# Supervised Learning with Similarity Functions

**Purushottam Kar**
Indian Institute of Technology
Kanpur, INDIA
purushot@cse.iitk.ac.in

**Prateek Jain**
Microsoft Research Lab
Bangalore, INDIA
prajain@microsoft.com

## Abstract

We address the problem of general supervised learning when data can only be accessed through an (indefinite) similarity function between data points. Existing work on learning with indefinite kernels has concentrated solely on binary/multi-class classification problems. We propose a model that is generic enough to handle any supervised learning task and also subsumes the model previously proposed for classification. We give a "goodness" criterion for similarity functions w.r.t. a given supervised learning task and then adapt a well-known landmarking technique to provide efficient algorithms for supervised learning using "good" similarity functions. We demonstrate the effectiveness of our model on three important supervised learning problems: a) real-valued regression, b) ordinal regression and c) ranking where we show that our method guarantees bounded generalization error. Furthermore, for the case of real-valued regression, we give a natural goodness definition that, when used in conjunction with a recent result in sparse vector recovery, guarantees a sparse predictor with bounded generalization error. Finally, we report results of our learning algorithms on regression and ordinal regression tasks using non-PSD similarity functions and demonstrate the effectiveness of our algorithms, especially that of the sparse landmark selection algorithm that achieves significantly higher accuracies than the baseline methods while offering reduced computational costs.

## 1 Introduction

The goal of this paper is to develop an extended framework for supervised learning with similarity functions. Kernel learning algorithms [1] have become the mainstay of discriminative learning with an incredible amount of effort having been put in, both from the theoretician's as well as the practitioner's side. However, these algorithms typically require the similarity function to be a positive semi-definite (PSD) function, which can be a limiting factor for several applications. Reasons being: 1) the Mercer's condition is a formal statement that is hard to verify, 2) several natural notions of similarity that arise in practical scenarios are not PSD, and 3) it is not clear as to why an artificial constraint like PSD-ness should limit the usability of a kernel.

Several recent papers have demonstrated that indefinite similarity functions can indeed be successfully used for learning [2, 3, 4, 5]. However, most of the existing work focuses on classification tasks and provides specialized techniques for the same, albeit with little or no theoretical guarantees. A notable exception is the line of work by [6, 7, 8] that defines a goodness criterion for a similarity function and then provides an algorithm that can exploit this goodness criterion to obtain provably accurate classifiers. However, their definitions are yet again restricted to the problem of classification as they take a "margin" based view of the problem that requires positive points to be more similar to positive points than to negative points by at least a constant margin.

In this work, we instead take a "target-value" point of view and require that target values of similar points be similar. Using this view, we propose a generic goodness definition that also admits the goodness definition of [6] for classification as a special case. Furthermore, our definition can be seen as imposing the existence of a smooth function over a generic space defined by similarity functions, rather than over a Hilbert space as required by typical goodness definitions of PSD kernels.

We then adapt the landmarking technique of [6] to provide an efficient algorithm that reduces learning tasks to corresponding learning problems over a linear space. The main technical challenge at this stage is to show that such reductions are able to provide good generalization error bounds for the learning tasks at hand. To this end, we consider three specific problems: a) regression, b) ordinal regression, and c) ranking. For each problem, we define appropriate surrogate loss functions, and show that our algorithm is able to, for each specific learning task, guarantee bounded generalization error with polynomial sample complexity. Moreover, by adapting a general framework given by [9], we show that these guarantees do not require the goodness definition to be overly restrictive by showing that our definitions admit all good PSD kernels as well.

For the problem of real-valued regression, we additionally provide a goodness definition that captures the intuition that usually, only a small number of landmarks are influential w.r.t. the learning task. However, to recover these landmarks, the uniform sampling technique would require sampling a large number of landmarks thus increasing the training/test time of the predictor. We address this issue by applying a sparse vector recovery algorithm given by [10] and show that the resulting sparse predictor still has bounded generalization error.

We also address an important issue faced by algorithms that use landmarking as a feature constructions step viz [6, 7, 8], namely that they typically assume separate landmark and training sets for ease of analysis. In practice however, one usually tries to overcome paucity of training data by reusing training data as landmark points as well. We use an argument outlined in [11] to theoretically justify such "double dipping" in our case. The details of the argument are given in Appendix B.

We perform several experiments on benchmark datasets that demonstrate significant performance gains for our methods over the baseline of kernel regression. Our sparse landmark selection technique provides significantly better predictors that are also more efficient at test time.

**Related Work**: Existing approaches to extend kernel learning algorithms to indefinite kernels can be classified into three broad categories: a) those that use indefinite kernels directly with existing kernel learning algorithms, resulting in non-convex formulations [2, 3]. b) those that convert a given indefinite kernel into a PSD one by either projecting onto the PSD-cone [4, 5] or performing other spectral operations [12]. The second approach is usually expensive due to the spectral operations involved apart from making the method inherently transductive. Moreover, any domain knowledge stored in the original kernel is lost due to these task oblivious operations and consequently, no generalization guarantees can be given. c) those that use notions of "task-kernel alignment" or equivalently, notions of "goodness" of a kernel, to give learning algorithms [6, 7, 8]. This approach enjoys several advantages over the other approaches listed above. These models are able to use the indefinite kernel directly with existing PSD kernel learning techniques; all the while retaining the ability to give generalization bounds that quantitatively parallel those of PSD kernel learning models. In this paper, we adopt the third approach for general supervised learning problem.

