proofs proceed mainly identically. Note that we use the notation

\[
\gamma_{\text{max}} := \max\{\gamma_1, \ldots, \gamma_m\} \quad \text{and} \quad \gamma_{\text{min}} := \min\{\gamma_1, \ldots, \gamma_m\}
\]

in the following theorem and the associated proof.

**Theorem 14** Let us fix some \( q \in [1, \infty) \). Assume that \( \nu \) is a finite measure on \( \mathbb{R}^d \) with \( \text{supp} \nu =: X \subset \mathbb{R}^d \) and let \((A_j)_{j=1, \ldots, m}\) be a partition of \( X \). Furthermore, let \( f : \mathbb{R}^d \to \mathbb{R} \) be such that \( f \in B_{q,\infty}^{\alpha}(\nu) \) for some \( \alpha \geq 1 \). For the functions \( K_j : \mathbb{R}^d \to \mathbb{R}, \ j \in \{1, \ldots, m\} \), defined by (32), where \( s := [\alpha] + 1 \) and \( \gamma_1, \ldots, \gamma_m > 0 \), we then have

\[
\left\| \sum_{j=1}^m 1_{A_j} \cdot (K_j \ast f) - f \right\|_{L_q(\nu)}^q \leq C_{\alpha,q} \left( \frac{\gamma_{\text{max}}}{\gamma_{\text{min}}} \right)^d \gamma_{\text{max}}^{\frac{q}{\gamma_{\text{max}}}},
\]

where \( C_{\alpha,q} := \| f \|_{B_{q,\infty}^{\alpha}(\nu)} \left( \frac{\gamma_{\text{max}}}{\gamma_{\text{min}}} \right)^d \gamma_{\text{max}}^{\frac{q}{\gamma_{\text{max}}}} \).

**Proof** In the following, we write \( J := \{1, \ldots, m\} \). To show

\[
\left\| \sum_{j \in J} 1_{A_j} \cdot (K_j \ast f) - f \right\|_{L_q(\nu)}^q \leq \left\| f \right\|_{B_{q,\infty}^{\alpha}(\nu)} \left( \frac{\gamma_{\text{max}}}{\gamma_{\text{min}}} \right)^d \gamma_{\text{max}}^{\frac{q}{\gamma_{\text{max}}}},
\]

we have to proceed in a similar way as in the proof of (Eberts and Steinwart, 2013, Theorem 2.2). First of all, we use the translation invariance of the Lebesgue measure and \( \exp(-\|u\|_2^2) = \exp\left(-\|u\|_2^2\right) \) (\( u \in \mathbb{R}^d \)) to obtain, for \( x \in X \) and \( j \in J \),

\[
K_j \ast f (x) = \int_{\mathbb{R}^d} \sum_{\ell=1}^s \left( \frac{s}{\ell} \right) (-1)^{1-\ell} \frac{1}{\ell^d} \left( \frac{2}{2 \gamma_j^2 \pi} \right) \left( \frac{4}{\ell^2 \gamma_j^2} \right) \exp \left( -\frac{2\|x-t\|_2^2}{\ell^2 \gamma_j^2} \right) f(t) \, dt
\]

\[
= \int_{\mathbb{R}^d} \left( \frac{2}{2 \gamma_j^2 \pi} \right)^{\frac{d}{2}} \exp \left( -\frac{2\|x\|_2^2}{\gamma_j^2} \right) \left( \sum_{\ell=1}^s \left( \frac{s}{\ell} \right) (-1)^{1-\ell} f(x + \ell h) \right) \, dh.
\]

With this we can derive, for \( q \geq 1 \),

\[
\left\| \sum_{j \in J} 1_{A_j} \cdot (K_j \ast f) - f \right\|_{L_q(\nu)}^q = \int_{\mathbb{R}^d} \left\| \sum_{j \in J} 1_{A_j}(x) (K_j \ast f)(x) - f(x) \right\|_{L_q(\nu)}^q \, d\nu(x)
\]

\[
\leq \int_{\mathbb{R}^d} \left( \sum_{j \in J} 1_{A_j}(x) |K_j \ast f(x) - f(x)| \right)^q \, d\nu(x)
\]

48
Then Hölder's inequality and \( \int_{\mathbb{R}^d} \exp\left(-2\gamma^{-2}\|h\|_2^2\right) \, dh = \left(\frac{\gamma^2\pi}{2}\right)^{d/2} \) yield, for \( q > 1 \),

\[
\left\| \sum_{j \in J} \mathbb{1}_{A_j} \cdot (K_j * f) - f \right\|_{L_q(\nu)}^q \\
\leq \sum_{j \in J} \int_{\mathbb{R}^d} \mathbb{1}_{A_j}(x) \left( \int_{\mathbb{R}^d} \left( \frac{2}{\gamma_j^2 \pi} \right)^{d/2} \exp\left(-\frac{2\|h\|_2^2}{\gamma_j^2}\right) \, dh \right)^{\frac{s+1}{s}} \left( \int_{\mathbb{R}^d} \left( \frac{2}{\gamma_j^2 \pi} \right)^{d/2} \exp\left(-\frac{2\|h\|_2^2}{\gamma_j^2}\right) |\Delta^s_h(f,x)|^q \, dh \right) \, d\nu(x)
\]

\[
= \sum_{j \in J} \int_{\mathbb{R}^d} \mathbb{1}_{A_j}(x) \int_{\mathbb{R}^d} \left( \frac{2}{\gamma_j^2 \pi} \right)^{d/2} \exp\left(-\frac{2\|h\|_2^2}{\gamma_j^2}\right) |\Delta^s_h(f,x)|^q \, dh \, d\nu(x)
\]

\[
\leq \sum_{j \in J} \int_{\mathbb{R}^d} \mathbb{1}_{A_j}(x) \int_{\mathbb{R}^d} \left( \frac{2}{\gamma_j^2 \pi} \right)^{d/2} \exp\left(-\frac{2\|h\|_2^2}{\gamma_j^2}\right) \|\Delta^s_h(f,\cdot)\|_{L_q(\nu)}^q \, dh \, d\nu(x)
\]

\[
= \int_{\mathbb{R}^d} \left( \frac{2}{\gamma_{\min}^2 \pi} \right)^{d/2} \exp\left(-\frac{2\|h\|_2^2}{\gamma_{\max}^2}\right) \|\Delta^s_h(f,\cdot)\|_{L_q(\nu)}^q \, dh
\]

\[
\leq \int_{\mathbb{R}^d} \left( \frac{2}{\gamma_{\min}^2 \pi} \right)^{d/2} \exp\left(-\frac{2\|h\|_2^2}{\gamma_{\max}^2}\right) \omega_{s,L_q(\nu)}(f,\|h\|_2) \, dh.
\]
Moreover, for \( q = 1 \), we have

\[
\left\| \sum_{j \in J} \mathbb{1}_{A_j} \cdot (K_j * f) - f \right\|_{L_1(\nu)} \leq \sum_{j \in J} \int_{\mathbb{R}^d} \mathbb{1}_{A_j}(x) \int_{\mathbb{R}^d} \left( \frac{2}{\gamma_j^2} \right)^{\frac{d}{2}} \exp \left( -\frac{2\|h\|_2^2}{\gamma_j^2} \right) |\Delta_h^j (f, x)| \, dh \, d\nu(x)
\]

\[
\leq \int_{\mathbb{R}^d} \left( \frac{2}{\pi \gamma_{\min}^2} \right)^{\frac{d}{2}} \exp \left( -\frac{2\|h\|_2^2}{\gamma_{\max}^2} \right) \int_{\mathbb{R}^d} \sum_{j \in J} \mathbb{1}_{A_j}(x) |\Delta_h^j (f, x)| \, d\nu(x) \, dh
\]

\[
\leq \int_{\mathbb{R}^d} \left( \frac{2}{\pi \gamma_{\min}^2} \right)^{\frac{d}{2}} \exp \left( -\frac{2\|h\|_2^2}{\gamma_{\max}^2} \right) \omega_{s, L_1(\nu)} (\|h\|_2) \, dh.
\]

