Learning Safe Prediction for Semi-Supervised Regression[*]

Yu-Feng Li and **Han-Wen Zha** and **Zhi-Hua Zhou**
National Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210023, China
Collaborative Innovation Center of Novel Software Technology and Industrialization, Nanjing 210023, China
{liyf,zhouzh}@lamda.nju.edu.cn; zhahw12@gmail.com

Abstract

Semi-supervised learning (SSL) concerns how to improve performance via the usage of unlabeled data. Recent studies indicate that the usage of unlabeled data might even deteriorate performance. Although some proposals have been developed to alleviate such a fundamental challenge for semi-supervised classification, the efforts on semi-supervised regression (SSR) remain to be limited. In this work we consider the learning of a *safe* prediction from multiple semi-supervised regressors, which is not worse than a direct supervised learner with only labeled data. We cast it as a geometric projection issue with an efficient algorithm. Furthermore, we show that the proposal is provably safe and has already achieved the maximal performance gain, if the ground-truth label assignment is realized by a convex linear combination of base regressors. This provides insight to help understand safe SSR. Experimental results on a broad range of datasets validate the effectiveness of our proposal.

Introduction

Semi-supervised learning (SSL) concerns the problem on how to improve learning performance via the usage of additional unlabeled data. Such a learning framework has received a great deal of attention owing to immense demands in real-world applications from medical diagnosis to intrusion detection (Zhu and Goldberg 2009). Many SSL methods have been developed, e.g., generative model (Miller and Uyar 1997; Nigam et al. 2000), graph-based method (Blum and Chawla 2001; Zhu, Ghahramani, and Lafferty 2003; Zhou et al. 2004), disagreement-based method (Blum and Mitchell 1998; Zhou and Li 2010) and semi-supervised SVMs (Bennett and Demiriz 1999; Joachims 1999).

Despite the success of SSL, however, a considerable amount of empirical studies reveal that SSL with the exploitation of unlabeled data might even deteriorate learning performance (Cozman, Cohen, and Cirelo 2002; Chawla and Karakoulas 2005; Chapelle, Schölkopf, and Zien 2006; Zhu and Goldberg 2009; Balcan and Blum 2010; Yang and Priebe 2011; Li et al. 2013). It is highly desirable to study *safe* SSL scheme that on one side could often improve performance, on the other side will not hurt performance, since the users of SSL wont expect that SSL with the usage of more data performs worse than certain direct supervised learning with only labeled data. Recently several proposals (Li and Zhou 2005; 2015; Balsubramani and Freund 2015; Krijthe and Loog 2015; Li, Kwok, and Zhou 2016; Li, Wang, and Zhou 2016) have been developed to alleviate such a fundamental challenge for semi-supervised classification (SSC), while the efforts on semi-supervised regression (SSR) remain to be limited.

In this work we consider the question of how to learn a safe prediction from multiple semi-supervised regressors. To our knowledge, such a question has not been thoroughly studied. Specifically, let $\{\mathbf{f}_1, \dots, \mathbf{f}_b\}$ be multiple SSR predictions and $\mathbf{f}_0$ be the prediction of certain direct supervised learner, where $\mathbf{f}_i \in \mathbb{R}^u$, $i = 0, \dots, b$; b and u refer to the number of regressors and unlabeled instances. How to derive a final prediction $g(\mathbf{f}_1, \dots, \mathbf{f}_b, \mathbf{f}_0)$, such that for regression measurement $g(\mathbf{f}_1, \dots, \mathbf{f}_b, \mathbf{f}_0)$ could often be better than $\mathbf{f}_0$, meanwhile it would not be worse than $\mathbf{f}_0$.

We present a SAFER (SAFE semi-supervised Regression) method. Without domain knowledge about the reliabilities of learners, SAFER proposes to maximize the performance gain of $g(\mathbf{f}_1, \dots, \mathbf{f}_b, \mathbf{f}_0)$ against $\mathbf{f}_0$ assuming that the weights of SSR learners are from a convex set. This is formulated as a saddle-point convex-concave optimization (Nesterov 2013). In order to alleviate the computational overhead and to understand how the proposal works, we represent safe SSR problem as a geometric projection issue, which brings two prominent advantages. i) The resultant new formulation is a simple convex quadratic program and much easier to solve. ii) One can show that SAFER is provably safe and have already achieved the maximal performance gain, if the ground-truth label assignment is realized by a convex linear combination of base SSR learners, which provides insight to help understand safe SSR. Experimental results on a broad range of datasets validate that SAFER clearly improves safeness of SSR, in addition obtains highly comparable performance with state-of-the-art approaches.

In the following, we first review several related works and then present the SAFER method. Next we show the experiment. Finally we conclude this paper.

[*]This research was supported by the NSFC (61333014, 61403186), JiangsuSF (BK20140613), MSRA research fund and 863 Program (2015AA015406).

Related Work

This work is related to two branches of studies. One is **safe SSC**. This line of research is raised in very recent. (Li and Zhou 2011; 2015) is one early work to build safe semi-supervised SVMs. They optimize the worst-case performance gain given a set of candidate low-density separators, showing that the proposed S4VM (Safe Semi-Supervised SVM) is provably safe given that low-density assumption (Chapelle, Schölkopf, and Zien 2006) holds. (Krijthe and Loog 2015) presents to build a safe semi-supervised classifier, which learns a projection of a supervised least square classifier from all possible semi-supervised least square classifiers. (Balsubramani and Freund 2015) proposes to learn a robust prediction with the highest accuracy given that the ground-truth label assignment is restricted to specific candidate set. Most recently, (Li, Kwok, and Zhou 2016) concerns to build a generic safe SSC framework for variants of performance measures, e.g., AUC, F_1 score, Top_k precision. (Li, Wang, and Zhou 2016) is one early work studying the quality of graph in graph-based SSC. They propose a large margin principle to help judge the graph quality and empirically achieve promising performance via excluding poor quality graphs.

