Universität Stuttgart

Fachbereich Mathematik

An SVM-like Approach for Expectile Regression

Muhammad Farooq, Ingo Steinwart

An SVM-like Approach for Expectile Regression

Muhammad Farooq,[*] Ingo Steinwart

July 7, 2015

Abstract

Expectile regression is a nice tool for investigating conditional distributions beyond the conditional mean. It is well-known that expectiles can be described with the help of the asymmetric least square loss function, and this link makes it possible to estimate expectiles in a non-parametric framework by a support vector machine like approach. In this work we develop an efficient sequential-minimal-optimization-based solver for the underlying optimization problem. The behavior of the solver is investigated by conducting various experiments and the results are compared with the recent R-package ER-Boost.

1 Introduction

In standard nonparametric regression analysis, most of the methods developed so far are based on the least square loss function for estimating conditional expectations. In many applications, however, it is required to study conditional distributions beyond means. A nice tool for this purpose was offered by [20] in the form of quantile regression, which allows both the location and the spread of the response variable to be studied by using asymmetric least absolute deviation loss function (ALAD). We refer the reader to [19, 37, 9, 33] and references therein, for details description and different estimation methods for quantile regression. Following the spirit of quantile regression, [21] proposed the asymmetric least square (ALS) loss function

$$L_\tau(t) = \begin{cases} \tau t^2 & \text{if } t \geqslant 0 \\ (1-\tau)t^2 & \text{if } t < 0, \end{cases} \tag{1}$$

to compute *conditional expectiles*, also called *regression expectiles*. These expectiles were found an interesting alternative to quantiles in many applications due to the computational advantages. For example, [3] used the expectile-order to determine the conditional ordering of individual values relative to other members of data sets, [31] developed an expectile-based technique to compute the distribution of treatment effects on the tail of the outcome variable in the presence of confounding mechanism, and [14] compared expectile regression with quantile regression for forecast evaluation under asymmetric loss functions and showed that expectile treatment effects provide more efficient estimates. There are some other areas where expectlies have been applied successfully, for instance, in demography, see [23] and in education, we refer to [29]. Moreover, in finance expectiles play an important role for risk measures of financial asserts, see for instance [2, 15, 26, 42]. For example, it has been shown recently that expectiles are the only coherent risk measures, see [5, 36]. Moreover, the frequently used expected shortfall (ES) is a conditional mean of a random variable given that it is less than a certain quantile. In other words, ES can be written as a function of both quantiles

---

and expectiles, which requires to establish a connection between quantiles and expectiles.This leads to the expectile-based quantile estimates, which can be more efficient than empirical quantiles [41]. In this regard, recall that, there is one-to-one mapping of expectiles to quantiles that was explored by [12] and further supported by [1, 45, 38]. Moreover, [7] embedded both quantiles and expectiles in the general class of M-estimators by proposing asymmetric M-estimators.

Some semiparametric and nonparametric expectile estimation methods have already been proposed in literature. For example, [24] considered penalized splines to compute smooth expectile estimates, [28] proposed a couple of different procedures including least asymmetrically weighted squares in combination with mixed models, boosting within an empirical risk minimization framework, and a restricted expectiles regression model. Moreover, [27] derived asymptotic properties of expectile regression estimates and used them to construct corresponding confidence intervals. Furthermore, a kernel method based on local linear fits was considered in [45], and a boosting method using regression trees was proposed in [44]. Finally, two expectile regression packages, ER-Boost [44] and expectreg [30], have recently been made available.

Another family of non-parametric estimation methods are the so-called kernel based regularized empirical risk minimizers, which include the well known *support vector machines (SVMs)* [39, p. 138ff]. These kernel-based methods often enjoy state-of-the-art empirical performance, relatively simple implementations, and a high flexibility. Recall that their flexibility is based on two main ingredients, namely the reproducing kernel Hilbert space (RKHS) H and the loss function L. Namely, the RKHS can be used to adapt to the nature of the input domain X, or more precisely, enables us to use both standard $\mathbb{R}^n$-valued data and non-standard data such as strings and graphs. Moreover, due to the so-called kernel-trick [25], the choice of H has little to no algorithmic consequences for solving SVM optimization problems. On the other hand, the choice of L determines the learning goal [32, Chapter 3]. For example, the so-called hinge loss is used for classification, the least squares loss leads to conditional mean regression, and the ALAD is used to estimate quantiles. Unfortunately, however, different L lead to different optimization problems, which in turn require different solvers. For the above mentioned loss functions various solvers have been designed, see for example [8, 10, 13, 18, 37] and references therein for more detail, but besides [16], who considered a kernelized iteratively reweighted strategy, no solver for the ALS has been proposed. In this paper, we derive a sequential minimal optimization (SMO) based solver, see [10] and particularly [22], for the ALS, which enables us to handle large data set efficiently. In addition, we consider different initialization methods and working set strategies in detail and validate them empirically, to further speed up the solver. Finally, we report some experiments that compare our solver with the ER-Boost package.

The rest of the paper is organized as follows: Section 2 presents the formulation of the primal and the dual optimization problem of SVMs. Section 3 proposes an algorithm to perform one dual variable update per iteration along with the stopping criteria and initialization methods. The exact two dimensional optimization problem with some working set selection strategies is discussed in Section 4. Some experiments and discussion on the results can be found in section 5. Finally, the appendices contain proofs of theorems and lemmas, and detailed results from experiments.

2 Primal and Dual Optimization Problem

Let us consider a training set $D := ((x_1, y_1), (x_2, y_2), \cdots (x_n, y_n)) \in (X \times \mathbb{R})^n$ that is sampled from some unknown distribution P on $X \times Y$, where X is an arbitrary set and $Y \subset \mathbb{R}$. In addition, we assume that $f : X \to \mathbb{R}$ is a function and $L : Y \times \mathbb{R} \to [0, \infty)$ is an arbitrary convex loss function defined in (1). Then the goal of supervised statistical learning is to find a function f such that the

risk

$$\mathcal{R}_{L,P}(f) := \int_{X \times Y} L(y, f(x)) dP(x, y) \,,$$

is small. This means that $\mathcal{R}_{L,P}(f)$ has to be close to the optimal risk

$$\mathcal{R}_{L,P}^* := \inf\{\mathcal{R}_{L,P}(f) | f : X \to \mathbb{R} \text{ measurable}\} \,,$$

which is called the Bayes risk with respect to P and L. Since the data generating distribution P is unknown, we replace $\mathcal{R}_{L,P}(f)$ by its empirical counterpart

$$\mathcal{R}_{L,D}(f) := \frac{1}{n} \sum_{i=1}^{n} L(y_i, f(x_i)) \,. \tag{2}$$

Now, recall that the support vector machines (SVMs) solve the regularized problem

$$f_{D,\lambda} = \arg\min_{f \in H} \lambda \|f\|_H^2 + \mathcal{R}_{L,D}(f) \,, \tag{3}$$

where $\lambda > 0$ is a user specified regularization parameter and H is the reproducing kernel Hilbert space (RKHS) over X with reproducing kernel $k : X \times X \to \mathbb{R}$, see e.g. [6, 4, 32]. For example, for input domains $X \subset \mathbb{R}^d$, one often uses SVMs that are equipped with Gaussian radial basis (RBF) kernels. Recall that the latter are defined by

$$k_\gamma(x, x') := \exp(-\gamma^2 \|x - x'\|_2^2) \,, \qquad x, x' \in \mathbb{R}^d \tag{4}$$

where $\gamma > 0$ is called the width parameter that is usually determined in a data-dependent way, e.g., by cross-validation. Note that k_γ is normalized, that is, $k_\gamma(x, x) = 1$ for all $x \in \mathbb{R}^d$, and all kernels we consider below are also normalized. By [32, Theorem 4.56], k_γ is also universal on every compact subset $X \in \mathbb{R}^n$ and in particular strictly positive definite. Furthermore, the RKHS H_γ induced by k_γ is dense in $L_p(\mu)$ [32, Chapter 4], where μ is a finite measure of $\mathbb{R}^n$ and $p \in [1, \infty)$. Therefore, the following consistency result applies to Gaussian kernels.

Theorem 1. *Let P be a distribution on $X \times \mathbb{R}$ with $\int_X y^2 P(dy|x) dP_X(x) < \infty$, L be the τ-asymmetric least squares loss, and $f_{L,P}^*$ be the conditional τ-expectile function. Moreover, let k be a bounded, measurable kernel whose RKHS is separable and dense in $L_2(P_X)$. Then for all sequences $\lambda_n \to 0$ with $\lambda_n^4 n \to 0$ and all $\varepsilon > 0$, we have*

$$\lim_{n \to \infty} P^n \big(D \in (X \times \mathbb{R})^n : \mathcal{R}_{L,P}(f_{D,\lambda_n}) - \mathcal{R}_{L,P}^* > \varepsilon \big) = 0 \,,$$

and

$$\lim_{n \to \infty} P^n \big(D \in (X \times \mathbb{R})^n : \|f_{D,\lambda_n} - f_{L,P}^*\|_0 > \varepsilon \big) = 0 \,,$$

where $\|g\|_0 := \int \min\{1, |g|\} dP_X$ is a translation-invariant metric describing convergence in probability P_X.

To deal with (3) algorithmically, we fix a feature space H_0 and a feature map $\Phi : X \to H_0$ of $\mathbb{R}$. Then every $f \in H$ can be represented by $w \in H_0$ via

$$f(\mathbf{x}_i) = \langle \mathbf{w}, \phi(\mathbf{x}_i) \rangle \,, \tag{5}$$

see [32, Theorem 4.21] for further details. Note that the latter theorem also shows that

$$\|f\|_H = \inf\{\|\mathbf{w}\|_{H_0} : \mathbf{w} \in H_0 \text{ with } f = \langle \mathbf{w}, \phi(\cdot) \rangle\} \,, \tag{6}$$

3

where $\phi := X \to H$ is the canonical feature map from the input space to RKHS. Using (2) and (6) in the objective function (3), we obtain the standard regularized problem for SVMs without offset

$$\arg\min_{w \in H_0} \lambda \|\mathbf{w}\|_{H_0}^2 + \frac{1}{n} \sum_{i=1}^{n} L(y_i, f(x_i)). \tag{7}$$

If L is the hinge loss function, then it is shown by [35] that the SVM without offset not only faster but also achieves accuracy that is comparable to SVM with offset. One reason for the faster training time was that the offset leads to an additional equality constraint for the dual problem and as a consequence, SMO type solvers can only update certain pairs of dual variables. In addition, the offset makes it relatively expensive to calculate the duality gap [10], which may serve as a stopping criterion for these solvers.

In the following, we will adapt the ideas of [35] to design a solver for (7) in the case of L being an asymmetric least squares loss. To this end, we first reformulate the primal optimization problem (7) such as

$$\arg\min_{\substack{(\mathbf{w}, \xi_+, \xi_-) \\ \mathbf{w} \in H}} P_C(\mathbf{w}, \xi_+, \xi_-) := \frac{1}{2}\|\mathbf{w}\|^2 + C\tau \sum_{i=1}^{n} \xi_{i,+}^2 + C(1-\tau) \sum_{i=1}^{n} \xi_{i,-}^2,$$

$$\text{such that} \qquad \xi_{i,+} \geq y_i - \langle \mathbf{w}, \phi(\mathbf{x}_i) \rangle, \tag{8}$$

$$\xi_{i,-} \geq \langle \mathbf{w}, \phi(\mathbf{x}_i) \rangle - y_i,$$

$$\xi_{i,+}, \xi_{i,-} \geq 0, \qquad \forall i = 1, \dots, n$$

where $C := \frac{1}{2n\lambda} > 0$. Using standard Langrangian techniques, see e.g. [10, Chapter 6], one can easily obtain the dual optimization problem

$$\arg\max_{(\alpha, \beta)} W(\alpha, \beta) := \langle \alpha - \beta, \mathbf{y} \rangle - \frac{1}{2}\langle \alpha - \beta, K(\alpha - \beta) \rangle - \frac{1}{4C\tau}\langle \alpha, \alpha \rangle - \frac{1}{4C(1-\tau)}\langle \beta, \beta \rangle \tag{9}$$

$$\alpha_i \geq 0, \beta_i \geq 0. \qquad \forall i = 1, \dots, n$$

Here $\mathbf{y}$ is the $n \times 1$ vector of labels and K is the $n \times n$ matrix with entries $K_{i,j} := k(x_i, x_j)$, $i, j = 1, \dots, n$. Note that (8) is a convex function as the loss function (1) is a convex suffices. Analogously, it is not hard to see that the dual optimization problem (9) is concave. This ensures the fulfillment of the strong duality assumptions [10, Chapter 5] and consequently, the primal optimal solution can be obtain from the dual optimal solution using the simple transformation, which is

$$\mathbf{w} := \sum_{i=1}^{n} (\alpha_i - \beta_i)\phi(\mathbf{x}_i).$$

In addition, the quadratic nature of (9) allows us to solve it using the quadratic programming (QP) techniques. However, many QP techniques that are implemented to solve dual optimization problems, for example, interior point methods [43, 25], are impractical for large scale problems. Decomposition methods, such as *chunking* [39] have been designed to handle this difficulty by breaking the optimization problem into smaller subproblems and solving them iteratively. The limiting case of decomposition methods is the Sequential Minimal Optimization (SMO) methods that optimizes two coordinates at each iteration [22] for SVMs with offset and hence, does not require storage of the entire kernel matrix. Section 4 presents this idea in more detail in view of expectile regression without offset. It is also worth noting that SVMs without offset allows us to develop an SMO type algorithm that performs one dual variable update per iteration as a starting point [35]. In the following section, we introduce this algorithm in details.

4

3 One Working Set Solution

Our goal in this section is to develop an SMO type algorithm that updates a single coordinate at each iteration. For this, we first compute one working set solution. Then we establish a rule to select a direction in which update should be performed, and a criterion to stop the algorithm. In the end, we present the procedures to initialize the coordinates.

Let us first compute the gradients for α_i and β_i from (9) that will be used throughout this paper. For this, we take the partial derivatives of (9) w.r.t. α_i and β_i and obtain the following

$$
\begin{aligned}
\nabla W_{\alpha_i}(\alpha, \beta) &= \langle e_i, \mathbf{y} \rangle - \langle e_i, K(\alpha - \beta) \rangle - \frac{\langle e_i, \alpha \rangle}{2C\tau}, \\
\nabla W_{\beta_i}(\alpha, \beta) &= -\langle e_i, \mathbf{y} \rangle + \langle e_i, K(\alpha - \beta) \rangle - \frac{\langle e_i, \beta \rangle}{2C(1 - \tau)}.
\end{aligned}
\tag{10}
$$

We now recall [10, p. 131ff] and reformulate the dual objective function (9). For $\alpha, \beta \in \mathbb{R}^n$ and an index $i \in \{1, \ldots, n\}$, we write $\alpha^{\backslash i} := \alpha - \alpha_i e_i$ and $\beta^{\backslash i} := \beta - \beta_i e_i$ where e_i is the i-th vector of standard basis of $\mathbb{R}^n$. Now the basic calculus together with $K_{i,i} = 1$ for normalized kernels leads to the following dual objective function for the 1D-problem

$$
\begin{aligned}
W(\alpha^{\backslash i} + \alpha_i e_i, \beta^{\backslash i} + \beta_i e_i) &:= W(\alpha^{\backslash i}, \beta^{\backslash i}) + (\alpha_i - \beta_i)\langle e_i, \mathbf{y} \rangle - \frac{1}{2}(\alpha_i - \beta_i)^2 \\
&\quad - (\alpha_i - \beta_i)\langle e_i, K(\alpha^{\backslash i} - \beta^{\backslash i}) \rangle - \frac{\alpha_i^2}{4C\tau} - \frac{\beta_i^2}{4C(1 - \tau)}.
\end{aligned}
\tag{11}
$$

Taking partial derivative of (11) w.r.t. α_i and β_i and setting them to zero yields the system of equations

$$
\begin{aligned}
b_1 \alpha_i - \beta_i &= c_i, \\
\alpha_i - b_2 \beta_i &= c_i,
\end{aligned}
\tag{12}
$$

where

$$
\begin{aligned}
b_1 &= \frac{2C\tau + 1}{2C\tau}, \\
b_2 &= \frac{2C(1 - \tau) + 1}{2C(1 - \tau)}, \\
c_i &= \langle e_i, \mathbf{y} \rangle - \langle e_i, K(\alpha^{\backslash i} - \beta^{\backslash i}) \rangle = \nabla W_{\alpha_i}(\alpha, \beta) + b_1\langle e_i, \alpha \rangle - \langle e_i, \beta \rangle.
\end{aligned}
\tag{13}
$$

After solving (12), we obtain the global solution

$$
\alpha_i^* = \frac{b_2 - 1}{b_1 b_2 - 1} c_i, \qquad \beta_i^* = \frac{1 - b_1}{b_1 b_2 - 1} c_i.
\tag{14}
$$

Note that $b_1, b_2 \in (1, \infty)$ for all $C > 0$ and $\tau \in (0, 1)$. Therefore, it is not hard to see from (14) that $\alpha_i^* = \beta_i^* = 0$ if and only if $c_i = 0$. On the other hand, for all $c_i \in \mathbb{R} \setminus \{0\}$, (14) leads to the relation

$$
\alpha_i^* = -\frac{\tau}{1 - \tau} \beta_i^*,
\tag{15}
$$

which implies that the global solution (α_i^*, β_i^*) violates the constraints of the dual problem (9). In other words, the global maximum that is attained by (9) does not lie in the set of feasible vectors. The following general theorem describes the way to find the solution in this situation.

Theorem 2. *Let $W : \mathbb{R}^m \to \mathbb{R}$ be a concave and twice continuous differentiable function and $\mathcal{A} \subset \mathbb{R}^m$ be a closed convex set. Assume that there is exactly one $\alpha^* \in \mathbb{R}^m$ with $W'(\alpha^*) = 0$. Then the following statements hold:*

i) For all $\alpha \neq \alpha^*$ we have $W(\alpha^*) > W(\alpha)$.

ii) If $\alpha^* \notin \mathcal{A}$, then there exists an $\alpha^\star \in \partial \mathcal{A}$ such that $W(\alpha^\star) \geq W(\alpha)$ for all $\alpha \in \mathcal{A}$.

Theorem 2 says that either α^* is the optimal feasible solution or there is an optimal feasible solution on boundary $\{(0, \beta_i) : \beta_i > 0\} \cup \{(\alpha_i, 0) : \alpha_i > 0\}$. Now (14) shows that we have exactly one value (α_i^*, β_i^*) at which derivative vanishes and (15) shows that (α_i^*, β_i^*) is not feasible. Consequently, we need to look at the boundaries to search for an optimal feasible solution. To this end, we split the problem into two cases. In the first case, we plug $\alpha_i = 0$ in (11) and then differentiate w.r.t. β_i, which provides

$$\frac{\partial W(\alpha^{\backslash i}, \beta^{\backslash i} + \beta_i e_i)}{\partial \beta_i} = -\langle e_i, \mathbf{y} \rangle + \langle e_i, K(\alpha^{\backslash i} - \beta^{\backslash i}) \rangle - b_2 \langle e_i, \beta \rangle \,.$$

Setting it to zero gives

$$\alpha_i^+ = 0 \,, \qquad \beta_i^+ = -\frac{c_i}{b_2} \,. \tag{16}$$

Similarly, for the second case, plugging $\beta_i = 0$ in (11) and differentiating w.r.t. α_i yields

$$\frac{\partial W(\alpha^{\backslash i} + \alpha_i e_i, \beta^{\backslash i})}{\partial \alpha_i} = \langle e_i, \mathbf{y} \rangle - \langle e_i, K(\alpha^{\backslash i} - \beta^{\backslash i}) \rangle - b_1 \langle e_i, \alpha \rangle \,.$$

Equating it to zero provides

$$\beta_i^+ = 0 \,, \qquad \alpha_i^+ = \frac{c_i}{b_1} \,. \tag{17}$$

Since $b_1, b_2 \in (1, \infty)$ are fixed constants for certain τ, therefore, (16) and (17) solely depend on c_i. In particular, if $c_i \neq 0$, then we show in the following theorem that either (16) or (17) gives the feasible optimal solution.

Theorem 3. *For $i = \{1, \ldots, n\}$, let $c_i \in \mathbb{R}$ and $b_1, b_2 \in (1, \infty)$ be defined by (13). Then the following implications holds:*

i) If $c_i < 0$, then (16) is the feasible solution.

ii) If $c_i = 0$, then (16) and (17) are the same feasible solution.

iii) If $c_i > 0$, then (17) is the feasible solution.