Consequently, we can proceed in the same way for all \( q \geq 1 \). To this end, note that the assumption \( f \in B_{q, \infty}^\alpha (\nu) \) implies \( \omega_{s, L_q(\nu)} (\|h\|_2) \leq \|f\|_{B_{q, \infty}^\alpha} t^\alpha \) for \( t > 0 \). The latter together with Hölder’s inequality yields

\[
\left\| \sum_{j \in J} \mathbb{1}_{A_j} \cdot (K_j * f) - f \right\|_{L_q(\nu)}^q \leq \int_{\mathbb{R}^d} \left( \frac{2}{\pi \gamma_{\min}^2} \right)^{\frac{d}{2}} \exp \left( -\frac{2\|h\|_2^2}{\gamma_{\max}^2} \right) \omega_{s, L_q(\nu)}^q (\|h\|_2) \, dh
\]

\[
\leq \|f\|_{B_{q, \infty}^\alpha}^q \left( \frac{2}{\pi \gamma_{\min}^2} \right)^{\frac{d}{2}} \left( \int_{\mathbb{R}^d} \|h\|_2^{q\alpha} \exp \left( -\frac{2\|h\|_2^2}{\gamma_{\max}^2} \right) \, dh \right) \frac{1}{q} \left( \int_{\mathbb{R}^d} \|h\|_2^{2\alpha} \exp \left( -\frac{2\|h\|_2^2}{\gamma_{\max}^2} \right) \, dh \right)^{\frac{1}{2}}
\]

\[
= \|f\|_{B_{q, \infty}^\alpha}^q \left( \frac{2\gamma_{\max}^2}{\pi \gamma_{\min}^4} \right)^{\frac{d}{2}} \left( \int_{\mathbb{R}^d} \|h\|_2^{2\alpha} \exp \left( -\frac{2\|h\|_2^2}{\gamma_{\max}^2} \right) \, dh \right)^{\frac{1}{2}}.
\]

Using the embedding constant \( d_{q\alpha} \) of \( \ell_2^d \) to \( \ell_2^d \), we obtain

\[
\int_{\mathbb{R}^d} \|h\|_2^{2\alpha} \exp \left( -\frac{2\|h\|_2^2}{\gamma_{\max}^2} \right) \, dh \leq d_{q\alpha} \sum_{\ell=1}^d \int_{\mathbb{R}} h_{\ell}^{2\alpha} \prod_{l=1}^d \exp \left( -\frac{2h_{l}^2}{\gamma_{\max}^2} \right) \, dh
\]

\[
= d_{q\alpha} \sum_{\ell=1}^d \left( \frac{\gamma_{\max}^2}{2} \right)^{\frac{d-1}{2}} \int_{\mathbb{R}} h_{\ell}^{2\alpha} \exp \left( -\frac{2h_{\ell}^2}{\gamma_{\max}^2} \right) \, dh
\]

\[
= 2d_{q\alpha} \left( \frac{\gamma_{\max}^2}{2} \right)^{\frac{d-1}{2}} \int_0^\infty t^{2\alpha} \exp \left( -\frac{2t^2}{\gamma_{\max}^2} \right) \, dt.
\]
for $\gamma > 0$. With the substitution $t = \left(\frac{1}{2} \gamma_{\text{max}}^2 u\right)^{\frac{1}{2}}$, the functional equation $\Gamma(t + 1) = t \Gamma(t)$ of the Gamma function $\Gamma$, and $\Gamma \left(\frac{1}{2}\right) = \sqrt{\pi}$ we further have

$$
\int_0^\infty i^{2q\alpha} \exp \left(-\frac{2t^2}{\gamma_{\text{max}}^2}\right) dt = \frac{\gamma_{\text{max}}}{\sqrt{2}} \left(\frac{\gamma_{\text{max}}^2}{2}\right)^{q\alpha} \int_0^\infty u^{(q\alpha+\frac{1}{2})-1} \exp (-u) du \\
= \frac{\gamma_{\text{max}}}{\sqrt{2}} \left(\frac{\gamma_{\text{max}}^2}{2}\right)^{q\alpha} \Gamma \left(q\alpha + \frac{1}{2}\right).
$$

Altogether, we finally obtain

$$
\left\| \sum_{j \in J} 1_{A_j} \cdot (K_j * f) - f \right\|_{L_q(\nu)}^q \\
\leq \|f\|_{B_{q,\infty}^d(\nu)}^q \left(\frac{2\gamma_{\text{max}}^2}{\pi \gamma_{\text{min}}^4}\right)^{\frac{q}{2}} \left(\int_{\mathbb{R}^d} \|h\|_{L_q(\nu)}^2 \exp \left(-\frac{2\|h\|_{L_q(\nu)}^2}{\gamma_{\text{max}}^2}\right) dh\right)^{\frac{1}{2}} \\
\leq \|f\|_{B_{q,\infty}^d(\nu)}^q \left(\frac{2\gamma_{\text{max}}^2}{\pi \gamma_{\text{min}}^4}\right)^{\frac{q}{2}} \left(\frac{d}{2}\right)^{q\alpha} \left(\frac{\pi d^{-1}}{2d}\right)^{\frac{1}{2}} \gamma_{\text{max}}^{2q\alpha + d} \Gamma \left(q\alpha + \frac{1}{2}\right)^{\frac{1}{2}} \\
= \|f\|_{B_{q,\infty}^d(\nu)}^q \left(\frac{d}{2}\right)^{q\alpha} \pi^{-\frac{1}{2}} \Gamma \left(q\alpha + \frac{1}{2}\right)^{\frac{1}{2}} \left(\frac{\gamma_{\text{max}}}{\gamma_{\text{min}}^d}\right) \gamma_{\text{max}}^d.
$$

Based on Theorems 5 and 14 we can now show Theorem 7.

**Proof** [Proof of Theorem 7] First, we have to choose a function $f_0 \in H$. To this end, we define functions $K_j : \mathbb{R}^d \rightarrow \mathbb{R}$, $j \in \{1, \ldots, m\}$, by (32), where $s := |\alpha| + 1$ and $\gamma_j > 0$. Then we define $f_0$ by convolving each $K_j$ with the Bayes decision function $f_{*L,P}^j$, that is

$$
f_0(x) := \sum_{j \in J_T} 1_{A_j}(x) \cdot (K_j * f_{*L,P}^j)(x), \quad x \in \mathbb{R}^d.
$$

Now, to show that $f_0$ is indeed a suitable function to bound the approximation error, we first need to ensure that $f_0$ is contained in $H$. In addition, we need to derive bounds for both, the regularization term and the excess risk of $f_0$. To this end, we apply (Eberts and Steinwart, 2013, Theorem 2.3) and obtain, for every $j \in J_T$,

$$
(K_j * f_{*L,P}^j)_{|A_j} \in H_{\gamma_j}(A_j)
$$

with

$$
\|1_{A_j} f_0\|_{H_{\gamma_j}(A_j)} = \|1_{A_j}(K_j * f_{*L,P}^j)\|_{H_{\gamma_j}(A_j)} \\
= \|(K_j * f_{*L,P}^j)_{|A_j}\|_{H_{\gamma_j}(A_j)} \\
\leq (\gamma_j \sqrt{\pi})^{-\frac{d}{2}} (2^s - 1) \|f_{*L,P}^j\|_{L_2(\mathbb{R}^d)}.
$$

51
This implies
\[ f_0 = \sum_{j \in J_T} \mathbb{1}_{A_j}(K_j * f_{L,P}^\ast) \in H_{J_T}. \]

Besides, note that \( 0 \in \hat{H}_{\gamma_j}(A_j) \) for every \( j \in \{1, \ldots, m\} \) such that \( f_0 \) can be written as \( f_0 = \sum_{j=1}^m f_j \), where
\[ f_j := \begin{cases} \mathbb{1}_{A_j}(K_j * f_{L,P}^\ast), & j \in J_T, \\ 0, & j \notin J_T. \end{cases} \]