The other one is **SSR**. Two representative SSR studies were developed by (Zhou and Li 2005) and (Brefeld et al. 2006), which are both based on the employment of multiple regressors. (Zhou and Li 2005) proposes to use the co-training (Blum and Mitchell 1998) style algorithm for the learning of two semi-supervised regressors, so as to improve the final regression performance. (Brefeld et al. 2006) considers multi-view training data, proposing a co-regularization framework that enforces the predictive results of unlabeled data from multiple views to be consistent.

In comparison to safe SSC, in this paper we consider safe SSR. Furthermore, our proposal yields a better condition of safeness than that of (Li and Zhou 2015). Moreover, unlike (Li and Zhou 2015) whose formulation is non-convex and hard to achieve the global optima, our proposal is convex and easy to derive the optimal solution. Different from (Krijthe and Loog 2015), our proposal does not restrict the learner to be least square classifier and is applicable to various learners. We explicitly consider to maximize the performance gain, which is not taken into account in (Balsubramani and Freund 2015). In comparison to previous SSR studies, which typically work on improving performance, the deficiency on performance degradation when using unlabeled data, has not been studied in literatures.

The Proposed Method

In this section we first present problem setting and formulation, and then give its representation to geometric projection, finally study how the proposal works.

Problem Setting and Formulation

Remind that in SSR, particularly for the scenario of multiple semi-supervised regressors, we obtain b SSR predictions $\{\mathbf{f}_1, \dots, \mathbf{f}_b\}$ for unlabeled instances where $\mathbf{f}_i \in \mathbb{R}^u$, $i = 1, \dots, b$ and u refers to the number of unlabeled instances. On the other hand, one can train a direct supervised regression model (e.g., k nearest neighbor) using only labeled data and consequently yield another prediction $\mathbf{f}_0 \in \mathbb{R}^u$ for unlabeled instances. The underlying challenge is the learning of safe prediction $g(\{\mathbf{f}_1, \dots, \mathbf{f}_b\}, \mathbf{f}_0)$, which often outperforms $\mathbf{f}_0$, meanwhile could not be worse than $\mathbf{f}_0$.

We start with a simple scenario to alleviate this challenge, where the weights of SSR regressors are known. Specifically, let $\boldsymbol{\alpha} = [\alpha_1; \alpha_2; \dots; \alpha_b] \geq \mathbf{0}$ be the weights of individual regressors $\mathbf{f}_i$'s, reflecting how close $\mathbf{f}_i$'s are to the ground-truth. The larger the weight, the closer the regressor is to the ground-truth. We employ the difference of Mean Square Error (a popular criterion in regression task) to measure the performance gain against $\mathbf{f}_0$, i.e., $\Big(\|\mathbf{f}_0 - \mathbf{f}^*\|^2 - \|\mathbf{f} - \mathbf{f}^*\|^2\Big)$ where $\mathbf{f}^*$ refers to the ground-truth label assignment. Obviously $\mathbf{f}^*$ is unknown, otherwise, trivial optimal solution can be easily derived. Observed that the weights of individual regressors are known, alternatively one optimizes the following functional instead:

$$\max_{\mathbf{f} \in \mathbb{R}^u} \sum_{i=1}^{b} \alpha_i \left(\|\mathbf{f}_0 - \mathbf{f}_i\|^2 - \|\mathbf{f} - \mathbf{f}_i\|^2\right) \tag{1}$$

Eq.(1) maximizes a combined performance gain as shown.

In reality, however, the explicit weights of individual regressors (i.e., $\boldsymbol{\alpha}$) is difficult to know and to make the proposal more practical, we assume that $\boldsymbol{\alpha}$ is from a candidate set. For the sake of simplicity, one assumes that $\boldsymbol{\alpha}$ is from a convex linear set $\mathcal{M} = \{\boldsymbol{\alpha} | \mathbf{A}^\top \boldsymbol{\alpha} \leq \mathbf{b}; \boldsymbol{\alpha} \geq \mathbf{0}\}$, which is a general form that reflects the relation of individual learners in ensemble learning (Zhou 2012), where $\mathbf{A}$ and $\mathbf{b}$ are task-dependent coefficients. For example, $\mathcal{M} = \{\boldsymbol{\alpha} | \mathbf{1}^\top \boldsymbol{\alpha} = 1; \boldsymbol{\alpha} \geq \mathbf{0}\}$ by assuming that the weights of individual learners are from a simplex; furthermore, suppose individual leaner $\mathbf{f}_i$ is more reliable than $\mathbf{f}_j$ and the set of all such indexes (i, j) is denoted as $\mathcal{S}$, $\mathcal{M}$ could be set to $\{\boldsymbol{\alpha} | \alpha_j - \alpha_i \leq 0, (i, j) \in \mathcal{S}; \boldsymbol{\alpha} \geq \mathbf{0}\}$. Without further knowledge to determine the weights of individual regressors, one aim to optimize the worst-case performance gain (Li and Zhou 2015; Balsubramani and Freund 2015) as follows.