In particular, exactly one of the two cases produces a feasible solution (α_i^+, β_i^+), and this is given by

$$\alpha_i^+ = \max\left(0, \frac{c_i}{b_1}\right) \,, \qquad \beta_i^+ = \max\left(0, -\frac{c_i}{b_2}\right) \,.$$

After finding the feasible optimal solution, the next task is to determine the coordinate i in which the update should be performed. Many approaches have been discussed so far for this purpose. A simple approach [10, p. 132-133] is to update for each coordinate $i = 1, \ldots, n$ iteratively. Another method [40] is to choose the coordinate for update that violates the Karush-Kuhn-Tucker (KKT) conditions of optimality most. The latter approach is implemented in SVMs packages, SVM^{light} [17] and LIBSVM [8]. Another idea, see [35], which is followed in this work, is to choose the coordinate i^* whose update achieves the largest improvement for the value of dual objective function W. In other words, it performs the update in the direction

$$i^* \in \arg \max_{i=1,\ldots,n} W(\alpha + \delta e_i, \beta + \eta e_i) - W(\alpha, \beta) \,, \tag{18}$$

where $\delta_i = \alpha_i^+ - \alpha_i$ and $\eta_i = \beta_i^+ - \beta_i$ denote the difference between the new and the old values of α_i and β_i respectively. Based on this idea, we establish a rule in the following lemma to compute the improvement in the value of dual objective function W.

Lemma 4. *Let $i \in \{1, \ldots, n\}$, $\alpha, \beta \in \mathbb{R}^n$, and $\delta, \eta \in \mathbb{R}$. Moreover let $b_1, b_2 \in (1, \infty)$ be defined by (13), then we have*

$$
\begin{aligned}
G(\delta, \eta) &:= W(\alpha + \delta e_i, \beta + \eta e_i) - W(\alpha, \beta) \\
&= \delta \left(\nabla W_{\alpha_i}(\alpha, \beta) - \frac{b_1 \delta}{2} \right) + \eta \left(\nabla W_{\beta_i}(\alpha, \beta) - \frac{b_2 \eta}{2} \right) + \delta \eta.
\end{aligned}
\tag{19}
$$

With the above lemma, the Procedure 1 solves (18) to search the best direction.

Procedure 1 Calculate $i^* \in \arg\max_{i=1,\ldots n} \left(W(\alpha + \delta e_i, \beta + \eta e_i) - W(\alpha, \beta) \right)$

> $bestgain \leftarrow -1$
> **for** $i = 1$ to n **do**
> $\quad \delta_i \leftarrow \max\left(0, \frac{c_i}{b_1}\right) - \alpha_i$
> $\quad \eta_i \leftarrow \max\left(0, -\frac{c_i}{b_2}\right) - \beta_i$
> $\quad gain \leftarrow G(\delta_i, \eta_i)$
> $\quad$ **if** $gain > bestgain$ **then**
> $\quad\quad bestgain \leftarrow gain$
> $\quad\quad i^* \leftarrow i$
> $\quad\quad \delta_{i^*} \leftarrow \delta_i$
> $\quad\quad \eta_{i^*} \leftarrow \eta_i$
> $\quad$ **end if**
> $\quad$ **return** $i^*, \delta_{i^*}, \eta_{i^*}$
> **end for**

3.1 Stopping Criteria

Solving problem (9) by some iteration method requires an appropriate stopping criteria. Several stopping criteria have been suggested so far for SVMs *with offset*. One method is to stop training when the KKT conditions are satisfied up to some predefined tolerance $\epsilon > 0$. Another method is to use the duality gap as a stopping criteria [10, p. 109 and 128]. This method is also adopted by [35] to formulate a duality gap for SVM *without offset*. Following this idea, we define for dual variables $\alpha \in \mathbb{R}_+$ and $\beta \in \mathbb{R}_+$

$$
f_{\alpha,\beta} := \langle \alpha - \beta, K e_i \rangle,
\tag{20}
$$

which gives $\|f_{\alpha,\beta}\|_H^2 = \langle \alpha - \beta, K(\alpha - \beta) \rangle$. As a result, the primal objective function (8) is

$$
P(f_{\alpha,\beta}, \xi_{i,+}, \xi_{i,-}) = \frac{1}{2} \langle \alpha - \beta, K(\alpha - \beta) \rangle + C\tau \sum_{i=1}^n \xi_{i,+}^2 + C(1-\tau) \sum_{i=1}^n \xi_{i,-}^2.
$$

Following [35], the duality gap of $P(f_{\alpha,\beta}, \xi_{i,+}, \xi_{i,-})$ and $W(\alpha, \beta)$ is defined as

$$
S(\alpha, \beta) := P(f_{\alpha,\beta}, \xi_{i,+}, \xi_{i,-}) - W(\alpha, \beta),
\tag{21}
$$

which tells us to stop the iteration method of solving problem (9) if $S(\alpha, \beta) < \epsilon$, where $\epsilon > 0$ is some predefined tolerance. To efficiently compute $S(\alpha, \beta)$, we split it into

$$T(\alpha, \beta) = \frac{1}{2} \langle (\alpha - \beta), K(\alpha - \beta) \rangle - W(\alpha, \beta),$$

$$E(\alpha, \beta) = \tau \sum_{i=1}^{n} \xi_{i,+}^2 + (1 - \tau) \sum_{i=1}^{n} \xi_{i,-}^2, \tag{22}$$

and as a result we have $S(\alpha, \beta) = T(\alpha, \beta) + C \cdot E(\alpha, \beta)$. The value $T(\alpha, \beta)$ can be obtained at each iteration by updating it in the chosen direction i, such as

$$T(\alpha + \delta e_i, \beta + \eta e_i) = T(\alpha, \beta) - U(\alpha_i, \beta_i, \delta, \eta),$$

where

$$U(\alpha_i, \beta_i, \delta, \eta) := \delta \left(2 \nabla W_{\alpha_i}(\alpha, \beta) + \langle y, e_i \rangle + \frac{\langle \alpha, e_i \rangle}{2C\tau} - \frac{(b_1 + 1)\delta}{2} \right)$$

$$+ \eta \left(\nabla W_{\beta_i}(\alpha, \beta) + \langle y, e_i \rangle + \frac{\langle \beta, e_i \rangle}{2C(1 - \tau)} - \frac{(b_2 + 1)\eta}{2} \eta \right) + 2\delta\eta. \tag{23}$$

Unlike $T(\alpha, \beta)$, the value $E(\alpha, \beta)$ can not be updated but needs to be computed from scratch at each iteration. To find an efficient formula, we first note that combining (8) with (20), we have

$$\xi_{i,+} = \max \left\{ 0, \langle y, e_i \rangle - \langle \alpha - \beta, Ke_i \rangle \right\} = \max \left\{ 0, \nabla W_{\alpha_i}(\alpha, \beta) + \frac{\langle \alpha, e_i \rangle}{2C\tau} \right\},$$

and

$$\xi_{i,-} = \max \left\{ 0, \langle \alpha - \beta, Ke_i \rangle - \langle y, e_i \rangle \right\} = \max \left\{ 0, -\nabla W_{\alpha_i}(\alpha, \beta) - \frac{\langle \alpha, e_i \rangle}{2C\tau} \right\}.$$

With these formulas, the computation of $E(\alpha, \beta)$ is an $O(n)$ operation. Let us now consider a little more involved stopping criteria based on [32, Chapter 7], that looks for an $f_{\alpha,\beta} \in H$ with

$$\lambda \|f_{\alpha,\beta}\|_H^2 + \mathcal{R}_{L,D}(\widehat{f}_{\alpha,\beta}) \leq \min_{f \in H} \lambda \|f\|_H^2 + \mathcal{R}_{L,D}(f) + \epsilon, \tag{24}$$

where $\widehat{f}_{\alpha,\beta}$ is clipped at $\pm M \in \mathbb{R}$. Formally speaking, the clipped value of $f_{\alpha,\beta} : X \to \mathbb{R}$ at $\pm M \in \mathbb{R}$ is defined by

$$\widehat{f}_{\alpha,\beta} = \begin{cases} -M & \text{if } f_{\alpha,\beta} < -M, \\ f_{\alpha,\beta} & \text{if } f_{\alpha,\beta} \in [-M, M], \\ -M & \text{if } f_{\alpha,\beta} > -M. \end{cases}$$

In other words, we restrict $f_{\alpha,\beta}$ to the interval $[-M, M]$, which in turns, reduces the risk $\mathcal{R}_{L,D}(f)$. However, clipping does not change the learning method since it is performed after the learning phase. Based on this idea, the clipped version of (20) after using (20) is

$$\widehat{f}_{\alpha,\beta}(x_i) = \left[\langle e_i, \mathbf{y} \rangle - \nabla W_{\alpha_i}(\alpha, \beta) - \frac{\langle \alpha, e_i \rangle}{2C\tau} \right]_{-M}^{M}, \tag{25}$$

which leads to the clipped $\xi_{i,+}$ and $\xi_{i,-}$ as

$$\widehat{\xi}_{i,+} = \max \left\{ 0, \langle \mathbf{y}, e_i \rangle - \left[\langle e_i, \mathbf{y} \rangle - \nabla W_{\alpha_i}(\alpha, \beta) - \frac{\langle \alpha, e_i \rangle}{2C\tau} \right]_{-M}^{M} \right\},$$

$$\widehat{\xi}_{i,-} = \max \left\{ 0, \left[\langle e_i, \mathbf{y} \rangle - \nabla W_{\alpha_i}(\alpha, \beta) - \frac{\langle \alpha, e_i \rangle}{2C\tau} \right]_{-M}^{M} - \langle \mathbf{y}, e_i \rangle \right\}. \tag{26}$$

8

We further define

$$\widehat{E}(\alpha, \beta) := \tau \sum_{i=1}^{n} \widehat{\xi}_{i,+}^{2} + (1 - \tau) \sum_{i=1}^{n} \widehat{\xi}_{i,-}^{2} \, .$$

Then we see that (24) is satisfied if

$$\widehat{S}(\alpha, \beta) := T(\alpha, \beta) + C \cdot \widehat{E}(\alpha, \beta) \leq \frac{\epsilon}{2\lambda} \, . \tag{27}$$

The *clipped* slack variables used in the stopping criteria (27) may provide a substantial decrease in duality gap in each iteration of learning algorithm compared to the unclipped slack variables used in (21), and hence the learning algorithm may require less number of iterations. [34] showed that the right hand side of the stopping criteria given in (21) should be replaced by $\frac{\epsilon}{2\lambda}$ as in (27), where ϵ has the same value for both. Furthermore, it is argued by [35] that unlike the duality gap stopping criteria for SVM *with offset* given by [10, p. 109f], both (21) and (27) are directly computable since they do not require the offset term. From this it is easy to derive an $O(n)$ procedure that updates $\nabla W_{\alpha}(\alpha, \beta)$, $\nabla W_{\beta}(\alpha, \beta)$ and calculate $S(\alpha, \beta)$. The pseudocode for this is presented in Procedure 2. The one for $\widehat{S}(\alpha, \beta)$ is an obvious modifications and therefore omitted.

Procedure 2 Update $\nabla W_{\alpha_i}(\alpha, \beta)$ and $\nabla W_{\beta_i}(\alpha, \beta)$ in direction i^* and calculate $S(\alpha, \beta)$

$T(\alpha, \beta) \leftarrow T(\alpha, \beta) - U(\alpha_i, \beta_i, \delta, \eta)$
$E(\alpha, \beta) \leftarrow 0$
for $k = 1$ to n **do**
$\quad \nabla W_{\alpha_k}(\alpha, \beta) \leftarrow \nabla W_{\alpha_k}(\alpha, \beta) - (\delta - \eta) K_{ik} - \frac{\delta^*}{2C\tau} \delta_{ik}$
$\quad \nabla W_{\beta_k}(\alpha, \beta) \leftarrow \nabla W_{\beta_k}(\alpha, \beta) + (\delta - \eta) K_{ik} - \frac{\eta^*}{2C(1-\tau)} \delta_{ik}$
$\quad \xi_{k,+} \leftarrow \max\{0, \nabla W_{\alpha_k}(\alpha, \beta) + \frac{\alpha_k}{2C\tau}\}$
$\quad \xi_{k,-} \leftarrow \max\{0, -\nabla W_{\alpha_k}(\alpha, \beta) - \frac{\alpha_k}{2C\tau}\}$
$\quad E(\alpha, \beta) \leftarrow E(\alpha, \beta) + \left(\tau \xi_{k,+}^2 + (1 - \tau) \xi_{k,-}^2 \right)$
end for
$S(\alpha, \beta) = T(\alpha, \beta) + C \cdot E(\alpha, \beta)$

With all the above computation, we now summarize the basic idea of the 1D-SVM in Algorithm 1. This tells us to look repeatedly for the best direction i^* and performs update in that direction until the predefined stopping criteria is satisfied.

Algorithm 1 1D-SVM solver
---
initialize $\alpha, \beta, \nabla W_{\alpha}(\alpha, \beta), \nabla W_{\beta}(\alpha, \beta)$ and $T(\alpha, \beta)$
while $S(\alpha, \beta) > \frac{\varepsilon}{2\lambda}$ **do**
$\quad (i^*, \delta_{i^*}, \eta_{i^*}) \leftarrow$ Procedure 1
$\quad \alpha_{i^*} \leftarrow \alpha_{i^*} + \delta_{i^*}$
$\quad \beta_{i^*} \leftarrow \beta_{i^*} + \eta_{i^*}$
$\quad$ use Procedure 2 to update $\nabla W_{\alpha}(\alpha, \beta), \nabla W_{\beta}(\alpha, \beta)$ in direction i^* by δ_{i^*} and η_{i^*} and calculate $S(\alpha, \beta)$
end while

A closer look of the Algorithm 1 reveals that there is still need to develop some procedures to initialize α and β, and the corresponding gradients. The following section presents some initialization methods to fulfill this requirement.

3.2 Initialization

Various approaches are available to initialize α and β and their corresponding gradients. We here briefly describe two approaches, namely, *cold start* and *warm start* that will be used in the implementation of the solver.

I0 & W0: Cold Start With Zeros. This is the most simplest initialization in which we take $\alpha \leftarrow 0$ and $\beta \leftarrow 0$ to initialize. After a simple calculation, it is not hard to initialize the corresponding gradients and the duality gap.

W1: Warm Start by Recycling Old Solution. Recall that typically the hyper-parameter λ is chosen by a search over a grid $\Lambda = \{\lambda_1, \ldots, \lambda_m\}$ of candidates values. If these values are ordered in the form $\lambda_1 > \ldots > \lambda_m$ and the SVM is trained in this order, then the resulting $C^{(1)}, \ldots, C^{(m)}$ satisfy the property that $C^{(j)} < C^{(j+1)}$ for all $j = 1, \ldots, m-1$. For $C^{(1)}$ we initialize the solver with the above cold start and for $j \geq 2$, we initialize it with a warm start $\alpha \leftarrow \alpha^*$ and $\beta \leftarrow \beta^*$ where α^*, β^* is the approximate solution obtained by training with $C^{old} = C^{j-1}$. Obviously, in this case, we can also recycle parts of $\nabla_\alpha(\alpha, \beta)$, $\nabla_\beta(\alpha, \beta)$ and $S(\alpha, \beta)$ such as described in the Procedure 3.

Procedure 3 Initialize by $\alpha \leftarrow \alpha^*$, $\beta \leftarrow \beta^*$, compute gradients and dual gap

$E(\alpha, \beta) \leftarrow 0$
for $i = 1$ to n **do**
$\quad \alpha_i \leftarrow \alpha_i^*$
$\quad \beta_i \leftarrow \beta_i^*$
$\quad \nabla_{\alpha_i}(\alpha, \beta) \leftarrow \nabla_{\alpha_i}(\alpha^*, \beta^*) + \frac{\alpha_i^*}{2\tau}\left(\frac{1}{C^{old}} - \frac{1}{C^{new}}\right)$
$\quad \nabla_{\beta_i}(\alpha, \beta) \leftarrow \nabla_{\beta_i}(\alpha^*, \beta^*) + \frac{\beta_i^*}{2(1-\tau)}\left(\frac{1}{C^{old}} - \frac{1}{C^{new}}\right)$
$\quad \xi_{i,+} \leftarrow \max\left(0, \nabla_{\alpha_i}(\alpha, \beta) + \frac{\alpha_i}{2\tau C^{new}}\right)$
$\quad \xi_{i,-} \leftarrow \max\left(0, -\nabla_{\alpha_i}(\alpha, \beta) - \frac{\alpha_i}{2\tau C^{new}}\right)$
$\quad E(\alpha, \beta) \leftarrow E(\alpha, \beta) + \left(\tau \xi_{i,+}^2 + (1-\tau)\xi_{i,-}^2\right)$
end for
$T(\alpha, \beta) \leftarrow T(\alpha, \beta) - \frac{1}{4}\left(\frac{1}{C^{old}} - \frac{1}{C^{new}}\right) \sum_{i=1}^n \left(\frac{\alpha_i^2}{\tau} + \frac{\beta_i^2}{1-\tau}\right)$
$S(\alpha, \beta) \leftarrow T(\alpha, \beta) + C^{new} E(\alpha, \beta)$

---

4 Working Set of Size Two

The Algorithm 1 performs an update for one coordinate per iteration. In this section, we extend this idea and develop an algorithm to perform an update for *two* coordinates per iteration. For this, we first solve the 2D- problem exactly in the following section. Then we will describe a low cost working set selection strategy based on the 1D-SVM solver. In the end, we establish a stopping criteria for the 2D-problem.