Obviously, the latter implies \( f_0 \in H \). Furthermore, for \( A_T := \bigcup_{j \in J_T} A_j \), (31) and Theorem 14 yield
\[
\mathcal{R}_{L, J_T, P}(f_0) - \mathcal{R}_{L, J_T, P}^* = \|f_0 - f_{L,P}^\ast\|_{L_2(P_{X|A_T})}^2 \\
= \sum_{j \in J_T} \mathbb{1}_{A_j}(K_j * f_{L,P}^\ast) - f_{L,P}^\ast\|_{L_2(P_{X|A_T})}^2 \\
\leq C_{\alpha, 2} \left( \max_{j \in J_T} \frac{\gamma_j}{\min_{j \in J_T} \gamma_j} \right)^d \max_{j \in J_T} \gamma_j^{2\alpha},
\]
where \( C_{\alpha, 2} \) is a constant only depending on \( \alpha, d, \) and \( \|f_{L,P}^\ast\|_{B_{2,\infty}^2(P_{X|A_T})} \). To utilize Theorem 3 it remains to examine the constants \( B, V, \vartheta, \) and \( B_0 \). Since we consider the least squares loss, which can be clipped at \( M \) with \( Y = [-M, M] \), the supremum bound (18) holds for \( B = 4M^2 \) and the variance bound (19) for \( V = 16M^2 \) and \( \vartheta = 1 \) (cf. Steinwart and Christmann, 2008a, Example 7.3). Next, we derive a bound for \( L \circ f_0 \|_{\infty} \) using (Eberts and Steinwart, 2013, Theorem 2.3) which provides, for every \( x \in X \), the supremum bound
\[
|f_0(x)| = \left| \sum_{j \in J_T} \mathbb{1}_{A_j}(x) \cdot (K_j * f_{L,P}^\ast)(x) \right| \leq \sum_{j \in J_T} \mathbb{1}_{A_j}(x) |K_j * f_{L,P}^\ast(x)| \leq (2^s - 1) \|f_{L,P}^\ast\|_{L_\infty(\mathbb{R}^d)}. \]

The latter implies
\[
\|L_{J_T} \circ f_0\|_{\infty} = \sup_{(x,y) \in X \times Y} |L(y, f_0(x))| \\
\leq \sup_{(x,y) \in X \times Y} \left( M^2 + 2M|f_0(x)| + |f_0(x)|^2 \right) \\
\leq 4^s \max \left\{ M^2, \|f_{L,P}^\ast\|_{L_\infty(\mathbb{R}^d)}^2 \right\},
\]
i.e. \( B_0 := 4^s \max \{ M^2, \|f_{L,P}^\ast\|_{L_\infty(\mathbb{R}^d)}^2 \} \). Moreover, since Theorem 4 provides \( e_i(\text{id} : H_{\gamma_j}(A_j) \to L_2(P_{X|A_j})) \leq a_j i^{-\frac{1}{2p}} \) for \( i \geq 1 \) with \( a_j = \hat{c}_p \sqrt{P_X(A_j)} r^{-\frac{1}{2p}} \gamma_j^{-\frac{d+2p}{2p}} \), we have
\[
\left( \max \left\{ c_p m \sum_{j=1}^m \lambda_j^{-p} a_j^{2p} \right\}^{\frac{1}{2p}}, B \right)^{2p},
\]
52
\[
\begin{align*}
&= \left( \max \left\{ c_p \tilde{c}_p \sqrt{m r} \frac{d + 2p}{2p} \left( \sum_{j=1}^{m} \left( \lambda_j^{-1} \gamma_j \frac{d + 2p}{p} P_X(A_j) \right)^{\frac{1}{2p}}, B \right) \right\} \right)^{2p} \\
&\leq \left( \max \left\{ c_p \tilde{c}_p m r \frac{d + 2p}{2p} \left( \sum_{j=1}^{m} \lambda_j^{-1} \gamma_j \frac{d + 2p}{p} P_X(A_j) \right)^{\frac{1}{2}}, B \right) \right)^{2p} \\
&\leq \left( \max \left\{ c_p \tilde{c}_p 8 \frac{d}{2p} r \left( \sum_{j=1}^{m} \lambda_j^{-1} \gamma_j \frac{d + 2p}{p} P_X(A_j) \right)^{\frac{1}{2}}, B \right) \right)^{2p} \\
&\leq C_p r^{2p} \left( \sum_{j=1}^{m} \lambda_j^{-1} \gamma_j \frac{d + 2p}{p} P_X(A_j) \right)^{2p} + B^{2p} \\
&=: a^{2p},
\end{align*}
\]

where we used the concavity of the function \( t \mapsto t^p \) for \( t \geq 0, m r^d \leq 8^d \) by (7), and \( C_p := c_p \tilde{c}_p 8^d \). Finally, applying Theorem 5 yields

\[
R_{L_j, P}(\tilde{f}_{D, \lambda, \gamma}) - R_{L_j, P}^* \\
\leq \sum_{j=1}^{m} \lambda_j \| f_{L_j, \gamma_j} \|_{L^2(\gamma_j, A_j)}^2 + R_{L_j, P}(\tilde{f}_{D, \lambda, \gamma}) - R_{L_j, P}^* \\
\leq 9 \left( \sum_{j=1}^{m} \lambda_j \| f_{L_j, \gamma_j} \|_{L^2(\gamma_j, A_j)}^2 + R_{L_j, P}(f_0) - R_{L_j, P}^* \right) \\
+ C \left( a^{2p} n^{-1} \right)^{\frac{1}{2p}} + 3 \left( \frac{72 V \tau}{n} \right)^{\frac{1}{2p}} + 15 B_0 \tau \\
\leq 9 \left( \sum_{j \in J_p} \lambda_j (\gamma_j \sqrt{\pi})^{-d} (2^s - 1)^2 \| f_{L_j} \|_{L^2(R^d)}^2 + C_{\alpha, 2} \left( \frac{\max_{j \in J_T} \gamma_j}{\min_{j \in J_T} \gamma_j} \right)^d \max_{j \in J_T} \gamma_j^{2\alpha} \right) \\
+ C C_p r^{2p} \left( \sum_{j=1}^{m} \lambda_j^{-1} \gamma_j \frac{d + 2p}{p} P_X(A_j) \right)^{p} n^{-1} + C B^{2p} n^{-1} + \frac{3456 M^2 \tau}{n} \\
+ 15 \cdot 4^s \max\{M^2, \| f_{L_j} \|_{L^2(R^d)}^2 \} \frac{\tau}{n} \\
\leq 9(2^s - 1)^2 \pi^{-\frac{d}{2}} \| f_{L_p} \|_{L^2(R^d)}^2 \sum_{j \in J_p} \lambda_j \gamma_j^{-d} + 9 C_{\alpha, 2} \left( \frac{\max_{j \in J_T} \gamma_j}{\min_{j \in J_T} \gamma_j} \right)^d \max_{j \in J_T} \gamma_j^{2\alpha} \\
+ C C_p r^{2p} \left( \sum_{j=1}^{m} \lambda_j^{-1} \gamma_j \frac{d + 2p}{p} P_X(A_j) \right)^{p} n^{-1} + 16^p C M^{4p} n^{-1} \\
+ \frac{3456 M^2 + 15 \cdot 4^s \max\{M^2, \| f_{L_p} \|_{L^2(R^d)}^2 \}}{n} \frac{\tau}{n}
\]

53
with probability $P^n$ not less than $1 - 3e^{-\tau}$. Finally, for $\tau \geq 1$, a variable transformation implies

$$\sum_{j=1}^{m} \lambda_j \|f_{D_j, \lambda_j, \gamma_j}\|^2_{H_{\gamma_j}(A_j)} + \mathcal{R}_{L_{J_T}, p}(\hat{f}_{D, \lambda, \gamma}) - \mathcal{R}_{L_{J_T}, p}^* \leq C_{M, \alpha, p} \left( \sum_{j \in J_T} \lambda_j \gamma_j - d + \left( \frac{\max_{j \in J_T} \gamma_j}{\min_{j \in J_T} \gamma_j} \right) d \max_{j \in J_T} \gamma_j^{2\alpha} + r_n^{2\alpha} \left( \sum_{j=1}^{m} \lambda_j^{-1} \gamma_j^{-(d+2p)} P_X(A_j) \right)^{p} n^{-1 + \tau n^{-1}} \right)$$

with probability $P^n$ not less than $1 - e^{-\hat{\tau}}$, where the constant $C_{M, \alpha, p}$ is defined by

$$C_{M, \alpha, p} := \max \left\{ 9(2^s - 1)^2 \pi^{-\frac{d}{2}} \|f_{L, p}\|^2_{L_2(\mathbb{R}^d)}; 9\|f_{L, p}\|^2_{B_{2, \infty}}(P_X|_{A_T}) \left( \frac{d}{2} \right)^\alpha \pi^{-\frac{1}{4}} \Gamma \left( 2\alpha + \frac{1}{2} \right) \right\}.$$

Next, using the just proven oracle inequality presented in Theorem 7, we show the learning rates of Theorem 8 in only a few steps.