$$\max_{\mathbf{f} \in \mathbb{R}^u} \min_{\boldsymbol{\alpha} \in \mathcal{M}} \sum_{i=1}^{b} \alpha_i \left(\|\mathbf{f}_0 - \mathbf{f}_i\|^2 - \|\mathbf{f} - \mathbf{f}_i\|^2\right) \tag{2}$$

Representation to Geometric Projection

Note that Eq.(2) is concave to $\mathbf{f}$ and convex to $\boldsymbol{\alpha}$, and thus it is recognized as saddle-point convex-concave optimization (Nesterov 2013). The optimization of Eq.(2), nevertheless, meets some difficulties, since it is non-trivial to be solved efficiently because of poor convergence rate induced by typical gradient descent algorithms (Nesterov 2013).

In order to alleviate the computational overload and understand how Eq.(2) works, we in the following show that Eq.(2) can be formulated as a geometric projection issue that help address the above concerns.

Algorithm 1 The SAFER Method

```
Input: multiple SSR predictions $\{\mathbf{f}_i\}_{i=1}^b$ and certain direct supervised regression prediction $\mathbf{f}_0$
Output: the learned prediction $\bar{\mathbf{f}}$
Construct a linear kernel matrix $\mathbf{F}$ where $F_{ij} = \mathbf{f}_i^\top \mathbf{f}_j$, $\forall 1 \le i, j \le b$
Derive a vector $\mathbf{v} = [2\mathbf{f}_1^\top \mathbf{f}_0; \dots; 2\mathbf{f}_b^\top \mathbf{f}_0]$
Solve the convex quadratic optimization Eq.(5) and obtain the optimal solution $\boldsymbol{\alpha}^* = [\alpha_1^*, \dots, \alpha_b^*]$
Return $\bar{\mathbf{f}} = \sum_{i=1}^b \alpha_i^* \mathbf{f}_i$
```

Specifically, by setting the derivative of Eq.(2) w.r.t. $\mathbf{f}$ to zero, Eq.(2) has a closed-form solution w.r.t. $\mathbf{f}$ as

$$\mathbf{f} = \sum_{i=1}^{b} \alpha_i \mathbf{f}_i, \tag{3}$$

Through this property, by substituting Eq.(3) into Eq.(2), we then get the following equivalent form that only relates to $\boldsymbol{\alpha}$.

$$\min_{\boldsymbol{\alpha} \in \mathcal{M}} \| \sum_{i=1}^{b} \alpha_i \mathbf{f}_i - \mathbf{f}_0 \|^2 \tag{4}$$

It is evident that Eq.(4) turns out to be simple convex quadratic program. More specifically, by expanding the quadratic form in Eq.(4), it can be rewritten as

$$\min_{\boldsymbol{\alpha} \in \mathcal{M}} \quad \boldsymbol{\alpha}^\top \mathbf{F} \boldsymbol{\alpha} - \mathbf{v}^\top \boldsymbol{\alpha} \tag{5}$$

where $\mathbf{F} \in \mathbb{R}^{b \times b}$ is a linear kernel matrix of $\mathbf{f}_i$'s, i.e., $F_{ij} = \mathbf{f}_i^\top \mathbf{f}_j, \forall 1 \le i, j \le b$ and $\mathbf{v} = [2\mathbf{f}_1^\top \mathbf{f}_0; \dots; 2\mathbf{f}_b^\top \mathbf{f}_0]$. Since $\mathbf{F}$ is positive semi-definite, Eq.(5) is convex. It is often much more efficient to solve convex quadratic program than saddle-point convex-concave optimization (Nesterov 2013). For example, one can employ state-of-the-art optimization solvers, such as the MOSEK package[1], efficiently. After solving the optimal solution $\boldsymbol{\alpha}^*$, the optimal $\bar{\mathbf{f}} = \sum_{i=1}^b \alpha_i^* \mathbf{f}_i$ according to Eq.(3) is obtained. Algorithm 1 summarizes the pseudocode of the proposed method.

It is not hard to find that Eq.(4) is a geometric projection problem. Specifically, let $\Omega = \{\mathbf{f} | \sum_{i=1}^b \alpha_i \mathbf{f}_i, \boldsymbol{\alpha} \in \mathcal{M}\}$, Eq.(4) can be rewritten as,

$$\bar{\mathbf{f}} = \arg\min_{\mathbf{f} \in \Omega} \|\mathbf{f} - \mathbf{f}_0\|^2, \tag{6}$$

which learns a projection of $\mathbf{f}_0$ onto the convex set Ω. Figure 1 illustrates the intuition of our proposed method via the viewpoint of geometric projection. In the sequel we show that with the help of such a novel representation, one could study how the proposal works.

How the Proposal Works

Before going into the detail analysis, as can be observed from Figure 1, the distance between $\|\bar{\mathbf{f}} - \mathbf{f}^*\|$ should be smaller than $\|\mathbf{f}_0 - \mathbf{f}^*\|$ if $\mathbf{f}^* \in \Omega$. Such an observation motivates us to derive the following results.