## 2 Problem formulation and Preliminaries

The goal in similarity-based supervised learning is to closely approximate a *target* predictor $y : \mathcal{X} \rightarrow \mathcal{Y}$ over some domain $\mathcal{X}$ using a *hypothesis* $\hat{f}(\,\cdot\,; K) : \mathcal{X} \rightarrow \mathcal{Y}$ that restricts its interaction with data points to computing similarity values given by $K$. Now, if the similarity function $K$ is not discriminative enough for the given task then we cannot hope to construct a predictor out of it that enjoys good generalization properties. Hence, it is natural to define the "goodness" of a given similarity function with respect to the learning task at hand.

**Definition 1** (Good similarity function: preliminary)**.** *Given a learning task $y : \mathcal{X} \rightarrow \mathcal{Y}$ over some distribution $\mathcal{D}$, a similarity function $K : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ is said to be $(\epsilon_0, B)$-good with respect to this task if there exists some bounded weighing function $w : \mathcal{X} \rightarrow [-B, B]$ such that for at least a $(1 - \epsilon_0)$ $\mathcal{D}$-fraction of the domain, we have* $y(\mathbf{x}) = \underset{\mathbf{x}' \sim \mathcal{D}}{\mathbb{E}} [\![ w(\mathbf{x}')y(\mathbf{x}')K(\mathbf{x}, \mathbf{x}') ]\!]$ .

The above definition is inspired by the definition of a "good" similarity function with respect to classification tasks given in [6]. However, their definition is tied to class labels and thus applies only

**Algorithm 1** Supervised learning with Similarity functions

```
Input: A target predictor $y : \mathcal{X} \to \mathcal{Y}$ over a distribution $\mathcal{D}$, an $(\epsilon_0, B)$-good similarity function $K$, labeled
    training points sampled from $\mathcal{D}$: $\mathcal{T} = \left\{(\mathbf{x}_1^t, y_1), \ldots, (\mathbf{x}_n^t, y_n)\right\}$, loss function $\ell_S : \mathbb{R} \times \mathcal{Y} \to \mathbb{R}^+$.
Output: A predictor $\hat{f} : \mathcal{X} \to \mathbb{R}$ with bounded true loss over $\mathcal{D}$
 1: Sample $d$ unlabeled landmarks from $\mathcal{D}$: $\mathcal{L} = \left\{\mathbf{x}_1^l, \ldots, \mathbf{x}_d^l\right\}$
    // Else subsample $d$ landmarks from $\mathcal{T}$ (see Appendix B for details)
 2: $\Psi_{\mathcal{L}} : \mathbf{x} \mapsto 1/\sqrt{d}\left(K(\mathbf{x}, \mathbf{x}_1^l), \ldots, K(\mathbf{x}, \mathbf{x}_d^l)\right) \in \mathbb{R}^d$
 3: $\hat{\mathbf{w}} = \underset{\mathbf{w} \in \mathbb{R}^d : \|\mathbf{w}\|_2 \leq B}{\arg\min} \sum_i^n \ell_S\left(\left\langle \mathbf{w}, \Psi_{\mathcal{L}}(\mathbf{x}_i^t)\right\rangle, y_i\right)$
 4: return $\hat{f} : \mathbf{x} \mapsto \langle \hat{\mathbf{w}}, \Psi_{\mathcal{L}}(\mathbf{x})\rangle$
```

to classification tasks. Similar to [6], the above definition calls a similarity function $K$ "good" if the target value $y(\mathbf{x})$ of a given point $\mathbf{x}$ can be approximated in terms of (a weighted combination of) the target values of the $K$-"neighbors" of $\mathbf{x}$. Also, note that this definition automatically enforces a smoothness prior on the framework.

However the above definition is too rigid. Moreover, it defines goodness in terms of violations, a non-convex loss function. To remedy this, we propose an alternative definition that incorporates an arbitrary (but in practice always convex) loss function.

**Definition 2** (Good similarity function: final)**.** *Given a learning task $y : \mathcal{X} \to \mathcal{Y}$ over some distribution $\mathcal{D}$, a similarity function $K$ is said to be $(\epsilon_0, B)$-good with respect to a loss function $\ell_S : \mathbb{R} \times \mathcal{Y} \to \mathbb{R}$ if there exists some bounded weighing function $w : \mathcal{X} \to [-B, B]$ such that if we define a predictor as $f(\mathbf{x}) := \underset{\mathbf{x}' \sim \mathcal{D}}{\mathbb{E}} [\![w(\mathbf{x}')K(\mathbf{x}, \mathbf{x}')]\!]$, then we have $\underset{\mathbf{x} \sim \mathcal{D}}{\mathbb{E}} [\![\ell_S(f(\mathbf{x}), y(\mathbf{x}))]\!] \leq \epsilon_0$.*

Note that Definition 2 reduces to Definition 1 for $\ell_S(a, b) = \mathbb{1}_{\{a \neq b\}}$. Moreover, for the case of binary classification where $y \in \{-1, +1\}$, if we take $\ell_S(a, b) = \mathbb{1}_{\{ab \leq B\gamma\}}$, then we recover the $(\epsilon_0, \gamma)$-goodness definition of a similarity function, given in Definition 3 of [6]. Also note that, assuming $\sup_{\mathbf{x} \in \mathcal{X}} \{|y(\mathbf{x})|\} < \infty$ we can w.l.o.g. merge $w(\mathbf{x}')y(\mathbf{x}')$ into a single term $w(\mathbf{x}')$.

Having given this definition we must make sure that "good" similarity functions allow the construction of effective predictors (Utility property). Moreover, we must make sure that the definition does not exclude commonly used PSD kernels (Admissibility property). Below, we formally define these two properties and in later sections, show that for each of the learning tasks considered, our goodness definition satisfies these two properties.