4.1 Exact Solution of Two Dimensional Problem

Let us fix two coordinates $i, j \in \{1, \ldots, n\}$ with $i \neq j$. We further assume that e_i and e_j are the i-th and j-th vectors of standard basis of $\mathbb{R}^n$, and write $\alpha^{\backslash i,j} := \alpha - \alpha_i e_i - \alpha_j e_j$ and $\beta^{\backslash i,j} := \beta - \beta_i e_i - \beta_j e_j$. By this and using $K_{ii} = K_{jj} = 1$ for normalized kernels, the dual objective function for 2D-problem is

$$
\begin{aligned}
\tilde{W} &:= W(\alpha^{\backslash i,j} + \alpha_i e_i + \alpha_j e_j, \beta^{\backslash i,j} + \beta_i e_i + \beta_j e_j) \\
&= W(\alpha^{\backslash i,j}, \beta^{\backslash i,j}) + W(\alpha_i, \beta_i) + W(\alpha_j, \beta_j) - (\alpha_i - \beta_i)(\alpha_j - \beta_j)K_{ij},
\end{aligned}
\tag{28}
$$

where

$$W(\alpha_i, \beta_i) := (\alpha_i - \beta_i)\langle e_i, \mathbf{y} \rangle - (\alpha_i - \beta_i)\langle e_i, K(\alpha^{\backslash i,j} - \beta^{\backslash i,j}) \rangle - \frac{1}{2}(\alpha_i - \beta_i)^2$$
$$- \frac{1}{4C\tau(1-\tau)}((1-\tau)\alpha_i^2 + \tau\beta_i^2),$$

$$W(\alpha_j, \beta_j) := (\alpha_j - \beta_j)\langle e_j, \mathbf{y} \rangle - (\alpha_j - \beta_j)\langle e_j, K(\alpha^{\backslash i,j} - \beta^{\backslash i,j}) \rangle - \frac{1}{2}(\alpha_j - \beta_j)^2$$
$$- \frac{1}{4C\tau(1-\tau)}((1-\tau)\alpha_j^2 + \tau\beta_j^2).$$

Taking partial derivatives of (28) w.r.t. $\alpha_i, \alpha_j, \beta_i$ and β_j, we obtain the gradients

$$\begin{aligned}
\nabla\tilde{W}_{\alpha_i} &= \langle e_i, \mathbf{y} \rangle - \langle e_i, K(\alpha^{\backslash i,j} - \beta^{\backslash i,j}) \rangle - b_1\alpha_i + \beta_i - (\alpha_j - \beta_j)K_{i,j}, \\
\nabla\tilde{W}_{\beta_i} &= -\langle e_i, \mathbf{y} \rangle + \langle e_i, K(\alpha^{\backslash i,j} - \beta^{\backslash i,j}) \rangle + \alpha_i - b_2\beta_i + (\alpha_j - \beta_j)K_{i,j}, \\
\nabla\tilde{W}_{\alpha_j} &= \langle e_j, \mathbf{y} \rangle - \langle e_j, K(\alpha^{\backslash i,j} - \beta^{\backslash i,j}) \rangle - b_1\alpha_j + \beta_j - (\alpha_i - \beta_i)K_{i,j}, \\
\nabla\tilde{W}_{\beta_j} &= -\langle e_j, \mathbf{y} \rangle + \langle e_j, K(\alpha^{\backslash i,j} - \beta^{\backslash i,j}) \rangle + \alpha_j - b_2\beta_j + (\alpha_i - \beta_i)K_{i,j},
\end{aligned} \tag{29}$$

where b_1, b_2 are defined in (13). By setting partial derivatives (29) to zero, we obtain the following system of equations

$$\begin{aligned}
b_1\alpha_i - \beta_i + k\alpha_j - k\beta_j &= c_i, \\
\alpha_i - b_2\beta_i + k\alpha_j - k\beta_j &= c_i, \\
k\alpha_i - k\beta_i + b_1\alpha_j - \beta_j &= c_j, \\
k\alpha_i - k\beta_i + \alpha_j - b_2\beta_j &= c_j,
\end{aligned} \tag{30}$$

where

$$\begin{aligned}
k &:= K_{ij}, \\
c_i &:= \langle e_i, \mathbf{y} \rangle - \langle e_i, K(\alpha^{\backslash i,j} - \beta^{\backslash i,j}) \rangle, \\
&= \nabla W_{\alpha_i}(\alpha, \beta) + b_1\langle \alpha, e_i \rangle - \langle \beta, e_i \rangle + \langle \alpha - \beta, e_j \rangle k, \\
c_i &:= \langle e_j, \mathbf{y} \rangle - \langle e_j, K(\alpha^{\backslash i,j} - \beta^{\backslash i,j}) \rangle \\
&= \nabla W_{\alpha_j}(\alpha, \beta) + b_1\langle \alpha, e_j \rangle - \langle \beta, e_j \rangle + \langle \alpha - \beta, e_i \rangle k.
\end{aligned}$$

Let $\alpha_i^*, \alpha_j^*, \beta_i^*$ and β_j^* be the solution of (30). Then solving (30) by matrix operations leads to the following global solution

$$\begin{aligned}
|M|\,\alpha_i^* &= (b_2 - 1)(b_1b_2 - 1)c_i + (1 - b_2)(b_1 + b_2 - 2)kc_j, \\
|M|\,\beta_i^* &= (1 - b_1)(b_1b_2 - 1)c_i + (b_1 - 1)(b_1 + b_2 - 2)kc_j, \\
|M|\,\alpha_j^* &= (b_2 - 1)(b_1b_2 - 1)c_j + (1 - b_2)(b_1 + b_2 - 2)kc_i, \\
|M|\,\beta_j^* &= (1 - b_1)(b_1b_2 - 1)c_j + (b_1 - 1)(b_1 + b_2 - 2)kc_i.
\end{aligned} \tag{31}$$

Here

$$|M| := b_1^2(b_2^2 - k^2) - 2b_1(b_2k^2 + b2 - 2k^2) - (b_2 - 2)^2k^2 + 1,$$

is always positive. This is shown in the following lemma

Lemma 5. For $b_1, b_2 \in [1, \infty)$ and $|k| \leq 1$, we have $|M| > 0$.

11

Note that, in the case of $c_i = c_j = 0$, we have $\alpha_i^* = \beta_i^* = \alpha_j^* = \beta_j^* = 0$. On the other hand, if $c_i \neq 0$ or $c_j \neq 0$, then (31) together with Lemma 5 leads, after some calculations, to the following equations

$$\alpha_i^* = -\frac{\tau}{1-\tau}\beta_i^*, \qquad \alpha_j^* = -\frac{\tau}{1-\tau}\beta_j^*.$$

Since $\tau \in (0,1)$, the global solution (31) thus violates the constraints of (9) iff $c_i \neq 0$ or $c_j \neq 0$, that is, the solution is not feasible. To obtain the feasible solution, we know by the Theorem 2 that we need to look at the boundaries of the feasible region. In our case, this means that we need to set some of the dual variables to zero. Note that this is a simple extension of the idea that is presented in 1D-problem. Let us begin by setting one dual variable to zero, say $\alpha_i = 0$. Computing the gradients with the remaining variables, we get the last three expressions of (29) where we set $\alpha_i = 0$. After setting the gradients to zero, we obtain the system of equations

$$\begin{aligned}
-b_2\beta_i + k\alpha_j - k\beta_j &= c_i\,, \\
-k\beta_i + b_1\alpha_j - \beta_j &= c_j\,, \\
-k\beta_i + \alpha_j - b_2\beta_j &= c_j\,,
\end{aligned} \tag{32}$$

where k, c_i, c_j, b_1 and b_2 are the same as in (30). Let us write α_j^+, β_i^+ and β_j^+ be the solution of (32). Then, by subtracting the last two equations of (32), we obtain

$$\alpha_j^+ = -\frac{\tau}{1-\tau}\beta_j^+\,, \tag{33}$$

and hence this solution is again not feasible. In a similar way, setting $\beta_i = 0$ provides the following system of equations

$$\begin{aligned}
b_1\alpha_i + k\alpha_j - k\beta_j &= c_i\,, \\
k\alpha_i + b_1\alpha_j - \beta_j &= c_j\,, \\
k\alpha_i + \alpha_j - b_2\beta_j &= c_j\,,
\end{aligned}$$

which again leads to (33) and thus the same conclusion. The remaining two cases where $\alpha_j = 0$ and $\beta_j = 0$ can be treated analogously. After this, we now consider the situation where two variables are set to zero. For this, we split the problem into six subcases. Let us consider the first subcase where we set $\alpha_i = 0$ and $\beta_i = 0$ in (28). Taking derivatives w.r.t. α_j and β_j provides

$$\begin{aligned}
\nabla W_{\alpha_j}(\alpha^{\backslash i}, \beta^{\backslash i}) &= \langle e_j, \mathbf{y}\rangle - \langle e_j, K(\alpha^{\backslash i,j} - \beta^{\backslash i,j})\rangle + \beta_j - b_1\alpha_j\,, \\
\nabla W_{\beta_j}(\alpha^{\backslash i}, \beta^{\backslash i}) &= -\langle e_j, \mathbf{y}\rangle + \langle e_j, K(\alpha^{\backslash i,j} - \beta^{\backslash i,j})\rangle + \alpha_j - b_2\beta_j\,.
\end{aligned} \tag{34}$$

Setting (34) to zero, we obtain the system of equations

$$\begin{aligned}
b_1\alpha_j - \beta_j &= c_j\,, \\
\alpha_j - b_2\beta_j &= c_j\,.
\end{aligned} \tag{35}$$

Let α_j^+ and β_j^+ be the solution of (35). Then subtracting equations of (35) leads to

$$\alpha_j^+ = -\frac{\tau}{1-\tau}\beta_j^+\,, \tag{36}$$

which shows that the solution is not feasible. Analogously, the second subcase where $\alpha_j = 0$ and $\beta_j = 0$ leads to the same conclusion. In the third subcase, we set $\alpha_i = 0$ and $\alpha_j = 0$ in (28) and differentiate w.r.t. β_i and β_j which gives

$$\begin{aligned}
\nabla W_{\beta_i}(\alpha^{\backslash i,j}, \beta) &= -\langle e_i, \mathbf{y}\rangle + \langle e_i, K(\alpha^{\backslash i,j} - \beta^{\backslash i,j})\rangle - \beta_j K_{ij} - b_2\beta_i\,, \\
\nabla W_{\beta_j}(\alpha^{\backslash i,j}, \beta) &= -\langle e_j, \mathbf{y}\rangle + \langle e_j, K(\alpha^{\backslash i,j} - \beta^{\backslash i,j})\rangle - \beta_i K_{ij} - b_2\beta_j\,.
\end{aligned} \tag{37}$$

Setting (37) to zero, we obtain a system of equations which, after some calculations, provides the solution

$$\alpha_i^+ = 0 \,, \quad \alpha_j^+ = 0 \,, \quad \beta_i^+ = \left|B_1\right|^{-1} (kc_j - b_2c_i) \,, \quad \beta_j^+ = \left|B_1\right|^{-1} (kc_i - b_2c_j), \tag{38}$$

where $\left|B_1\right| := b_2^2 - k^2 > 0$. Considering the forth subcase, we set $\beta_i = 0$ and $\beta_j = 0$. Analogous to third subcase, the gradients are

$$\nabla W_{\alpha_i}(\alpha, \beta^{\backslash i,j}) = \langle e_i, \mathbf{y}\rangle - \langle e_i, K(\alpha^{\backslash i,j} - \beta^{\backslash i,j})\rangle - \alpha_j K_{ij} - b_1\alpha_i \,,$$

$$\nabla W_{\alpha_j}(\alpha, \beta^{\backslash i,j}) = \langle e_j, \mathbf{y}\rangle - \langle e_j, K(\alpha^{\backslash i,j} - \beta^{\backslash i,j})\rangle - \alpha_i K_{ij} - b_1\alpha_j \,,$$

which leads to the solution

$$\beta_i^+ = 0 \,, \quad \beta_j^+ = 0 \,, \quad \alpha_i^+ = \left|B_2\right|^{-1} (b_1c_i - kc_j) \,, \quad \alpha_j^+ = \left|B_2\right|^{-1} (b_1c_j - kc_i) \,, \tag{39}$$

where $\left|B_2\right| := b_1^2 - k^2 > 0$. For fifth subcase, we set $\alpha_i = 0$ and $\beta_j = 0$ and obtain the following solution

$$\alpha_i^+ = 0 \,, \quad \beta_j^+ = 0 \,, \quad \beta_i^+ = \left|B_3\right|^{-1} (b_1c_i - kc_j) \,, \quad \alpha_j^+ = \left|B_3\right|^{-1} (kc_i - b_2c_j) \,, \tag{40}$$

where $\left|B_3\right| := k^2 - b_1b_2 < 0$. Finally, for the last subcase where $\alpha_j = 0$ and $\beta_i = 0$, the solution can be obtained by interchanging i with j in the solution of fifth subcase, which is

$$\beta_i^+ = 0 \,, \quad \alpha_j^+ = 0 \,, \quad \alpha_i^+ = \left|B_3\right|^{-1} (kc_j - b_2c_i) \,, \quad \beta_j^+ = \left|B_3\right|^{-1} (b_1c_j - kc_i) \,. \tag{41}$$

It is interesting to note that the solutions (38), (39), (40) and (41) have the following common expressions

$$T_1 := kc_j - b_2c_i \,, \tag{42}$$
$$T_2 := kc_i - b_2c_j \,, \tag{43}$$
$$T_3 := b_1c_i - kc_j \,, \tag{44}$$
$$T_4 := b_1c_j - kc_i \,. \tag{45}$$

The following lemma investigates the behavior of the above four expressions.

Lemma 6. *Assume that $c_i \neq 0$ or $c_j \neq 0$. Then the following implications hold:*

   i) If $T_1 \geq 0$ and $T_2 \geq 0$ then we have $c_i < 0$ and $c_j < 0$.

   ii) If $T_3 \geq 0$ and $T_4 \geq 0$ then we have $c_i > 0$ and $c_j > 0$.

In particular, the expressions T_1, T_2, T_3 and T_4 are not simultaneously positive or negative.

Using Lemma 6, the following theorem shows that only one case from (38), (39), (40) and (41) provides the feasible optimal solution.

Theorem 7. *Assume that $c_i \neq 0$ or $c_j \neq 0$, then exactly one of the four cases (38), (39), (40) and (41) produces a feasible solution. Moreover, the following implications hold:*

   i) If $T_1 \geq 0$ and $T_2 \geq 0$, then (38) is the feasible solution.

   ii) If $T_3 \geq 0$ and $T_4 \geq 0$, then (39) is the feasible solution.

   iii) If $T_1 \leq 0$ and $T_3 \leq 0$, then (40) is the feasible solution.

   iv) If $T_2 \leq 0$ and $T_4 \leq 0$, then (41) is the feasible solution.

Theorem 7 also suggests to impose *if* conditions based on expressions (42), (43), (44) and (45) in the implementation of the algorithm for 2D-SVM solver. This helps to reach directly to the feasible optimum solution.

4.2 Working Set Selection Strategies

In this section, we address the question how to choose the directions i^* and j^* in which the 2D-SVM solver performs an update. Several possibilities are available for this task. A straightforward approach is to consider all pairs of directions (i, j) and choose the one for which the 2D-gain of W is maximum. Note that the 2D-gain is simply an extension of the idea presented in Lemma 4. Formally, for $\alpha, \beta \in \mathbb{R}^n$ and $\delta, \eta \in \mathbb{R}$, the 2D-gain is

$$W(\alpha + \delta e_i + \delta e_j, \beta + \eta e_i + \eta e_j) - W(\alpha, \beta) = G(\delta_i, \eta_i) + G(\delta_j, \eta_j) - (\delta_i - \eta_i)(\delta_j - \eta_j)K_{i,j}, \quad (46)$$

where $G(\delta_k, \eta_k)$ for $k = i, j$ is the 1D-gain defined in Lemma 4.

It is worth noting that the above described working set selection strategy is not a good choice because the search is $O(n^2)$. However it may be viewed as an "optimal" two dimensional strategy and served as a baseline to all other subset selection strategy that can be interpreted as the low cost approximations to this approach. In the following, we describe two low cost working set selection strategies that we consider in this work.

WSS 1: Two 1D-direction With Maximal Gain From Separate Subsets. A simple way to preserve the low cost search from 1D-solver is to split the index set $\{1, \ldots, n\}$ into two parts $\{1, \ldots, \frac{n}{2}\}$ and $\{\frac{n}{2} + 1, \ldots, n\}$ and search for the 1D directions with maximum gain over these two parts separately. In other words, we can choose the directions i^* and j^* by

$$
\begin{aligned}
i^* &\in \arg \max_{i \le n/2} W(\alpha + \delta e_i, \beta + \eta e_i) - W(\alpha, \beta), \\
j^* &\in \arg \max_{i > n/2} W(\alpha + \delta e_i, \beta + \eta e_i) - W(\alpha, \beta),
\end{aligned}
\quad (47)
$$

where δ and η are defined in 1D-SVM solver. These chosen directions are used for the first iteration. For the subsequent iterations, we search for the new 1D directions, i^*_{new} and j^*_{new}, again by using (47). Then we compute the 2D-gain of W for all pairs of old and new directions of previous and current iterations respectively and look for a pair for which this gain is maximum.

WSS 2: 1D-direction With Maximal Gain And A Direction Of A Nearby Sample. This is simply an extension of *WSS 1*. After determining (i^*, j^*) by *WSS 1*, we fix i^* and then search for another direction j^* from k-nearest neighbors of x_{i^*} with respect to the metric

$$d(x, x') := \|x - x'\|^2.$$

4.3 Stopping Criteria

To formulate the stopping criteria for 2D-problem, we follow the idea that is presented in Section 3.1. Let us first consider the component $T(\alpha, \beta)$ of (22) and by using (46), we find the following update of $T(\alpha, \beta)$ in the directions of i and j

$$
\begin{aligned}
T(\alpha + \delta e_i + \delta e_j, \beta + \eta e_i + \eta e_j) = {}&T(\alpha, \beta) - U(\alpha_i, \beta_i, \delta_i, \eta_i) - U(\alpha_j, \beta_j, \delta_j, \eta_j) \\
&+ 2(\delta_i - \eta_i)(\delta_j - \eta_j)K_{i,j},
\end{aligned}
$$

where $U(\alpha_k, \beta_k, \delta_k, \eta_k)$ for $k = i, j$ is defined in (23). To compute $E(\alpha, \beta)$, we first obtain the updated gradients in the directions of i and j, and then subsequently compute $\xi_{l,+}, \xi_{l,-}$. Moreover, $\widehat{E}(\alpha, \beta)$ can also be computed for the 2D-problem similar to 1D-problem by using (26). With all above computations, we now summarize the idea of 2D-SVM solver in Algorithm 2.

Algorithm 2 2D-SVM Solver

---
    initialize $\alpha, \beta, \nabla W_\alpha(\alpha, \beta), \nabla W_\beta(\alpha, \beta)$ and $T(\alpha, \beta)$
    while $S(\alpha, \gamma) > \frac{\varepsilon}{2\alpha}$ **do**
        select directions i^* and j^*
        use procedure 5 to obtain the optimum solution for direction i^* and j^*
        update α and β in the direction i^* and j^*
        update $\nabla W_\alpha(\alpha, \beta), \nabla W_\beta(\alpha, \beta)$ in the directions (i^*, j^*) and calculate $S(\alpha, \beta)$
    end while

---

5 Experiments

To evaluate the performance of the proposed solver for expectile regression, we perform several experiments to address the following questions:

1. Which subset selection strategy leads to the smallest number of iterations or shortest run time?

2. What is the number of nearest neighbors that leads to the smallest number of iterations and shortest run time?

3. Is there advantage of warm start initialization when the parameter search is performed over a grid?

4. Does the clipping provide a significant reduction in the training time and iterations?

5. How well does the 2D-SVM-solver work as compared to ER-Boost that is proposed by [44]?

To answer these questions, we implemented the 2D-SVM-solver in C^{++}. The algorithm was compiled by LINUX's gcc version 4.7.2 with various software and hardware optimization enabled. All experiments were conducted on a computer with INTEL CORE i7 950 (3.07 GHz) and 8GB RAM under 64bit version of Debian Linux 7.8 (Debian 3.2.0-4-amd64). During all experiments that incorporated measurement of run time, one core was used solely for the experiments, and the number of other processes running on the system were minimized.

In order to perform the experiments, we have considered nine data sets that were downloaded from different sources. These data sets comprises various number of features and vary in sample sizes from 630 to 20639. The data sets CONCRETE-COMP, UPDRS-MOTOR, CYCLE-PP, AIRFOIL-NOISE and HOUR were downloaded from UCI repository. The two data sets NC-CRIME and HEAD-CIRCUM are available and documented in R packages Ecdat and AGD respectively. The remaining two data sets CAL-HOUSING and MUNICH-RENT were downloaded from StatLib and the data archive of the Institute of Statistics, Ludwig-Maximilians-University of Munich respectively. We scaled the data sets componentwise such that all the samples including labels lie in $[-1, 1]^{d+1}$, where d is the dimension of the input data. In addition to that, we generated a random split for all data sets that contained approximately 70% training and 30% test samples. Table 1 describes the characteristics of the considered data sets.

In all our experiments with the SVM solver, we used Gaussian kernels (4). To determine the hyper-parameters, we have considered a geometrically spaced 10 by 10 grid for λ and γ over the interval $[c_1 n^{-1}, 1]$ and $[c_2 n^{-1/d}, c_3]$ respectively, where n is the number of training samples, d is the input dimension, and $c_1 := 0.001, c_2 := 0.1$ and $c_3 := 0.2$. Here, the values of the constants were chosen with the help of our experience with least square SVMS [11]. To choose the best values of

these hyper-parameters, we used k-fold cross validation with randomly generated folds. In our case, we have considered $k = 5$. During the k-fold cross validation, the hyper-parameter γ was internally converted to $\tilde{\gamma} := \frac{(k-1)n\gamma}{k}$ and λ to $C := \frac{k}{2(k-1)n\lambda}$, where $(k-1)n/k$ is approximate *actual* training set size for k-fold cross validation.

data	sample sizes	training size	test size	dimension
NC-CRIME	630	441	189	19
CONCRETE-COMP	1030	721	309	8
AIRFOIL-NOISE	1503	1052	451	5
MUNICH-RENT	2053	1437	616	12
UPDRS-MOTOR	5875	4112	1763	19
HEAD-CIRCUM	7020	4914	2106	4
CYCLE-PP	9568	6697	2871	5
HOUR	17379	12165	5214	12
CAL-HOUSING	20639	14447	6192	8

Table 1: Characteristics of data sets together with the training sizes and the test sizes that refer to the splits used in the run time experiments.

Let us now explore the answers of the above stated questions one by one. To address the first question, we performed experiments with warm start initialization method and clipped duality gap. In addition, we have considered $N = 15$ nearest neighbors for WSS 2. The results are presented in Figure 2 and 3, which depict that WSS 2 needs substantially less iterations as well as training time than WSS 1 on all data sets. For larger data sets such as UPDRS-MOTOR, HEAD-CIRCUM, CYCLE-PP, HOURS and CAL-HOUSING, the run time and iterations with WSS 2 is at least 50% less than WSS 1. Moreover, a closer analysis, see Figure 4 and 5 shows that the savings are obtained at the hyper-parameters pairs for which training is particularly expensive, that is, for small λ and medium to small γ.