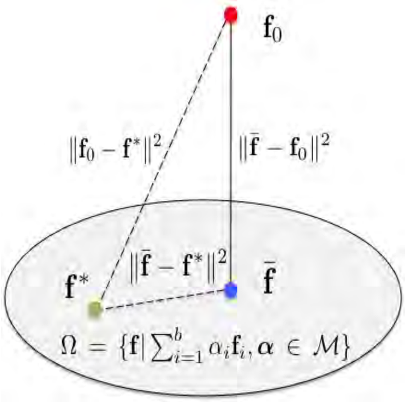
$\mathbf{f}_0$
$\|\mathbf{f}_0 - \mathbf{f}^*\|^2$
$\|\bar{\mathbf{f}} - \mathbf{f}_0\|^2$
$\mathbf{f}^*$
$\|\bar{\mathbf{f}} - \mathbf{f}^*\|^2$
$\bar{\mathbf{f}}$
$\Omega = \{\mathbf{f} | \sum_{i=1}^b \alpha_i \mathbf{f}_i, \boldsymbol{\alpha} \in \mathcal{M}\}$

Figure 1: Illustration of the intuition of our proposal via the projection viewpoint. Intuitively, the proposal learns a projection of $\mathbf{f}_0$ onto a convex feasible set Ω.

Theorem 1. *$\|\bar{\mathbf{f}} - \mathbf{f}^*\|^2 \le \|\mathbf{f}_0 - \mathbf{f}^*\|^2$, if the ground truth label assignment $\mathbf{f}^* \in \Omega = \{\mathbf{f} | \sum_{i=1}^b \alpha_i \mathbf{f}_i, \boldsymbol{\alpha} \in \mathcal{M}\}$.*

Proofs of the theorems in this paper are in supplemental material. Theorem 1 reveals that the proposal is provably safe, when ground-truth label assignment is realized by a convex linear combination of base regressors. Such a safeness condition improves the one proposed in (Li and Zhou 2015), which requires that ground-truth $\mathbf{f}^*$ is form one of base learners. Another advantage is that unlike (Li and Zhou 2015) which is theoretically hard to achieve optimality because of non-convexity, our proposed solution is naturally optimal as the formulation is convex.

In addition to safeness, it is also important to study the ability in performance improvement. The following Theorem indicates that our proposal has already achieved the maximal performance gain, with the same condition.

Theorem 2. *$\bar{\mathbf{f}}$ has already achieved the maximal worst-case performance gain against $\mathbf{f}_0$, if the ground truth $\mathbf{f}^* \in \Omega$. Specifically, $\bar{\mathbf{f}}$ is the optimal solution of the following functional,*

$$\bar{\mathbf{f}} = \underset{\mathbf{f} \in \mathbb{R}^u}{\operatorname{argmax}} \min_{\mathbf{f}^* \in \Omega} \left(\|\mathbf{f}_0 - \mathbf{f}^*\|^2 - \|\mathbf{f} - \mathbf{f}^*\|^2 \right)$$

With Theorems 1-2, it is now clear that the proposed method is provably safe and achieves the maximal worst-case performance gain, if the ground-truth label assignment is realized by a convex linear combination of base regressors. To understand our proposal more comprehensively, in the following we investigate how the performance of our proposal is affected when the condition previously mentioned is violated. Specifically, let $\boldsymbol{\lambda}^* = [\lambda_1^*, \dots, \lambda_b^*] \in \mathcal{M}$ be the optimal solution of the following functional,

$$\boldsymbol{\lambda}^* = \underset{\boldsymbol{\lambda} \in \mathcal{M}}{\operatorname{argmin}} \| \sum_{i=1}^{b} \lambda_i \mathbf{f}_i - \mathbf{f}^* \|^2$$

and $\boldsymbol{\epsilon}$ be the residual, i.e., $\boldsymbol{\epsilon} = \mathbf{f}^* - \sum_{i=1}^b \lambda_i^* \mathbf{f}_i$, reflecting the degree of violation. Suppose $\mathbf{f}_i$'s are normalized into $[0, 1]$, we then have the following result for the proposed method.

[1] https://www.mosek.com/resources/downloads

Theorem 3. *The increased loss of the proposed method against* $\mathbf{f}_0$*, i.e.,* $\Big(\|\bar{\mathbf{f}} - \mathbf{f}^*\|^2 - \|\mathbf{f}_0 - \mathbf{f}^*\|^2\Big)$*, is at most* $\min\{2\|\boldsymbol{\epsilon}\|_1/u, 2\|\boldsymbol{\epsilon}\|_2/\sqrt{u}\}$.

Theorem 3 discloses that when the required safeness condition is violated, the worst-case increased loss of our proposed method is only related to the norm of the residual (in other words, the quality of regressors), and has nothing to do with other factors, e.g., the quantity of regressors.

Experiments

In order to validate the effectiveness of the proposed method, extensive experiments are conducted on a broad range of data sets[2] (Table 1) that cover diverse domains including physical measurements (*abalone*), health (*bodyfat*), economics (*cadata*), activity recognition (*mpg*), etc. The sample size ranges from around 100 (*pyrim*) to more than 20,000 (*cadta*). All the features and labels are normalized into $[0, 1]$.

Experimental Setup

SAFER[3] is compared with the following methods.