**Proof** [Proof of Theorem 8] First of all, we define sequences $\hat{\lambda}_n := c_2 n^{-1}$ and $\hat{\gamma}_n := c_3 n^{-\frac{1}{2(\alpha + d)}}$ to simplify the presentation. Then Theorem 7 $\sum_{j=1}^{m} P_X(A_j) = 1$, and $|J_T| \leq m_n \leq 8^d r_n^{-d}$ together with $\lambda_{n,j} = r_n^d \hat{\lambda}_n$ and $\gamma_{n,j} = \hat{\gamma}_n$ for all $j \in \{1, \ldots, m_n\}$ yield

$$\mathcal{R}_{L_{J_T}, p}(\hat{f}_{D, \lambda, \gamma}) - \mathcal{R}_{L_{J_T}, p}^* \leq C_{M, \alpha, p} \left( |J_T| d \hat{\lambda}_n \hat{\gamma}_n^{-d} + \sum_{j=1}^{m} \lambda_j^{-1} \gamma_j^{-(d+2p)} P_X(A_j) \right)^{p} n^{-1 + \tau n^{-1}}$$

$$= C_{M, \alpha, p} \left( |J_T| d \hat{\lambda}_n \hat{\gamma}_n^{-d} + \sum_{j=1}^{m} \lambda_j^{-1} \gamma_j^{-(d+2p)} P_X(A_j) \right)^{p} n^{-1 + \tau n^{-1}} \leq 8^d C_{M, \alpha, p} \left( \hat{\lambda}_n \hat{\gamma}_n^{-d} + \sum_{j=1}^{m} \lambda_j^{-1} \gamma_j^{-(d+2p)} P_X(A_j) \right)^{p} n^{-1 + \tau n^{-1}}.$$

Using the choices $\hat{\lambda}_n = c_2 n^{-1}$, $\hat{\gamma}_n = c_3 n^{-\frac{1}{2(\alpha + d)}}$, as well as $r_n = c_1 n^{-\frac{1}{\beta d}}$ finally implies

$$\mathcal{R}_{L_{J_T}, p}(\hat{f}_{D, \lambda, \gamma}) - \mathcal{R}_{L_{J_T}, p}^* \leq 8^d C_{M, \alpha, p} \left( \hat{\lambda}_n \hat{\gamma}_n^{-d} + \sum_{j=1}^{m} \lambda_j^{-1} \gamma_j^{-(d+2p)} P_X(A_j) \right)^{p} n^{-1 + \tau n^{-1}} \leq \hat{C}_{M, \alpha, p} \left( n^{-\frac{2\alpha}{2\alpha + d}} + n^{-\frac{2\alpha}{2\alpha + d}} + n^{p} n^{-\frac{d+2p}{\beta d}} n^{-1 + \tau n^{-1}} \right) \leq \hat{C}_{M, \alpha, p} \left( n^{-\frac{2\alpha}{2\alpha + d}} + n^{-\frac{2\alpha}{2\alpha + d}} + n^{-\frac{2\alpha}{2\alpha + d}} + \left( \frac{1}{2} - \frac{1}{\beta d} \right) \right)^{p} + \tau n^{-1}.$$
\[ \leq C \tau n^{-\frac{2\alpha}{\alpha+\beta}} + \xi \]

with probability \( P^n \) not less than \( 1 - e^{-\tau} \), where \( C > 0 \) is a constant and \( \xi \geq \left(1 + \frac{2}{\alpha+\beta} + \frac{\alpha}{\beta} - \frac{2}{\beta d}\right) p > 0. \)

**Proof** [Proof of Corollary \( \text{[2]} \)] For simplicity of notation, we write \( \lambda, \lambda_j, \gamma, \) and \( \gamma_j \) instead of \( \lambda_n, \lambda_{n,j}, \gamma_n, \) and \( \gamma_{n,j} \). Since \( \bigcup_{j \in J} A_j \subset T^+ \delta \) for all \( n \geq n_\delta \), the assumption \( f_{L,P}^* \in B_{2,\infty}^\alpha(P_X|\bigcup_{j \in J} A_j) \) implies

\[ f_{L,P}^* \in B_{2,\infty}^\alpha(P_X|\bigcup_{j \in J} A_j). \]

With this, Theorems \( \text{[7]} \) and \( \text{[8]} \) immediately yield

\[
\begin{align*}
\mathcal{R}_{L,P}(\hat{f}_{D,\lambda}) - &\mathcal{R}_{L,P}^* \\
\leq &\sum_{j=1}^m \lambda_j \|f_{D,\lambda_j}\|_{H_{\hat{\gamma}_j}}^2 + \mathcal{R}_{L,P}(\hat{f}_{D,\lambda}) - \mathcal{R}_{L,P}^*
\leq &\sum_{j=1}^m \lambda_j \|f_{D,\lambda_j}\|_{H_{\hat{\gamma}_j}}^2 + \mathcal{R}_{L,P}(\hat{f}_{D,\lambda}) - \mathcal{R}_{L,P}^*
\leq &C_{M,\alpha,p} \left( \sum_{j \in J} \lambda_j \gamma_j^{-d} + \left( \frac{\max_{j \in J} \gamma_j}{\min_{j \in J} \gamma_j} \right)^d \max_{j \in J} \gamma_j^{2\alpha} + r^{2p} \left( \sum_{j=1}^m \lambda_j^{-1} \gamma_j^{-\frac{d+2p}{p}} P_X(A_j) \right)^{p-1} \right)
\leq &C \tau n^{-\frac{2\alpha}{\alpha+\beta}} + \xi
\end{align*}
\]

with probability \( P^n \) not less than \( 1 - e^{-\tau} \), where \( \xi \geq \left(1 + \frac{2}{\alpha+\beta} + \frac{\alpha}{\beta} - \frac{2}{\beta d}\right) p > 0. \) Moreover, the constants \( C_{M,\alpha,p} > 0 \) and \( C > 0 \) coincide with those of Theorems \( \text{[7]} \) and \( \text{[8]} \).

It remains to prove Theorem \( \text{[10]} \). However, we previously have to consider the following technical lemma.

**Lemma 15** Let \( d \geq 1 \) and \( r_n := c n^{-\frac{1}{d\beta}} \) with \( \beta > 1 \) and a constant \( c > 0 \). We fix finite subsets \( \Lambda_n \subset (0,r_n^d) \) and \( \Gamma_n \subset (0,r_n] \) such that \( \Lambda_n \) is an \( (r_n^d) \)-net of \((0,r_n^d]\) and \( \Gamma_n \) is an \( \delta_n \)-net of \((0,r_n]\) with \( 0 < \varepsilon_n \leq n^{-1/2} \), \( \delta_n > 0 \), \( r_n \in \Lambda_n \), and \( r_n \in \Gamma_n \). Moreover, let \( J \subset \{1, \ldots, m_n\} \) be an arbitrary non-empty index set and \( |J| \leq m_n \leq 8^d r_n^{-d} \). Then, for all \( 0 < \alpha < \frac{\beta}{\beta+2-d} d, \) \( n \geq 1 \), and all \( p \in (0,1) \) with \( p \leq \frac{d-2a-2d}{2a+d+2d} \), we have

\[
\inf_{(\lambda_j, \gamma_j)^m_{j=1} \in (\Lambda_n \times \Gamma_n)^m} \left( \sum_{j \in J} \lambda_j \gamma_j^{-d} + \left( \frac{\max_{j \in J} \gamma_j}{\min_{j \in J} \gamma_j} \right)^d \max_{j \in J} \gamma_j^{2\alpha} + r_n^{2p} \left( \sum_{j=1}^m \lambda_j^{-1} \gamma_j^{-\frac{d+2p}{p}} P_X(A_j) \right)^{p-1} \right)
\leq C \left( n^{-\frac{2\alpha}{\alpha+\beta}} + \delta_n^2 \right),
\]

where \( \xi := \left( \frac{2\alpha(2a+d+2)}{(2a+d)(2a+d)(1+p)+2p} + \max \{ \frac{d-2}{\beta d} + 2, 0 \} \right) p \) and \( C > 0 \) is a constant independent of \( n, \Lambda_n, \varepsilon_n, \Gamma_n, \) and \( \delta_n \).
Proof Without loss of generality, we may assume that \( \Lambda_n \) and \( \Gamma_n \) are of the form \( \Lambda_n = \{ \Lambda^{(1)}, \ldots, \Lambda^{(u)} \} \) and \( \Gamma_n = \{ \Gamma^{(1)}, \ldots, \Gamma^{(v)} \} \) with \( \Lambda^{(u)} = r_n^d \) and \( \Gamma^{(v)} = r_n \) as well as \( \lambda^{(i-1)} < \lambda^{(i)} \) and \( \gamma^{(\ell-1)} < \gamma^{(\ell)} \) for all \( i = 2, \ldots, u \) and \( \ell = 2, \ldots, v \). With \( \lambda^{(0)} := 0 \) and \( \gamma^{(0)} := 0 \) it is easy to see that

\[
\lambda^{(i)} - \lambda^{(i-1)} \leq 2r_n \varepsilon_n \quad \text{and} \quad \gamma^{(\ell)} - \gamma^{(\ell-1)} \leq 2\delta_n
\]