We have fixed $N = 15$ for WSS 2 so far to address the previous question. To investigate how the computational requirements change with the number of nearest neighbors, we performed the experiments for N-nearest neighbors, where $N = 5, 10, 15, 20, 25, 30, 35, 40$ for each $\tau = 0.25, 0.50, 0.75$. Again we used warm start initialization and clipped duality gap. Here, it was observed that the number of iterations tends to decrease with increasing N. However, for $N \geq 25$, only a slight improvement in the number of iterations was found whereas the required run time tended to increase compared to smaller N. We therefore plotted the results for $N = 5, 10, 15, 20$ only. Figure 6 shows that the solver attains the minimum training time for $N = 15$ on almost all data sets. Moreover, Figure 7 shows that the number of iterations decreases with increasing N. However, this decrease becomes negligible when $N \geq 15$. All this together leads us to conclude that $N = 15$ is the best choice for our ER-SVM solver. Finally, Figure 8 and 9 illustrate the computational requirements for different hyper-parameters pairs. Again the largest savings for $N = 15$ were obtained for small λ.

To answer the third question regarding the initialization methods, we trained with $N = 15$ and clipped duality gap. The results, which are presented in Figure 10 and 11 show that using the warm start initialization saves between 20% and 40% of both training time and iterations. The detailed behavior for different hyper-parameter pairs is illustrated in Figure 12 and 13. Again the savings are more pronounced for smaller λ.
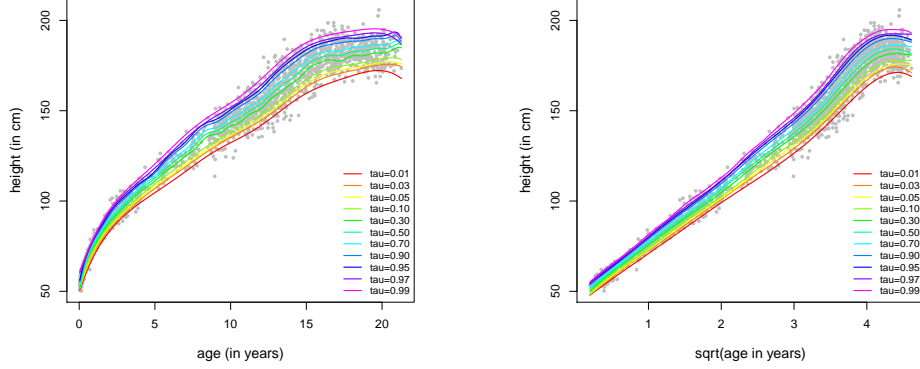
Figure 1: Estimated expectiles for $\tau = 0.01, 0.03, 0.05, 0.10, 0.30, 0.50, 0.70, 0.90, 0.95, 0.97, 0.99$ for height against age of HEAD-CIRCUM. The graphs comprises expectile curves for original data set (left) and data set with transformed age.

To answer the forth question, we considered stopping criteria with clipped duality gap and with unclipped duality gap. Here, we used the warm start initialization option and WSS 2 with $N = 15$ nearest neighbors. The corresponding results are shown in Figures 14 and 15. In the case of hinge loss function, [35] showed that using the clipped duality gap yields significant reduction, both in run times and iterations. However, in our case, we get only a small reduction in iterations, that is, 1% on almost all data sets. On the other hand, this stopping criteria causes 2% to 17% increase in run times on different data sets. This indicates that the unclipped duality gap is the better choice in our case. The per grid plot of hyper-parameters for data set CAL-HOUSING, as presented in Figure 16 and 17, shows that clipping reduces the run time only for few pairs of hyper-parameter when λ is small and γ is large. For rest of the pairs, unclipped duality gap leads to smaller run time.

Finally, we compare our SVM solver with ER-Boost on the basis of test error and training time. For this, we considered our 2D-SVM solver with unclipped duality gap (ER-SVM), our 2D-SVM solver with clipped duality gap (ER-SVM*) and ER-Boost [44]. Since the experiments using large data sets entail long run times, we splitted the data sets into three categories, namely, small ($n < 5000$), medium ($5000 \leq n < 10000$) and large ($n \geq 10000$). We then conducted experiments for ER-SVM, ER-SVM* and ER-Boost by repeating 5-fold cross validation 25, 10 and 5 times for the small, medium and large data sets respectively. For the 2D-SVM solvers, we used the 10 by 10 default grid of hyper-parameters described above. For ER-Boost, we used the default value of boosting steps ($M = 100$) and performed 5 fold cross validation to choose the best value of the interaction level (L) between variables, as by the ER-Boost manual. The resulting, average test error (standard deviation) and training time are shown in Table 2 and Table 3 respectively. It turns out that both SVMs solvers have better test performance than ER-Boost on all data sets, but all reported errors are relatively small. Examining the achieved training times for each data set, we observe that SVM solvers are more sensitive to the training set size and less sensitive to the dimensions of data set, whereas, ER-Boost behaves the other way around. In addition to that, the test performance of ER-SVM* is slightly better than ER-SVM at the cost of almost 10% longer training times.

In the end, Figure 1 presents the expectile curves for different τ considering height against age from data set HEAD-CIRCUM. On the left we see some crossing and wiggling problems. Following [24], the use of square root transformation on age resolves these issues as the right figure shows.

DATA	$\tau = 0.25$			$\tau = 0.50$			$\tau = 0.75$		
	ER-SVM	ER-SVM*	ER-Boost	ER-SVM	ER-SVM*	ER-Boost	ER-SVM	ER-SVM*	ER-Boost
NC-CRIME	0.00616 (0.00182)	0.00555 (0.00169)	0.00948 (0.00177)	0.00669 (0.00194)	0.00605 (0.00161)	0.01367 (0.00305)	0.00536 (0.00172)	0.00509 (0.00157)	0.01459 (0.00405)
CONCRETE-COMP	0.00901 (0.00130)	0.00893 (0.00128)	0.03961 (0.00365)	0.01021 (0.00122)	0.01013 (0.00117)	0.05038 (0.00417)	0.00889 (0.00112)	0.00879 (0.00101)	0.04556 (0.00339)
AIRFOIL-NOISE	0.00814 (0.00121)	0.00806 (0.00119)	0.04223 (0.00211)	0.00947 (0.00134)	0.00939 (0.00115)	0.04817 (0.00256)	0.00855 (0.00092)	0.00850 (0.00087)	0.03832 (0.00218)
MUNICH-RENT	0.00131 (0.00033)	0.00126 (0.00030)	0.01569 (0.00087)	0.00122 (0.00029)	0.00121 (0.00029)	0.01812 (0.00113)	0.00101 (0.00018)	0.00101 (0.00016)	0.01598 (0.00103)
UPDRS-MOTOR	0.02518 (0.00152)	0.02502 (0.00152)	0.05345 (0.00069)	0.02844 (0.00159)	0.02828 (0.00152)	0.06257 (0.001496)	0.02585 (0.00166)	0.02569 (0.00169)	0.015229 (0.001787)
HEAD-CIRCUM	0.00323 (0.00008)	0.00323 (0.00008)	0.02419 (0.00047)	0.00390 (0.00011)	0.00390 (0.00011)	0.02482 (0.00057)	0.00333 (0.00009)	0.00333 (0.00096)	0.01855 (0.00045)
CYCLE-PP	0.00420 (0.00009)	0.00421 (0.00011)	0.03588 (0.00079)	0.00516 (0.000197)	0.00516 (0.00019)	0.04536 (0.00097)	0.00479 (0.00027)	0.00477 (0.000289)	0.03930 (0.00076)
HOUR	0.01575 (0.00029)	0.01543 (0.00034)	0.02888 (0.00077)	0.01664 (0.00046)	0.01627 (0.00043)	0.04021 (0.00110)	0.01285 (0.00031)	0.01259 (0.00035)	0.03821 (0.00103)
CAL-HOUSING	0.02426 (0.00126)	0.02415 (0.00117)	0.05406 (0.00135)	0.02546 (0.00123)	0.02518 (0.00119)	0.07473 (0.00158)	0.01919 (0.00071)	0.01912 (0.00064)	0.07337 (0.00144)

Table 2: Average test error (standard deviation) for 2D-SVM with unclipped duality gap stopping criteria (ER-SVM), 2D-SVM with clipped duality gap stopping criteria (ER-SVM*) and ER-Boost. The average test error (standard deviation) was computed on 25 random splits for small data sets, 10 random splits for medium size data sets and 5 random split for larger size data sets.

DATA	$\tau = 0.25$			$\tau = 0.50$			$\tau = 0.75$		
	ER-SVM	ER-SVM*	ER-Boost	ER-SVM	ER-SVM*	ER-Boost	ER-SVM	ER-SVM*	ER-Boost
NC-CRIME	0.305	0.317	20.954	0.323	0.318	21.545	0.298	0.311	21.595
CONCRETE-COMP	0.983	1.028	1.861	1.027	1.089	1.899	0.964	1.018	1.8025
AIRFOIL-NOISE	2.078	2.173	0.645	2.234	2.342	0.656	2.122	2.232	0.649
MUNICH-RENT	2.413	2.485	9.288	2.385	2.476	9.542	2.364	2.426	9.460
UPDRS-MOTOR	43.874	46.737	110.853	47.819	47.819	114.967	42.614	45.537	114.335
HEAD-CIRCUM	34.352	36.173	1.7826	36.928	39.029	1.7529	36.744	37.256	1.796
CYCLE-PP	83.452	85.893	2.7473	91.127	93.897	2.758	85.690	87.309	2.714
HOUR	307.249	318.357	70.376	315.692	327.897	69.576	281.972	288.479	68.536
CAL-HOUSING	506.679	529.945	39.913	535.835	550.364	39.974	458.223	479.735	38.880

Table 3: Training time (in seconds) for 2D-SVM with unclipped duality gap stopping criteria (ER-SVM), 2D-SVM with clipped duality gap stopping criteria (ER-SVM*) and ER-Boost.

A Proofs

The proofs of Lemma 4 and Lemma 5 are trivial and therefore omitted. The rest of the proofs are given below.

Proof of Theorem 1. The first convergence follows from [32, Theorem 9.1] and the second convergence is a consequence of the first convergence and [32, Corollary 3.62], where we note that we do not need the completeness of $\mathbf{X}$ since we already know the existence and uniqueness of $f_{L,P}^*$. $\qquad\square$

Proof of Theorem 2. i) We first show that W has a global maximum at α^*. To do this, we proceed by contradiction, that is, we assume that there exists an $\alpha \in \mathbb{R}^m$ with

$$W(\alpha^*) < W(\alpha). \tag{48}$$

By concavity of W, we conclude that for $t \in [0,1]$

$$W((1-t)\alpha^* + t\alpha) \geq (1-t)W(\alpha^*) + tW(\alpha). \tag{49}$$

On the other hand, $h := t(\alpha - \alpha^*) \in \mathbb{R}^m$ and Taylor's theorem in the multiple dimensional version yields

$$W((1-t)\alpha^* + t\alpha) = W(\alpha^* + h),$$

$$= W(\alpha^*) + \langle W'(\alpha^*), h \rangle + \frac{1}{2}\langle h, W''(\alpha^*)h \rangle + O(\| h^2 \|),$$

$$= W(\alpha^*) + \frac{t^2}{2}\langle \alpha - \alpha^*, W''(\alpha^*)(\alpha - \alpha^*) \rangle + O(t^2).$$

Using this in (49) we obtain

$$W(\alpha^*) + t\big(W(\alpha) - W(\alpha^*)\big) \leq W(\alpha^*) + \frac{t^2}{2}\langle \alpha - \alpha^*, W''(\alpha^*)(\alpha - \alpha^*) \rangle + O(t^2), \tag{50}$$

and thus

$$c_1 t \leq \frac{c_2}{2}t^2 + O(t^2),$$

where $c_1 := W(\alpha) - W(\alpha^*)$ and $c_2 := \langle \alpha - \alpha^*, W''(\alpha^*)(\alpha - \alpha^*) \rangle$. Furthermore, we have $c_2 \leq 0$ since W is concave and $c_1 > 0$ by (48). For sufficiently small $t > 0$, (50) is therefore impossible and hence (48) can not be true. Let us now show that W has no other global maximum. To show this, we assume the converse, that is, W has a global maximum at some $\alpha^{**} \neq \alpha^*$. Then we obtain $W'(\alpha^{**}) = 0$ by usual calculus, and hence our assumptions are violated. Consequently, W has its only global maximum at α^*.

ii) If $\alpha^* \notin \mathcal{A}$ then we also have $\alpha^* \notin \mathring{\mathcal{A}}$, where $\mathring{\mathcal{A}}$ denotes the interior of $\mathcal{A}$, and for $\alpha \in \mathring{\mathcal{A}}$ we thus have $\alpha \neq \alpha^*$. Let us now show that for all $\alpha \in \mathring{\mathcal{A}}$ there exists an $\alpha^\star \in \partial\mathcal{A}$ with

$$W(\alpha^\star) > W(\alpha). \tag{51}$$

To this end, we fix an $\alpha \in \mathring{\mathcal{A}}$ and consider the function

$$\gamma : [0,1] \to \mathbb{R}^m$$
$$t \mapsto (1-t)\alpha^* + t\alpha.$$

Furthermore, we set

$$h := W \circ \gamma.$$

19

Then it is easy to see that h is concave. Moreover, since $\alpha \neq \alpha^*$, we find $\gamma(t) \neq \alpha^*$ for all $t \in (0, 1]$ and thus $h(t) < h(0)$ for all $t \in (0, 1]$. By the concavity of h we conclude that h is strictly decreasing. We now show that there exists $t^\star \in (0, 1]$ with $\gamma(t^\star) \in \partial\mathcal{A}$. Let us assume the converse, that is, $\Gamma \cup \partial\mathcal{A} = \emptyset$, where $\Gamma := \gamma([0, 1])$. Considering the partition $\mathring{\mathcal{A}}$, $\partial\mathcal{A}$, $\mathbb{R}^m \backslash \bar{\mathcal{A}}$, where $\bar{\mathcal{A}}$ denotes the closure of $\mathcal{A}$, we then find by the assumed $\mathcal{A} = \bar{\mathcal{A}}$ and $\Gamma \cup \partial\mathcal{A} = \emptyset$ that

$$B_1 := \Gamma \cap \mathring{\mathcal{A}}$$
$$B_2 := \Gamma \cap \mathbb{R}^m \backslash \bar{\mathcal{A}} = \Gamma \cap (\mathbb{R}^m \backslash \mathcal{A}),$$

is a partition of Γ. Since $\alpha \in \mathring{\mathcal{A}}$ and $\alpha^* \notin \mathcal{A}$, we further find $B_1 \neq \emptyset$ and $B_2 \neq \emptyset$. Moreover, since $\mathbb{R}^m \backslash \mathcal{A}$ is open, the sets B_1 and B_2 are relatively open in Γ and Γ. However, the continuous image of a connected set, is connected and thus Γ is connected. This leads to a contradiction, and hence there exists a $t^\star \in [0, 1]$ with $\gamma(t^\star) \in \partial\mathcal{A}$. Clearly, we have $t^\star < 1$ since $\alpha \notin \partial\mathcal{A}$. For $\alpha^\star := \gamma(t^\star)$, the already established strict monotonicity of h then shows

$$W(\alpha^\star) = h(t^\star) > h(1) = W(\alpha).$$

Consequently we have shown (51) and thus

$$\sup_{\alpha \in \partial\mathcal{A}} W(\alpha) = \sup_{\alpha \in \mathcal{A}} W(\alpha).$$

In other words, it suffices to show that the supremum over $\partial\mathcal{A}$ is attained at some $\alpha^\star \in \partial\mathcal{A}$. To this end, we first show that $\{W \geq \rho\}$ is bounded for all $\rho < W^* := W(\alpha^*)$. For $\alpha \in S$, where $S \subset \mathbb{R}^m$ denotes the Euclidean unit sphere, we define

$$\begin{aligned} h_\alpha : [0, \infty) &\to \mathbb{R}^m \\ t &\mapsto W(\alpha^* + t\alpha). \end{aligned}$$

Then h_α is concave and continuously differentiable, and has a global maximum at $t = 0$. Moreover, h_α is strictly decreasing with $\lim_{t \to \infty} h_\alpha(t) = -\infty$. We define

$$t_\alpha := \max\{t \geq 0 : h_\alpha(t) \geq \rho\},$$

where we note that the maximum is indeed attained by the continuity of h_α and $t_\alpha < \infty$. Our next intermediate goal is to show that $\alpha \mapsto t_\alpha$ is continuous. To this end, we fix an $\alpha_0 \in S$, and an $\varepsilon > 0$ with $\sqrt{\varepsilon} < -h'_{\alpha_0}(t_{\alpha_0})$, where we note that $h'_{\alpha_0}(t_{\alpha_0}) < 0$ since h_{α_0} is strictly decreasing and $W^* > \rho$. Since W is continuous differentiable, then there exist a $\delta > 0$ such that for all $\alpha \in S$ with $\|\alpha_0 - \alpha\| \leq \delta$ we have

$$|h'_{\alpha_0}(t_{\alpha_0}) - h'_\alpha(t_{\alpha_0})| \leq \varepsilon.$$

For $t_\alpha \geq t_{\alpha_0}$, the concavity, or more precisely, the subdifferential inequality of $-h'_\alpha(t_{\alpha_0})$, then gives

$$\begin{aligned} h_\alpha(t_\alpha) &\leq h_\alpha(t_{\alpha_0}) + h'_\alpha(t_{\alpha_0})(t_\alpha - t_{\alpha_0}), \\ &\leq h_{\alpha_0}(t_{\alpha_0}) + \varepsilon + (h'_{\alpha_0}(t_{\alpha_0}) + \varepsilon)(t_\alpha - t_{\alpha_0}), \\ &\leq h_{\alpha_0}(t_{\alpha_0}) + \varepsilon + \frac{1}{2}h'_{\alpha_0}(t_{\alpha_0})(t_\alpha - t_{\alpha_0}). \end{aligned}$$

Now recall that $h_\alpha(t_\alpha) = \rho = h_{\alpha_0}(t_{\alpha_0})$. Thus we obtain

$$0 \leq \varepsilon + \frac{1}{2}h'_{\alpha_0}(t_{\alpha_0})(t_\alpha - t_{\alpha_0}),$$

20

and since $h'_{\alpha_0}(t_{\alpha_0}) < 0$, we conclude that

$$\frac{-2\varepsilon}{h'_{\alpha_0}(t_{\alpha_0})} \geq t_\alpha - t_{\alpha_0} \,,$$

and thus

$$t_\alpha \leq t_{\alpha_0} + \frac{-2\varepsilon}{h'_{\alpha_0}(t_{\alpha_0})} \leq 2\sqrt{\varepsilon} \,.$$

Since an analogous bound can be established in the case $t \leq t_{\alpha_0}$, we conclude that $\alpha \mapsto t_\alpha$ is continuous. Consequently, there exist an $\alpha_0 \in S$ with $t_{\alpha_0} = \sup_{\alpha \in S} t_\alpha$, and thus $\{W \geq \rho\}$ is bounded.