### 2.1 Utility

**Definition 3** (Utility)**.** *A similarity function $K$ is said to be $\epsilon_0$-useful w.r.t. a loss function $\ell_{actual}(\cdot, \cdot)$ if the following holds: there exists a learning algorithm $\mathcal{A}$ that, for any $\epsilon_1, \delta > 0$, when given poly$(1/\epsilon_1, \log(1/\delta))$ "labeled" and "unlabeled" samples from the input distribution $\mathcal{D}$, with probability at least $1 - \delta$ , generates a hypothesis $\hat{f}(\mathbf{x}; K)$ s.t. $\underset{\mathbf{x} \sim \mathcal{D}}{\mathbb{E}} \left[\!\!\left[\ell_{actual}\left(\hat{f}(\mathbf{x}), y(\mathbf{x})\right)\right]\!\!\right] \leq \epsilon_0 + \epsilon_1$. Note that $\hat{f}(\mathbf{x}; K)$ is restricted to access the data solely through $K$.*

Here, the $\epsilon_0$ term captures the *misfit* or the bias of the similarity function with respect to the learning problem. Notice that the above utility definition allows for learning from unlabeled data points and thus puts our approach in the semi-supervised learning framework.

All our utility guarantees proceed by first using unlabeled samples as *landmarks* to construct a landmarked space. Next, using the goodness definition, we show the existence of a good linear predictor in the landmarked space. This guarantee is obtained in two steps as outlined in Algorithm 1: first of all we choose $d$ unlabeled landmark points and construct a map $\Psi : \mathcal{X} \to \mathbb{R}^d$ (see Step 1 of Algorithm 1) and show that there exists a linear predictor over $\mathbb{R}^d$ that closely approximates the predictor $f$ used in Definition 2 (see Lemma 15 in Appendix A). In the second step, we learn a predictor (over the landmarked space) using ERM over a fresh labeled training set (see Step 3 of Algorithm 1). We then use individual task-specific arguments and Rademacher average-based generalization bounds [13] thus proving the utility of the similarity function.

### 2.2 Admissibility

In order to show that our models are not too rigid, we would prove that they admit good PSD kernels. The notion of a good PSD kernel for us will be one that corresponds to a prevalent large margin technique for the given problem. In general, most notions correspond to the existence of a linear operator in the RKHS of the kernel that has small loss at large margin. More formally,

**Definition 4** (Good PSD Kernel)**.** *Given a learning task $y : \mathcal{X} \rightarrow \mathcal{Y}$ over some distribution $\mathcal{D}$, a PSD kernel $K : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ with associated RKHS $\mathcal{H}_K$ and canonical feature map $\Phi_K : \mathcal{X} \rightarrow \mathcal{H}_K$ is said to be $(\epsilon_0, \gamma)$-good with respect to a loss function $\ell_K : \mathbb{R} \times \mathcal{Y} \rightarrow \mathbb{R}$ if there exists $\mathbf{W}^* \in \mathcal{H}_K$ such that $\|\mathbf{W}^*\| = 1$ and*

$$\underset{\mathbf{x} \sim \mathcal{D}}{\mathbb{E}} \left[\!\!\left[ \ell_K \left( \frac{\langle \mathbf{W}^*, \Phi_K(\mathbf{x}) \rangle}{\gamma}, y(\mathbf{x}) \right) \right]\!\!\right] < \epsilon_0$$

We will show, for all the learning tasks considered, that every $(\epsilon_0, \gamma)$-good PSD kernel, when treated as simply a similarity function with no consideration of its RKHS, is also $(\epsilon + \epsilon_1, B)$-good for arbitrarily small $\epsilon_1$ with $B = h(\gamma, \epsilon_1)$ for some function $h$. To prove these results we will adapt techniques introduced in [9] with certain modifications and task-dependent arguments.

## 3 Applications

We will now instantiate the general learning model described above to real-valued regression, ordinal regression and ranking by providing utility and admissibility guarantees. Due to lack of space, we relegate all proofs as well as the discussion on ranking to the supplementary material (Appendix F).

### 3.1 Real-valued Regression

Real-valued regression is a quintessential learning problem [1] that has received a lot of attention in the learning literature. In the following we shall present algorithms for performing real-valued regression using non-PSD similarity measures. We consider the problem with $\ell_{\text{actual}}(a, b) = |a - b|$ as the true loss function. For the surrogates $\ell_S$ and $\ell_K$, we choose the $\epsilon$-insensitive loss function [1] defined as follows:

$$\ell_\epsilon(a, b) = \ell_\epsilon(a - b) = \begin{cases} 0, & \text{if } |a - b| < \epsilon, \\ |a - b| - \epsilon, & \text{otherwise.} \end{cases}$$

The above loss function automatically gives us notions of good kernels and similarity functions by appealing to Definitions 4 and 2 respectively. It is easy to transfer error bounds in terms of absolute error to those in terms of mean squared error (MSE), a commonly used performance measure for real-valued regression. See Appendix D for further discussion on the choice of the loss function.

Using the landmarking strategy described in Section 2.1, we can reduce the problem of real regression to that of a linear regression problem in the landmarked space. More specifically, the ERM step in Algorithm 1 becomes the following: $\underset{\mathbf{w} \in \mathbb{R}^d : \|\mathbf{w}\|_2 \leq B}{\arg\min} \ \sum_i^n \ell_\epsilon(\langle \mathbf{w}, \Psi_{\mathcal{L}}(\mathbf{x}_i) \rangle - y_i)$.

There exist solvers (for instance [14]) to efficiently solve the above problem on linear spaces. Using proof techniques sketched in Section 2.1 along with specific arguments for the $\epsilon$-insensitive loss, we can prove generalization guarantees and hence utility guarantees for the similarity function.