(34)

hold for all \( i = 1, \ldots, u \) and \( \ell = 1, \ldots, v \). Furthermore, define \( \lambda^* := n^{-(2\alpha+d)/(1+p)+2p} \) and \( \gamma^* := cn^{-2\alpha/(1+p)+2p} \). Then there exist indices \( i \in \{1, \ldots, u\} \) and \( \ell \in \{1, \ldots, v\} \) with \( \lambda^{(i-1)} \leq r_n^d \lambda^* \leq \lambda^{(i)} \) and \( \gamma^{(\ell-1)} \leq \gamma^* \leq \gamma^{(\ell)} \). Together with (33), this yields

\[
r_n^d \lambda^* \leq \lambda^{(i)} \leq r_n^d \lambda^* + 2r_n \varepsilon_n \quad \text{and} \quad \gamma^* \leq \gamma^{(\ell)} \leq \gamma^* + 2\delta_n.
\]

(35)

Moreover, the definition of \( \lambda^* \) implies \( \varepsilon_n \leq \lambda^* \) and the one of \( \gamma^* \) implies \( \gamma^* \leq r_n \) for \( \alpha < \frac{\beta-1}{2} \) and \( p \in (0, p^*) \), where \( p^* := \frac{\beta-d-2}{2\alpha+d+2} \). Additionally, it is easy to check that

\[
\lambda^* (\gamma^*)^{-d} + (\gamma^*)^{-p} \leq \gamma^* \quad \text{and} \quad \lambda^* (\gamma^*)^{-d} + (\gamma^*)^{-p} \leq (\gamma^*)^{-d+2p} r_n^{-2p} n^{-1} \leq \tilde{c} n^{-2\alpha/(1+p)+2p} + \max\{\frac{d-2}{p}, 0\} p,
\]

(36)

where \( \tilde{c} \) is a positive constant. Using (35), the bound \( |J| \leq m_n \leq 8^d r_n^{-d} \), and (33), we obtain

\[
\inf_{(\lambda_j, \gamma_j)} \left( \sum_{j \in J} \lambda_j \gamma_j^{-d} + \frac{\max_{j \in J} \gamma_j}{\min_{j \in J} \gamma_j} \right)^{\frac{d}{d+2\alpha}} \max_{j \in J} \gamma_j^{2\alpha} + r_n^{2p} \left( \sum_{j=1}^{m_n} \lambda_j^{-1} \gamma_j^{-\frac{d+2p}{p}} \mathbb{P}(A_j) \right)^{\frac{p}{p-1}} n^{-1} \leq \tilde{c} n^{-2\alpha/(1+p)+2p} + \max\{\frac{d-2}{p}, 0\} p + \tilde{c} \delta_n^{2\alpha} \leq C \left( n^{-2\alpha/(1+p)+\xi} + \delta_n^{2\alpha} \right)
\]

with \( \xi := \frac{2\alpha(2\alpha+d+2)}{2\alpha+2d+1} \) and \( \Gamma_n \) as well as Lemma [15].

Proof [Proof of Theorem [10]] Let \( l \) be defined by \( l := \left\lceil \frac{n}{l} \right\rceil + 1 \), i.e. \( l \geq \frac{n}{l} \). With this, Theorem [7] yields with probability \( \mathbb{P}^l \) not less than \( 1 - |\Lambda_n \times \Gamma_n|^{m_n} e^{-\frac{l}{2}} \)

\[
\mathcal{R}_{L,l_{*}}(\widehat{f}_1, \lambda, \gamma) - \mathcal{R}_{L,l_{*}}^*
\]

56
for all \((\lambda_j, \gamma_j) \in \Lambda_n \times \Gamma_n, j \in \{1, \ldots, m_n\}\), simultaneously, where \(c_1 > 0\) is a constant independent of \(n, \tau, \lambda, \text{ and } \gamma\). Furthermore, the oracle inequality of (Steinwart and Christmann, 2008a, Theorem 7.2) for empirical risk minimization, \(n - l \geq \frac{n}{2} - 1 \geq \frac{n}{4}\), and 

\[
\tau_n := \tau + \ln(1 + |\Lambda_n \times \Gamma_n|^{m_n})
\]

yield

\[
\begin{align*}
R_{L,J_T,P}(\hat{f}_{D_1, A_{D_2}, \gamma_{D_2}}) - R_{L,J_T,P}^* & \\
& \leq c_1 \left( \sum_{j \in J_T} \lambda_j \gamma_j^{-d} + \left( \max_{j \in J_T} \frac{\gamma_j}{\min_{j \in J_T} \gamma_j} \right)^d \max_{j \in J_T} \gamma_j^{2\alpha} + r_n^{2p} \left( \sum_{j=1}^{m_n} \lambda_j^{-1} \gamma_j^{-\frac{d+2p}{p}} P_X(A_j) \right)^p n^{-1} + \tau n^{-1} \right) \\
& \leq c_1 \left( \sum_{j \in J_T} \lambda_j \gamma_j^{-d} + \left( \max_{j \in J_T} \frac{\gamma_j}{\min_{j \in J_T} \gamma_j} \right)^d \max_{j \in J_T} \gamma_j^{2\alpha} + r_n^{2p} \left( \sum_{j=1}^{m_n} \lambda_j^{-1} \gamma_j^{-\frac{d+2p}{p}} P_X(A_j) \right)^p n^{-1} + \tau n^{-1} \right)
\end{align*}
\]

(37)

with probability \(P^{n-l} \) not less than \(1 - e^{-\tau}\). With (37), (39) and Lemma 15 we can conclude

\[
\begin{align*}
& R_{L,J_T,P}(\hat{f}_{D_1, A_{D_2}, \gamma_{D_2}}) - R_{L,J_T,P}^* \\
& \leq 6 \left( \inf_{(\lambda_j, \gamma_j)^{m_n}_{j=1} \in (\Lambda_n \times \Gamma_n)^{m_n}} \sum_{j \in J_T} \lambda_j \gamma_j^{-d} + \left( \max_{j \in J_T} \frac{\gamma_j}{\min_{j \in J_T} \gamma_j} \right)^d \max_{j \in J_T} \gamma_j^{2\alpha} + r_n^{2p} \left( \sum_{j=1}^{m_n} \lambda_j^{-1} \gamma_j^{-\frac{d+2p}{p}} P_X(A_j) \right)^p n^{-1} + \tau n^{-1} \right) \\
& \leq 6c_1 \left( \inf_{(\lambda_j, \gamma_j)^{m_n}_{j=1} \in (\Lambda_n \times \Gamma_n)^{m_n}} \sum_{j \in J_T} \lambda_j \gamma_j^{-d} + \left( \max_{j \in J_T} \frac{\gamma_j}{\min_{j \in J_T} \gamma_j} \right)^d \max_{j \in J_T} \gamma_j^{2\alpha} + r_n^{2p} \left( \sum_{j=1}^{m_n} \lambda_j^{-1} \gamma_j^{-\frac{d+2p}{p}} P_X(A_j) \right)^p n^{-1} + \tau n^{-1} \right) + 2048M^2 \tau_n \frac{\tau_n}{n}
\end{align*}
\]

(38)

(39)

with probability \(P^n \) not less than \(1 - (1 + |\Lambda_n \times \Gamma_n|^{m_n}) e^{-\tau}\). Finally, a variable transformation yields

\[
\begin{align*}
& R_{L,J_T,P}(\hat{f}_{D_1, A_{D_2}, \gamma_{D_2}}) - R_{L,J_T,P}^* \\
& \leq 12c_1Cn^{-\frac{2\alpha+\delta}{2\alpha+d}} + (6c_1 \tau + \ln(1 + |\Lambda_n \times \Gamma_n|^{m_n})) n^{-1} \\
& \leq 12c_1Cn^{-\frac{2\alpha+\delta}{2\alpha+d}} + (6c_1 + 2048M^2) (\tau + 2 \ln(1 + |\Lambda_n \times \Gamma_n|) n^{-1}
\end{align*}
\]

57
\[
= 12c_1Cn^{-\frac{2\alpha}{2\alpha + \xi}} + (6c_1 + 2048M^2) \left( \tau n^{-1} + 2 \cdot 8^n c^{-d} \ln(1 + |A_n \times \Gamma_n|) n^{-\frac{\beta - 1}{\beta}} \right) \\
< \left( 12c_1C + (6c_1 + 2048M^2) \left( \tau + 2 \cdot 8^n c^{-d} \ln(1 + |A_n \times \Gamma_n|) \right) \right) n^{-\frac{\alpha}{2\alpha + \xi}}
\]

with probability $P^n$ not less than $1 - e^{-\tau}$, where we used

\[
\alpha < \frac{\beta - 1}{2} \quad \iff \quad n^{-\frac{\beta - 1}{\beta}} < n^{-\frac{\alpha}{2\alpha + \xi}}
\]

in the last step.

