DOI: 10.1111/mice.12617

RESEARCH ARTICLE

COMPUTER-AIDED CIVIL AND INFRASTRUCTURE ENGINEERING WILEY

Reducing the effect of sample bias for small data sets with double-weighted support vector transfer regression

Huan Luo | **Stephanie German Paal**

Zachry Department of Civil & Environmental Engineering, Texas A&M University, College Station, TX, USA

Correspondence
Stephanie German Paal, Zachry Department of Civil & Environmental Engineering, Texas A&M University, College Station, TX 77843, USA.
Email: spaal@civil.tamu.edu

Abstract

Small data sets are an extremely challenging problem in the machine learning (ML) realm, and in specific, in regression scenarios, as the lack of relevant data can lead to ML models that have large bias. However, there are many applications for which a purely data-driven procedure would be advantageous, but a large amount of data are not available. This article proposes a novel regression-based transfer learning (TL) model to address this challenge, where TL is defined as knowledge transfer from a large, relevant data set (source domain data) to a small data set (target domain data). The proposed TL model is termed double-weighted support vector transfer regression (DW-SVTR), which couples least squares support vector machines for regression (LS-SVMR) with two weight functions. The first weight function uses kernel mean matching (KMM) to reweight the source domain data such that the mean values of the source and target domain data in a reproduced kernel Hilbert space (RKHS) are close. In this way, the source domain data points relevant to the target domain points have a larger weight than irrelevant source domain points. The second weight is a function of estimated residuals, which aims to further reduce the negative interference of irrelevant source domain points. The proposed approach is assessed and validated via simulated data and by enhanced shear strength prediction of nonductile columns based on limited availability of nonductile column data. Specifically, the results for the latter show that the proposed DW-SVTR can reduce the root mean square error (RMSE) by 34% and enhance the coefficient of determination (R^2) by 229%. These numerical results demonstrate that the DW-SVTR significantly reduces the effect of small sample bias and improves prediction performance compared to standard ML methods.

1 | INTRODUCTION

Despite the fact that machine learning (ML) techniques have reformed the world of numerical modeling and achieved great success in many other engineering and science disciplines (Adeli, 2001; Cha, Choi, & Büyüköztürk, 2017; Cha, Choi, Suh, Mahmoudkhani, & Büyüköztürk, 2018; Reich, 1997), certain challenges remain unsolved. One challenge, which significantly affects their performance, is how to reduce the negative effect induced by sample bias of small data sets, specifically in regression scenarios. This is because regression-based ML techniques usually require a large, high-quality training data set to adaptively fit the data and form an accurate, robust model

for prediction (Ahangar-Asr, Faramarzi, Javadi, & Giustolisi, 2011; Aminian, Javid, Asghari, Gandomi, & Esmaeili, 2011; Cheng & Cao, 2014; Chou & Pham, 2015; Gandomi, Mohammadzadeh, Pérez-Ordóñez, & Alavi, 2014; Jeon, Shafieezadeh, & DesRoches, 2014; Luo & Paal, 2018, 2019; Pal & Deswal, 2011; Rafiei & Adeli, 2016, 2018; Rafiei, Khushefati, Demirboga, & Adeli, 2017; Yuen, Ortiz, & Huang, 2016). Typically, the sample points in a training data set can reasonably represent the distribution of a target domain. In this case, the sample bias induced by the training data set is negligible (Quionero-Candela, Sugiyama, Schwaighofer, & Lawrence, 2009). However, once the size of the training data set is not sufficient (note that we call this a small data set in this article), the effect of small sample bias is no longer negligible.

This is because, when a data set is small, it may lead to a biased sample. This means that the sample points in the small data set cannot accurately represent the distribution of a target domain and cannot reflect the underlying patterns in the target domain data (Quionero-Candela et al., 2009), leading to large bias in the final, fully trained ML model for prediction in the target domain. Transfer learning (TL) aims to address the problems with sample bias induced by small data sets by transferring ML models trained with a relevant large data set to improve prediction (Pan & Yang, 2009; Weiss, Khoshgoftaar, & Wang, 2016). In this article, the following terminology is employed: the "small data set" is from the "target domain" and the "large data set" is from the "source domain." In many TL approaches it is typically assumed that the target and source domains are somewhat different from one another, but still related to a certain extent (Pan & Yang, 2009). Thus, the ML models, fully trained based on the source domain data, can be applied to prediction in the target domain. This seems to deviate the default assumption in many standard ML settings, where the training and test data sets are independently and identically distributed (i.i.d), as the data set is shifted (Cortes & Mohri, 2014; Gretton et al., 2009; Huang, Gretton, Borgwardt, Schölkopf, & Smola, 2007; Quionero-Candela et al., 2009). Mathematically speaking, data set shift happens when two data sets are drawn from two different distributions (Quionero-Candela et al., 2009). Specifically, given the distributions of the source and target data, one can sample the training data set $\{(\boldsymbol{x}_j^S, y_j^S)\}_{j=1}^n$ from the source data distribution $p^S(\boldsymbol{x}, y)$ and the test data set $\{(\boldsymbol{x}_k^T, y_k^T)\}_{k=1}^m$ from the target data distribution $p^T(\boldsymbol{x}, y)$, where $\boldsymbol{x} \in R^p$ and $y \in R$. A data set shift is present when $p^S(\boldsymbol{x}, y) \neq p^T(\boldsymbol{x}, y)$.

Currently, the majority of TL approaches have been developed for classification problems (Gao & Mosalam, 2018; Pan & Yang, 2009), but less attention has been paid on regression problems (Pardoe & Stone, 2010; Salaken, Khosravi, Nguyen, & Nahavandi, 2019). The main difference between classification and regression problems is that the response variable for classification problems is discrete, whereas that for regression problems, it is continuous (James, Witten, Hastie, & Tibshirani, 2013). This difference strictly restricts the direct use of some existing TL approaches for addressing regression problems (i.e., some TL methods for classification must be modified for their use in regression settings, e.g., the work in Pardoe & Stone, 2010). Besides, existing regression-based TL methods generally assume that the target and source domains are related to each other (Garcke & Vanck, 2014; Karbalayghareh, Qian, & Dougherty, 2018; Pardoe & Stone, 2010). Therefore, these TL methods may work well for regression problems when the source and target domain data are related but will most likely work poorly when they are unrelated. The relevance is represented by the joint distributions of two domains (Garcke & Vanck, 2014; Huang et al., 2007). According to Bayes rule, the joint distribution can be written as $p(\boldsymbol{x}, y) = p(\boldsymbol{x}|y)p(y) = p(y|\boldsymbol{x})p(\boldsymbol{x})$. The equation $p(\boldsymbol{x}, y) = p(\boldsymbol{x}|y)p(y)$ is called the generative model, whereas $p(\boldsymbol{x}, y) = p(y|\boldsymbol{x})p(\boldsymbol{x})$ is called the discriminative model (Garcke & Vanck, 2014; Quionero-Candela et al., 2009). The majority of existing TL approaches focus on the discriminative approach. Thus, $p^S(y|\boldsymbol{x})p^S(\boldsymbol{x}) \neq p^T(y|\boldsymbol{x})p^T(\boldsymbol{x})$ (i.e., the source and target domain distributions are different) is achieved via different marginal distributions, that is, $p^S(\boldsymbol{x}) \neq p^T(\boldsymbol{x})$ (also called covariate shift) (Quionero-Candela et al., 2009), different posterior distributions, that is, $p^S(y|\boldsymbol{x}) \neq p^T(y|\boldsymbol{x})$, or both.

The case where the posterior distributions of source and target domains are different (i.e., $p^S(y|\boldsymbol{x}) \neq p^T(y|\boldsymbol{x})$) is very challenging, because the two terms could be arbitrarily far apart. It is even more difficult when both the marginal and posterior distributions of two domains are different. This is because both the marginal and the posterior distributions of source and target domains could be arbitrarily far away, which makes source and target data completely unrelated. Almost all of the existing regression-based TL approaches fail to solve this case, as this case is analogous to, for example, the relation that a well-trained ML model with a sufficiently large data set in the economic field is applied to the prediction on a problem in an engineering discipline. Therefore, at first glance, there is no way, for instance, to utilize a well-trained ML model for housing price prediction to predict the structural strength for engineering structures (i.e., two unrelated domains with different tasks and/or different nonlinear relations). In this article, a novel regression-based TL model, termed double-weighted support vector transfer regression (DW-SVTR), is proposed to reduce the effect of sample bias of small data sets and sufficiently exploit the useful

information provided by small data sets in civil engineering (CE). Further, the proposed DW-SVTR is also attempted to solve the most challenging case, where both the marginal and the posterior distributions of source and target data are different. The final numerical results in this article demonstrate that the proposed approach is even effective under these circumstances. The rest of this article is organized as follows: Section 2 presents the literature review to introduce existing work on TL, whereas the methodology of the proposed DW-SVTR is introduced in Section 3. Section 4 details the implementation procedure of the proposed DW-SVTR. The illustrative examples to validate the proposed approach are given in Section 5. Finally, conclusions are made in Section 6.

2 | LITERATURE REVIEW

There are many TL approaches that have been proposed to deal with the problems associated with small data sets. These approaches can be expressed as instance based or feature based. Instance-based transfer approaches (such that a portion of sample points from the source domain data can be used in the target domain) have been created, which use boosting (Dai, Yang, Xue, & Yu, 2007; Pardoe & Stone, 2010), multiple input sources (Tan, Zhong, Xiang, & Yang, 2014), and reweighting approaches based on covariate shift setting (Cortes & Mohri, 2014; Gretton et al., 2009; Huang et al., 2007; Sugiyama, Nakajima, Kashima, Buenau, & Kawanabe, 2008). In feature-based transfer approaches, the source and target data are mapped into a space where the shared information from both data can be applied to the target domain (Argyriou, Evgeniou, & Pontil, 2007, 2008). However, as mentioned previously, the majority of these approaches have been used to deal with classification problems, and only a few recent research efforts have focused on regression problems.