**Theorem 5.** *Every similarity function that is $(\epsilon_0, B)$-good for a regression problem with respect to the insensitive loss function $\ell_\epsilon(\cdot, \cdot)$ is $(\epsilon_0 + \epsilon)$-useful with respect to absolute loss as well as $(B\epsilon_0 + B\epsilon)$-useful with respect to mean squared error. Moreover, both the dimensionality of the landmarked space as well as the labeled sample complexity can be bounded by $\mathcal{O}\left(\frac{B^2}{\epsilon_1^2} \log \frac{1}{\delta}\right)$.*

We are also able to prove the following (tight) admissibility result:

**Theorem 6.** *Every PSD kernel that is $(\epsilon_0, \gamma)$-good for a regression problem is, for any $\epsilon_1 > 0$, $\left(\epsilon_0 + \epsilon_1, \mathcal{O}\left(\frac{1}{\epsilon_1 \gamma^2}\right)\right)$-good as a similarity function as well. Moreover, for any $\epsilon_1 < 1/2$ and any $\gamma < 1$, there exists a regression instance and a corresponding kernel that is $(0, \gamma)$-good for the regression problem but only $(\epsilon_1, B)$-good as a similarity function for $B = \Omega\left(\frac{1}{\epsilon_1 \gamma^2}\right)$.*

### 3.2 Sparse regression models

An artifact of a random choice of landmarks is that very few of them might turn out to be "informative" with respect to the prediction problem at hand. For instance, in a network, there might exist *hubs* or *authoritative* nodes that yield rich information about the learning problem. If the relative abundance of such nodes is low then random selection would compel us to choose a large number of landmarks before enough "informative" ones have been collected.

However this greatly increases training and testing times due to the increased costs of constructing the landmarked space. Thus, the ability to prune away irrelevant landmarks would speed up training and test routines. We note that this issue has been addressed before in literature [8, 12] by way of landmark selection heuristics. In contrast, we guarantee that our predictor will select a small number of landmarks while incurring bounded generalization error. However this requires a careful restructuring of the learning model to incorporate the "informativeness" of landmarks.

**Definition 7.** *A similarity function $K$ is said to be $(\epsilon_0, B, \tau)$-good for a real-valued regression problem $y : \mathcal{X} \rightarrow \mathbb{R}$ if for some bounded weight function $w : \mathcal{X} \rightarrow [-B, B]$ and* choice function *$R : \mathcal{X} \rightarrow \{0, 1\}$ with $\underset{\mathbf{x}\sim\mathcal{D}}{\mathbb{E}}[\![R(\mathbf{x})]\!] = \tau$, the predictor $f : \mathbf{x} \mapsto \underset{\mathbf{x}'\sim\mathcal{D}}{\mathbb{E}}[\![w(\mathbf{x}')K(\mathbf{x}, \mathbf{x}')|R(\mathbf{x}')]\!]$ has bounded $\epsilon$-insensitive loss i.e. $\underset{\mathbf{x}\sim\mathcal{D}}{\mathbb{E}}[\![\ell_\epsilon\left(f(\mathbf{x}), y(\mathbf{x})\right)]\!] < \epsilon_0$.*

The role of the choice function is to single out informative landmarks, while $\tau$ specifies the relative density of informative landmarks. Note that the above definition is similar in spirit to the goodness definition presented in [15]. While the motivation behind [15] was to give an improved admissibility result for binary classification, we squarely focus on the utility guarantees; with the aim of accelerating our learning algorithms via landmark pruning.

We prove the utility guarantee in three steps as outlined in Appendix D. First, we use the usual landmarking step to project the problem onto a linear space. This step guarantees the following:

**Theorem 8.** *Given a similarity function that is $(\epsilon_0, B, \tau)$-good for a regression problem, there exists a randomized map $\Psi : \mathcal{X} \rightarrow \mathbb{R}^d$ for $d = \mathcal{O}\left(\frac{B^2}{\tau\epsilon_1^2}\log\frac{1}{\delta}\right)$ such that with probability at least $1 - \delta$, there exists a linear operator $\tilde{f} : \mathbf{x} \mapsto \langle \mathbf{w}, \mathbf{x} \rangle$ over $\mathbb{R}^d$ such that $\left\|\mathbf{w}\right\|_1 \leq B$ with $\epsilon$-insensitive loss bounded by $\epsilon_0 + \epsilon_1$. Moreover, with the same confidence we have $\left\|\mathbf{w}\right\|_0 \leq \frac{3d\tau}{2}$.*

Our proof follows that of [15], however we additionally prove sparsity of $\mathbf{w}$ as well. The number of landmarks required here is a $\Omega\left(1/\tau\right)$ fraction greater than that required by Theorem 5. This formally captures the intuition presented earlier of a small fraction of dimensions (read landmarks) being actually relevant to the learning problem. So, in the second step, we use the *Forward Greedy Selection* algorithm given in [10] to learn a sparse predictor. The use of this learning algorithm necessitates the use of a different generalization bound in the final step to complete the utility guarantee given below. We refer the reader to Appendix D for the details of the algorithm and its utility analysis.

**Theorem 9.** *Every similarity function that is $(\epsilon_0, B, \tau)$-good for a regression problem with respect to the insensitive loss function $\ell_\epsilon\left(\cdot, \cdot\right)$ is $(\epsilon_0 + \epsilon)$-useful with respect to absolute loss as well; with the dimensionality of the landmarked space being bounded by $\mathcal{O}\left(\frac{B^2}{\tau\epsilon_1^2}\log\frac{1}{\delta}\right)$ and the labeled sampled complexity being bounded by $\mathcal{O}\left(\frac{B^2}{\epsilon_1^2}\log\frac{B}{\epsilon_1\delta}\right)$. Moreover, this utility can be achieved by an $\mathcal{O}\left(\tau\right)$-sparse predictor on the landmarked space.*

We note that the improvements obtained here by using the sparse learning methods of [10] provide $\Omega\left(\tau\right)$ increase in sparsity. We now prove admissibility results for this sparse learning model. We do this by showing that the dense model analyzed in Theorem 5 and that given in Definition 7 are interpretable in each other for an appropriate selection of parameters. The guarantees in Theorem 6 can then be invoked to conclude the admissibility proof.