References

R. A. Adams and J. J. F. Fournier. *Sobolev Spaces*. Academic Press, New York, 2nd edition, 2003.

N. Aronszajn. Theory of reproducing kernels. *Trans. Amer. Math. Soc.*, 68:337–404, 1950.

K.P. Bennett and J. A. Blue. A support vector machine approach to decision trees. In *Neural Networks Proceedings, 1998. IEEE World Congress on Computational Intelligence. The 1998 IEEE International Joint Conference on*, volume 3, pages 2396–2401 vol.3, 1998. doi: 10.1109/IJCNN.1998.687237.

H. Berens and R. DeVore. Quantitative Korovkin Theorems for Positive Linear Operators on $L^p$-Spaces. *Trans. Amer. Math. Soc.*, 245:pp. 349–361, 1978. ISSN 00029947. URL http://www.jstor.org/stable/1998871

A. Berlinet and C. Thomas-Agnan. *Reproducing Kernel Hilbert Spaces in Probability and Statistics*. Kluwer, Boston, 2004.

E. Blanzieri and F. Melgani. Nearest Neighbor Classification of Remote Sensing Images With the Maximal Margin Principle. *Geoscience and Remote Sensing, IEEE Transactions on*, 46(6):1804–1811, 2008. ISSN 0196-2892. doi: 10.1109/TGRS.2008.916090.

Enrico Blanzieri and Anton Bryl. Instance-based spam filtering using SVM nearest neighbor classifier. In *In Proceedings of FLAIRS 2007*, pages 441–442, 2007a.

Enrico Blanzieri and Anton Bryl. Evaluation of the Highest Probability SVM Nearest Neighbor Classifier with Variable Relative Error Cost. In *In Proceedings of Fourth Conference on Email and Anti-Spam, CEAS’2007*, 2007b.

Léon Bottou and Vladimir Vapnik. Local Learning Algorithms. *Neural Computation*, 4: 888–900, 1992.

A. Caponnetto and E. De Vito. Optimal rates for regularized least squares algorithm. *Found. Comput. Math.*, 7:331–368, 2007.
B. Carl and I. Stephani. *Entropy, Compactness, and the Approximation of Operators*. Cambridge Tracts in Mathematics. Cambridge University Press, Cambridge, 1990. ISBN 9780521330114. URL [http://books.google.de/books?id=x1BNHvgP7PIC](http://books.google.de/books?id=x1BNHvgP7PIC)

Chih-Chung Chang and Chih-Jen Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011. Software available at [http://www.csie.ntu.edu.tw/~cjlin/libsvm](http://www.csie.ntu.edu.tw/~cjlin/libsvm)

Fu Chang, Chien-Yang Guo, Xiao-Rong Lin, and Chi-Jen Lu. Tree Decomposition for Large-Scale SVM Problems. *J. Mach. Learn. Res.*, 11:2935–2972, December 2010. ISSN 1532-4435. URL [http://dl.acm.org/citation.cfm?id=1756006.1953027](http://dl.acm.org/citation.cfm?id=1756006.1953027)

Haibin Cheng, Pang-Ning Tan, and Rong Jin. Localized Support Vector Machine and Its Efficient Algorithm. In *SIAM International Conference on Data Mining*, 2007.

Haibin Cheng, Pang-Ning Tan, and Rong Jin. Efficient Algorithm for Localized Support Vector Machine. *Knowledge and Data Engineering, IEEE Transactions on*, 22(4):537–549, 2010. ISSN 1041-4347. doi: 10.1109/TKDE.2009.116.

Ronan Collobert, Samy Bengio, and Yoshua Bengio. A Parallel Mixture of SVMs for Very Large Scale Problems. In *Advances in Neural Information Processing Systems*, pages 633–640, 2001.

F. Cucker and S. Smale. On the mathematical foundations of learning. *Bull. Amer. Math. Soc.*, 39:1–49, 2002.

Sanjoy Dasgupta. Lecture 1: Clustering in metric spaces. CSE 291: Topics in unsupervised learning, 2008. URL [http://cseweb.ucsd.edu/~dasgupta/291/lec1.pdf](http://cseweb.ucsd.edu/~dasgupta/291/lec1.pdf)

E. De Vito, A. Caponnetto, and L. Rosasco. Model selection for regularized least-squares algorithm in learning theory. *Found. Comput. Math.*, 5:59–85, 2005.

R.A. DeVore and V.A. Popov. Interpolation of Besov Spaces. *Trans. Amer. Math. Soc.*, 305(1):pp. 397–414, 1988.

Ronald Alvin DeVore and George G. Lorentz. *Constructive Approximation*. Grundlehren der mathematischen Wissenschaften. Springer-Verlag, Berlin, 1993. ISBN 3-540-50627-6. URL [http://books.google.de/books?id=cDqNW6kZwC](http://books.google.de/books?id=cDqNW6kZwC)

Mona Eberts and Ingo Steinwart. Optimal learning rates for least squares SVMs using Gaussian kernels. In J. Shawe-Taylor, R.S. Zemel, P. Bartlett, F.C.N. Pereira, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 24*, pages 1539–1547. 2011.

Mona Eberts and Ingo Steinwart. Optimal regression rates for SVMs using Gaussian kernels. *Electron. J. Statist.*, 7:1–42, 2013. ISSN 1935-7524. doi: 10.1214/12-EJS760.

D. E. Edmunds and H. Triebel. *Function Spaces, Entropy Numbers, Differential Operators*. Cambridge University Press, Cambridge, 1996.
Teofilo F. Gonzalez. Clustering to minimize the maximum intercluster distance. *Theoretical Computer Science*, 38(0):293–306, 1985. ISSN 0304-3975. doi: http://dx.doi.org/10.1016/0304-3975(85)90224-5. URL http://www.sciencedirect.com/science/article/pii/0304397585902245

Hans Peter Graf, Eric Cosatto, Leon Bottou, Igor Durdanovic, and Vladimir Vapnik. Parallel Support Vector Machines: The Cascade SVM. In *Advances in Neural Information Processing Systems*, pages 521–528. MIT Press, 2005.

L. Györfi, M. Kohler, A. Krzyżak, and H. Walk. *A Distribution-Free Theory of Nonparametric Regression*. Springer-Verlag, New York, 2002.

Robert Hable. Universal consistency of localized versions of regularized kernel methods. *J. Mach. Learn. Res.*, 14(1):153–186, January 2013. ISSN 1532-4435. URL http://dl.acm.org/citation.cfm?id=2502581.2502586

Marius Kloft and Gilles Blanchard. On the convergence rate of ℓ_p-norm multiple kernel learning. *J. Mach. Learn. Res.*, 13:2465–2502, Aug 2012.

R. Koenker. *Quantile Regression*. Econometric Society Monographs. Cambridge University Press, 1st edition, 2005. ISBN 9780521608275. URL http://books.google.de/books?id=hdkt7V4NXsgC

S. Mendelson and J. Neeman. Regularization in kernel learning. *Ann. Statist.*, 38:526–565, 2010.

Nicola Segata and Enrico Blanzieri. Empirical assessment of classification accuracy of local SVM. Technical report, University of Trento, Information Engineering and Computer Science, 2008. URL eprints.biblio.unitn.it/1398/1/014.pdf

Nicola Segata and Enrico Blanzieri. Fast and Scalable Local Kernel Machines. *J. Mach. Learn. Res.*, 11:1883–1926, August 2010. ISSN 1532-4435. URL http://dl.acm.org/citation.cfm?id=1756006.1859915

J. S. Simonoff. *Smoothing Methods in Statistics*. Springer, New York, 1996.

S. Smale and D.-X. Zhou. Learning theory estimates via integral operators and their approximations. *Constr. Approx.*, 26:153–172, 2007.