Pardoe and Stone (2010) modified two existing boosting-based classification TL models, ExpBoost (Rettinger, Zinkevich, & Bowling, 2006) and TrAdaBoost (Dai et al., 2007), to form two TL models called ExpBoost.R2 and Two-stage TrAdaBoost.R2 for regression problems. Both of these TL models are based on AdaBoost.R2 (Drucker, 1997), where the reweighting of instances (i.e., data points) that have larger residuals predicted by a learner (i.e., ML model) are achieved by normalizing errors into adjusted errors within the range [0, 1] in each boosting iteration. The proposed boosting-based transfer regression models are validated effectively by numerical experiments. Garcke and Vanck (2014) proposed two approaches for inductive transfer regression based on importance weighting. These two methods are to estimate a weight that is a density ratio between the target and source data. The first one relies on the prediction performance of an ML model learned from the data in the source domain, whereas the second one minimizes the Kullback–Leibler divergence (Sugiyama et al., 2008) between two distributions of the target and source data. Numerical experiments are performed and results indicate that the former is better than the latter. A seed-based TL model for regression problems is proposed by Salaken et al. (2019). In this approach, each sample point in the target domain is regarded as a seed for initiating the transfer of the source data. An auto-encoder deep learning technique is used to transform the source data into an abstracted feature space, where the number of features for the data in the source domain matches that in the target domain. Then, a k-means clustering algorithm, with the number of clusters equal to the number of sample points in the target domain, is applied to cluster the source domain data, and each target domain sample point is appended with a relevant cluster by minimizing the Euclidean distance. The effectiveness of this method is verified by numerical results.

Although these mentioned regression-based TL approaches can reduce the effect of small sample bias and thus improve prediction performance for small data sets, such capabilities may be limited to the transfer between two related domain data, as validated in the numerical experiments. If the source and target domain data are far apart and unrelated, these methods may no longer be valid because these approaches may not be able to extract the shared information from two unrelated domains. To alleviate this limitation, we propose a novel TL approach for regression problems.

3 | DOUBLE-WEIGHTED SUPPORT VECTOR TRANSFER REGRESSION

This section presents a novel regression-based TL approach, which is a new variant of least squares support vector machines for regression (LS-SVMR), by coupling LS-SVMR with two weight functions. Thus, the proposed approach is called DW-SVTR. The two weight functions have different effects. The first weight is obtained using kernel mean matching (KMM), which accords more weight to the source domain points that are relevant to target domain points than irrelevant source domain points. In this way, the first weight function can augment the small data set in the target domain with the relevant source domain points to reduce the small sample bias. The second weight is a function of residuals and thus, serves the purpose to further reduce the negative interference of irrelevant source domain points analogous to outliers for the target domain training sample. The detailed information is presented as follows.

Suppose the data set $\{(\boldsymbol{x}_j^S, y_j^S)\}_{j=1}^n$ is sampled from the source domain distribution $p^S(\boldsymbol{x}, y)$ and the data set $\{(\boldsymbol{x}_k^T, y_k^T)\}_{k=1}^m$ is sampled from the target domain distribution $p^T(\boldsymbol{x}, y)$, where $\boldsymbol{x}_j^S$ and $\boldsymbol{x}_k^T$ both $\in R^p$ have the same dimension, y_j^S and y_k^T both $\in R$ also have the same dimension, and $m \ll n$. In the proposed TL method, we do not have *a priori* assumption that the source and target data are related. Therefore, the source and target data could be unrelated (e.g., both the marginal and posterior distributions of the two domains are different). Because the source and target data could be unrelated and arbitrarily far apart, this means that the units of the predictors and response variables between these two domains may vary greatly, leading to a significant discrepancy in numeric values in the original space. In this case, there is no way to utilize the information from the source data to improve the prediction for the target domain in the original space. Thus, the first step is to eliminate the effect of different ranges of values due to the different units. For both domain data, we first transform the predictors $\boldsymbol{x}_t^o \in R^p$ and response $y_t^o \in R$ of the data set $\{(\boldsymbol{x}_t^o, y_t^o)\}_{t=1}^d$ to zero mean and unit variance by using the following formulas:

$$\boldsymbol{x}_t = \left(\boldsymbol{x}_t^o - \bar{\boldsymbol{x}}\right) ./\boldsymbol{\sigma}_{\boldsymbol{x}} \tag{1}$$

$$y_t = \frac{y_t^o - \bar{y}}{\sigma_y} \tag{2}$$

where "./" operator represents element division of two vectors, $\bar{\boldsymbol{x}} \in R^p$ is the mean of the predictors, $\boldsymbol{\sigma}_{\boldsymbol{x}} \in R^p$ is the standard deviation of the predictors, $\bar{y} \in R$ is the mean of the response variable, $\sigma_y \in R$ is the standard deviation of the response variable.

After successfully transforming the data, the transformed data in both domains will be within the space with zero mean and unit variance. Denote $\boldsymbol{z}_j^S = (\boldsymbol{x}_j^S, y_j^S)$ as a point from the transformed data set in the source domain and $\boldsymbol{z}_k^T = (\boldsymbol{x}_k^T, y_k^T)$ as a point from the transformed data set in the target domain. Because the data set in the target domain is small and not sufficient in size, it cannot be directly employed to train a good ML model due to the potential small sample bias. Thus, we need to reweight the source domain data such that partial points with appropriate weights in the source domain can be utilized by small data set in the target domain to reduce its sample bias. Denote $\boldsymbol{z}_i = (\boldsymbol{x}_i, y_i)$ as a point from the augmented data set $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^{m+n}$ that is formed by combining the transformed source domain data set $\{(\boldsymbol{x}_j^S, y_j^S)\}_{j=1}^n$ and the transformed target domain data set $\{(\boldsymbol{x}_k^T, y_k^T)\}_{k=1}^m$. Given the augmented data set, the learning objective of the proposed DW-SVTR is to find optimal model parameters $\boldsymbol{w} = (w_1, w_2, \ldots, w_h)^T \in R^h$ and $b \in R$ that minimize the following objective function:

$$J(\boldsymbol{w}, e_i) = \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + \frac{1}{2}\gamma \sum_{i=1}^{m+n} \beta(\boldsymbol{z}_i)\, \upsilon(\boldsymbol{x}_i)\, e_i^2 \tag{3}$$

$$\text{Subject to}: \; y_i = w^T \varphi(x_i) + b + e_i, \quad i = 1, \ldots, (m+n) \tag{4}$$

where $e_i \in R, i = 1, \ldots, m+n$ is the error term; $\gamma \in R$ is a regularization parameter; $\beta(\boldsymbol{z}_i)$, $\upsilon(\boldsymbol{x}_i) \in R, i = 1, \ldots, m+n$ are weights that can take any value in the range $[\varepsilon, 1]$, $\beta(\boldsymbol{z}_i)$ is a weight to determine the importance of each data point in the augmented data set and $\upsilon(\boldsymbol{x}_i)$ is a weight, which is a function of residual where data points having large residuals have smaller weights and those having small residuals have larger weights; the determination of these two types of weight functions will be introduced in detail; $\varepsilon \in$ R is a real number approaching 0; $\varphi(\boldsymbol{x}_i)$ is a feature vector, and $\varphi(\cdot) : R^p \to R^h$ is a mapping function from p dimensions to a higher h-dimensional feature space.

If $\beta(\boldsymbol{z}_i)$ takes a value approaching ε, it means the point $\boldsymbol{z}_i$ is irrelevant to the data points in the target domain and plays a lesser role in prediction for the target domain; otherwise, if $\beta(\boldsymbol{z}_i)$ takes a value approaching one, it means the point $\boldsymbol{z}_i$ is highly relevant to the target domain and plays an important role in prediction for the target domain.

The Lagrangian function is established to solve Equation (3) and Equation (4):

$$L(\boldsymbol{w}, b, e_i; \alpha_i) = J(\boldsymbol{w}, e_i) - \sum_{i=1}^{m+n} \alpha_i \left((\boldsymbol{w})^T \varphi(\boldsymbol{x}_i) + b + e_i - y_i\right) \tag{5}$$

where $\alpha_i \in R, i = 1, \ldots, m+n$ is a Lagrange multiplier (also called support values).

The Karush–Kuhn–Tucker (KKT) conditions for optimality are used by differentiating the variables in Equation (5) above, which results in the following:

$$\begin{cases} \frac{\partial L}{\partial \boldsymbol{w}} = 0 \to \boldsymbol{w} = \sum_{i=1}^{m+n} \alpha_i \varphi(\boldsymbol{x}_i) \\ \frac{\partial L}{\partial b} = 0 \to 0 = \sum_{i=1}^{m+n} \alpha_i \\ \frac{\partial L}{\partial e_i} = 0 \to e_i = \frac{\alpha_i}{\gamma \upsilon(\boldsymbol{x}_i)\beta(\boldsymbol{z}_i)}, i = 1, \ldots, m+n \\ \frac{\partial L}{\partial \alpha_i} = 0 \to \mathrm{y}_i = \boldsymbol{w}^T \varphi(\boldsymbol{x}_i) + b + e_i, i = 1, \ldots, m+n \end{cases} \tag{6}$$

Rearranging Equation (6) and eliminating $\boldsymbol{w}$ and e_i, using a kernel function to replace the inner product of the

feature vectors, the following matrix equation (7) can be obtained:

$$\begin{bmatrix} 0 & 1 & 1 & \cdots & 1 \\ 1 & K(\boldsymbol{x}_1,\boldsymbol{x}_1)+\frac{1}{\gamma v(\boldsymbol{x}_1)\beta(\boldsymbol{z}_1)} & K(\boldsymbol{x}_1,\boldsymbol{x}_2) & \cdots & K(\boldsymbol{x}_1,\boldsymbol{x}_{m+n}) \\ 1 & K(\boldsymbol{x}_2,\boldsymbol{x}_1) & K(\boldsymbol{x}_2,\boldsymbol{x}_2)+\frac{1}{\gamma v(\boldsymbol{x}_2)\beta(\boldsymbol{z}_2)} & \cdots & K(\boldsymbol{x}_2,\boldsymbol{x}_{m+n}) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & K(\boldsymbol{x}_{m+n},\boldsymbol{x}_1) & K(\boldsymbol{x}_{m+n},\boldsymbol{x}_2) & \cdots & K(\boldsymbol{x}_{m+n},\boldsymbol{x}_{m+n})+\frac{1}{\gamma v(\boldsymbol{x}_{m+n})\beta(\boldsymbol{z}_{m+n})} \end{bmatrix} \begin{bmatrix} b \\ \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_{m+n} \end{bmatrix} = \begin{bmatrix} 0 \\ y_1 \\ y_2 \\ \vdots \\ y_{m+n} \end{bmatrix} \tag{7}$$

where the kernel function is $K(\mathbf{x}_i,\mathbf{x}_t)=\varphi^T(\mathbf{x}_i)\varphi(x_t), i=1,\dots,m+n; t=1,\dots,m+n$.