**Theorem 10.** *Every $(\epsilon_0, B)$-good similarity function $K$ is also $\left(\epsilon_0, B, \frac{\bar{w}}{B}\right)$-good where $\bar{w} = \underset{\mathbf{x}\sim\mathcal{D}}{\mathbb{E}}[\![|w(\mathbf{x})|]\!]$. Moreover, every $(\epsilon_0, B, \tau)$-good similarity function $K$ is also $(\epsilon_0, B/\tau)$-good.*

Using Theorem 6, we immediately have the following corollary:

**Corollary 11.** *Every PSD kernel that is $(\epsilon_0, \gamma)$-good for a regression problem is, for any $\epsilon_1 > 0$, $\left(\epsilon_0 + \epsilon_1, \mathcal{O}\left(\frac{1}{\epsilon_1\gamma^2}\right), 1\right)$-good as a similarity function as well.*

### 3.3 Ordinal Regression

The problem of ordinal regression requires an accurate prediction of (discrete) labels coming from a finite ordered set $[r] = \{1, 2, \ldots, r\}$. The problem is similar to both classification and regression, but has some distinct features due to which it has received independent attention [16, 17] in domains such as product ratings etc. The most popular performance measure for this problem is the absolute loss which is the absolute difference between the predicted and the true labels.

A natural and rather tempting way to solve this problem is to relax the problem to real-valued regression and threshold the output of the learned real-valued predictor using predefined thresholds $b_1, \ldots, b_r$ to get discrete labels. Although this approach has been prevalent in literature [17], as the discussion in the supplementary material shows, this leads to poor generalization guarantees in our model. More specifically, a goodness definition constructed around such a direct reduction is only able to ensure $(\epsilon_0 + 1)$-utility i.e. the absolute error rate is always greater than 1.

One of the reasons for this is the presence of the thresholding operation that makes it impossible to distinguish between instances that would not be affected by small perturbations to the underlying real-valued predictor and those that would. To remedy this, we enforce a (soft) margin with respect to thresholding that makes the formulation more robust to noise. More formally, we expect that if a point belongs to the label $i$, then in addition to being sandwiched between the thresholds $b_i$ and $b_{i+1}$, it should be separated from these by a margin as well i.e. $b_i + \gamma \leq f(\mathbf{x}) \leq b_{i+1} - \gamma$.

This is a direct generalization of the margin principle in classification where we expect $\mathbf{w}^\top \mathbf{x} > b + \gamma$ for positively labeled points and $\mathbf{w}^\top \mathbf{x} < b - \gamma$ for negatively labeled points. Of course, wherein classification requires a single threshold, we require several, depending upon the number of labels. For any $x \in \mathbb{R}$, let $[x]_+ = \max\{x, 0\}$. Thus, if we define the $\gamma$-margin loss function to be $[x]_\gamma := [\gamma - x]_+$ (note that this is simply the well known hinge loss function scaled by a factor of $\gamma$), we can define our goodness criterion as follows:

**Definition 12.** *A similarity function $K$ is said to be $(\epsilon_0, B)$-good for an ordinal regression problem $y : \mathcal{X} \to [r]$ if for some bounded weight function $w : \mathcal{X} \to [-B, B]$ and some (unknown but fixed) set of thresholds $\{b_i\}_{i=1}^r$ with $b_1 = -\infty$, the predictor $f : \mathbf{x} \mapsto \mathop{\mathbb{E}}_{\mathbf{x}' \sim \mathcal{D}} [\![ w(\mathbf{x}')K(\mathbf{x}, \mathbf{x}') ]\!]$ satisfies*

$$\mathop{\mathbb{E}}_{\mathbf{x} \sim \mathcal{D}} \left[\!\!\left[ \left[f(\mathbf{x}) - b_{y(\mathbf{x})}\right]_\gamma + \left[b_{y(\mathbf{x})+1} - f(\mathbf{x})\right]_\gamma \right]\!\!\right] < \epsilon_0.$$

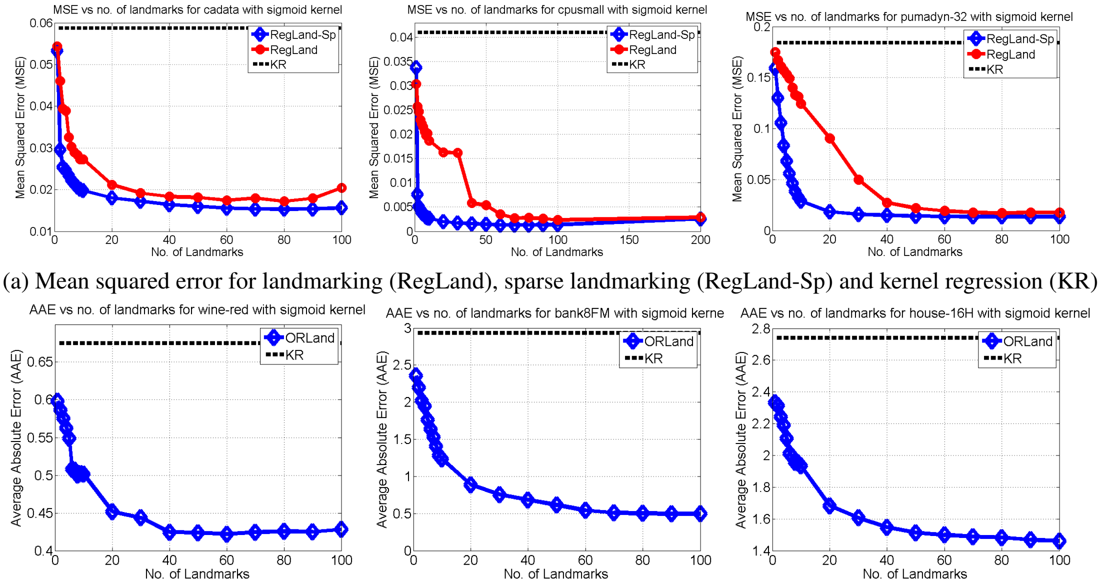
We now give utility guarantees for our learning model. We shall give guarantees on both the misclassification error as well as the absolute error of our learned predictor. We say that a set of points $x_1, \ldots, x_i \ldots$ is $\Delta$-spaced if $\min_{i \neq j} \{|x_i - x_j|\} \geq \Delta$. Define the function $\psi_\Delta(x) = \frac{x + \Delta - 1}{\Delta}$.