Now we show that there exist $\alpha^\star \in \mathcal{A}$ with

$$W^\star := \sup_{\alpha \in \mathcal{A}} W(\alpha) = W(\alpha^\star) \,.$$

Clearly there is an $(\alpha_n) \subset \mathcal{A}$ with

$$W(\alpha_n) \to W^\star \,,$$

and since $\{W \geq \rho\}$ is bounded, the sequence α_n is also bounded. Then there is a subsequence α_{n_k} and an $\alpha^\star$ with $\alpha_{n_k} \to \alpha^\star$ and the continuity of W yields $W(\alpha_{n_k}) \to W(\alpha^\star)$. Consequently, we have shown $W(\alpha^\star) = W(\alpha)$. Finally $\alpha^\star = \lim \alpha_{n_k} \in \mathcal{A}$ follows from $\mathcal{A} = \bar{\mathcal{A}}$. $\qquad\square$

Proof of Theorem 3. If $c_i = 0$, there is nothing to prove. Let us assume that $c_i > 0$. Since $b_1, b_2 \in (1, \infty)$, then only (17) provides a feasible solution $c_i > 0$ because $\beta_i^+ < 0$ in (16). Similarly, if we assume that $c_i < 0$, then $\alpha_i^+ < 0$ in (17) while $\beta_i^+ > 0$ in (16) which makes it feasible solution. We finally conclude that only one of two cases provides the feasible optimal solution when $c_i \neq 0$. $\qquad\square$

Proof of Lemma 6. i) Since $T_1 \geq 0$ and $T_2 \geq 0$, we have $\frac{b_2}{k} c_i \leq c_j \leq \frac{k}{b_2} c_i$. Since we assumed that $c_i \neq 0$ or $c_j \neq 0$, we conclude from the latter and $b_2, k \geq 0$ that we actually have $c_i \neq 0$ and $c_j \neq 0$. Moreover, $b_2 > 1$ and $k \leq 1$ shows that $c_i < 0$ and $c_j < 0$.

ii) Since $T_3 \geq 0$ and $T_4 \geq 0$, we have $\frac{k}{b_1} c_i \leq c_j \leq \frac{b_1}{k} c_i$. Since we assumed that $c_i \neq 0$ or $c_j \neq 0$, we conclude from the latter and $b_1, k \geq 0$ that we actually have $c_i \neq 0$ and $c_j \neq 0$. Moreover, $b_1 > 1$ and $k \leq 1$ shows that $c_i > 0$ and $c_j > 0$.

Finally, this leads to conclude that T_1, T_2, T_3 and T_4 are not simultaneously positive. By similar arguments, it can be shown that these expressions are not simultaneously negative. $\qquad\square$

Proof of Theorem 7. Our first goal is to show that at most one of the four cases (38), (39), (40) and (41) leads to a feasible solution. To this end we note that (38) is feasible if and only if T_1 and T_2 are non-negative. Similar consideration from (39) to (41) leads to the Table 4. Now let

Optimal Solution	T_1	T_2	T_3	T_4
(38) feasible	≥ 0	≥ 0	–	–
(39) feasible	–	–	≥ 0	≥ 0
(40) feasible	–	≤ 0	≤ 0	–
(41) feasible	≤ 0	–	–	≤ 0

Table 4: Behavior of expressions T_1, T_2, T_3 and T_4 when any optimal solutions is feasible

assume that (38) is feasible. By Lemma 6 we then see that (39) is not feasible. Moreover, if (40) was feasible, we would have

$$kc_i = b_2 c_j \,,$$

which implies $c_i = c_j = 0$ by $k \le 1$ and $b_2 > 1$. Since the latter contradicts the assumed $c_i \ne 0$ or $c_j \ne 0$, we therefore conclude that (40) is not feasible. Analogously, (41) is not feasible. Hence, we have shown that if (38) is feasible, the remaining cases (39) to (41) are not feasible. Since the arguments can be repeated using Table 4 when one of the remaining cases (39) to (41) is considered feasible, we finally conclude that at most one of the four cases is feasible, that is, we have shown our intermediate result.

Let us now assume that none of the four cases yield a feasible solution. Then we obtain Table 5, where in each row, at least one of the inequalities needs to be true. Let us assume that $T_1 < 0$,

Optimal Solution	T_1	T_2	T_3	T_4
(38) not feasible	< 0	< 0	$-$	$-$
(39) not feasible	$-$	$-$	< 0	< 0
(40) not feasible	$-$	> 0	> 0	$-$
(41) not feasible	> 0	$-$	$-$	> 0

Table 5: Behavior of expressions T_1, T_2, T_3 and T_4 when none of the optimal solutions is feasible

then by Table 5, we conclude that we have following set of inequalities

$$
\begin{align}
kc_j \quad &< \quad b_2 c_i \,, \tag{52}\\
kc_i \quad &> \quad b_2 c_j \,, \tag{53}\\
b_1 c_i \quad &< \quad kc_j \,, \tag{54}\\
b_1 c_j \quad &> \quad kc_i \,. \tag{55}
\end{align}
$$

Combining (52) and (54) as well as (53) and (55), we obtain

$$
\begin{align}
b_1 c_i \quad &< \quad b_2 c_i \,, \tag{56}\\
b_2 c_j \quad &< \quad b_1 c_j \,. \tag{57}
\end{align}
$$

Now if $c_i < 0$, we find $c_j < 0$ by (52). Moreover (56) together with $c_i < 0$ implies $b_2 < b_1$, while (57) together with $c_j < 0$ implies $b_1 < b_2$, that is, we have found a contradiction. Analogously, we obtain a contradiction in the case $T_1 \ge 0$. As a consequence exactly one of the four cases produces a feasible solution. Finally, the implications are a direct consequence of the form of the solutions in (38) to (41) and the fact that only one case provides a feasible solution. □

References

[1] B. Abdous and B. Remillard. Relating quantiles and expectiles under weighted-symmetry. *Ann. Inst. Statist. Math.*, 47:371–384, 1995.

[2] A. L. Anderson. *A Study on Expectiles: Measuring Risk in Finance.* PhD thesis, University of Georgia, 2012.

[3] Y. Aragon, S. Casanova, R. Chambers, and E. Leconte. Conditional ordering using nonparametric expectiles. *J. Off. Stat.*, 21:617–633, 2005.

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

[5] F. Bellini, B. Klar, A. Müller, and R. E. Gianin. Generalized quantiles as risk measures. *Insurance Math. Econom.*, 54:41–48, 2014.

[6] A. Berlinet and C. Thomas-Agnan. *Reproducing kernel Hilbert spaces in probability and statistics*, volume 3. Kluwer Academic Boston, 2004.

[7] J. Breckling and R. Chambers. M-quantiles. *Biometrika*, 75:761–771, 1988.

[8] C-C. Chang and C-J. Lin. LIBSVM: a library for support vector machines. *ACM Trans. Intell. Syst. Technol.*, 2:27, 2011.

[9] A. Christmann and I. Steinwart. How SVMs can estimate quantiles and the median. In *Advances in neural information processing systems*, pages 305–312, 2007.

[10] N. Cristianini and J. Shawe-Taylor. *An introduction to support vector machines and other kernel-based learning methods.* Cambridge University Press, Cambridge, 2000.

[11] M. Eberts and I. Steinwart. Optimal learning rates for least squares SVMs using gaussian kernels. In *Advances in neural information processing systems*, pages 1539–1547, 2011.

[12] B. Efron. Regression percentiles using asymmetric squared error loss. *Statist. Sci.*, 1:93–125, 1991.

[13] T. Glasmachers and C. Igel. Maximum-gain working set selection for SVMs. *J. Mach. Learn. Res.*, 7:1437–1466, 2006.

[14] K. Guler, P. T. Ng, and Z. Xiao. Mincer-Zarnovitz quantile and expectile regressions for forecast evaluations under asymmetric loss functions. *Northern Arizona University, The WA Franke College of Business. Working Paper Series 14-01*, 2014.

[15] B. Hamidi, B. Maillet, and J-L. Prigent. A dynamic autoregressive expectile for time-invariant portfolio protection strategies. *J. Econom. Dynam. Control*, 46:1–29, 2014.

[16] X. Huang, L. Shi, and J. AK Suykens. Asymmetric least squares support vector machine classifiers. *Comput. Statist. Data Anal.*, 70:395–405, 2014.

[17] T. Joachims. Making large-scale SVM learning practical. In *Advances in Kernel Methods - Support Vector Learning.* MIT Press, Cambridge, MA, USA, 1999.

[18] S. S. Keerthi and S. K. Shevade. SMO algorithm for least-squares SVM formulations. *Neural comput.*, 15:487–507, 2003.

[19] R. Koenker. *Quantile regression.* Cambridge University Press, Cambridge, 2005.

[20] R. Koenker and G. Bassett Jr. Regression quantiles. *Econometrica*, 46:33–50, 1978.

[21] W. K. Newey and J. L. Powell. Asymmetric least squares estimation and testing. *Econometrica*, 55:819–847, 1987.

[22] J. Platt. Fast training of support vector machines using sequential minimal optimization. In *Advances in kernel methods-Support Vector Learning*, pages 185–208. MIT press, Cambridge, MA., 1999.

[23] S. Schnabel and P. Eilers. An analysis of life expectancy and economic production using expectile frontier zones. *Demographic Res.*, 21:109–134, 2009.

[24] S. K. Schnabel and P. H. Eilers. Optimal expectile smoothing. *Comput. Statist. Data Anal.*, 53:4168–4177, 2009.

[25] B. Schölkopf and A.J. Smola. *Learning with kernels: support vector machines, regularization, optimization, and beyond.* MIT press, Cambridge, MA., 2002.

[26] J. Shim and C. Hwang. Expected shortfall estimation using kernel machines. *Journal of Korean Data & Information Science Society*, 24:12–20, 2013.

[27] F. Sobotka, G. Kauermann, L. S. Waltrup, and T. Kneib. On confidence intervals for semiparametric expectile regression. *Stat. Comput.*, 23:135–148, 2013.

[28] F. Sobotka and T. Kneib. Geoadditive expectile regression. *Comput. Statist. Data Anal.*, 56:755–767, 2012.

[29] F. Sobotka, R. Radice, G. Marra, and T. Kneib. Estimating the relationship between women's education and fertility in Botswana by using an instrumental variable approach to semiparametric expectile regression. *J. Roy. Stat. Soc. C- App.*, 62:25–45, 2013.

[30] F. Sobotka, S. Schnabel, L. S. Waltrup, P. Eilers, T. Kneib, and G. Kauermann. expectreg: Expectile and qauntile regression. R package version 0.39. `http://cran.r-project.org/web/packages/expectreg/index.html`, 2014.

[31] S. Stahlschmidt, M. Eckardt, and W. K. Härdle. Expectile treatment effects: An efficient alternative to compute the distribution of treatment effects. Technical report, Sonderforschungsbereich 649, Humboldt University, Berlin, Germany, 2014.

[32] I. Steinwart and A. Christmann. *Support vector machines*. Springer, New York, 2008.

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

[34] I. Steinwart, D. Hush, and C. Scovel. An oracle inequality for clipped regularized risk minimizers. In *Advances in neural information processing systems*, pages 1321–1328, 2006.

[35] I. Steinwart, D. Hush, and C. Scovel. Training SVMs without offset. *J. Mach. Learn. Res.*, 12:141–202, 2011.

[36] I. Steinwart, C. Pasin, R. Williamson, and S. Zhang. Elicitation and identification of properties. In M. F. Balcan and C. Szepesvari, editors, *JMLR Workshop and Conference Proceedings Volume 35: Proceedings of the 27th Conference on Learning Theory 2014*, pages 482–526, 2014.

[37] I. Takeuchi, Q. V. Le, T. D. Sears, and A. J. S. Nonparametric quantile estimation. *J. Mach. Learn. Res.*, 7:1231–1264, 2006.

[38] J. W. Taylor. Estimating value at risk and expected shortfall using expectiles. *J. Financ. Econ.*, 6:231–252, 2008.

[39] V. Vapnik. *The nature of statistical learning theory*. Springer-Verlag, New York, 2000.

[40] M. Vogt. SMO algorithms for support vector machines without bias term. *Technische Univ. Darmstadt, Inst. Automat. Contr., Lab. Contr. Syst. Process Automat., Darmstadt, Germany*, 2002.

[41] L. S. Waltrup, F. Sobotka, T. Kneib, and G. Kauermann. Expectile and quantile regression–David and Goliath? *Stat. Model.*, page 1471082X14561155, 2014.

[42] Y. Wang, S. Wang, and K. K. Lai. Measuring financial risk with generalized asymmetric least squares regression. *Appl. Soft Comput.*, 11(8):5793–5800, 2011.

[43] S.J. Wright and J. Nocedal. *Numerical optimization*, volume 2. Springer, New York, 1999.

[44] Y. Yang and H. Zou. Nonparametric multiple expectile regression via ER-Boost. *J. Stat. Comput. Simulation*, 85:1442–1458, 2015.

[45] Q. Yao and H. Tong. Asymmetric least squares regression estimation: a nonparametric approach. *J. Nonparametr. Statist.*, 6:273–292, 1996.