For the determination of $\beta(\boldsymbol{z}_i)\in R, i=1,\dots,m+n$, for each data point in the augmented data set, we wish to accord points relevant to the points in the target domain more weight than irrelevant points. In conjunction with the use of the kernel function, the relevance is evaluated by the Euclidean distance in a reproduced kernel Hilbert space (RKHS). Specifically, in a feature space, data points (e.g.,$\varphi(\boldsymbol{z}_i)$) close to the points in the target domain (e.g., $\varphi(\boldsymbol{z}_k^T)$) will acquire more weight than distant points. Because the small data set in the target domain has already been included in the augmented data set, $\beta(\boldsymbol{z}_i\cap\boldsymbol{z}_k^T)$ will be one. Thus, the problem is changed to determine $\beta(\boldsymbol{z}_i\cap\boldsymbol{z}_j^S)$. To obtain $\beta(\boldsymbol{z}_i\cap\boldsymbol{z}_j^S)$ for each data point in the source domain, we wish to reweight the data points in the source domain such that the mean of the weighted data points in the source domain (i.e., $\frac{1}{n}\sum_{j=1}^{n}\beta(\boldsymbol{z}_i\cap\boldsymbol{z}_j^S)\varphi(\boldsymbol{z}_j^S)$) is close to the mean of the data points in the target domain ($\frac{1}{m}\sum_{k=1}^{m}\varphi(\boldsymbol{z}_k^T)$). Denote $\boldsymbol{\beta}=\{\beta(\boldsymbol{z}_i\cap\boldsymbol{z}_j^S)\}_{j=1}^{n}$ as a weight vector containing the weight for each data point in the source domain. According to the KMM algorithm (Gretton et al., 2009; Huang et al., 2007), the weight vector $\boldsymbol{\beta}$ can be obtained by minimizing the discrepancy between the mean of the weighted source domain data and the mean of the target domain data subjected to two constraints as shown in the following:

$$\boldsymbol{\beta}=arg\min_{\beta}\left\|\frac{1}{n}\sum_{j=1}^{n}\beta\left(\boldsymbol{z}_i\cap\boldsymbol{z}_j^S\right)\varphi\left(\boldsymbol{z}_j^S\right)-\frac{1}{m}\sum_{k=1}^{m}\varphi\left(\boldsymbol{z}_k^T\right)\right\|^2 \tag{8}$$

By reformulating Equation (8) and using the kernel function to replace the inner product of the feature vectors, the following quadratic programming (QP) problem concerning the two constraints can be formulated:

$$\text{Minimize}:\ J(\beta)=\frac{1}{2}\ \beta^T K_1\beta-\kappa^T\beta \tag{9}$$

$$\text{subject to}:\ \left|\frac{1}{n}\sum_{j=1}^{n}\beta\left(z_i\cap z_j^S\right)-1\right|\le\epsilon \qquad 0\le\beta\left(z_i\cap z_j^S\right)\le B, j=1,\dots,n \tag{10}$$

where $\boldsymbol{K}_1=\boldsymbol{K}_{jt}=K(\boldsymbol{z}_j^S,\boldsymbol{z}_t^S)\in R^{n\times n}$, $j,t=1,\dots,n$ is a kernel matrix calculated based on the data in the source domain, $B=1{,}000$ is the upper boundary to reflect the scope of discrepancy between the source domain distribution $p^S(\boldsymbol{z})$ and the target domain distribution $p^T(z)$, $\epsilon=(\sqrt{n}-1)/\sqrt{n}$ is the normalization error, $\boldsymbol{\kappa}=\frac{n}{m}\boldsymbol{K}_2 1_{m\times 1}\in R^n$, where $\boldsymbol{K}_2=\boldsymbol{K}_{jk}=K(\boldsymbol{z}_j^S,\boldsymbol{z}_k^T)\in R^{n\times m}$, $j=1,\dots,n$ and $k=1,\dots,m$ is a kernel matrix calculated based on the source and target domain data.

After solving the QP problem and normalizing the weights $\boldsymbol{\beta}=\boldsymbol{\beta}/max(\boldsymbol{\beta})$, each data point in the source domain will have an associated weight $\beta(\boldsymbol{z}_i\cap\boldsymbol{z}_j^S)$. Because we have already determined the weight $\beta(\boldsymbol{z}_i\cap\boldsymbol{z}_k^T)$ for each data point in the target domain, we now have determined the weight $\beta(\boldsymbol{z}_i)$ for each data point in the augmented data set. The points having a large weight in the augmented data set will be more relevant to the target domain points than points having a small weight. Additionally, irrelevant data points are equivalent to outliers in this case, as they are distant from the target domain data points (De Brabanter et al., 2009; Mu & Yuen, 2015; Rousseeuw & Leroy, 1987; Suykens, De Brabanter, Lukas, & Vandewalle, 2002; Yuen & Mu, 2012; Yuen & Ortiz, 2017). Although these "outliers" already have a small weight, we wish to further reduce their negative effect. Thus, another weight $v(\boldsymbol{x}_i)$, which is a function of residuals, is incorporated as well, as presented in Equation (3). By imposing the weight $\beta(\boldsymbol{z}_i)$ to each data point in the augmented data set, the relevant points will have small residuals whereas the irrelevant points or "outliers" will have large residuals. Points having large residuals will have a small weight $v(\boldsymbol{x}_i)$, whereas points having small residuals will have a large weight $v(\boldsymbol{x}_i)$. Therefore, in this sense, the importance of the relevant points is further emphasized, whereas that of the irrelevant points is further diminished. According to Suykens, De Brabanter et al. (2002), the weight $v(\boldsymbol{x}_i)$ is determined by the following:

$$v(x_i)=\begin{cases}1 & \text{if } |e_i/\delta|\le c_1\\ \frac{c_2-|e_i/\delta|}{c_2-c_1} & \text{if } c_1\le|e_i/\delta|\le c_2\\ \varepsilon & \text{otherwise}\end{cases} \tag{11}$$

where $c_1 = 2.5$, $c_2 = 3$, $\varepsilon = 10^{-4}$, and $\delta = 1.483MAD(\{e_i\}_{i=1}^{m+n})$ is a robust estimate where MAD is the median absolute deviation and other variables are defined previously.

After solving Equation (7) (Suykens, Van Gestel, De Brabanter, De Moor, & Vandewalle, 2002), the Lagrange multiplier $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_r)$ and parameter b can be obtained, which can then be utilized for prediction in the target domain (e.g., $\boldsymbol{x}^T$) using the following:

$$\hat{y}(\boldsymbol{x}^T) = \sum_{i=1}^{m+n} \alpha_i K(\boldsymbol{x}^T, \boldsymbol{x}_i) + b \tag{12}$$

The RBF kernel is utilized for all the mentioned kernel functions above, which is defined as follows:

$$K(\boldsymbol{x}_i, \boldsymbol{x}_t) = \exp\left(-\frac{\boldsymbol{x}_i - \boldsymbol{x}_t{}^2}{2\sigma^2}\right) \tag{13}$$

4 | HYPERPARAMETER TUNING AND IMPLEMENTATION

There are three hyperparameters in the proposed DW-SVTR—one regularization parameter γ, one kernel parameter σ^2 for the kernel function $K(\boldsymbol{x}_i, \boldsymbol{x}_t)$, and one kernel parameter σ_β^2 for kernel functions $K(\boldsymbol{z}_j^S, \boldsymbol{z}_t^S)$ and $K(\boldsymbol{z}_j^S, \boldsymbol{z}_k^T)$—that need to be accurately defined before the training process because they can significantly affect the accuracy level of the predicted results. In this article, the hyperparameter tuning procedure is performed by evaluating the performance of the proposed DW-SVTR using leave-one-out (LOO) cross-validation on the target domain data points $\{(\boldsymbol{x}_k^T, y_k^T)\}_{k=1}^m$ in the augmented training data set $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^{m+n}$ introduced in Section 3. The optimal values are those that minimize the mean square error (MSE).

The implementation procedure of the proposed DW-SVTR approach for reducing the effect of sample bias of small data sets in the target domain, which is also applicable for unrelated domains, is summarized as follows:

Algorithm 1: Implementation of the proposed DW-SVTR model

Require: Training data sets in the source domain $\{\boldsymbol{z}_j^S\}_{j=1}^n = \{(\boldsymbol{x}_j^S, y_j^S)\}_{j=1}^n$ and target domain $\{\boldsymbol{z}_k^T\}_{k=1}^m = \{(\boldsymbol{x}_k^T, y_k^T)\}_{k=1}^m$, test data in the target domain $\boldsymbol{x}^T$, and optimal hyper-parameter combination $(\gamma, \sigma^2, \sigma_\beta^2)$.