**Theorem 13.** *Let $K$ be a similarity function that is $(\epsilon_0, B)$-good for an ordinal regression problem with respect to $\Delta$-spaced thresholds and $\gamma$-margin loss. Let $\bar{\gamma} = \max\{\gamma, 1\}$. Then $K$ is $\psi_{(\Delta/\bar{\gamma})}\left(\frac{\epsilon_0}{\bar{\gamma}}\right)$-useful with respect to ordinal regression error (absolute loss). Moreover, $K$ is $\left(\frac{\epsilon_0}{\bar{\gamma}}\right)$-useful with respect to the zero-one mislabeling error as well.*

We can bound, both dimensionality of the landmarked space as well as labeled sampled complexity, by $\mathcal{O}\left(\frac{B^2}{\epsilon_1^2} \log \frac{1}{\delta}\right)$. Notice that for $\epsilon_0 < 1$ and large enough $d, n$, we can ensure that the ordinal regression error rate is also bounded above by 1 since $\sup_{x \in [0,1], \Delta > 0} (\psi_\Delta(x)) = 1$. This is in contrast with the direct reduction to real valued regression which has ordinal regression error rate bounded *below* by 1. This indicates the advantage of the present model over a naive reduction to regression.

We can show that our definition of a good similarity function admits all good PSD kernels as well. The kernel goodness criterion we adopt corresponds to the large margin framework proposed by [16]. We refer the reader to Appendix E.3 for the definition and give the admissibility result below.

**Theorem 14.** *Every PSD kernel that is $(\epsilon_0, \gamma)$-good for an ordinal regression problem is also $\left(\gamma_1 \epsilon_0 + \epsilon_1, \mathcal{O}\left(\frac{\gamma_1^2}{\epsilon_1 \gamma^2}\right)\right)$-good as a similarity function with respect to the $\gamma_1$-margin loss for any $\gamma_1, \epsilon_1 > 0$. Moreover, for any $\epsilon_1 < \gamma_1/2$, there exists an ordinal regression instance and a corresponding kernel that is $(0, \gamma)$-good for the ordinal regression problem but only $(\epsilon_1, B)$-good as a similarity function with respect to the $\gamma_1$-margin loss function for $B = \Omega\left(\frac{\gamma_1^2}{\epsilon_1 \gamma^2}\right)$.*



(a) Mean squared error for landmarking (RegLand), sparse landmarking (RegLand-Sp) and kernel regression (KR)

(b) Avg. absolute error for landmarking (ORLand) and kernel regression (KR) on ordinal regression datasets

Figure 1: Performance of landmarking algorithms with increasing number of landmarks on real-valued regression (Figure 1a) and ordinal regression (Figure 1b) datasets.

| Datasets | Sigmoid kernel | | Manhattan kernel | |
|---|---|---|---|---|
| | KR | Land-Sp | KR | Land-Sp |
| Abalone [18]<br>$N = 4177$<br>$d = 8$ | 2.1e-02<br>(8.3e-04) | 6.2e-03<br>(8.4e-04) | 1.7e-02<br>(7.1e-04) | 6.0e-03<br>(3.7e-04) |
| Bodyfat [19]<br>$N = 252$<br>$d = 14$ | 4.6e-04<br>(6.5e-05) | 9.5e-05<br>(1.3e-04) | 3.9e-04<br>(2.2e-05) | 3.5e-05<br>(1.3e-05) |
| CAHousing [19]<br>$N = 20640$<br>$d = 8$ | 5.9e-02<br>(2.3e-04) | 1.6e-02<br>(6.2e-04) | 5.8e-02<br>(1.9e-04) | 1.5e-02<br>(1.4e-04) |
| CPUData [20]<br>$N = 8192$<br>$d = 12$ | 4.1e-02<br>(1.6e-03) | 1.4e-03<br>(1.7e-04) | 4.3e-02<br>(1.6e-03) | 1.2e-03<br>(3.2e-05) |
| PumaDyn-8 [20]<br>$N = 8192$<br>$d = 8$ | 2.3e-01<br>(4.6e-03) | 1.4e-02<br>(4.5e-04) | 2.3e-01<br>(4.5e-03) | 1.4e-02<br>(4.8e-04) |
| PumaDyn-32 [20]<br>$N = 8192$<br>$d = 32$ | 1.8e-01<br>(3.6e-03) | 1.4e-02<br>(3.7e-04) | 1.8e-01<br>(3.6e-03) | 1.4e-02<br>(3.1e-04) |