I. Steinwart and A. Christmann. *Support Vector Machines*. Springer, New York, 2008a.

I. Steinwart and A. Christmann. How SVMs can estimate quantiles and the median. In J.C. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems 20*, pages 305–312, Cambridge, MA, 2008b. MIT Press.

I. Steinwart and A. Christmann. Estimating conditional quantiles with the help of the pinball loss. *Bernoulli*, 17:211–225, 2011.

I. Steinwart, D. Hush, and C. Scovel. Optimal rates for regularized least squares regression. In S. Dasgupta and A. Klivans, editors, *Proceedings of the 22nd Annual Conference on Learning Theory*, pages 79–93, 2009.
I. Steinwart, D. Hush, and C. Scovel. Training SVMs without offset. *J. Mach. Learn. Res.*, 12:141–202, 2011.

Taiji Suzuki. Unifying Framework for Fast Learning Rate of Non-Sparse Multiple Kernel Learning. In J. Shawe-Taylor, R.S. Zemel, P. Bartlett, F.C.N. Pereira, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 24*, pages 1575–1583, 2011.

Hans Triebel. *Theory of Function Spaces*. Birkhäuser, Basel, repr. of the 1983 edition, 2010. ISBN 978-3-0346-0415-4.

Ivor W. Tsang, James T. Kwok, and Pak-Ming Cheung. Core Vector Machines: Fast SVM Training on Very Large Data Sets. *J. Mach. Learn. Res.*, 6:363–392, December 2005. ISSN 1532-4435. URL http://dl.acm.org/citation.cfm?id=1046920.1058114

Ivor W. Tsang, Andras Kocsor, and James T. Kwok. Simpler core vector machines with enclosing balls. In *Proceedings of the 24th international conference on Machine learning*, ICML ’07, pages 911–918, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-793-3. doi: 10.1145/1273496.1273611. URL http://doi.acm.org/10.1145/1273496.1273611

Vladimir Vapnik and Léon Bottou. Local Algorithms for Pattern Recognition and Dependencies Estimation. *Neural Computation*, 5:893–909, 1993. doi: 10.1162/neco.1993.5.6.893.

Donghui Wu, Kristin P. Bennett, Nello Cristianini, and John Shawe-Taylor. Large Margin Trees for Induction and Transduction. In *Proceedings of the Sixteenth International Conference on Machine Learning*, ICML ’99, pages 474–483, San Francisco, CA, USA, 1999. Morgan Kaufmann Publishers Inc. ISBN 1-55860-612-2. URL http://dl.acm.org/citation.cfm?id=645528.657622

Alon Zakai and Ya’acov Ritov. Consistency and Localizability. *J. Mach. Learn. Res.*, 10:827–856, June 2009. ISSN 1532-4435. URL http://dl.acm.org/citation.cfm?id=1577069.1577099

Hao Zhang, A.C. Berg, M. Maire, and J. Malik. SVM-KNN: Discriminative Nearest Neighbor Classification for Visual Category Recognition. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 2, pages 2126–2136, 2006. doi: 10.1109/CVPR.2006.301.
Appendix A

For the sake of completeness, we present in the following some tables containing the computational results achieved by the LS-, VP-, and RC-SVMs for all real and artificial data set types. Here, the training and test times, given in seconds, are averaged over all successful runs. Moreover, for the test and $L_2$-errors, we also stated the mean of all runs plus/minus the standard deviation. The same is true for the number of working sets (# of ws), except for the LS-SVMs, where we always have one working set by construction. The last two columns contain median, minimum, and maximum of the working set sizes appearing during the various runs.

| data set sizes | runs | train time | test time | test error | # of ws | ws size: median | ws size: range |
|----------------|------|------------|-----------|------------|---------|----------------|---------------|
| 1000 (1000)    | 100  | 1.38       | 0.48      | 0.7142 ± 0.0097 | 1.00    |                 |               |
| 2500 (1000)    | 100  | 7.44       | 1.15      | 0.6002 ± 0.0082 | 101.92 ± 0.04 | 10             |               |
| 5000 (1000)    | 100  | 30.19      | 1.55      | 0.5707 ± 0.0070 | 1.00    |                 |               |
| 10000 (5000)   | 10   | 138.61     | 12.14     | 0.4909 ± 0.0099 | 1.00    |                 |               |
| 25000 (5000)   | 10   | 922.90     | 34.19     | 0.3816 ± 0.0042 | 1.00    |                 |               |
| 50000 (5000)   | 3    | 3788       | 176.68    | 0.3117 ± 0.0012 | 1.00    |                 |               |
| 100000 (5000)  | 3    | 16353      | 507.38    | 0.2417 ± 0.0012 | 1.00    |                 |               |

| data set sizes | runs | train time | test time | test error | # of ws | ws size: median | ws size: range |
|----------------|------|------------|-----------|------------|---------|----------------|---------------|
| 1000 (1000)    | 100  | 5.85       | 0.73      | 0.7521 ± 0.0088 | 83.73 ± 0.186 | 6              | [1 , 1.56]    |
| 2500 (1000)    | 100  | 7.44       | 1.15      | 0.6002 ± 0.0082 | 101.92 ± 0.04 | 10             | [1 , 1.45]    |
| 5000 (1000)    | 100  | 9.37       | 1.83      | 0.6011 ± 0.0079 | 106.72 ± 0.195 | 24             | [1 , 1.987]   |
| 10000 (5000)   | 10   | 13.45      | 13.68     | 0.5238 ± 0.0076 | 120.20 ± 0.199 | 40             | [1 , 1.952]   |
| 25000 (5000)   | 10   | 26.73      | 31.07     | 0.4134 ± 0.0067 | 131.00 ± 0.226 | 80             | 1, 3.3204     |
| 50000 (5000)   | 3    | 3788       | 176.68    | 0.3117 ± 0.0012 | 1.00    |                 |               |
| 100000 (5000)  | 3    | 16353      | 507.38    | 0.2417 ± 0.0012 | 1.00    |                 |               |

Table 3: LS- and VP-SVM results relating to the COVTYPE data sets
| data set sizes | runs | train time | test time | test error | # of ws | ws size: median | ws size: range |
|----------------|------|------------|-----------|------------|---------|----------------|----------------|
| 100 (1000)     | 100  | 1.76       | 0.30      | 0.7159     | 0.00    | 100            | [1000, 10000]  |
| 2500 (5000)    | 100  | 22.55      | 2.19      | 0.6234     | 0.00    | 250            | [2500, 25000]  |
| 500 (1000)     | 100  | 97.36      | 9.16      | 0.5707     | 0.00    | 500            | [5000, 50000]  |
| 1000 (2000)    | 10   | 741.46     | 111.75    | 0.4699     | 0.00    | 1000           | [10000, 100000]| 0.2395
| 2500 (5000)    | 10   | 8628       | 291.75    | 0.3866     | 0.00    | 2500           | [25000, 250000]| 0.2395
| 5000 (10000)   | 3    | 11416      | 521.84    | 0.3117     | 0.00    | 5000           | [50000, 500000]| 0.2395
| 10000 (20000)  | 3    | 19921      | 622.99    | 0.2417     | 0.00    | 10000          | [100000, 1000000]| 0.2395
| 25000 (50000)  | 0    | NA         | NA        | NA         | NA      | NA             | NA             | NA
| 50000 (100000) | 0    | NA         | NA        | NA         | NA      | NA             | NA             | NA

Table 4: RC-SVM results relating to the

| data set sizes | runs | train time | test time | test error | # of ws | ws size: median | ws size: range |
|----------------|------|------------|-----------|------------|---------|----------------|----------------|
| 100 (1000)     | 100  | 0.99       | 0.32      | 0.8143     | 0.00    | 100            | [100, 1000]    |
| 2500 (5000)    | 100  | 2.73       | 5.29      | 0.7185     | 0.00    | 250            | [2500, 25000]  |
| 500 (1000)     | 50   | 6.61       | 15.24     | 0.6852     | 0.00    | 500            | [5000, 50000]  |
| 1000 (2000)    | 10   | 31.94      | 69.23     | 0.6278     | 0.00    | 1000           | [10000, 100000]| 0.2395
| 2500 (5000)    | 10   | 259.18     | 304.44    | 0.5494     | 0.00    | 2500           | [25000, 250000]| 0.2395
| 5000 (10000)   | 3    | 867.30     | 645.72    | 0.4904     | 0.00    | 5000           | [50000, 500000]| 0.2395
| 10000 (20000)  | 3    | 2665       | 1204      | 0.4120     | 0.00    | 10000          | [100000, 1000000]| 0.2395
| 25000 (50000)  | 3    | 9534       | 1398      | 0.3093     | 0.00    | 25000          | [250000, 2500000]| 0.2395
| 50000 (100000) | 3    | 4146       | 2689      | 0.2300     | 0.00    | 50000          | [500000, 5000000]| 0.2395