1. Initialization stage:

(a) Transform the training data sets in the source and target domains individually using Eqs. (1-2);

(b) Record the means $\bar{\boldsymbol{x}}_{tr}^T$, $\bar{y}_{tr}^T$ and standard deviations $\boldsymbol{\sigma}_{\boldsymbol{x}_{tr}^T}$, $\sigma_{y_{tr}^T}$ for the target domain training data set $\{(\boldsymbol{x}_k^T, y_k^T)\}_{k=1}^m$;

(c) Combine the transformed data sets in the source and target domains as an augmented data set $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^{m+n}$;

2. Reweighting stage:

(a) Calculate the $\boldsymbol{K}_1$ and $\boldsymbol{\kappa}$ in Eq. (9) using Eq. (13) with the parameter σ_β^2;

(b) Set $\beta(\boldsymbol{z}_i \cap \boldsymbol{z}_k^T) = 1, k = 1, \ldots, m$;

(c) Solve Eq. (9-10) to obtain $\boldsymbol{\beta} = \{\beta(\boldsymbol{z}_i \cap \boldsymbol{z}_j^S)\}_{j=1}^n$ and normalize it as $\boldsymbol{\beta} = \boldsymbol{\beta}/max(\boldsymbol{\beta})$;

(d) Combine $\{\beta(\boldsymbol{z}_i \cap \boldsymbol{z}_k^T)\}_{k=1}^m$ and $\{\beta(\boldsymbol{z}_i \cap \boldsymbol{z}_j^S)\}_{j=1}^n$ as $\{\beta(\boldsymbol{z}_i)\}_{i=1}^{m+n}$;

(e) Set weight $\upsilon(\boldsymbol{x}_i)$ in Eq. (7) for each data point in the augmented data set $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^{m+n}$ to 1;

(f) Solve Eq. (7) to obtain $\boldsymbol{\alpha}$, b, and compute $e_i = \alpha_i/(\gamma\upsilon(\boldsymbol{x}_i)\beta(\boldsymbol{z}_i)), i = 1, \ldots, m+n$;

3. Iterative stage:

Set the maximum iterative number S, tolerance tol, count $s = 0$, and $t = Inf$

while $t > tol$ & $s < S$ do

(a) Set $\boldsymbol{\alpha}^{(s)} = \boldsymbol{\alpha}, b^{(s)} = b, e_i^{(s)} = e_i$, and $\upsilon^{(s)}(\boldsymbol{x}_i) = \upsilon(\boldsymbol{x}_i)$;

(b) Compute the robust estimate $\delta^{(s)} = 1.483MAD(e_i^{(s)})$;

(c) Update the weight $\upsilon^{(s+1)}(\boldsymbol{x}_i)$ from $\delta^{(s)}$ and $e_i^{(s)}$ using Eq. (11);

(d) Solve Eq. (7) to obtain the $\boldsymbol{\alpha}^{(s+1)}$ and $b^{(s+1)}$;

(e) Update the $e_i^{(s+1)} = \alpha_i^{(s+1)}/(\gamma\upsilon^{(s+1)}(\boldsymbol{x}_i)\beta(\boldsymbol{z}_i))$;

(f) Calculate $t = \boldsymbol{\alpha}^{(s+1)} - \boldsymbol{\alpha}^{(s)}$;

(g) Set $\boldsymbol{\alpha} = \boldsymbol{\alpha}^{(s+1)}, b = b^{(s+1)}, e_i = e_i^{(s+1)}$, and $\upsilon(\boldsymbol{x}_i) = \upsilon^{(s+1)}(\boldsymbol{x}_i), i = 1, \ldots, m+n$;

(h) Set $s = s + 1$

end while

4. Output stage:

(a) Transform the target test data $\boldsymbol{x}^T$ with the recorded mean $\bar{\boldsymbol{x}}_{tr}^T$ and standard deviation $\boldsymbol{\sigma}_{\boldsymbol{x}_{tr}^T}$ using Eq. (1);

(b) Output the final $\boldsymbol{\alpha}$ and b from the stage 3;

(c) Given $\boldsymbol{\alpha}$ and b, predict the response value $\hat{y}(\boldsymbol{x}^T)$ of the transformed data $\boldsymbol{x}^T$ using Eq. (12);

(d) Transform the predicted $\hat{y}(\boldsymbol{x}^T)$ back by $\hat{y}(\boldsymbol{x}^T) = \hat{y}(\boldsymbol{x}^T) \times \sigma_{y_{tr}^T} + \bar{y}_{tr}^T$;

5 | ILLUSTRATIVE EXAMPLES

To thoroughly assess the performance of the proposed DW-SVTR approach, two examples using simulated and multidimensional real data are carried out. First, the simulated example is used to illustrate the general performance for the most challenging case via data sets where both the marginal and posterior distributions of the two domains

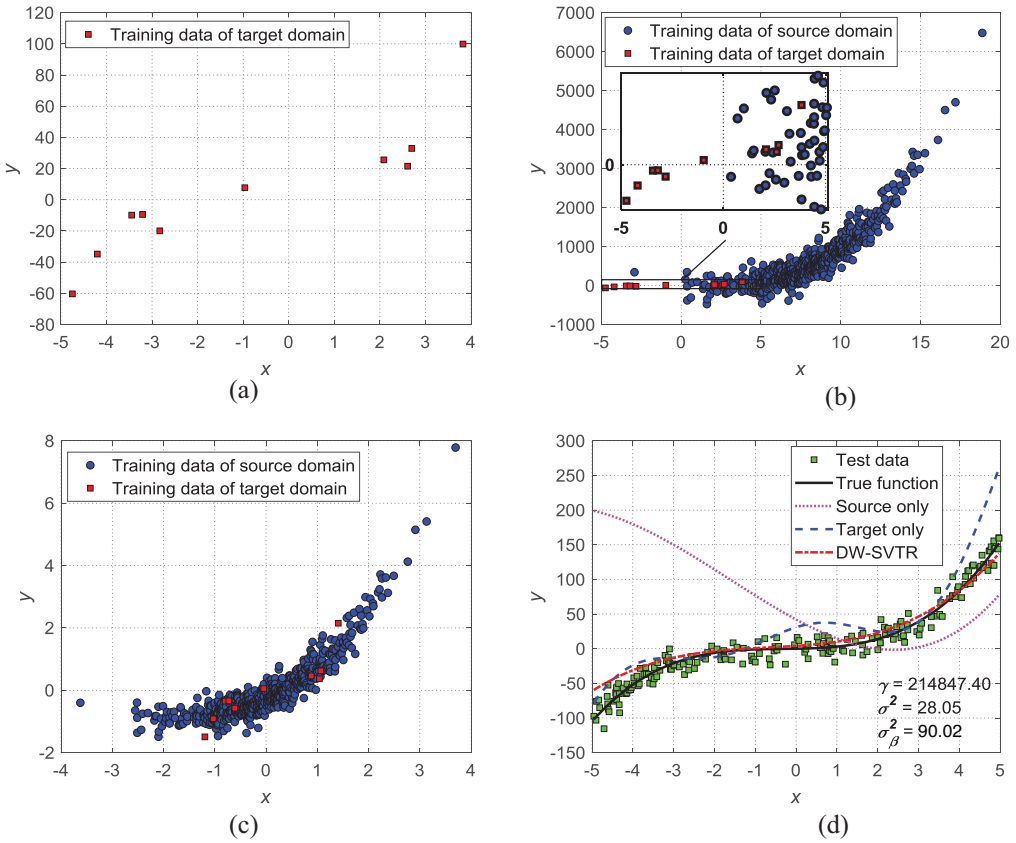

FIGURE 1 A typical representation of 10 random trials for the three analytical cases: (a) target domain training sample points in the original space; (b) combined source and target domain training data sets in the original space; (c) combined source and target domain training data sets in the transformed space; (d) result comparison of three analytical cases in the original space

are different. Then, the proposed approach is employed to predict the shear strength of nonductile reinforced concrete (RC) columns to illustrate the real-world utilization of the approach when sufficient large data sets are not available.

5.1 | Example 1: Simulated data sets

This example is designed to illustrate how the proposed DW-SVTR works in an especially challenging case. In this example, the data sets in the source and target domains are generated from different joint distributions, where both the marginal and posterior distributions are different, that is, $p^S(\boldsymbol{x}) \neq p^T(\boldsymbol{x})$ and $p^S(y|\boldsymbol{x}) \neq p^T(y|\boldsymbol{x})$. The source domain has a sufficient number of data points, whereas the target domain only has a few data points. Thus, the target domain data have a potentially large sample bias. This case is more challenging as both the predictor and the response values in the data sets for the source and the target domains may be significantly different, more likely leading to the case where there is no relevance between the source and the target domains. In the context of regression settings, it is commonly thought that there is no way to use an ML model trained with data from one domain to improve the prediction on another, seemingly, completely irrelevant domain. However, the theory presented in the previous section along with the following experimental results demonstrates that the proposed DW-SVTR can still transfer useful information to reduce the negative effect induced by sample bias due to small data and improve the predictive performance in this case.

The marginal distributions of the data sets in the source and target domains are assumed as normal and uniform distributions, respectively, where $x^S \sim \text{Normal}(8, 3^2)$ and $x^T \sim \text{Uniform}(-5, 5)$. The responses for the data set in the source domain are generated from $y^S = -6x^S + (x^S)^3 + \varepsilon^S$, whereas those for the data set in the target domain are generated according to $y^T = x^T + (x^T)^2 + (x^T)^3 + \varepsilon^T$. The distribution of the error term for the source data is $\varepsilon^S \sim \text{Normal}(0, 200^2)$, and for the target data, it is $\varepsilon^T \sim \text{Normal}(0, 12^2)$. Thus, in this sense, both the posterior and the marginal distributions between the source and the target domain data are different. Ten points (e.g., red squares in Figure 1a) randomly sampled from the target domain serve as the training data from the target domain and 600

points (e.g., blue circles in Figure 1b) randomly sampled from the source domain are the training data from the source domain. An individual test data set including 200 points (e.g., green squares in Figure 1d) is randomly generated from the target domain.

In this example, based on the simulated data sets presented above, three analytical cases are designed and compared to demonstrate how the proposed method reduces the negative effect induced by a biased sample, thus, improving the overall prediction performance. For these three cases, the training data set varies, but the test data set is held constant: (1) Target only: the 10 training sample points in the target domain (e.g., squares in Figure 1a) are used to train an ML model, and this trained ML model is then used to predict the 200 test sample points in the target domain (e.g., squares in Figure 1d); (2) Source only: the 600 training sample points in the source domain (e.g., circles in Figure 1b) are used to train an ML model, and this ML model is used to predict the 200 test sample points in the target domain; and, (3) DW-SVTR: all 610 training sample points (i.e., 10 sample points from target domain as introduced in Case 1 and 600 sample points from the source domain as introduced in Case 2) are used as the training data set for the proposed DW-SVTR, and the trained DW-SVTR model is then utilized to predict the 200 test sample points in the target domain. The LS-SVMR (Suykens, Van Gestel et al., 2002) is employed for Cases (1) and (2). Therefore, the cases of *target only* and *source only* are regarded as the benchmarks in comparison to the *DW-SVTR* case. It should be noted that both LS-SVMR and proposed DW-SVTR are nonparametric regression methods.