- **1NN**: Direct supervised nearest neighbor algorithm with only labeled data.
- **COREG** (Zhou and Li 2005): Representative semi-supervised regression method. Two kNN regressors based on two different distance metrics are employed.
- **Self-kNN**: Semi-supervised extension of the supervised kNN method based on self-training (Yarowsky 1995). It first trains a supervised kNN method based on only labeled instances, and then predict the label of unlabeled instances. After that, by adding the predicted labels on the unlabeled data as “ground-truth”, another supervised kNN method is trained. This process is repeated until predictions on the unlabeled data no longer change or a maximum number of iterations achieves.
- **Self-LS**: Semi-supervised extension of the supervised least square method (Hastie, Tibshirani, and Friedman 2001) based on self-training. The algorithm is similar to Self-kNN except that supervised method is adapted to least square regression.
- **Voting**: We also compare with the voting method, which uniformly weights multiple regressors. This approach is found promising in practice (Zhou 2012).

The co-regularization method (Brefeld et al. 2006) is not compared because it requires multiple views of data, which is not the case in our experimental data sets. For the baseline 1NN method, the Euclidean distance is used to locate the nearest neighbor. For the Self-kNN method, the Euclidean distance is used and k is set to 3. The maximum number of iterations is set to 5 and further increasing it does not improve performance. For the Self-LS method, the parameters related to the importance for the labeled and unlabeled instances are set to 1 and 0.1, respectively. For the COREG method, the parameters are set to the recommended one in the package and the two distance metrics are employed by the Euclidean and Mahalanobis distances. For the Voting and the proposed SAFER method, 3 semi-supervised regressors are used where one is from the Self-LS method and the other two are from the Self-kNN methods employing the Euclidean and the Cosine distance, respectively. Without sufficient domain knowledge, $\mathcal{M} = \{\boldsymbol{\alpha}|\mathbf{1}^\top\boldsymbol{\alpha} = 1; \boldsymbol{\alpha} \geq \mathbf{0}\}$.

For each data set, 5 and 10 labeled instances are randomly selected and the rest ones are unlabeled data. Experiment for each dataset is repeated for 30 times, and the average performance (mean±std) on the unlabeled data is reported.

Comparison Results

Table 1 shows the Mean Square Error of compared methods and the proposal on 5 and 10 labeled instances, respectively. We have the following observations from Table 1.

- Self-kNN generally improves the performance, however it causes serious performance degradation in 2 cases.
- Self-LS is not effective. One possible reason is the performance of supervised LS is not as good as that of kNN in our experimental data sets.
- COREG achieves good performance, whereas it also will significantly decrease the performance in some cases.
- The Voting method improves both the average performance of Self-kNN and Self-LS, but in 6 cases it significantly decreases the performance.
- The proposed method achieves significant improvement in 6 and 8 cases, which are the most among all the compared methods on 5 and 10 labeled instances, respectively. It also obtains the best average performance. What is more importantly, it does not seriously reduce the performance.

Overall the proposed method effectively improves safeness of SSR, in addition obtains highly comparable performance with state-of-the-art approaches.

Study Robustness to Safeness Condition

Figure 2 studies the robustness of our proposal w.r.t. safeness condition presented in Theorem 1. Figure 2 shows a comparison between the reduced Mean Square Error of the proposal against 1NN (the higher the better) and the lower bound of increased loss (i.e., $-\min\{2\|\boldsymbol{\epsilon}\|_1/u, 2\|\boldsymbol{\epsilon}\|_2/\sqrt{u}\}$) shown in Theorem 3 based on 30 random splits of data using 10 labeled instances. Each subfigure corresponds to one data set. It appears that even that the optimal convex combination of individual regressors is usually not equal to the ground truth, the proposal still works well. This reflects that our proposal works quite robust to safeness condition.

Study Robustness to Performance Measures

It is interesting to study whether the proposal SAFER is effective in other regression performance measures. Table 2 studies the experimental comparison on Mean Abso-

[2] https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/

[3] http://lamda.nju.edu.cn/code_SAFER.ashx

Table 1: Mean Square Error (mean±std) for the compared methods and SAFER using 5 and 10 labeled instances. For all methods, if the performance is significantly better/worse than the baseline 1NN method, the corresponding entries are bolded/boxed (paired t-tests at 95% significance level). The average mean square error on all the experimental data sets is listed for comparison. The win/tie/loss counts are summarized and the method with the smallest number of losses against 1NN is bolded.

5 labeled instances

Dataset	1NN	Self-kNN	Self-LS	COREG	Voting	SAFER
abalone	.017 ± .007	**.014 ± .003**	**.013 ± .004**	**.013 ± .003**	**.012 ± .003**	**.013 ± .003**
bodyfat	.024 ± .008	.025 ± .009	$\boxed{.054 \pm .016}$	.026 ± .008	$\boxed{.031 \pm .011}$	.025 ± .009
cadata	.090 ± .031	**.073 ± .023**	**.067 ± .022**	**.069 ± .028**	**.069 ± .022**	**.070 ± .023**
cpusmall	.027 ± .012	$\boxed{.031 \pm .008}$	$\boxed{.050 \pm .021}$	$\boxed{.031 \pm .009}$	.024 ± .006	.028 ± .009
eunite2001	.052 ± .017	**.037 ± .015**	**.024 ± .012**	**.037 ± .011**	**.031 ± .013**	**.032 ± .010**
housing	.042 ± .007	.043 ± .009	$\boxed{.048 \pm .012}$	.041 ± .008	.042 ± .009	.041 ± .009
mg	.071 ± .035	**.057 ± .015**	**.053 ± .011**	**.054 ± .019**	**.054 ± .013**	**.053 ± .013**
mpg	.029 ± .012	.030 ± .012	$\boxed{.040 \pm .014}$	.031 ± .012	.031 ± .012	.030 ± .012
pyrim	.032 ± .009	**.027 ± .005**	$\boxed{.063 \pm .012}$	.029 ± .011	**.025 ± .007**	**.025 ± .005**
space_ga	.005 ± .002	.005 ± .003	$\boxed{.030 \pm .005}$	**.004 ± .002**	$\boxed{.008 \pm .002}$	**.004 ± .002**
Ave. Mse.	.039	.034	.044	.033	.033	.032
win/tie/loss against 1NN		5/4/1	4/0/6	5/4/1	5/3/2	**6/4/0**