Optimal Learning Rates for Localized SVMs

| data set sizes | runs | train time | test time | test error | # of ws | ws size: median | ws size: range |
|----------------|------|------------|-----------|------------|---------|----------------|----------------|
| 100 (1000)     | 100  | 0.99       | 0.32      | 0.8143     | 0.00    | 100            | [100, 1000]    |
| 2500 (5000)    | 100  | 2.73       | 5.29      | 0.7185     | 0.00    | 250            | [2500, 25000]  |
| 500 (1000)     | 50   | 6.61       | 15.24     | 0.6852     | 0.00    | 500            | [5000, 50000]  |
| 1000 (2000)    | 10   | 31.94      | 69.23     | 0.6278     | 0.00    | 1000           | [10000, 100000]| 0.2395
| 2500 (5000)    | 10   | 259.18     | 304.44    | 0.5494     | 0.00    | 2500           | [25000, 250000]| 0.2395
| 5000 (10000)   | 3    | 867.30     | 645.72    | 0.4904     | 0.00    | 5000           | [50000, 500000]| 0.2395
| 10000 (20000)  | 3    | 2665       | 1204      | 0.4120     | 0.00    | 10000          | [100000, 1000000]| 0.2395
| 25000 (50000)  | 3    | 9534       | 1398      | 0.3093     | 0.00    | 25000          | [250000, 2500000]| 0.2395
| 50000 (100000) | 3    | 4146       | 2689      | 0.2300     | 0.00    | 50000          | [500000, 5000000]| 0.2395

Optimal Learning Rates for Localized SVMs
Table 5: Experimental results relating to the COD-RNA data sets

| data set sizes | runs | train time | test time | test error | # of ws | ws size: median | ws size: range |
|----------------|------|------------|-----------|------------|---------|----------------|---------------|
| 100 (1000)     | 100  | 1.21       | 0.25      | 0.3936 ± 0.005 | 1.00    |                |               |
| 2500 (10000)   | 100  | 8.25       | 0.57      | 0.1842 ± 0.003 | 1.00    |                |               |
| 5000 (25000)   | 10   | 33.99      | 1.03      | 0.6722 ± 0.003 | 1.00    |                |               |
| 10000 (50000)  | 10   | 129.26     | 11.75     | 0.1506 ± 0.002 | 1.00    |                |               |
| 25000 (125000) | 10   | 791.14     | 52.75     | 0.1451 ± 0.003 | 1.00    |                |               |
| 50000 (250000) | 10   | 4092.93    | 109.78    | 0.1410 ± 0.003 | 1.00    |                |               |
| 100000 (500000)| 3    | 11078.20   | 201.54    | 0.1302 ± 0.003 | 1.00    |                |               |

| data set sizes | runs | train time | test time | test error | # of ws | ws size: median | ws size: range |
|----------------|------|------------|-----------|------------|---------|----------------|---------------|
| 100 (1000)     | 100  | 0.87       | 0.39      | 0.2036 ± 0.003 | 3.00 ± 0.00 |                |               |
| 2500 (10000)   | 100  | 2.65       | 0.63      | 0.1925 ± 0.003 | 3.00 ± 0.00 |                |               |
| 5000 (25000)   | 10   | 9.58       | 0.99      | 0.1786 ± 0.002 | 3.00 ± 0.00 |                |               |
| 10000 (50000)  | 10   | 34.21      | 11.14     | 0.1660 ± 0.002 | 3.00 ± 0.00 |                |               |
| 25000 (125000) | 10   | 240.20     | 20.02     | 0.1502 ± 0.003 | 3.00 ± 0.00 |                |               |
| 50000 (250000) | 10   | 972.97     | 40.13     | 0.1470 ± 0.003 | 3.00 ± 0.00 |                |               |
| 100000 (500000)| 3    | 3725.80    | 178.60    | 0.1534 ± 0.003 | 3.00 ± 0.00 |                |               |
Table 6: Experimental results relating to the IJCNN1 data sets
### Table 7: Experimental results relating to the artificial data of Type I

| data set sizes | runs | train time | test time | test error | \(L^2\)-error | \(\sigma\) of ws | ws size: median | ws size: range |
|----------------|------|------------|-----------|------------|----------------|----------------|----------------|----------------|
| 1000 (1000)    | 100  | 1.00       | 1.00      | 1.00       | 0.00           | 0.00           | 1000           | 1000           |
| 2000 (1000)    | 100  | 1.00       | 1.00      | 1.00       | 0.00           | 0.00           | 2000           | 2000           |
| 5000 (2000)    | 100  | 1.00       | 1.00      | 1.00       | 0.00           | 0.00           | 5000           | 5000           |
| 10000 (5000)   | 100  | 1.00       | 1.00      | 1.00       | 0.00           | 0.00           | 10000          | 10000          |

### Table 8: Experimental results relating to the artificial data of Type II

| data set sizes | runs | train time | test time | test error | \(L^2\)-error | \(\sigma\) of ws | ws size: median | ws size: range |
|----------------|------|------------|-----------|------------|----------------|----------------|----------------|----------------|
| 1000 (1000)    | 100  | 1.00       | 1.00      | 1.00       | 0.00           | 0.00           | 1000           | 1000           |
| 2000 (1000)    | 100  | 1.00       | 1.00      | 1.00       | 0.00           | 0.00           | 2000           | 2000           |
| 5000 (2000)    | 100  | 1.00       | 1.00      | 1.00       | 0.00           | 0.00           | 5000           | 5000           |
| 10000 (5000)   | 100  | 1.00       | 1.00      | 1.00       | 0.00           | 0.00           | 10000          | 10000          |
### Table 9: Experimental results relating to the artificial data of Type III

| Data set size | runs | train time | test time | test error | L_2-error | # of ws | ws size: median | ws size: range |
|---------------|------|------------|-----------|------------|-----------|---------|----------------|----------------|
| 1000 (100)    | 100  | 0.30       | 0.05      | 0.0088      | 0.0782     | 1.00    |                |                |
| 2500 (100)    | 100  | 1.82       | 0.07      | 0.0057      | 0.0077     | 1.00    |                |                |
| 5000 (100)    | 100  | 7.67       | 0.11      | 0.0032      | 0.0060     | 1.00    |                |                |
| 10000 (100)   | 100  | 29.56      | 0.24      | 0.0059      | 0.0056     | 1.00    |                |                |

### Table 10: Experimental results relating to the artificial data of Type IV

| Data set size | runs | train time | test time | test error | L_2-error | # of ws | ws size: median | ws size: range |
|---------------|------|------------|-----------|------------|-----------|---------|----------------|----------------|
| 1000 (100)    | 100  | 1.53       | 0.20      | 0.0580      | 0.0716     | 1.00    |                |                |
| 2500 (100)    | 100  | 0.53       | 0.16      | 0.0000      | 0.0000     | 1.00    |                |                |
| 5000 (100)    | 100  | 0.77       | 0.19      | 0.0047      | 0.0042     | 1.00    |                |                |
| 10000 (100)   | 100  | 5.11       | 0.59      | 0.0054      | 0.0024     | 1.00    |                |                |
| Data set | Train | Test | Test error | $\bar{E}_{\text{error}}$ | $\#$ of ws | WS size | WS size range |
|----------|-------|------|------------|------------------------|-----------|---------|---------------|
| 1000     | 100   | 0.53 | 0.05       | 0.028 ± 0.004          | 100       | 0.694   | [0.53, 0.74]  |
| 2000     | 100   | 3.26 | 0.09       | 0.027 ± 0.003          | 200       | 0.690   | [0.53, 0.74]  |
| 5000     | 100   | 12.71| 0.25       | 0.048 ± 0.009          | 500       | 0.692   | [0.53, 0.74]  |
| 10000    | 100   | 49.29| 0.28       | 0.040 ± 0.001          | 1000      | 0.691   | [0.53, 0.74]  |

Table 11: Experimental results relating to the artificial data of Type V.