We individually run the experiment 10 times by setting 10 different random seeds to statistically reflect the performance of the proposed DW-SVTR. A typical representative of the results within the 10 runs is presented in Figure 1. Figure 1a shows the small training data set in the target domain, which only includes 10 training sample points and thus has a potentially large sample bias. Figure 1b presents the training data sets in the source and target domains that are combined in a figure. It is found that in Figure 1b only four target domain points are near to the points in the source domain in the original space. This illustrates the significant lack of relevance between the two domains. Figure 1c shows the combined training data set in the transformed space. Note that the transformation for the data sets in the source and target domains is first performed separately using Equations (1) and (2). Then, the transformed data sets in the source and target domains are combined, as described in Algorithm 1 in Section 4. It is observed in Figure 1c that the relevance between the two domains significantly increases after transformation.

Figure 1d shows the comparison of results among the three analytical cases. For analytical Case 1, from Figure 1d, it is observed that the LS-SVMR model trained with 10 training sample points has a large bias in some areas where the training sample points are not available. This is demonstrated in Figure 1d by the dashed line (i.e., *target only*), which has an apparent discrepancy from the solid line (i.e., *true function*) in the areas where the training sample points are not available as shown in Figure 1a. For analytical Case 2, as the source domain training data set is not relevant to the target domain, and thus, the LS-SVMR model trained with the 600 source domain training sample points has a significantly large bias for prediction on the target domain. This is illustrated by the significant discrepancy between the dotted line (i.e., *source only*) and the solid line across almost all the areas represented by the test data set in the target domain. For analytical Case 3, the proposed DW-SVTR model is used and trained with the combined training data set in the transformed space. The proposed DW-SVTR model attributes more weight to the source domain sample points that are close to the 10 target domain training sample points than the distant source domain points. Hence, the proposed approach can borrow more relevant source domain sample points to augment the small set of target domain training sample points, reducing the effect of small sample bias without sustaining negative effects from those distant source domain points. Also, the negative interferences of these distant source domain points are further diminished by another weight in the proposed DW-SVTR model, as introduced previously. The obtained three hyperparameters of the proposed DW-SVTR for this typical representative is presented in Figure 1d. The numerical experiment result predicted by the DW-SVTR (dash-dot line in Figure 1d) agrees well with the true function (solid line in Figure 1d), demonstrating that the proposed DW-SVTR can reasonably predict all test sample points in the target domain regardless of the unrelated nature of the two domains, and further, illuminating the powerful TL capabilities of the proposed DW-SVTR approach.

The result comparisons of 10 random trials among these three analytical cases are presented in Figure 2. Figures 2a and 2b show their predictive performance comparison over the 10 random trials using box plots in terms of coefficient of determination (R^2) and root mean square error (RMSE), respectively. By observation of Figure 2, the proposed DW-SVTR has the highest R^2 and the lowest RMSE among these three analytical cases in terms of the median of the 10 random trials. Further, the proposed DW-SVTR achieves the smallest performance variation among these three analytical cases over the 10 random trials. Additionally, the obtained mean values of R^2 and RMSE over the 10 random trials for the *source only* case are −7.57 and 135.64, for the *target only* case are 0.88 and 16.06, and for the proposed *DW-SVTR* case are

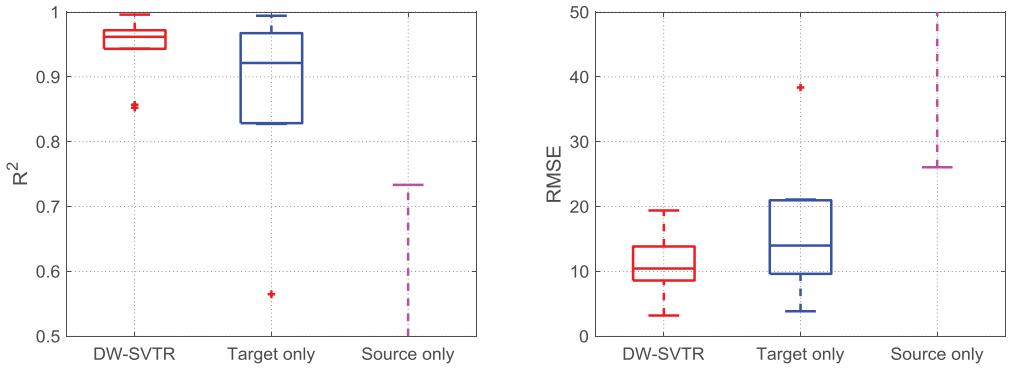

FIGURE 2 Comparison of the performance of the TL approaches over the 10 random trials using box plots in terms of R^2 and RMSE. For visual reasons, not all data are displayed for *source only* case

0.94 and 11.36. Therefore, it is evident that the proposed DW-SVTR (i.e., Case 3) statistically performs the best in comparison to the other two analytical cases, and analytical Case 2 (i.e., *source only*) statistically has the worst performance.

5.2 | Example 2: Shear strength prediction of nonductile RC columns

In structural and earthquake engineering, ductile and nonductile RC columns under earthquake loads have different physical behaviors and failure modes. Ductile columns typically have good seismic performance and deformation capacity and will most likely experience flexure failures under large earthquakes, whereas nonductile columns have worse seismic-resistant capabilities, leading to flexure-shear and shear failures under earthquakes (Moehle, 2014). Nonductile columns will easily cause the global collapse of RC frame buildings under large earthquakes due to their associated shear strength deficiency. Thus, it is critical and necessary to identify the shear strength of nonductile columns before the occurrence of large earthquakes such that these nonductile columns can be reinforced and retrofitted to enhance their seismic performance, avoiding the global collapse of RC frame buildings. The challenge is that there are a greater number of tests performed on ductile columns than nonductile columns. Thus, if nonductile columns are considered in isolation, significant bias may be present in an ML model due to the smaller size of this data set. Therefore, this example intends to predict the shear strength of nonductile columns using the proposed DW-SVTR model when the availability of the training data set of nonductile columns is limited, by transferring knowledge from a data set composed of ductile columns.

5.2.1 | Data sets

In this example, two column data sets, including rectangular RC columns and circular RC columns, are used to further assess the proposed DW-SVTR model in real-world applications. Both of these two data sets are taken from physical experiments. For the rectangular RC column data set, there are a total of 262 sample points where 208 of them are flexure-critical columns, which are classified as ductile columns and the remaining 54 are shear- and flexure-shear–critical columns, which are categorized as nonductile columns. For the circular RC column data set, there are a total of 160 sample points where 98 of them are ductile columns (i.e., flexure-critical columns) and the remaining 62 are nonductile columns (i.e., flexure-shear- and shear-critical columns). For each data set, the input predictors (i.e., explanatory variables) are column gross sectional area (X_1), concrete compressive strength (X_2), column cross-sectional effective depth (X_3), longitudinal reinforcement yield stress (X_4) and area (X_5), transverse reinforcement yield stress (X_6) and area (X_7), stirrup spacing to effective depth ratio (X_8), shear span to effective depth ratio (X_9), and applied axial load (X_{10}), and the response variables are lateral strength (y_1) and drift capacity (y_2). Thus, for either rectangular or circular section RC columns, the data set is comprised of the same predictors and response variables. The input predictors are selected to cover all the aspects that can affect the seismic performance of an RC column (Hua, Eberhard, Lowes, & Gu, 2019; Moehle, 2014). The statistical properties for the 208 and 54 rectangular RC ductile and nonductile columns and the 98 and 62 circular RC ductile and nonductile columns are summarized in Tables 1 and 2, respectively. Note that some of the input predictors are normalized in Tables 1 and 2 to maintain commonly used terminologies. Because only the drift capacity (i.e., y_2) of ductile columns will be used to constitute the

TABLE 1 Statistical range of material and geometric properties for the rectangular RC ductile and nonductile column database

Property	Minimum	Maximum	Mean	Std. dev
Shear span to effective depth ratio	1.5 (1.08)	8.40 (3.76)	4.19 (2.49)	1.52 (0.93)
Stirrup spacing to effective depth ratio	0.12 (0.11)	0.9 (1.14)	0.28 (0.5)	0.13 (0.33)
Concrete compressive strength (MPa)	20.6 (16)	118 (86)	55.4 (31.1)	29.7 (12.4)
Longitudinal reinforcement yield stress (MPa)	339 (318)	635 (510)	445.7 (406.4)	62.35 (70.87)
Transverse reinforcement yield stress (MPa)	255 (249)	1,424 (559)	509.4 (400.3)	235.97 (82.99)
Longitudinal reinforcement ratio	0.01 (0.013)	0.06 (0.04)	0.023 (0.024)	0.01 (0.005)
Transverse reinforcement ratio	0.0011 (0.0006)	0.03 (0.012)	0.009 (0.004)	0.006 (0.003)
Axial load ratio	0 (0)	0.8 (0.9)	0.25 (0.3)	0.17 (0.28)
Maximum shear force (kN)	32.16 (29.56)	1,338.80 (604.6)	218.84 (187.5)	191.82 (135.78)
Drift capacity (%)	0.72	9.39	3.93	1.91

Note: The values in the parentheses are for nonductile columns.