10 labeled instances

Dataset	1NN	Self-kNN	Self-LS	COREG	Voting	SAFER
abalone	.020 ± .010	**.014 ± .005**	**.013 ± .004**	**.012 ± .003**	**.012 ± .003**	**.013 ± .005**
bodyfat	.019 ± .005	.019 ± .007	$\boxed{.041 \pm .013}$	.020 ± .006	$\boxed{.023 \pm .009}$	.018 ± .007
cadata	.083 ± .029	**.063 ± .012**	**.056 ± .007**	**.054 ± .010**	**.057 ± .009**	**.060 ± .013**
cpusmall	.024 ± .012	$\boxed{.027 \pm .008}$	$\boxed{.042 \pm .004}$	$\boxed{.028 \pm .008}$	**.020 ± .005**	.025 ± .008
eunite2001	.044 ± .014	**.037 ± .013**	**.020 ± .006**	**.031 ± .009**	**.029 ± .009**	**.029 ± .007**
housing	.039 ± .010	.036 ± .009	.036 ± .009	**.035 ± .005**	**.034 ± .008**	**.035 ± .009**
mg	.062 ± .019	**.046 ± .015**	**.048 ± .011**	**.045 ± .015**	**.043 ± .014**	**.045 ± .014**
mpg	.022 ± .007	.020 ± .006	$\boxed{.030 \pm .014}$	.021 ± .007	.021 ± .008	.020 ± .006
pyrim	.023 ± .006	**.021 ± .005**	$\boxed{.052 \pm .014}$	.022 ± .006	**.020 ± .007**	**.020 ± .006**
space_ga	.004 ± .001	**.003 ± .001**	$\boxed{.028 \pm .002}$	**.003 ± .001**	$\boxed{.006 \pm .001}$	**.003 ± .001**
Ave. Mse.	.034	.029	.037	.027	.026	.027
win/tie/loss against 1NN		6/3/1	4/1/5	6/3/1	7/1/2	**7/3/0**

lute Error (MAE) (Willmott and Matsuura 2005) and Mean ϵ-insensitive Error (MEE) (Smola and Schölkopf 2004). As Table 2 shows, the proposal achieves competitive performance in a variety of performance measures. This result reveals that although SAFER is designed through Mean Square Error, it has a certain degree of robustness to the change of performance measures.

Conclusion

Despite remarkable progress of SSL, however, it is plagued with the problem of performance degeneration when using unlabeled data. In this paper we present an efficient and effective projection algorithm to learn a *safe* prediction from multiple semi-supervised regressors. We show that the proposal is provably safe and achieves the worst-case performance gain, when the ground-truth is realized as a convex linear combination of individual regressors. Extensive experiments validate encouraging performance. In our ongoing work, as stated in Theorem 3, generating high quality regressors is crucial. Accuracy estimation from unlabeled data (Platanios, Blum, and Mitchell 2014) might be a possible solution.

References

Balcan, M. F., and Blum, A. 2010. A discriminative model for semi-supervised learning. *Journal of the ACM* 57(3).