A Detailed Results of Experiments

A.1 Results for Different Working Set Selection Methods
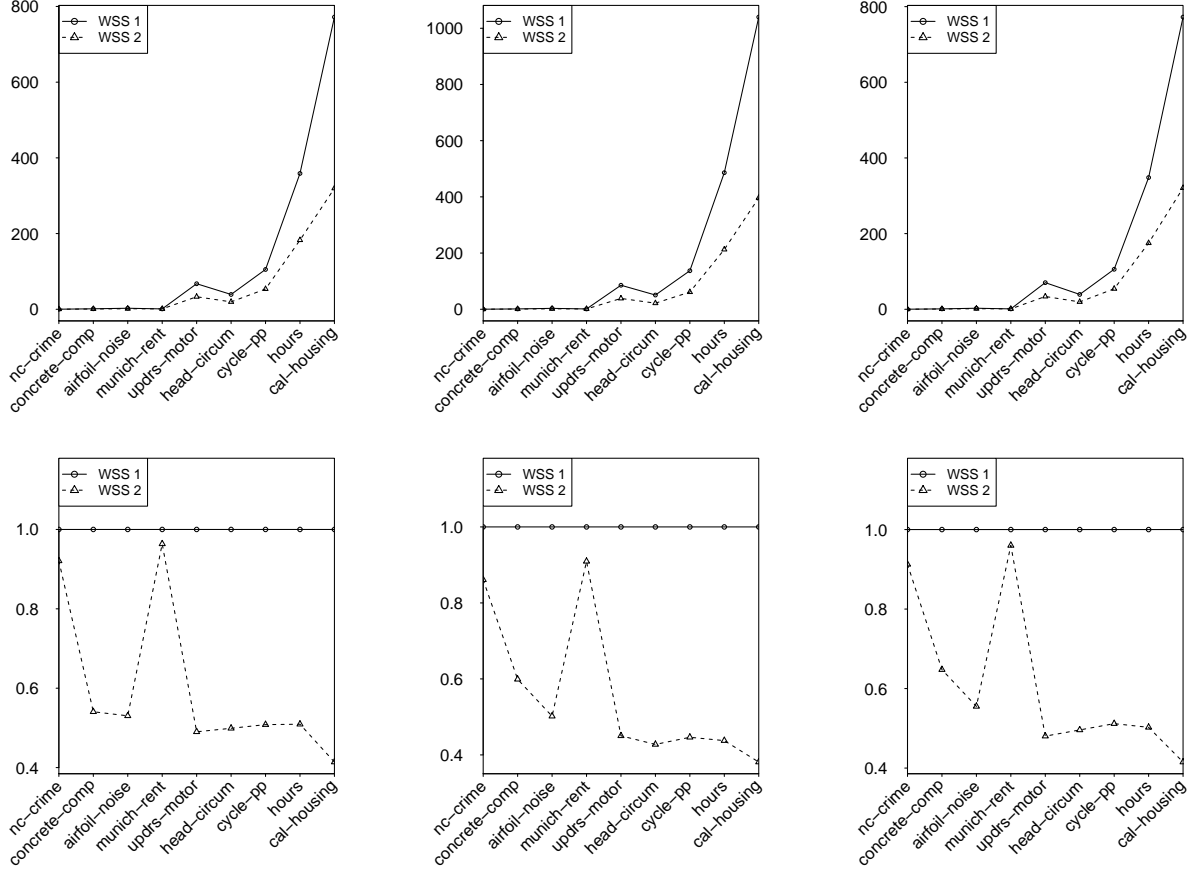
Figure 2: Train time (top) and corresponding ratio (bottom) of different data sets for different working set selection methods after fixing warm start initialization and stopping criteria with clipped duality gap. The graphs comprises of $\tau = 0.25$ (left), $\tau = 0.50$ (middle) and $\tau = 0.75$ (right).
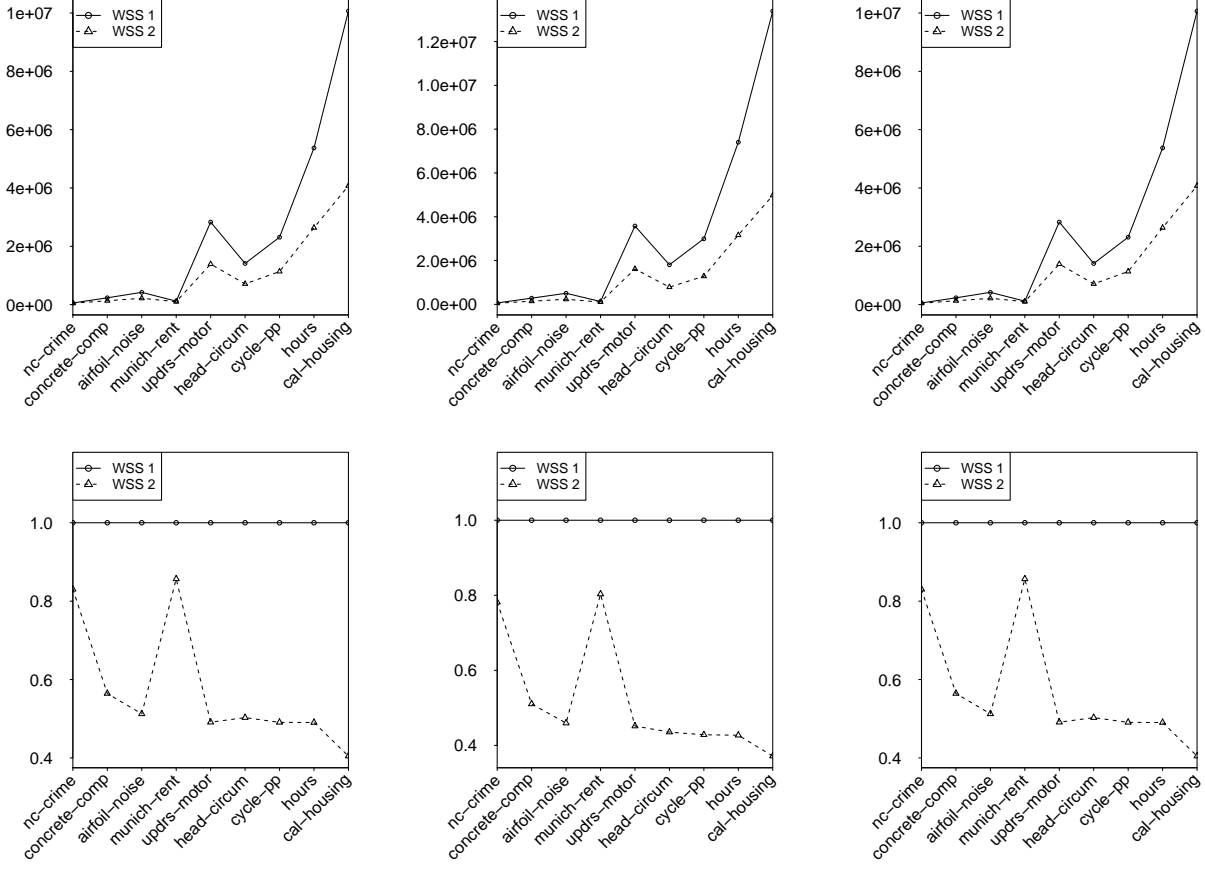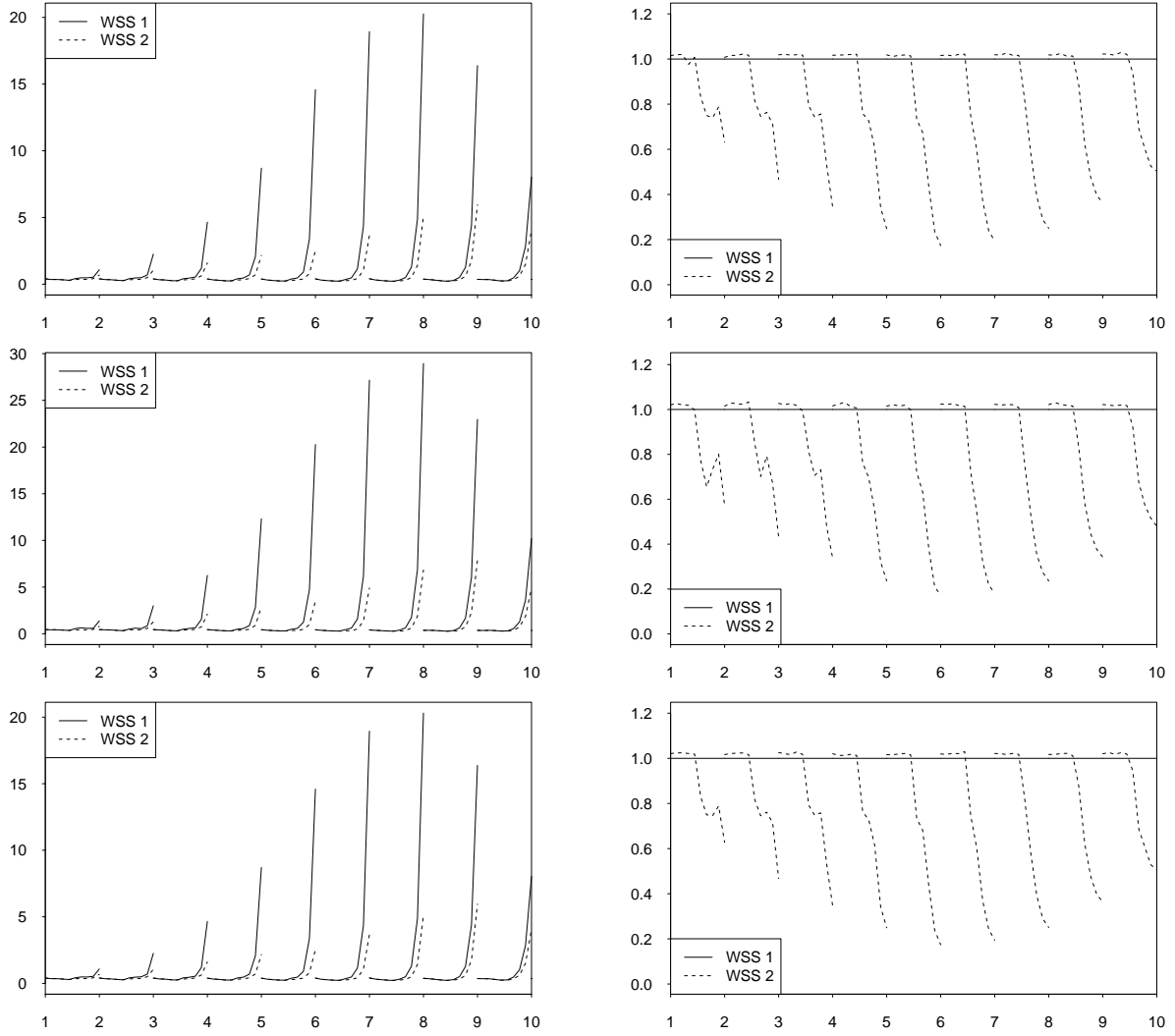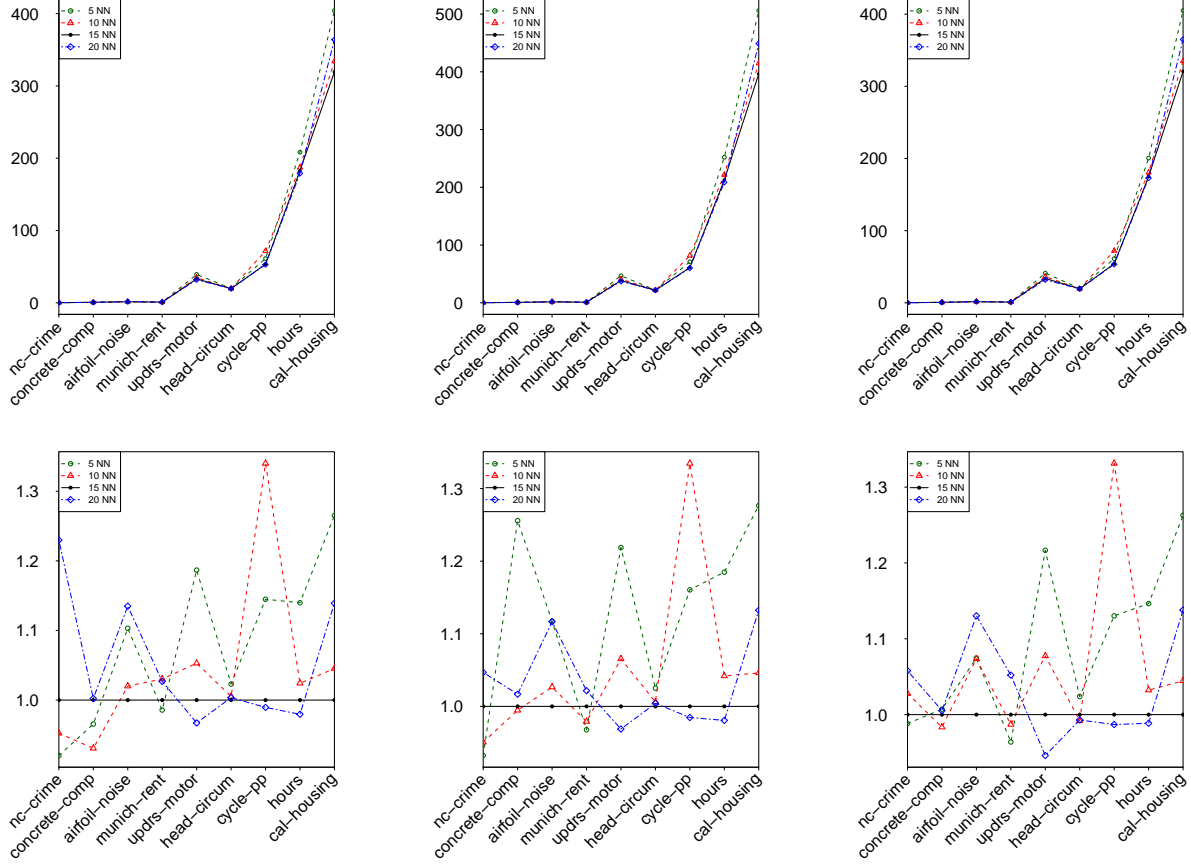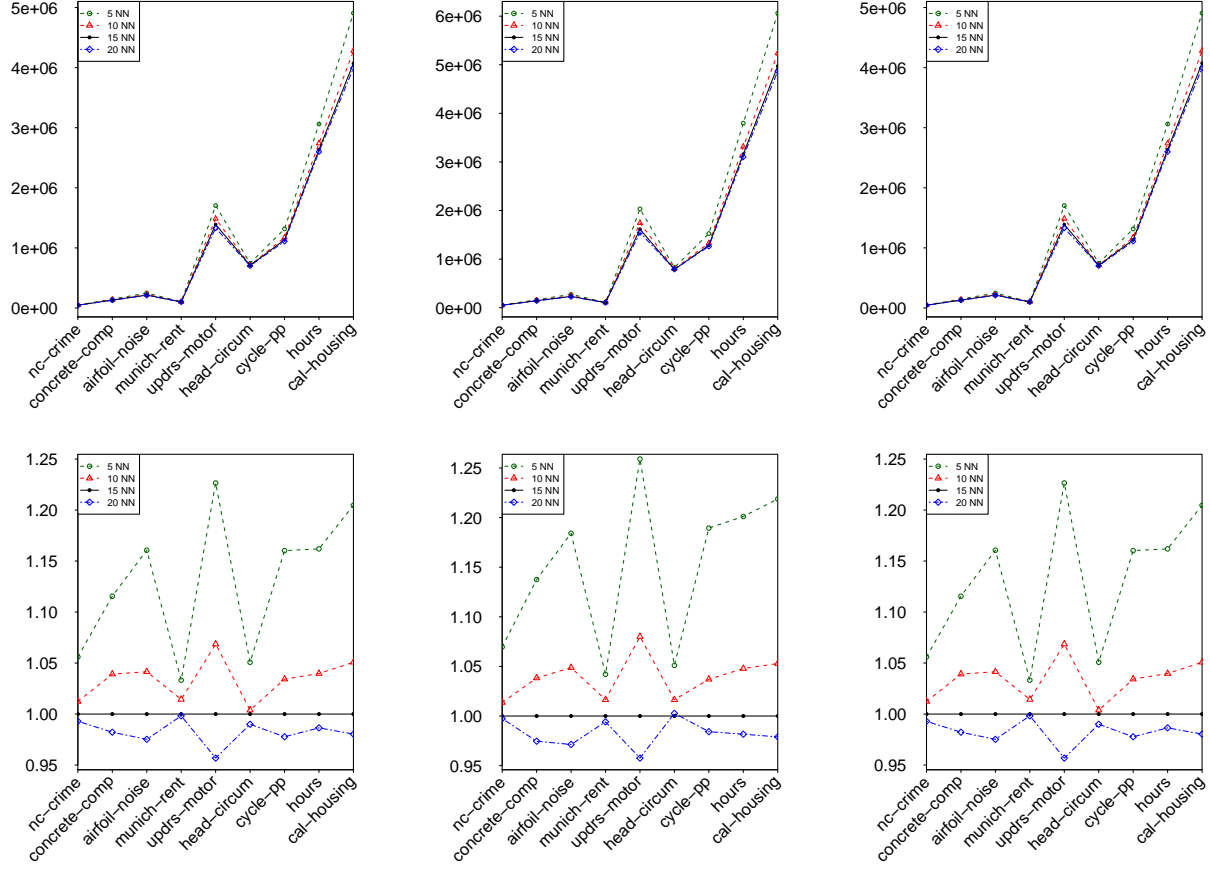
Figure 3: Train iterations (top) and corresponding ratio (bottom) of different data sets for different working set selection methods after fixing warm start initialization and stopping criteria with clipped duality gap. The graphs comprises of $\tau = 0.25$ (left), $\tau = 0.50$ (middle) and $\tau = 0.75$ (right).

Figure 4: Average train time (left) and corresponding ratio (right) per grid point for different working set selection strategies using clipped duality gap criteria and initializing solver with warm start for data set CAL-HOUSING. For WSS 2, 15 nearest neighbors are considered. The graphs comprises for $\tau = 0.25$ (top), $\tau = 0.50$ (middle) and $\tau = 0.75$ (bottom).

27
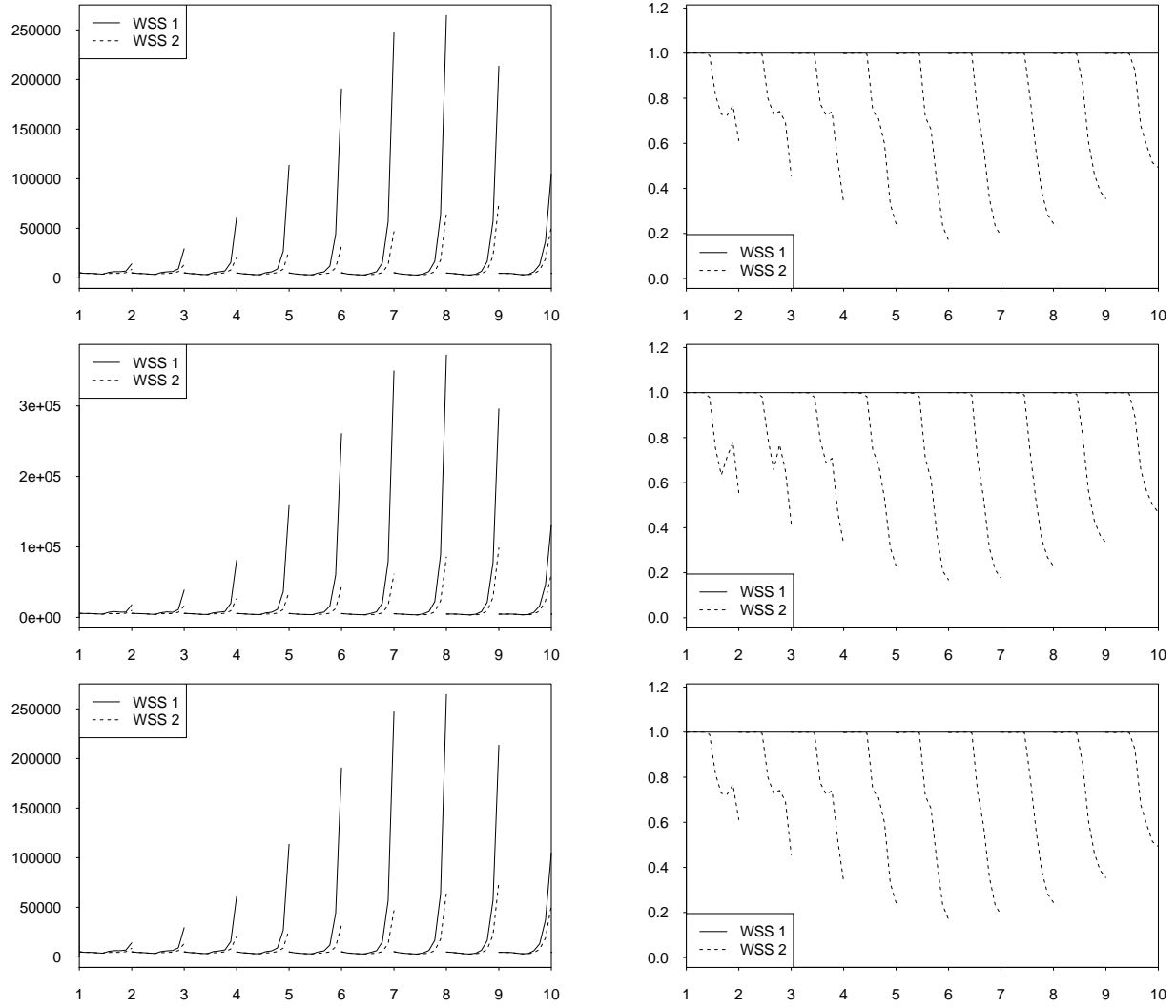
Figure 5: Average number of iterations (left) and corresponding ratio (right) per grid point for different working set selection strategies using clipped duality gap criteria and initializing solver with warm start for data set CAL-HOUSING. For WSS 2, 15 nearest neighbors are considered. The graphs comprises for $\tau = 0.25$ (top), $\tau = 0.50$ (middle) and $\tau = 0.75$(bottom).

A.2 Results for Different Number of Nearest Neighbors

Figure 6: Train time (top) and corresponding ratio (bottom) of different data sets for different number of nearest neighbors after fixing warm start initialization and stopping criteria with clipped duality gap. The graphs comprises of $\tau = 0.25$ (left), $\tau = 0.50$ (middle) and $\tau = 0.75$ (right).