TABLE 2 Statistical range of material and geometric properties for the circular RC ductile and nonductile column database

Property	Minimum	Maximum	Mean	Std. dev
Shear span to effective depth ratio	1.76 (1.18)	10.49 (3.32)	4.78 (1.84)	1.98 (0.52)
Stirrup spacing to effective depth ratio	0.04 (0)	0.73 (0.58)	0.14 (0.21)	0.098 (0.12)
Concrete compressive strength (MPa)	22 (18.9)	90 (42.2)	40.64 (31.65)	17.31 (4)
Longitudinal reinforcement yield stress (MPa)	240 (240)	565.4 (482)	425.35 (400)	65.68 (54.89)
Transverse reinforcement yield stress (MPa)	207 (0)	1,000 (691.5)	460.47 (334.4)	152.2 (125.1)
Longitudinal reinforcement ratio	0.0046 (0.005)	0.0558 (0.05)	0.024 (0.03)	0.0098 (0.01)
Transverse reinforcement ratio	0.0013 (0)	0.0349 (0.0427)	0.012 (0.007)	0.0073 (0.007)
Axial load ratio	0.00 (0)	0.74 (0.57)	0.17 (0.11)	0.16 (0.12)
Maximum shear force (kN)	19.00 (75)	2,968.00 (985)	251.43 (295.39)	360.18 (156.58)
Drift capacity (%)	1.59	14.66	6.04	2.72

Note: The values in the parentheses are for nonductile columns.

source domain data (see Sections 5.2.2 and 5.2.3 for more detailed information), the statistical properties for the drift capacity of ductile columns are given. More detailed information for the rectangular and circular RC column data sets can be found in Luo and Paal (2018, 2019), respectively.

For each data set, we select the nonductile columns as the target domain and the ductile columns as the source domain. The main difference between ductile and nonductile columns is that the lateral strength for the ductile columns is governed by flexural strength, whereas that for nonductile columns is dominated by shear strength (Moehle, 2014). The lateral strength is defined at the maximum shear force (kN) in the hysteretic force–deformation curve. We designed 10 numerical experiments to sufficiently assess the performance of the proposed DW-SVTR approach based on these two data sets. For each data set, the task for the target domain will always be the shear strength prediction of nonductile columns, but the source domain training data set will vary. The detailed information is as follows.

5.2.2 | Validation for rectangular columns

For the rectangular columns, the target domain data set is comprised of the 54 nonductile RC rectangular columns with shear strength (i.e., y_1) as the response variable. Five numerical experiments are designed to evaluate four different transfer strategies in comparison to one baseline model. *Experiment 1* corresponds to the scenario where the source domain training data set consists of the 208 rectangular ductile columns with the flexural strength (i.e., y_1) as the response variable. *Experiment 2* corresponds to the scenario where the source domain training data set consists of the 208 rectangular

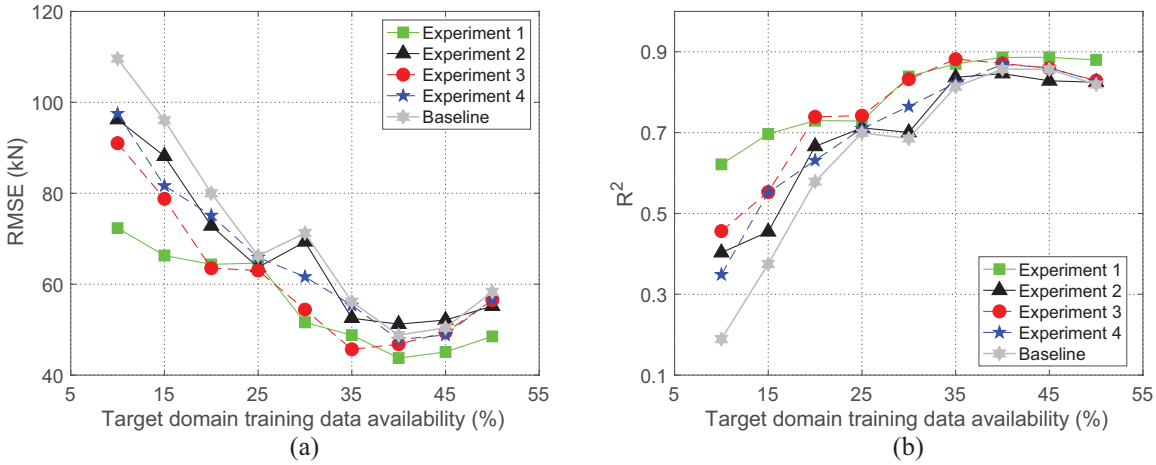

FIGURE 3 Performance versus size of target domain training data availability in terms of (a) mean RMSE and (b) mean R^2 for rectangular columns over the 10 random trials

ductile columns with the drift capacity (i.e., y_2) as the response variable. *Experiment 3* corresponds to the scenario where the source domain training data set consists of the 98 circular ductile columns with the flexural strength (i.e., y_1) as the response variable. *Experiment 4* corresponds to the scenario where the source domain training data set consists of the 98 circular ductile columns with the drift capacity (i.e., y_2) as the response variable. Finally, *Experiment 5* corresponds to the baseline, where only the target domain training data set is used and no transfer strategy is applied. It should be noted that drift capacity (%) has an entirely different physical meaning from shear strength (kN) (Moehle, 2014), which translates to a large discrepancy between the corresponding numeric values. Further, the reinforcement layouts and cross-section shapes of rectangular and circular columns are also different. In this sense, *Experiment 1* is analogous to related joint distributions between the source and target domains; *Experiment 2* is analogous to related marginal distributions but unrelated posterior distributions (i.e., $p^S(y|\boldsymbol{x}) \neq p^T(y|\boldsymbol{x})$); *Experiment 3* is analogous to unrelated marginal distributions (i.e., $p^S(\boldsymbol{x}) \neq p^T(\boldsymbol{x})$) but related posterior distributions; and, *Experiment 4* is analogous to unrelated marginal and posterior distributions (i.e., $p^S(\boldsymbol{x}) \neq p^T(\boldsymbol{x})$ and $p^S(y|\boldsymbol{x}) \neq p^T(y|\boldsymbol{x})$). Therefore, these five numerical experiments can thoroughly and effectively assess the performance of the proposed DW-SVTR model.

For each experiment, the availability of the target domain training data is apportioned as 10%, 15%, 20%, 25%, 30%, 35%, 40%, 45%, and 50% of the total target domain data, and the test set for the target domain will be maintained at 50% of the total target domain data (mutually exclusive from the target domain training data). For each case of data availability, each experiment is run 10 times with different random seeds to measure the performance variability for the proposed DW-SVTR model for different split of target domain training and test sets. This ensures the results are statistically reliable. It should be noted that for each run, the same target domain training and test sets are applied for all five experiments mentioned above. For *Experiments 1* through *4*, the proposed DW-SVTR model is used, whereas for *Experiment 5* (i.e., the baseline), LS-SVMR is used. The comparison of results for these transfer strategies and the baseline in each case of target domain training data availability is shown in Figure 3, where both R^2 and RMSE are taken as the averages of the R^2 and RMSE over the 10 random trials.

From Figure 3, it is observed that, compared to the baseline, both R^2 and RMSE suggest that the proposed DW-SVTR model significantly improves the prediction performance when the target domain training data is very small (i.e., only 10% availability). The RMSE is decreased from 109.59 kN (in the baseline model) to 72.34 kN (*Experiment 1*), 96.22 kN (*Experiment 2*), 91.02 kN (*Experiment 3*), and 97.52 kN (*Experiment 4*), resulting in a reduction of 34%, 12%, 17%, and 11%, respectively. The R^2 is increased from 0.19 (*Baseline*) to 0.62 (*Experiment 1*), 0.40 (*Experiment 2*), 0.46 (*Experiment 3*), and 0.35 (*Experiment 4*), enhancing the performance by 229%, 110%, 142%, and 84%, respectively. With the increase in the size of the target domain training data, the prediction performance in terms of average RMSE and R^2 values over the 10 random trials for all five experiments globally increases (though a few locally decreases), and the improved performance of the proposed DW-SVTR globally decreases. This is because, with the increase of available target domain training data, the target domain sample bias decreases, and thus, the performance difference between the baseline and the proposed approach also

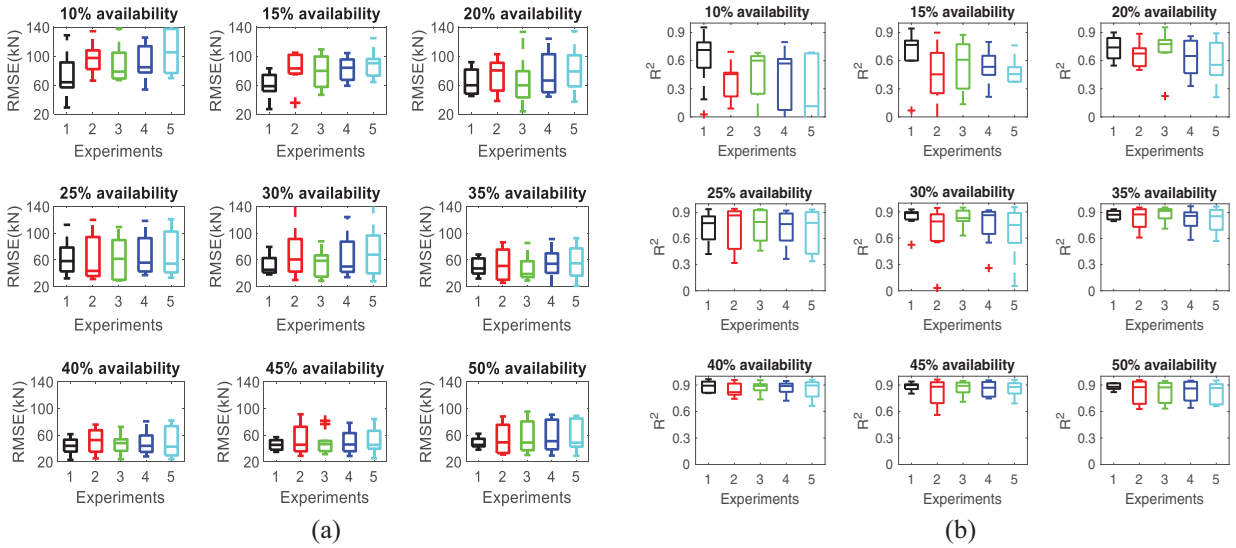

FIGURE 4 Boxplots for rectangular columns over 10 random trials based on four different transfer situations and one baseline in terms of (a) RMSE and (b) R^2. The values on the x-axis represent the experiment number as described in Section 5.2.2. For visual reasons, not all data is displayed for some experiments

decreases. According to different transfer strategies, the improved performance by the proposed DW-SVTR also varies. The most significant performance improvement in terms of both RMSE and R^2 is in *Experiment 1*, followed by *Experiment 3*, and *Experiment 2* is comparable to *Experiment 4*. However, both *Experiments 2* and *4* are outperformed by *Experiment 3*. It is worth noting that the proposed DW-SVTR model also works for *Experiment 2* where the posterior distributions between the source and target domains are unrelated and for *Experiment 4* where both the marginal and posterior distributions are unrelated, as introduced previously. This further demonstrates that the proposed approach is effective even if the source and target domains are unrelated. The comparison of performance variability over the 10 random trials in each case is reported by way of boxplots in Figure 4. From Figure 4, it is observed that, compared to the baseline, the proposed DW-SVTR statistically improves the performance in terms of the median of 10 random trials for all four transfer strategies.