Balsubramani, A., and Freund, Y. 2015. Optimally Com-

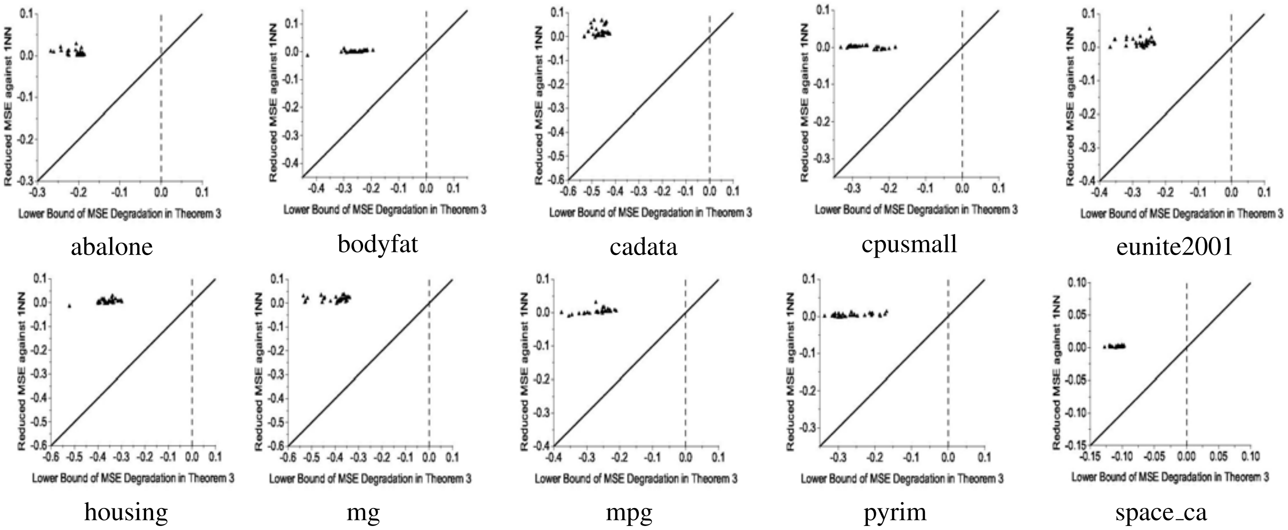

Figure 2: Reduced Mean Square Error (MSE) of the proposal against 1NN (the higher the better) v.s. the lower bound of increased loss (i.e., $-\min\{2\|\epsilon\|_1/u, 2\|\epsilon\|_2/\sqrt{u}\}$) shown in Theorem 3 based on 30 random splits of data using 10 labeled instances, where each subfigure corresponds to one data set used in the experiment.

Table 2: Mean Absolute Error (mean±std) and Mean ϵ-insensitive Error (mean±std, $\epsilon = 0.05$) for the compared methods and SAFER using 10 labeled instances.

Mean Absolute Error						
Dataset	1NN	Self-kNN	Self-LS	COREG	Voting	SAFER
abalone	.100 ± .025	**.089 ± .020**	**.086 ± .018**	**.083 ± .015**	**.081 ± .018**	**.086 ± .019**
bodyfat	.108 ± .013	.107 ± .018	$\boxed{.164 \pm .026}$	$\boxed{.114 \pm .015}$	$\boxed{.119 \pm .023}$	.105 ± .018
cadata	.216 ± .037	**.195 ± .022**	**.189 ± .016**	**.182 ± .023**	**.189 ± .019**	**.192 ± .023**
cpusmall	.073 ± .014	$\boxed{.078 \pm .007}$	$\boxed{.168 \pm .010}$	$\boxed{.081 \pm .008}$	$\boxed{.092 \pm .008}$	.076 ± .007
eunite2001	.162 ± .023	**.152 ± .027**	**.108 ± .016**	**.138 ± .018**	**.132 ± .021**	**.133 ± .017**
housing	.137 ± .018	.135 ± .023	.140 ± .023	.135 ± .016	.131 ± .022	.132 ± .022
mg	.188 ± .029	**.166 ± .025**	**.176 ± .017**	**.168 ± .026**	**.163 ± .023**	**.164 ± .025**
mpg	.110 ± .014	.107 ± .018	$\boxed{.138 \pm .029}$	.112 ± .020	.109 ± .022	.105 ± .018
pyrim	.105 ± .014	.107 ± .011	$\boxed{.174 \pm .021}$	$\boxed{.111 \pm .012}$	**.095 ± .016**	**.099 ± .014**
space_ga	.050 ± .005	**.043 ± .005**	$\boxed{.131 \pm .004}$	**.041 ± .005**	$\boxed{.060 \pm .004}$	**.042 ± .004**
Ave. Mae.	.125	.118	.147	.116	.117	.114
win/tie/loss against 1NN		5/4/1	4/1/5	5/2/3	5/2/3	**6/4/0**

Mean ϵ-insensitive Error						
Dataset	1NN	Self-kNN	Self-LS	COREG	Voting	SAFER
abalone	.062 ± .023	**.049 ± .017**	**.046 ± .015**	**.044 ± .012**	**.042 ± .014**	**.046 ± .016**
bodyfat	.065 ± .013	.065 ± .017	$\boxed{.118 \pm .025}$	$\boxed{.070 \pm .014}$	$\boxed{.076 \pm .021}$	.063 ± .017
cadata	.170 ± .037	**.149 ± .021**	**.143 ± .015**	**.136 ± .022**	**.143 ± .018**	**.146 ± .023**
cpusmall	.039 ± .013	$\boxed{.043 \pm .007}$	$\boxed{.122 \pm .009}$	$\boxed{.046 \pm .007}$	$\boxed{.053 \pm .007}$	.041 ± .007
eunite2001	.117 ± .023	**.108 ± .026**	**.066 ± .015**	**.094 ± .017**	**.089 ± .020**	**.089 ± .016**
housing	.095 ± .017	.092 ± .022	.096 ± .021	.092 ± .014	.088 ± .020	.089 ± .021
mg	.143 ± .029	**.121 ± .024**	**.130 ± .017**	**.123 ± .025**	**.118 ± .023**	**.119 ± .024**
mpg	.069 ± .013	.066 ± .016	$\boxed{.094 \pm .028}$	.069 ± .019	.067 ± .020	.064 ± .016
pyrim	.066 ± .012	.065 ± .010	$\boxed{.129 \pm .020}$	.068 ± .012	**.055 ± .015**	**.059 ± .013**
space_ga	.016 ± .004	**.011 ± .003**	$\boxed{.087 \pm .004}$	**.010 ± .003**	$\boxed{.023 \pm .003}$	**.010 ± .002**
Ave. Mee.	.084	.077	.103	.075	.075	.073
win/tie/loss against 1NN		5/4/1	4/1/5	5/3/2	5/2/3	**6/4/0**

bining Classifiers Using Unlabeled Data. In *Proceedings of International Conference on Learning Theory*, 211–225.

Bennett, K., and Demiriz, A. 1999. Semi-supervised support vector machines. In Kearns, M. J.; Solla, S. A.; and Cohn, D. A., eds., *Advances in Neural Information Processing Systems 11*. Cambridge, MA: MIT Press. 368–374.