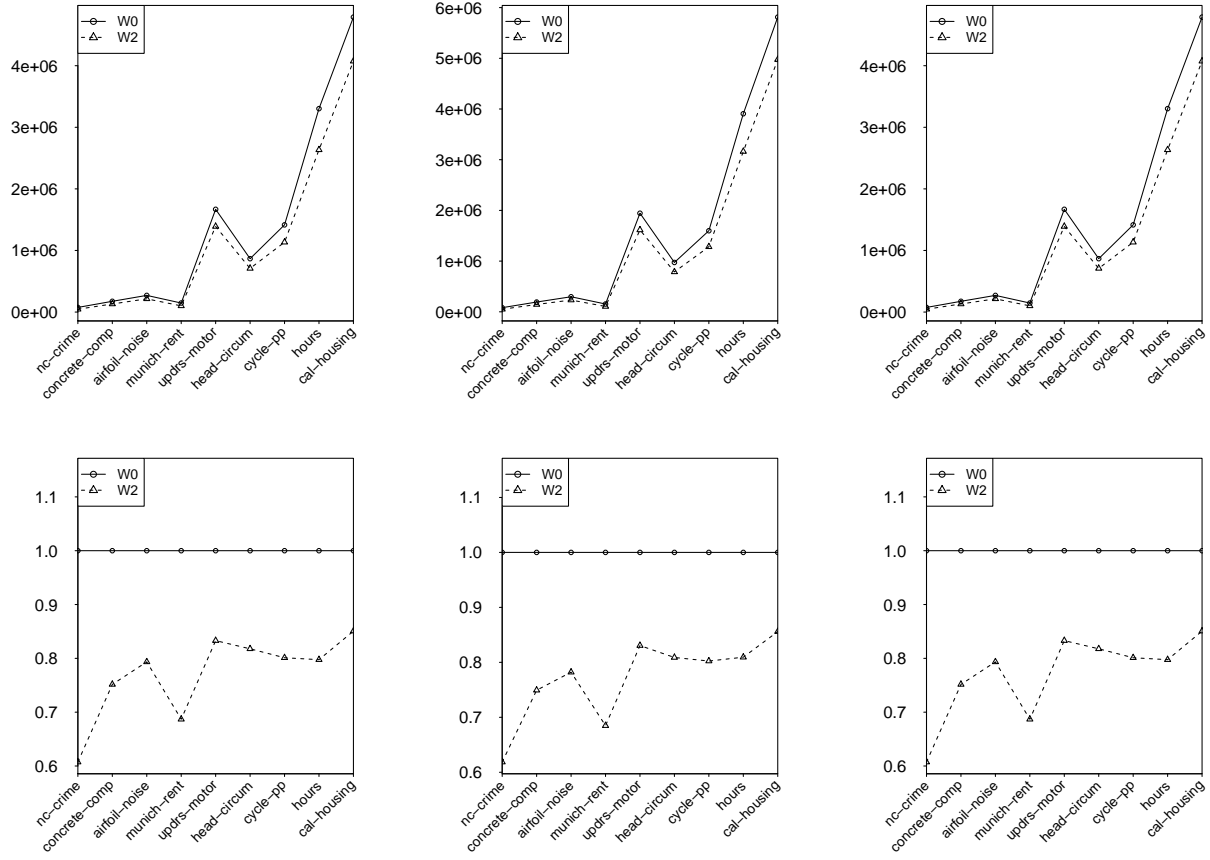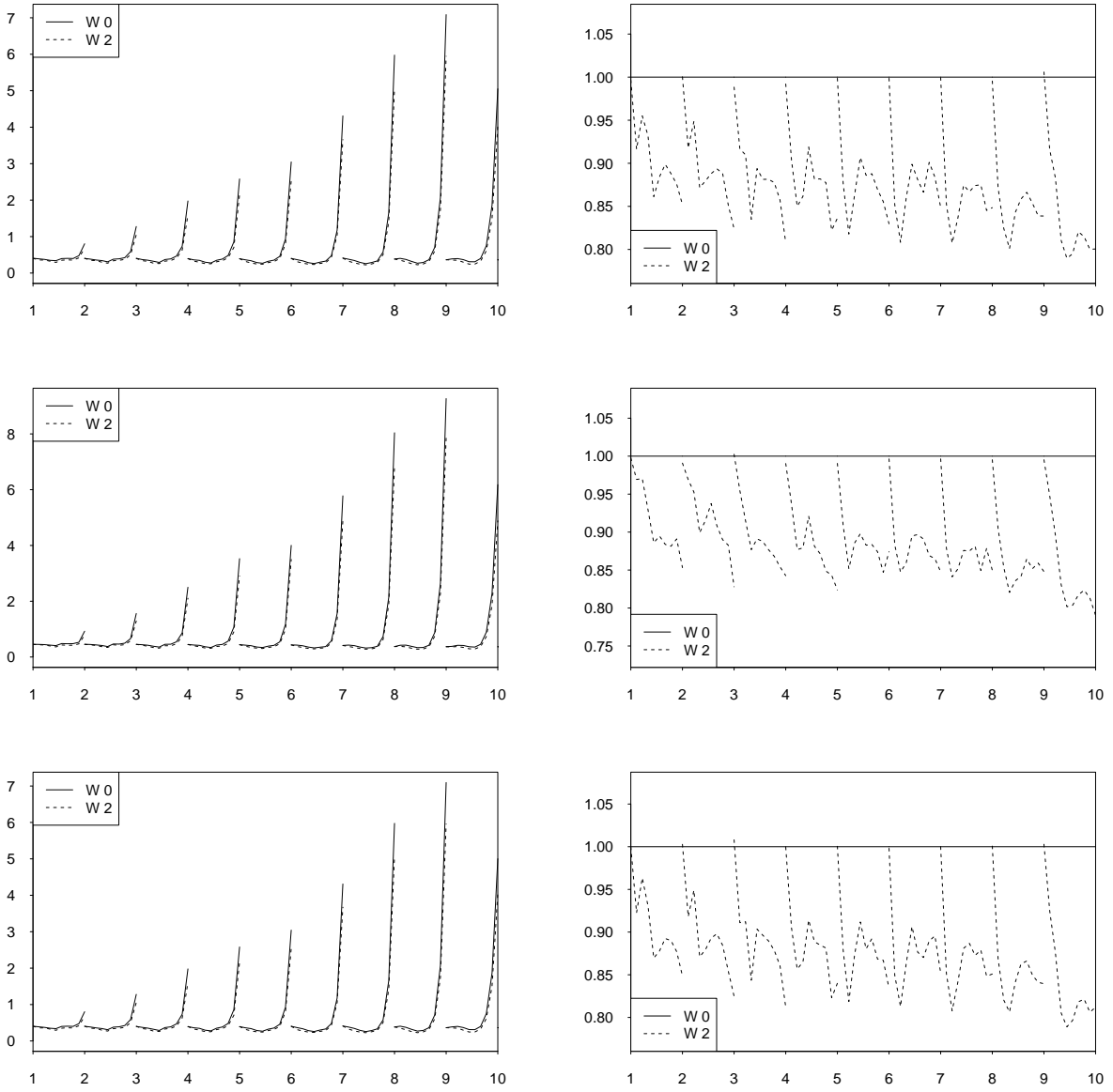Figure 7: Train iterations (top) and corresponding ratio (bottom) of different data sets for different number of nearest neighbors after fixing with warm start initialization and stopping criteria with clipped duality gap. The graphs comprises of $\tau = 0.25$ (left), $\tau = 0.50$ (middle) and $\tau = 0.75$ (right).
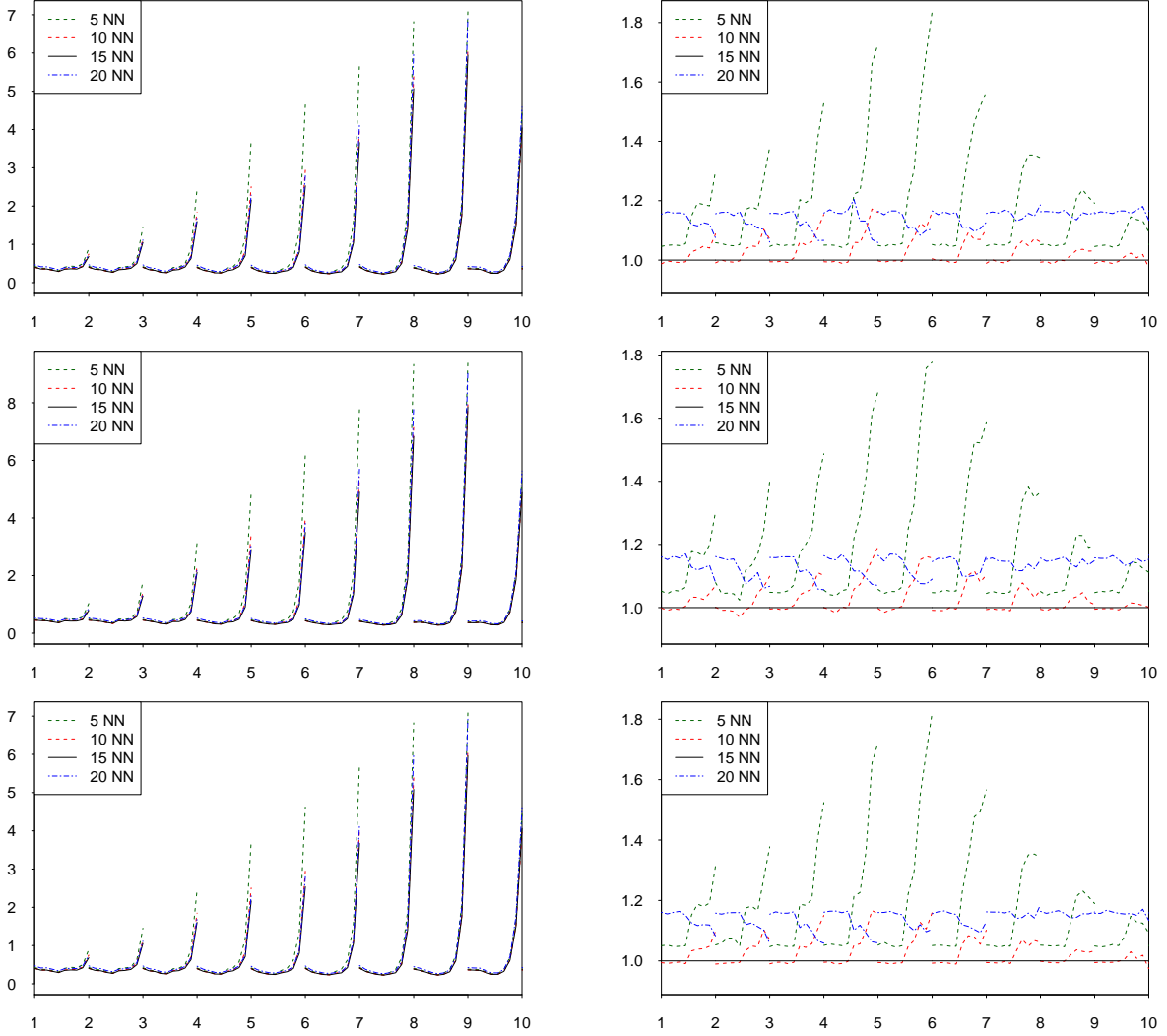
Figure 8: Average train time (left) and corresponding ratio (right) per grid point for different different number of nearest neighbors considering warm start and stopping criteria with clipped duality gap for the data set CAL-HOUSING. The graphs comprises for $\tau = 0.25$ (top), $\tau = 0.50$ (middle) and $\tau = 0.75$ (bottom).
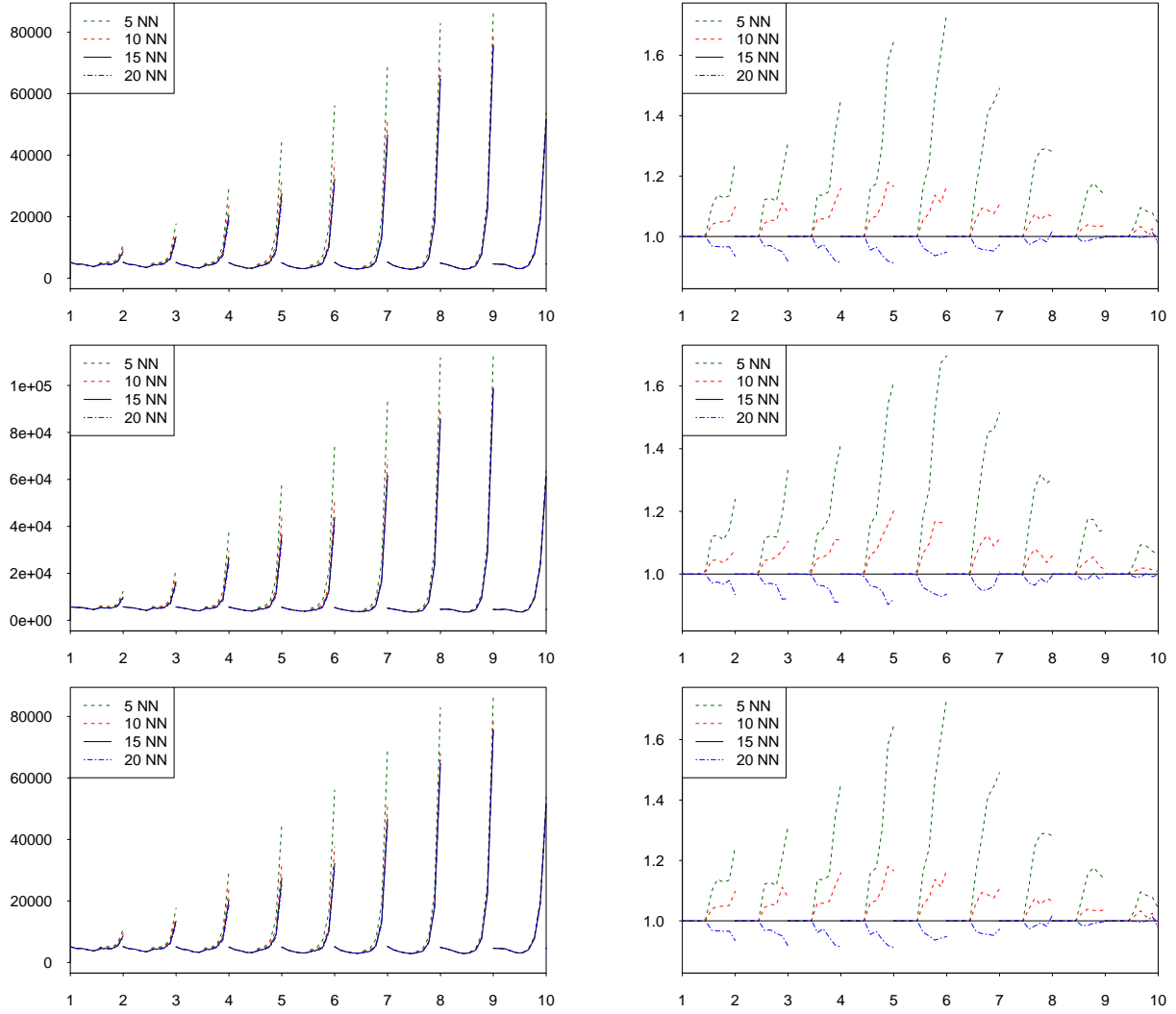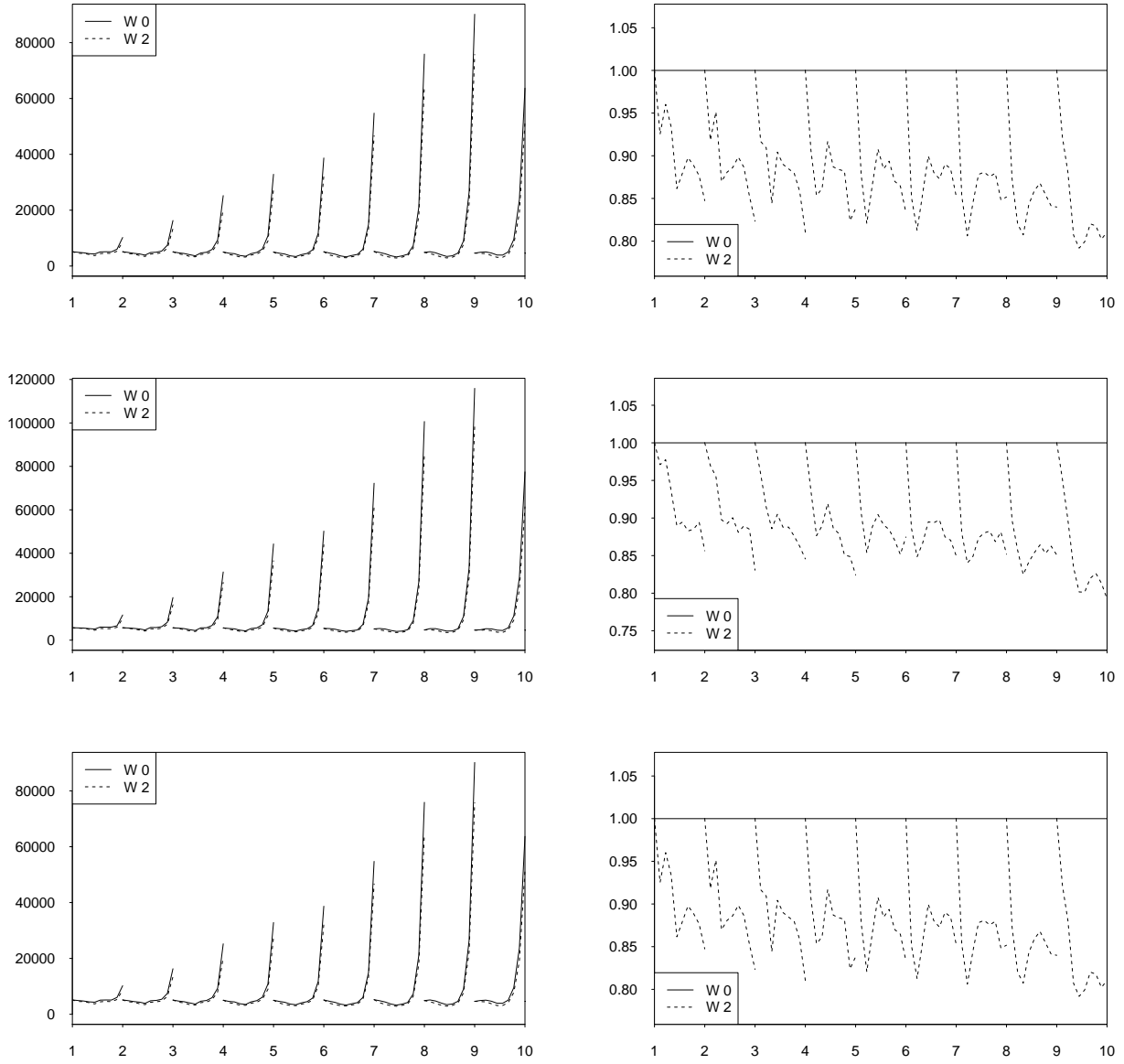
Figure 9: Average number of iterations (left) and corresponding ratio (right) per grid point for different different number of nearest neighbors considering warm start and stopping criteria with clipped duality gap for the data set CAL-HOUSING. The graphs comprises for $\tau = 0.25$ (top), $\tau = 0.50$ (middle) and $\tau = 0.75$ (bottom).
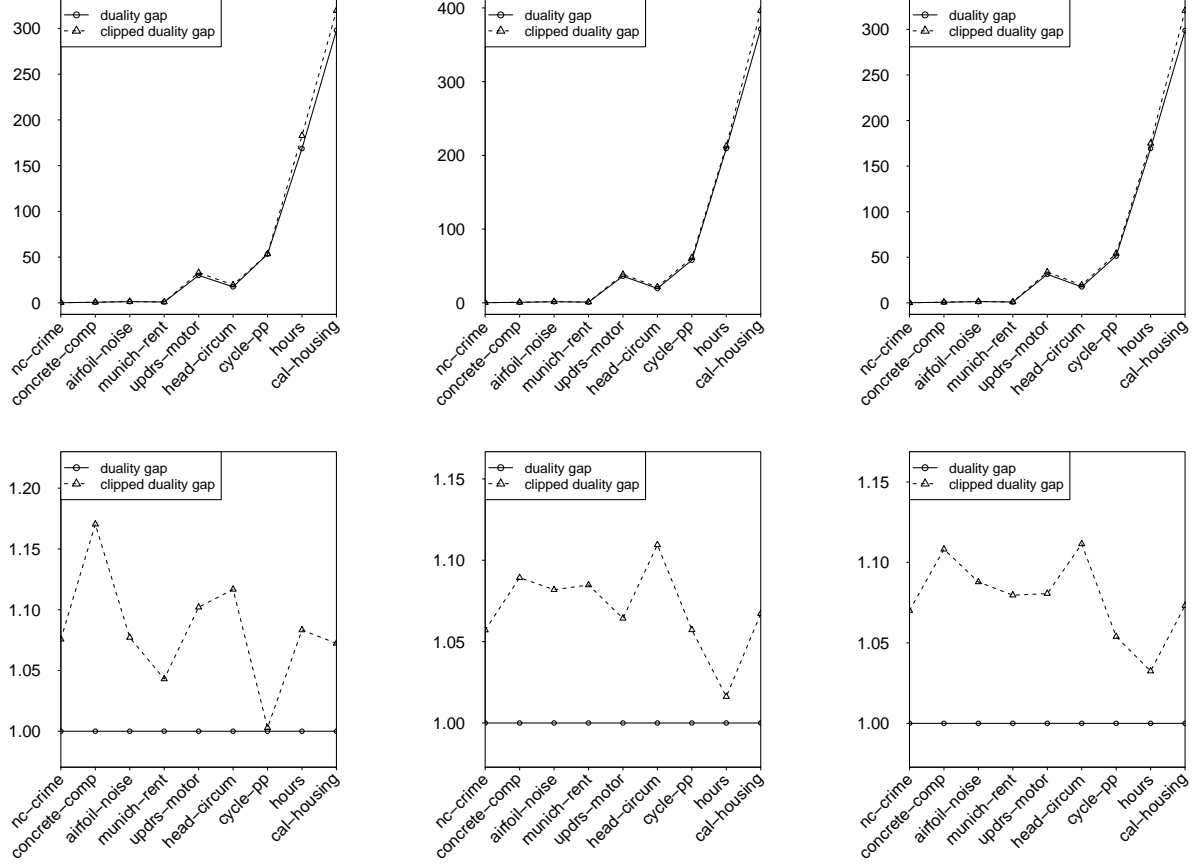
A.3 Results for Different Initialization Methods
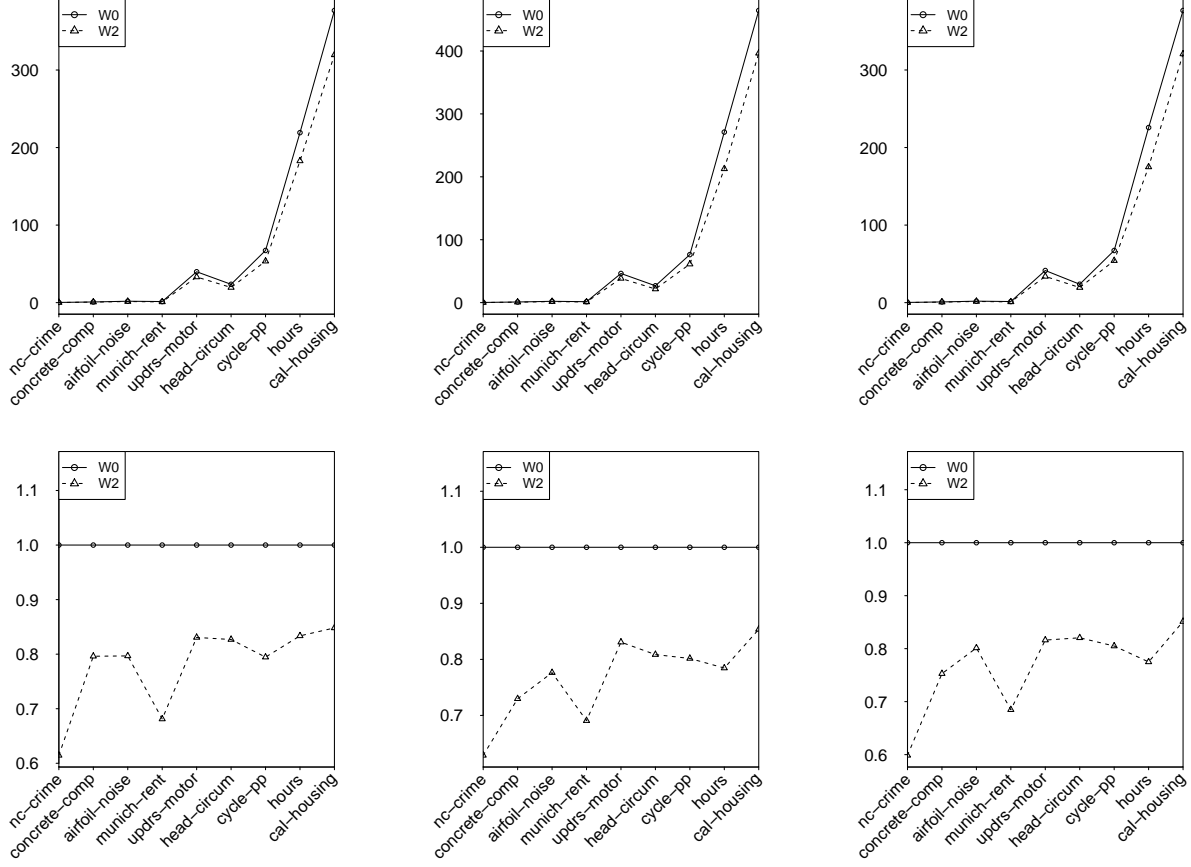
Figure 10: Train time (top) and corresponding ratio (bottom) of different data sets for different initialization methods after fixing stopping criteria with clipped duality gap and $NN = 15$. The graphs comprises of $\tau = 0.25$ (left), $\tau = 0.50$ (middle) and $\tau = 0.75$ (right).
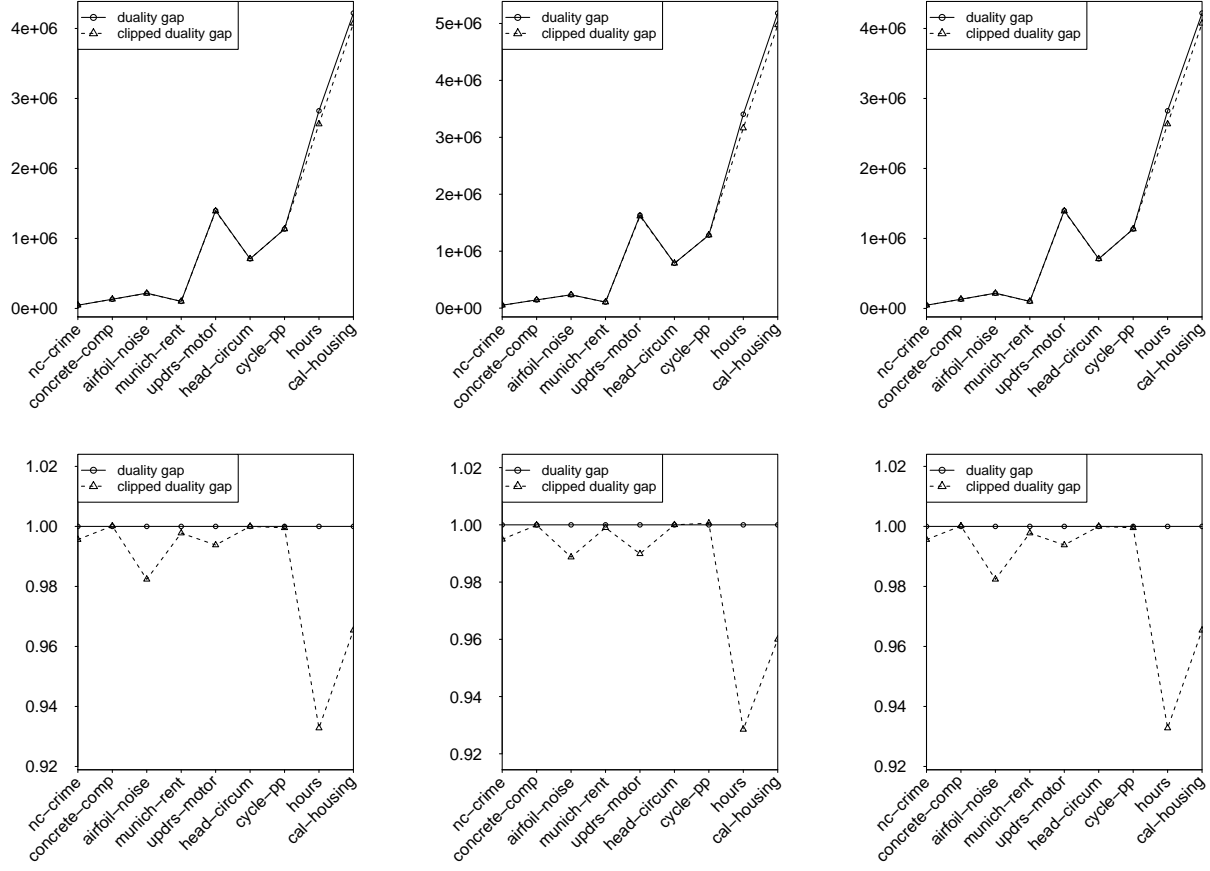
Figure 11: Train iterations (top) and corresponding ratio (bottom) of different data sets for different initialization methods after fixing stopping criteria with clipped duality gap and $NN = 15$. The graphs comprises of $\tau = 0.25$ (left), $\tau = 0.50$ (middle) and $\tau = 0.75$ (right).