5.2.3 | Validation for circular columns

For the circular columns, the target domain data set is comprised of the 62 nonductile circular columns with shear strength (i.e., y_1) as the response variable. *Experiment 1* corresponds to the scenario where the source domain training data set consists of the 98 circular ductile columns with the flexural strength (i.e., y_1) as the response variable. *Experiment 2* corresponds to the scenario where the source domain training data set consists of the 98 circular ductile columns with the drift capacity (i.e., y_2) as the response variable. *Experiment 3* corresponds to the scenario where the source domain training data set consists of the 208 rectangular ductile columns with the flexural strength (i.e., y_1) as the response variable. *Experiment 4* corresponds to the scenario where the source domain training data set consists of the 208 rectangular ductile columns with the drift capacity (i.e., y_2) as the response variable. *Experiment 5* also corresponds to the baseline, as introduced in Section 5.2.2. The same validation procedure described in the previous section is also utilized here. The comparison among these four transfer strategies and the baseline in each case of target domain training data availability is shown in Figure 5, where both R^2 and RMSE are taken as the averages of the R^2 and RMSE over 10 random trials.

From Figure 5, it is observed that when the availability of target domain training data is 10%, the R^2 for the baseline is negative, which means the fully trained LS-SVMR model for the baseline has a significantly large bias and thus breaks down. In this case, the proposed DW-SVTR can still improve the performance of the baseline. Additionally, when the availability of target domain training data is 15%, both R^2 and RMSE suggest that the proposed DW-SVTR approach significantly improves the prediction performance of the baseline. The RMSE is decreased from 143.38 kN (*Baseline*) to 110.48 kN (*Experiment 1*), 129.07 kN (*Experiment 2*), 128.57 kN (*Experiment 3*), and 135.66 kN (*Experiment 4*), resulting in a reduction of roughly 23%, 10%, 10%, and 5%, respectively. The R^2 value is increased from 0.16 (*Baseline*) to 0.49 (*Experiment 1*), 0.33 (*Experiment 2*), 0.31 (*Experiment 3*), and 0.19 (*Experiment 4*), enhancing the performance by roughly 206%, 106%, 94%,

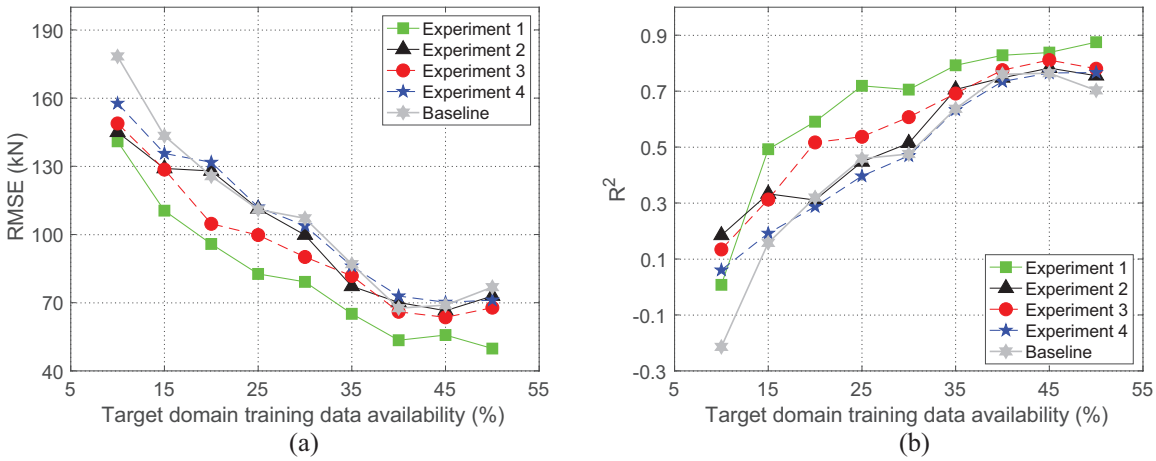

FIGURE 5 Performance versus size of target domain training data availability curve in terms of (a) mean RMSE and (b) mean R^2 for circular columns over the 10 random trials

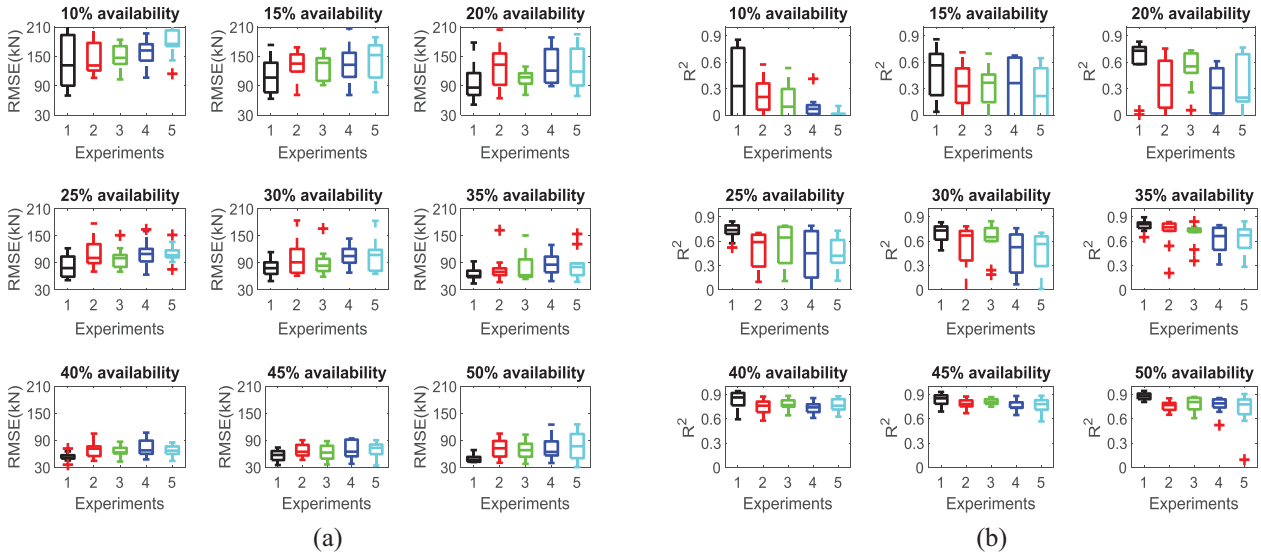

FIGURE 6 Boxplots for circular columns over 10 random trials based on four different transfer situations and one baseline in terms of (a) RMSE and (b) R^2. The values on the *x*-axis represent the experiment number as described in Section 5.2.3. For visual reasons, not all data is displayed for some experiments

and 19%, respectively. With the increase of the size of the target domain training data, a similar trend reflected in the rectangular columns is also captured for the circular columns. According to different transfer strategies, the improved performance by the proposed DW-SVTR also varies. The most significant improvement for both RMSE and R^2 is again in *Experiment 1*, followed by *Experiment 3*. *Experiment 2* is slightly better than *Experiment 4*, but both are outperformed by *Experiment 3*. This investigation agrees well with that for the rectangular column validation. The comparison of performance variability over the 10 random trials in each experiment is also reported as boxplots in Figure 6. By observation of Figures 6a and 6b, the median values of both RMSE and R^2 for all four transfer strategies (i.e., *Experiments 1–4*) are better than the baseline (i.e., *Experiment 5*) when the size of target training data is small. Therefore, it is evident that, compared to the baseline, the proposed DW-SVTR statistically improves the performance in terms of the median of 10 random trials for all four transfer strategies, as observed for the rectangular column data set.

5.3 | Discussion of results

The results obtained for both the simulated data and the multidimensional real-world data presented herein suggest that the proposed DW-SVTR approach can reduce

the effect of sample bias induced by a small data set and improve the overall prediction performance substantially. Further, the proposed DW-SVTR model is also validated for two unrelated domains with both marginal and posterior distributions, which are different (i.e., $p^S(\boldsymbol{x}) \neq p^T(\boldsymbol{x})$ and $p^S(y|\boldsymbol{x}) \neq p^T(y|\boldsymbol{x})$).

The simulated example clearly illustrates how the proposed approach reduces the effect of small sample bias and improves the prediction performance for two unrelated domains. The real-world examples explicitly investigate the performance of the proposed approach in terms of target domain training data availability and different transfer strategies. For the relation between performance variability and target domain training data availability over 10 random trials, it is observed that the apparent performance variability is present when the size of target domain training data is small (e.g., 10% availability) (Figures 4 and 6). Further, with an increase in the target domain training data availability, the performance variability decreases in general. This is because when the target domain training data set is small (e.g., 10% availability), different random seeds (i.e., 10 random trials) produce target domain training data that has different levels of small sample bias for corresponding test data. This causes the variation of improvement in performance, leading to apparent performance variability. When the size of the target domain training data increases, the difference among these levels of small sample bias decreases, producing relatively lower performance variability. For the relation between performance variability and different transfer strategies, all of the numerical results suggested that *Experiment 1* produces the best performance improvement. This could be explained by the fact that, compared to other transfer strategies, *Experiment 1* is associated with the source domain that is most related to the target domain as explained in Section 5.2.2, which means that the source domain data can provide more useful information for the proposed DW-SVTR model to reduce the effect of small sample bias and improve the prediction performance. Notably, even for the case where there are two irrelevant domains, the proposed approach is still able to seek useful information from the source domain data to enhance the prediction performance in the target domain if there is shared information in the transformed space.