Blum, A., and Chawla, S. 2001. Learning from labeled and unlabeled data using graph mincuts. In *Proceedings of the 8th International Conference on Machine Learning*, 19–26.

Blum, A., and Mitchell, T. 1998. Combining labeled and unlabeled data with co-training. In *Proceedings of the 7th Annual Conference on Computational Learning Theory*.

Brefeld, U.; Gärtner, T.; Scheffer, T.; and Wrobel, S. 2006. Efficient co-regularised least squares regression. In *Proceedings of the 23rd International Conference on Machine learning*, 137–144.

Chapelle, O.; Schölkopf, B.; and Zien, A., eds. 2006. *Semi-Supervised Learning*. MIT Press.

Chawla, N. V., and Karakoulas, G. 2005. Learning from labeled and unlabeled data: An empirical study across techniques and domains. *Journal of Artificial Intelligence Research* 23:331–366.

Cozman, F. G.; Cohen, I.; and Cirelo, M. 2002. Unlabeled data can degrade classification performance of generative classifiers. In *Proceedings of the 15th International Florida Artificial Intelligence Research Society Conference*, 327–331.

Hastie, T.; Tibshirani, R.; and Friedman, J. 2001. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer.

Joachims, T. 1999. Transductive inference for text classification using support vector machines. In *Proceedings of the 16th International Conference on Machine Learning*, 200–209.

Krijthe, J. H., and Loog, M. 2015. Implicitly constrained semi-supervised least squares classification. In *Advances in 14th International Symposium on Intelligent Data Analysis*, 158–169.

Li, M., and Zhou, Z.-H. 2005. SETRED: Self-training with editing. In *Proceedings of the 9th Pacific-Asia Conference on Knowledge Discovery and Data Mining*, 611–621.

Li, Y.-F., and Zhou, Z.-H. 2011. Towards making unlabeled data never hurt. In *Proceedings of the 28th International Conference on Machine learning*, 1081–1088.

Li, Y.-F., and Zhou, Z.-H. 2015. Towards making unlabeled data never hurt. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 37(1):175–188.

Li, Y.-F.; Tsang, I. W.; Kwok, J. T.; and Zhou, Z.-H. 2013. Convex and scalable weakly labeled svms. *Journal of Machine Learning Research* 14:2151–2188.

Li, Y.-F.; Kwok, J. T.; and Zhou, Z.-H. 2016. Towards safe semi-supervised learning for multivariate performance measures. In *Proceedings of 30th AAAI Conference on Artificial Intelligence*, 1816–1822.

Li, Y.-F.; Wang, S.-B.; and Zhou, Z.-H. 2016. Graph quality judgement: A large margin expedition. In *Proceedings of the 25th International Joint Confernece on Artificial Intelligence*, 1725–1731.

Miller, D. J., and Uyar, H. S. 1997. A mixture of experts classifier with learning based on both labelled and unlabelled data. In Mozer, M.; Jordan, M. I.; and Petsche, T., eds., *Advances in Neural Information Processing Systems 9*. Cambridge, MA: MIT Press. 571–577.

Nesterov, Y. 2013. *Introductory lectures on convex optimization: A basic course*, volume 87. Springer Science & Business Media.

Nigam, K.; McCallum, A. K.; Thrun, S.; and Mitchell, T. 2000. Text classification from labeled and unlabeled documents using EM. *Machine Learning* 39(2):103–134.

Platanios, E. A.; Blum, A.; and Mitchell, T. M. 2014. Estimating accuracy from unlabeled data. In *Proceedings of the 30th Conference on Uncertainty in Artificial Intelligence*, 682–691.

Smola, A. J., and Schölkopf, B. 2004. A tutorial on support vector regression. *Statistics and computing* 14(3):199–222.

Willmott, C. J., and Matsuura, K. 2005. Advantages of the mean absolute error (mae) over the root mean square error (rmse) in assessing average model performance. *Climate research* 30(1):79–82.

Yang, T., and Priebe, C. 2011. The effect of model misspecification on semi-supervised classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 33(10):2093–2103.

Yarowsky, D. 1995. Unsupervised word sense disambiguation rivaling supervised methods. In *Proceedings of the 33rd Annual Meeting on Association for Computational Linguistics*, 189–196.

Zhou, Z.-H., and Li, M. 2005. Semi-supervised regression with co-training. In *Proceedings of the 19th International Joint Conference on Artificial Intelligence*, volume 5, 908–913.

Zhou, Z.-H., and Li, M. 2010. Semi-supervised learning by disagreement. *Knowledge and Information Systems* 24(3):415–439.

Zhou, D.; Bousquet, O.; Lal, T. N.; Weston, J.; and Schölkopf, B. 2004. Learning with local and global consistency. In Thrun, S.; Saul, L.; and Schölkopf, B., eds., *Advances in Neural Information Processing Systems 16*. Cambridge, MA: MIT Press. 595–602.

Zhou, Z.-H. 2012. *Ensemble Methods: Foundations and Algorithms*. Boca Raton: FL: Chapman & Hall.

Zhu, X., and Goldberg, A. B. 2009. Introduction to semi-supervised learning. *Synthesis lectures on artificial intelligence and machine learning* 3(1):1–130.

Zhu, X.; Ghahramani, Z.; and Lafferty, J. 2003. Semi-supervised learning using Gaussian fields and harmonic functions. In *Proceedings of the 20th International Conference on Machine Learning*, 912–919.