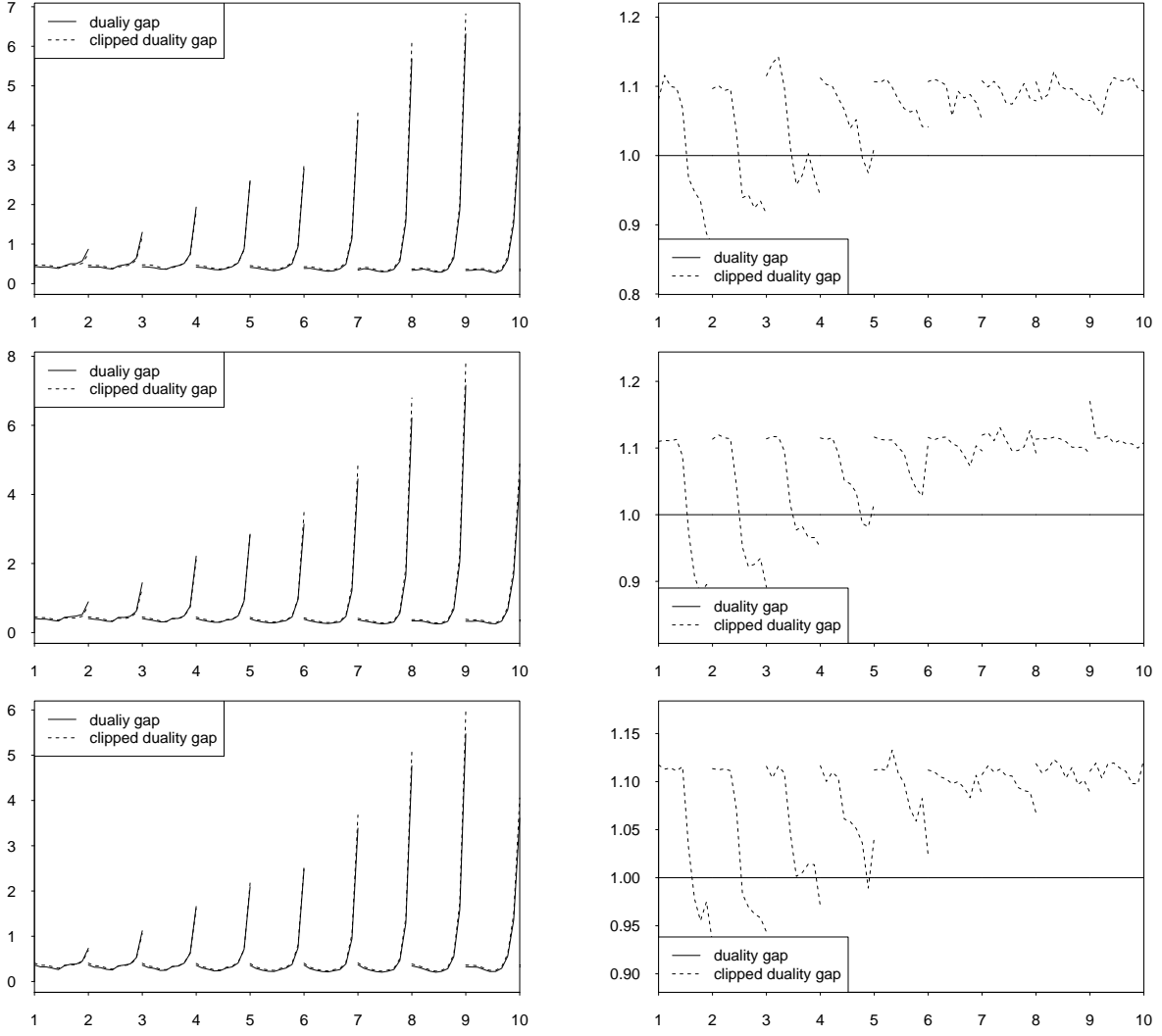Figure 12: Average train time (left) and corresponding ratio (right) per grid point for different initialization methods considering stopping criteria after with clipped duality gap and WSS 2 for the data set CAL-HOUSING. The graphs comprises for $\tau = 0.25$ (top), $\tau = 0.50$ (middle) and $\tau = 0.75$ (bottom).

Figure 13: Average train iterations (left) and corresponding ratio (right) per grid point for different initialization methods considering stopping criteria with clipped duality gap and WSS 2 for the data set CAL-HOUSING. The graphs comprises for $\tau = 0.25$ (top), $\tau = 0.50$ (middle) and $\tau = 0.75$ (bottom).
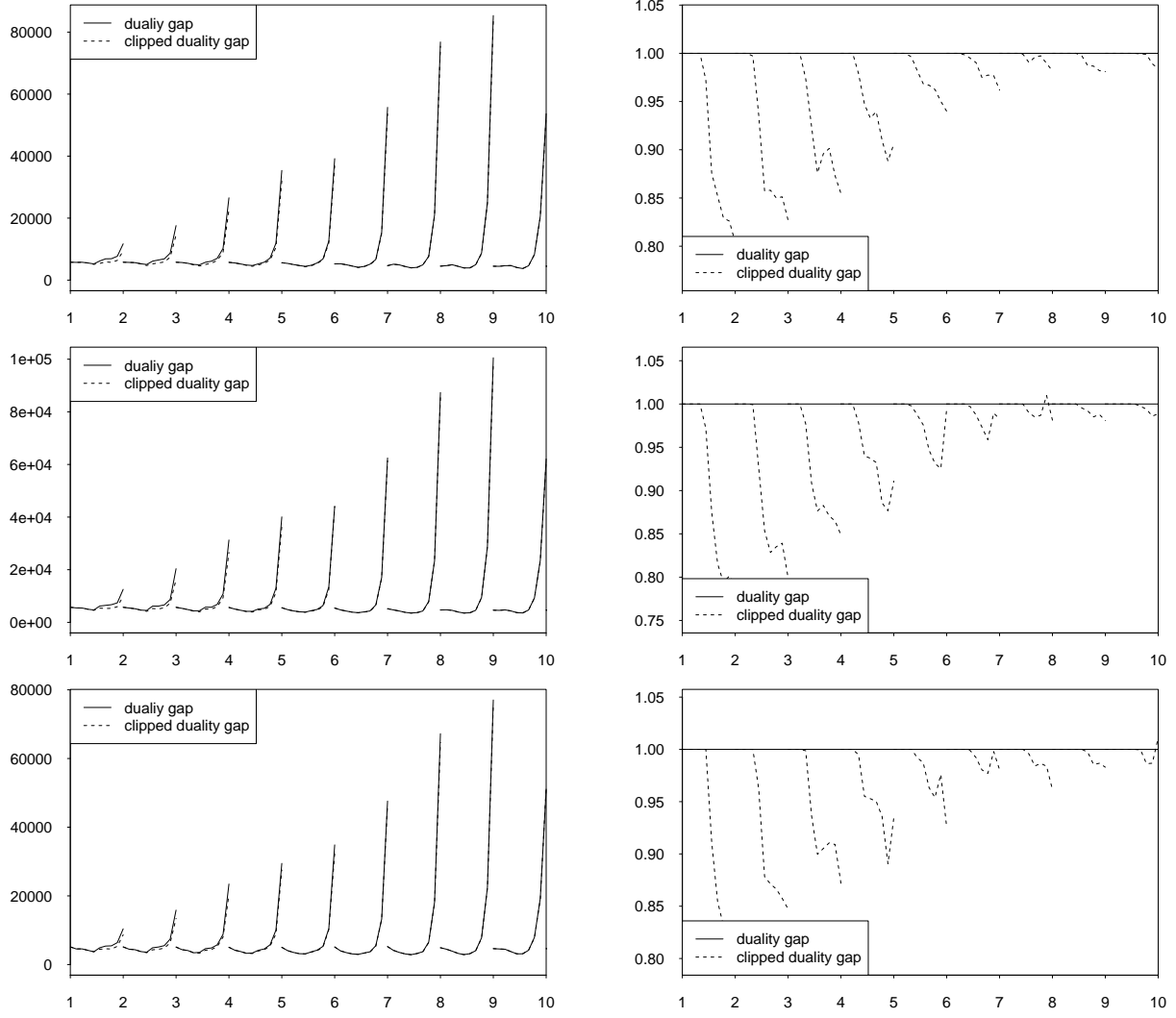
A.4 Results for two Different Stopping Criteria

Figure 14: Train time (top) and corresponding ratio (bottom) of different data sets for different stopping criteria after fixing initialization method as warm start and $NN = 15$. The graphs comprises of $\tau = 0.25$ (left), $\tau = 0.50$ (middle) and $\tau = 0.75$ (right).

Figure 15: Train iterations (top) and corresponding ratio (bottom) of different data sets for different stopping criteria after fixing initialization method as warm start and $NN = 15$. The graphs comprises of $\tau = 0.25$ (left), $\tau = 0.50$ (middle) and $\tau = 0.75$ (right).

Figure 16: Average train time (left) and corresponding ratio (right) per grid point for different stopping criteria using 15 NN and initializing solver with warm start for CAL-HOUSING. The graphs comprises for $\tau = 0.25$ (top), $\tau = 0.50$ (middle) and $\tau = 0.75$ (bottom).

Figure 17: Average number of iterations (left) and corresponding ratio (right) per grid point for different stopping criteria using 15 NN and initializing solver with warm start for CAL-HOUSING. The graphs comprises for $\tau = 0.25$ (top), $\tau = 0.50$ (middle) and $\tau = 0.75$ (bottom).

Muhammad Farooq

Fachbereich Mathematik, Universität Stuttgart, Pfaffenwaldring 57, 70569 Stuttgart, Germany

E-Mail: `Muhammad.Farooq@mathematik.uni-stuttgart.de`

Ingo Steinwart

Fachbereich Mathematik, Universität Stuttgart, Pfaffenwaldring 57, 70569 Stuttgart, Germany

E-Mail: `steinwart@mathematik.uni-stuttgart.de`

WWW: `http://www.isa.uni-stuttgart.de/Steinwart/`

Erschienene Preprints ab Nummer 2012-001

Komplette Liste: `http://www.mathematik.uni-stuttgart.de/preprints`

2015-016 *Hang, H.; Steinwart, I.:* A Bernstein-type Inequality for Some Mixing Processes and Dynamical Systems with an Application to Learning

2015-015 *Steinwart, I.:* Representation of Quasi-Monotone Functionals by Families of Separating Hyperplanes

2015-014 *Muhammad, F.; Steinwart, I.:* An SVM-like Approach for Expectile Regression

2015-013 *Nava-Yazdani, E.:* Splines and geometric mean for data in geodesic spaces

2015-012 *Kimmerle, W.; Köster, I.:* Sylow Numbers from Character Tables and Group Rings

2015-011 *Györfi, L.; Walk, H.:* On the asymptotic normality of an estimate of a regression functional

2015-010 *Gorodski, C, Kollross, A.:* Some remarks on polar actions

2015-009 *Apprich, C.; Höllig, K.; Hörner, J.; Reif, U.:* Collocation with WEB–Splines

2015-008 *Kabil, B.; Rodrigues, M.:* Spectral Validation of the Whitham Equations for Periodic Waves of Lattice Dynamical Systems

2015-007 *Kollross, A.:* Hyperpolar actions on reducible symmetric spaces

2015-006 *Schmid, J.; Griesemer, M.:* Well-posedness of Non-autonomous Linear Evolution Equations in Uniformly Convex Spaces

2015-005 *Hinrichs, A.; Markhasin, L.; Oettershagen, J.; Ullrich, T.:* Optimal quasi-Monte Carlo rules on higher order digital nets for the numerical integration of multivariate periodic functions

2015-004 *Kutter, M.; Rohde, C.; Sändig, A.-M.:* Well-Posedness of a Two Scale Model for Liquid Phase Epitaxy with Elasticity

2015-003 *Rossi, E.; Schleper, V.:* Convergence of a numerical scheme for a mixed hyperbolic-parabolic system in two space dimensions

2015-002 *Döring, M.; Györfi, L.; Walk, H.:* Exact rate of convergence of kernel-based classification rule

2015-001 *Kohler, M.; Müller, F.; Walk, H.:* Estimation of a regression function corresponding to latent variables

2014-021 *Neusser, J.; Rohde, C.; Schleper, V.:* Relaxed Navier-Stokes-Korteweg Equations for Compressible Two-Phase Flow with Phase Transition

2014-020 *Kabil, B.; Rohde, C.:* Persistence of undercompressive phase boundaries for isothermal Euler equations including configurational forces and surface tension

2014-019 *Bilyk, D.; Markhasin, L.:* BMO and exponential Orlicz space estimates of the discrepancy function in arbitrary dimension

2014-018 *Schmid, J.:* Well-posedness of non-autonomous linear evolution equations for generators whose commutators are scalar

2014-017 *Margolis, L.:* A Sylow theorem for the integral group ring of $PSL(2,q)$

2014-016 *Rybak, I.; Magiera, J.; Helmig, R.; Rohde, C.:* Multirate time integration for coupled saturated/unsaturated porous medium and free flow systems

2014-015 *Gaspoz, F.D.; Heine, C.-J.; Siebert, K.G.:* Optimal Grading of the Newest Vertex Bisection and H^1-Stability of the L_2-Projection

2014-014 *Kohler, M.; Krzyżak, A.; Walk, H.:* Nonparametric recursive quantile estimation

2014-013 *Kohler, M.; Krzyżak, A.; Tent, R.; Walk, H.:* Nonparametric quantile estimation using importance sampling

2014-012 *Györfi, L.; Ottucsák, G.; Walk, H.:*   The growth optimal investment strategy is secure, too.

2014-011 *Györfi, L.; Walk, H.:*   Strongly consistent detection for nonparametric hypotheses

2014-010 *Köster, I.:*   Finite Groups with Sylow numbers $\{q^x, a, b\}$

2014-009 *Kahnert, D.:*   Hausdorff Dimension of Rings

2014-008 *Steinwart, I.:*   Measuring the Capacity of Sets of Functions in the Analysis of ERM

2014-007 *Steinwart, I.:*   Convergence Types and Rates in Generic Karhunen-Loève Expansions with Applications to Sample Path Properties

2014-006 *Steinwart, I.; Pasin, C.; Williamson, R.; Zhang, S.:*   Elicitation and Identification of Properties

2014-005 *Schmid, J.; Griesemer, M.:*   Integration of Non-Autonomous Linear Evolution Equations

2014-004 *Markhasin, L.:* L_2- and $S_{p,q}^r B$-discrepancy of (order 2) digital nets

2014-003 *Markhasin, L.:*   Discrepancy and integration in function spaces with dominating mixed smoothness

2014-002 *Eberts, M.; Steinwart, I.:*   Optimal Learning Rates for Localized SVMs

2014-001 *Giesselmann, J.:*   A relative entropy approach to convergence of a low order approximation to a nonlinear elasticity model with viscosity and capillarity

2013-016 *Steinwart, I.:*   Fully Adaptive Density-Based Clustering

2013-015 *Steinwart, I.:*   Some Remarks on the Statistical Analysis of SVMs and Related Methods

2013-014 *Rohde, C.; Zeiler, C.:*   A Relaxation Riemann Solver for Compressible Two-Phase Flow with Phase Transition and Surface Tension

2013-013 *Moroianu, A.; Semmelmann, U.:*   Generalized Killling spinors on Einstein manifolds

2013-012 *Moroianu, A.; Semmelmann, U.:*   Generalized Killing Spinors on Spheres

2013-011 *Kohls, K; Rösch, A.; Siebert, K.G.:*   Convergence of Adaptive Finite Elements for Control Constrained Optimal Control Problems

2013-010 *Corli, A.; Rohde, C.; Schleper, V.:*   Parabolic Approximations of Diffusive-Dispersive Equations

2013-009 *Nava-Yazdani, E.; Polthier, K.:*   De Casteljau's Algorithm on Manifolds

2013-008 *Bächle, A.; Margolis, L.:*   Rational conjugacy of torsion units in integral group rings of non-solvable groups

2013-007 *Knarr, N.; Stroppel, M.J.:*   Heisenberg groups over composition algebras

2013-006 *Knarr, N.; Stroppel, M.J.:*   Heisenberg groups, semifields, and translation planes

2013-005 *Eck, C.; Kutter, M.; Sändig, A.-M.; Rohde, C.:*   A Two Scale Model for Liquid Phase Epitaxy with Elasticity: An Iterative Procedure

2013-004 *Griesemer, M.; Wellig, D.:*   The Strong-Coupling Polaron in Electromagnetic Fields

2013-003 *Kabil, B.; Rohde, C.:*   The Influence of Surface Tension and Configurational Forces on the Stability of Liquid-Vapor Interfaces

2013-002 *Devroye, L.; Ferrario, P.G.; Györfi, L.; Walk, H.:*   Strong universal consistent estimate of the minimum mean squared error

2013-001 *Kohls, K.; Rösch, A.; Siebert, K.G.:*   A Posteriori Error Analysis of Optimal Control Problems with Control Constraints

2012-013 *Diaz Ramos, J.C.; Dominguez Vázquez, M.; Kollross, A.:*   Polar actions on complex hyperbolic spaces