(a) Mean squared error for real regression

| Datasets | Sigmoid kernel | | Manhattan kernel | |
|---|---|---|---|---|
| | KR | ORLand | KR | ORLand |
| Wine-Red [18]<br>$N = 1599$<br>$d = 11$ | 6.8e-01<br>(2.8e-02) | 4.2e-01<br>(3.8e-02) | 6.7e-01<br>(3.0e-02) | 4.5e-01<br>(3.2e-02) |
| Wine-White [18]<br>$N = 4898$<br>$d = 11$ | 6.2e-01<br>(2.0e-02) | 8.9e-01<br>(8.5e-01) | 6.2e-01<br>(2.0e-02) | 4.9e-01<br>(1.5e-02) |
| Bank-8 [20]<br>$N = 8192$<br>$d = 8$ | 2.9e+0<br>(6.2e-02) | 6.1e-01<br>(4.4e-02) | 2.7e+0<br>(6.6e-02) | 6.3e-01<br>(1.7e-02) |
| Bank-32 [20]<br>$N = 8192$<br>$d = 32$ | 2.7e+0<br>(1.2e-01) | 1.6e+0<br>(2.3e-02) | 2.6e+0<br>(8.1e-02) | 1.6e+0<br>(9.4e-02) |
| House-8 [20]<br>$N = 22784$<br>$d = 8$ | 2.8e+0<br>(9.3e-03) | 1.5e+0<br>(2.0e-02) | 2.7e+0<br>(1.0e-02) | 1.4e+0<br>(1.2e-02) |
| House-16 [20]<br>$N = 22784$<br>$d = 16$ | 2.7e+0<br>(2.0e-02) | 1.5e+0<br>(1.0e-02) | 2.8e+0<br>(2.0e-02) | 1.4e+0<br>(2.3e-02) |

(b) Mean absolute error for ordinal regression

Table 1: Performance of landmarking-based algorithms (with 50 landmarks) vs. baseline kernel regression (KR). Values in parentheses indicate standard deviation values. Values in the first columns indicate dataset source (in parentheses), size (N) and dimensionality (d).

Due to lack of space we refer the reader to Appendix F for a discussion on ranking models that includes utility and admissibility guarantees with respect to the popular NDCG loss.

# 4 Experimental Results

In this section we present an empirical evaluation of our learning models for the problems of real-valued regression and ordinal regression on benchmark datasets taken from a variety of sources [18, 19, 20]. In all cases, we compare our algorithms against kernel regression (KR), a well known technique [21] for non-linear regression, whose predictor is of the form:

$$f : \mathbf{x} \mapsto \frac{\sum_{\mathbf{x}_i \in \mathcal{T}} y(\mathbf{x}_i) K(\mathbf{x}, \mathbf{x}_i)}{\sum_{\mathbf{x}_i \in \mathcal{T}} K(\mathbf{x}, \mathbf{x}_i)},$$

where $\mathcal{T}$ is the training set. We selected KR as the baseline as it is a popular regression method that does not require similarity functions to be PSD. For ordinal regression problems, we rounded off the result of the KR predictor to get a discrete label. We implemented all our algorithms as well as the baseline KR method in Matlab. In all our experiments we report results across $5$ random splits on the (indefinite) Sigmoid: $K(\mathbf{x},\mathbf{y}) = \tanh(a\langle\mathbf{x},\mathbf{y}\rangle + r)$ and Manhattan: $K(\mathbf{x},\mathbf{y}) = -\left\|\mathbf{x}-\mathbf{y}\right\|_1$ kernels. Following standard practice, we fixed $r = -1$ and $a = 1/d_{\text{orig}}$ for the Sigmoid kernel where $d_{\text{orig}}$ is the dimensionality of the dataset.

**Real valued regression**: For this experiment, we compare our methods (RegLand and RegLand-Sp) with the KR method. For RegLand, we constructed the landmarked space as specified in Algorithm 1 and learned a linear predictor using the LIBLINEAR package [14] that minimizes $\epsilon$-insensitive loss. In the second algorithm (RegLand-Sp), we used the sparse learning algorithm of [10] on the landmarked space to learn the best predictor for a given sparsity level. Due to its simplicity and good convergence properties, we implemented the *Fully Corrective* version of the Forward Greedy Selection algorithm with squared loss as the surrogate.

We evaluated all methods using Mean Squared Error (MSE) on the test set. Figure 1a shows the MSE incurred by our methods along with reference values of accuracies obtained by KR as landmark sizes increase. The plots clearly show that our methods incur significantly lesser error than KR. Moreover, RegLand-Sp learns more accurate predictors using the same number of landmarks. For instance, when learning using the Sigmoid kernel on the `CPUData` dataset, at $20$ landmarks, RegLand is able to guarantee an MSE of $0.016$ whereas RegLand-Sp offers an MSE of less than $0.02$ ; MLKR is only able to guarantee an MSE rate of $0.04$ for this dataset. In Table 1a, we compare accuracies of the two algorithms when given $50$ landmark points with those of KR for the Sigmoid and Manhattan kernels. We find that in all cases, RegLand-Sp gives superior accuracies than KR. Moreover, the Manhattan kernel seems to match or outperform the Sigmoid kernel on all the datasets.

**Ordinal Regression**: Here, we compare our method with the baseline KR method on benchmark datasets. As mentioned in Section 3.3, our method uses the EXC formulation of [16] along with landmarking scheme given in Algorithm 1. We implemented a gradient descent-based solver (ORLand) to solve the primal formulation of EXC and used fixed equi-spaced thresholds instead of learning them as suggested by [16]. Of the six datasets considered here, the two `Wine` datasets are ordinal regression datasets where the quality of the wine is to be predicted on a scale from 1 to 10. The remaining four datasets are regression datasets whose labels were subjected to equi-frequency binning to obtain ordinal regression datasets [16]. We measured the average absolute error (AAE) for each method. Figure 1b compares ORLand with KR as the number of landmarks increases. Table 1b compares accuracies of ORLand for $50$ landmark points with those of KR for Sigmoid and Manhattan kernels. In almost all cases, ORLand gives a much better performance than KR. The Sigmoid kernel seems to outperform the Manhattan kernel on a couple of datasets.

We refer the reader to Appendix G for additional experimental results.

## 5 Conclusion

In this work we considered the general problem of supervised learning using non-PSD similarity functions. We provided a goodness criterion for similarity functions w.r.t. various learning tasks. This allowed us to construct efficient learning algorithms with provable generalization error bounds. At the same time, we were able to show, for each learning task, that our criterion is not too restrictive in that it admits all good PSD kernels. We then focused on the problem of identifying influential landmarks with the aim of learning sparse predictors. We presented a model that formalized the intuition that typically only a small fraction of landmarks is influential for a given learning problem. We adapted existing sparse vector recovery algorithms within our model to learn provably sparse predictors with bounded generalization error. Finally, we empirically evaluated our learning algorithms on benchmark regression and ordinal regression tasks. In all cases, our learning methods, especially the sparse recovery algorithm, consistently outperformed the kernel regression baseline.

An interesting direction for future research would be learning good similarity functions á la metric learning or kernel learning. It would also be interesting to conduct large scale experiments on real-world data such as social networks that naturally capture the notion of similarity amongst nodes.

### Acknowledgments

P. K. is supported by a Microsoft Research India Ph.D. fellowship award. Part of this work was done while P. K. was an intern at Microsoft Research Labs India, Bangalore.

## References

[1] Bernhard Schölkopf and Alex J. Smola. *Learning with Kernels : Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, 2002.

[2] Bernard Haasdonk. Feature Space Interpretation of SVMs with Indefinite Kernels. *IEEE Transactions on Pattern Analysis and Machince Intelligence*, 27(4):482–492, 2005.

[3] Cheng Soon Ong, Xavier Mary, Stéphane Canu, and Alexander J. Smola. Learning with non-positive Kernels. In *21st Annual International Conference on Machine Learning*, 2004.

[4] Yihua Chen, Maya R. Gupta, and Benjamin Recht. Learning Kernels from Indefinite Similarities. In *26th Annual International Conference on Machine Learning*, pages 145–152, 2009.

[5] Ronny Luss and Alexandre d'Aspremont. Support Vector Machine Classification with Indefinite Kernels. In *21st Annual Conference on Neural Information Processing Systems*, 2007.

[6] Maria-Florina Balcan and Avrim Blum. On a Theory of Learning with Similarity Functions. In *23rd Annual International Conference on Machine Learning*, pages 73–80, 2006.

[7] Liwei Wang, Cheng Yang, and Jufu Feng. On Learning with Dissimilarity Functions. In *24th Annual International Conference on Machine Learning*, pages 991–998, 2007.

[8] Purushottam Kar and Prateek Jain. Similarity-based Learning via Data Driven Embeddings. In *25th Annual Conference on Neural Information Processing Systems*, 2011.

[9] Nathan Srebro. How Good Is a Kernel When Used as a Similarity Measure? In *20th Annual Conference on Computational Learning Theory*, pages 323–335, 2007.

[10] Shai Shalev-Shwartz, Nathan Srebro, and Tong Zhang. Trading Accuracy for Sparsity in Optimization Problems with Sparsity Constraints. *SIAM Journal on Optimization*, 20(6):2807–2832, 2010.

[11] Nathan Srebro Shai Ben-David, Ali Rahimi. Generalization Bounds for Indefinite Kernel Machines. In *NIPS 2008 Workshop: New Challenges in Theoretical Machine Learning*, 2008.

[12] Yihua Chen, Eric K. Garcia, Maya R. Gupta, Ali Rahimi, and Luca Cazzanti. Similarity-based Classification: Concepts and Algorithms. *Journal of Machine Learning Research*, 10:747–776, 2009.

[13] Sham M. Kakade, Karthik Sridharan, and Ambuj Tewari. On the Complexity of Linear Prediction : Risk Bounds, Margin Bounds, and Regularization. In *22nd Annual Conference on Neural Information Processing Systems*, 2008.

[14] Chia-Hua Ho and Chih-Jen Lin. Large-scale Linear Support Vector Regression. `http://www.csie.ntu.edu.tw/~cjlin/papers/linear-svr.pdf`, retrieved on May 18, 2012, 2012.

[15] Maria-Florina Balcan, Avrim Blum, and Nathan Srebro. Improved Guarantees for Learning via Similarity Functions. In *21st Annual Conference on Computational Learning Theory*, pages 287–298, 2008.

[16] Wei Chu and S. Sathiya Keerthi. Support Vector Ordinal Regression. *Neural Computation*, 19(3):792–815, 2007.

[17] Shivani Agarwal. Generalization Bounds for Some Ordinal Regression Algorithms. In *19th International Conference on Algorithmic Learning Theory*, pages 7–21, 2008.

[18] A. Frank and Arthur Asuncion. UCI Machine Learning Repository. `http://archive.ics.uci.edu/ml`, 2010. University of California, Irvine, School of Information and Computer Sciences.

[19] StatLib Dataset Repository. `http://lib.stat.cmu.edu/datasets/`. Carnegie Mellon University.

[20] Delve Dataset Repository. `http://www.cs.toronto.edu/~delve/data/datasets.html`. University of Toronto.

[21] Kilian Q. Weinberger and Gerald Tesauro. Metric Learning for Kernel Regression. In *11th International Conference on Artificial Intelligence and Statistics*, pages 612–619, 2007.