The successful validation of the proposed method for the knowledge transfer between two irrelevant domains under this premise in regression settings has emphasized its widespread potential. This approach can be employed regardless of the problem or discipline to reduce the effect of small sample bias in regression scenarios by augmenting small data sets with useful data from a relevant or irrelevant large data set. This will be extremely powerful in scenarios where large data are difficult to acquire, whether that be due to the high economic or computational cost of experimental tests or simulations, or due to the complex nature of acquiring real-world data. For example, in CE, due to the expensive cost associated with full-scale physical experiments of structures, it is often difficult to perform a sufficient number of experimental tests to investigate the structural behavior of a new design, material, construction method, or load. However, with the proposed method, it is possible to augment a small number of tests with useful data from an easily available large data set (e.g., data set from economic or medical domains or something more relevant such as the ductile vs. nonductile example). In this sense, the proposed approach can be employed in countless ways to reduce our reliance on physical testing, minimize computational expense, and minimize actual cost.

5.4 | Limitations and future works

Although the proposed TL method has shown good performance on both synthetic and real data for regression transfer between two domains, it does have some limitations. One of the key limitations is that the target domain data should be close to the source domain data in the transformed space. This is because in the proposed method, one coupled weight function assigns larger weights to the source domain data points close to the target domain data than those source domain points, which are more distant. The source domain data points with larger weights are more relevant to the target domain data and thus play an important role in prediction on the target domain. For source domain points with smaller weights, they are most likely irrelevant and play a lesser role. Additionally, these points with smaller weights can be thought of as outliers, because they are distant from the target domain data. Their negative interference can be further diminished by another coupled weight function. In this way, the performance improvement by the proposed approach for the prediction in the target domain can be achieved with the help of the source domain points with larger weights. Therefore, once there is no nearby source domain data in the transformed space (i.e., all the source domain data are distant from the target domain data), the coupled weight functions will assign smaller weights to all the source domain data points. The proposed approach is not able to improve the prediction performance on the target domain under this circumstance, because there would be no source domain points with larger weights that can be transferred to improve the prediction performance on target domain.

Another limitation of the proposed method is additional data transformation techniques were not tested. The use of appropriate data transformation methods has been proven to improve prediction performance (Han,

Pei, & Kamber, 2011). Further, data transformation techniques could be useful to eliminate the effects of different ranges of values, which can transform the data in different domains into the same transformed space. In this way, some information that is not shared between two domains in the original space may be shared in the transformed space. Future works will explore more data transformation methods such that more shared information between two different domains can be exploited in the transformed space.

Additionally, because the proposed TL method is also an ML model, some properties that ML methods have are also applicable to the proposed method. Like all the ML methods, the proposed method can accurately predict the response within the input ranges of the target domain training set. Outside of these ranges, it cannot necessarily reliably be used for prediction. In this case, the predicted results must be carefully checked with the physical knowledge or experts. This is because the proposed TL method can transfer source domain points with larger weights to improve the prediction performance on target domain. This means that only source domain points close to the target training data in the transformed space are utilized and other distant source domain points are abandoned due to their smaller weights. Thus, the useful information from source domain data is limited to the input ranges of the target training data.

6 | CONCLUSIONS

A novel regression-based TL approach is proposed to reduce the negative effect of sample bias for small data sets. The proposed TL model is termed DW-SVTR, which couples LS-SVMR with two weight functions. The model formulation and implementation are introduced in detail. Numerical experiments are performed on two types of data sets to comprehensively demonstrate the advantages of the proposed approach: simulated data sets and multidimensional real-world (experimental) data sets. The simulated data set example shows how the proposed approach reduces the negative effect of small sample bias and improves the prediction performance for two unrelated domains. Moreover, the real-world data set example explicitly investigates the performance of the proposed approach in terms of target domain training data availability and different transfer strategies. The results from both examples ultimately show that the proposed approach can transfer useful information from the source domain (large data set) to the target domain (small data set), effectively reducing the small sample bias of the target domain (small data set). Further, the results also demonstrate that the proposed approach is still valid for transfer between two irrelevant domains if there is shared information in the transformed space.

REFERENCES

Adeli, H. (2001). Neural networks in civil engineering: 1989–2000. *Computer-Aided Civil and Infrastructure Engineering*, *16*(2), 126–142.

Ahangar-Asr, A., Faramarzi, A., Javadi, A. A., & Giustolisi, O. (2011). Modelling mechanical behaviour of rubber concrete using evolutionary polynomial regression. *Engineering Computations*, *28*(4), 492–507.

Aminian, P., Javid, M. R., Asghari, A., Gandomi, A. H., & Esmaeili, M. A. (2011). A robust predictive model for base shear of steel frame structures using a hybrid genetic programming and simulated annealing method. *Neural Computing and Applications*, *20*(8), 1321.

Argyriou, A., Evgeniou, T., & Pontil, M. (2007). Multi-task feature learning. In B. Scholkopf, J. Platt, & T. Hofmann (Eds.), *Advances in neural information processing systems* (pp. 41–48). Cambridge, MA: MIT Press.

Argyriou, A., Evgeniou, T., & Pontil, M. (2008). Convex multi-task feature learning. *Machine Learning*, *73*(3), 243–272.

Cha, Y. J., Choi, W., & Büyüköztürk, O. (2017). Deep learning-based crack damage detection using convolutional neural networks. *Computer-Aided Civil and Infrastructure Engineering*, *32*(5), 361–378.

Cha, Y. J., Choi, W., Suh, G., Mahmoudkhani, S., & Büyüköztürk, O. (2018). Autonomous structural visual inspection using region-based deep learning for detecting multiple damage types. *Computer-Aided Civil and Infrastructure Engineering*, *33*(9), 731–747.

Cheng, M. Y., & Cao, M. T. (2014). Evolutionary multivariate adaptive regression splines for estimating shear strength in reinforced-concrete deep beams. *Engineering Applications of Artificial Intelligence*, *28*, 86–96.

Chou, J. S., & Pham, A. D. (2015). Smart artificial firefly colony algorithm-based support vector regression for enhanced forecasting in civil engineering. *Computer-Aided Civil and Infrastructure Engineering*, *30*(9), 715–732.

Cortes, C., & Mohri, M. (2014). Domain adaptation and sample bias correction theory and algorithm for regression. *Theoretical Computer Science*, *519*, 103–126.

Dai, W., Yang, Q., Xue, G., & Yu, Y. (2007). Boosting for transfer learning. In Z. Ghahramani (Ed.), *Proceedings of the 24th International conference on machine learning*, New York: Omni Press.

De Brabanter, K., Pelckmans, K., De Brabanter, J., Debruyne, M., Suykens, J. A., Hubert, M., & De Moor, B. (2009). Robustness of kernel based regression: A comparison of iterative weighting schemes. In C. Alippi M. Polycarpou C. Panayiotou & G. Ellinas (Eds.), *International conference on artificial neural networks* (pp. 100–110). Berlin, Heidelberg: Springer.

Drucker, H. (1997). Improving regressors using boosting techniques. In D. H. Fisher (Ed.), *ICML* (Vol. 97, pp. 107–115). San Francisco, CA: Morgan Kaufmann Publishers Inc.

Gandomi, A. H., Mohammadzadeh, D., Pérez-Ordóñez, J. L., & Alavi, A. H. (2014). Linear genetic programming for shear strength prediction of reinforced concrete beams without stirrups. *Applied Soft Computing*, *19*, 112–120.

Gao, Y., & Mosalam, K. M. (2018). Deep transfer learning for image-based structural damage recognition. *Computer-Aided Civil and Infrastructure Engineering*, *33*(9), 748–768.

Garcke, J., & Vanck, T. (2014). Importance weighted inductive transfer learning for regression. In T. Calders & F. Esposito (Eds.), *Joint European conference on machine learning and knowledge discovery in databases* (pp. 466–481). Berlin, Heidelberg: Springer.

Gretton, A., Smola, A., Huang, J., Schmittfull, M., Borgwardt, K., & Schölkopf, B. (2009). Covariate shift by kernel mean matching. *Dataset Shift in Machine Learning*, *3*(4), 5.

Han, J., Pei, J., & Kamber, M. (2011). *Data mining: Concepts and techniques*. Amsterdam: Elsevier.

Hua, J., Eberhard, M. O., Lowes, L. N., & Gu, X. (2019). Modes, mechanisms, and likelihood of seismic shear failure in rectangular reinforced concrete columns. *Journal of Structural Engineering*, *145*(10), 04019096.

Huang, J., Gretton, A., Borgwardt, K., Schölkopf, B., & Smola, A. J. (2007). Correcting sample selection bias by unlabeled data. In B. Scholkopf J. Platt & T. Hofmann (Eds.), *Advances in neural information processing systems* (pp. 601–608). Cambridge, MA: MIT Press.

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). *An introduction to statistical learning* (Vol. 112, pp. –). New York: Springer.

Jeon, J. S., Shafieezadeh, A., & DesRoches, R. (2014). Statistical models for shear strength of RC beam-column joints using machine-learning techniques. *Earthquake Engineering & Structural Dynamics*, *43*(14), 2075–2095.

Karbalayghareh, A., Qian, X., & Dougherty, E. R. (2018). Optimal Bayesian transfer regression. *IEEE Signal Processing Letters*, *25*(11), 1655–1659.

Luo, H., & Paal, S. G. (2018). Machine learning–based backbone curve model of reinforced concrete columns subjected to cyclic loading reversals. *Journal of Computing in Civil Engineering*, *32*(5), 04018042.

Luo, H., & Paal, S. G. (2019). A locally weighted machine learning model for generalized prediction of drift capacity in seismic vulnerability assessments. *Computer-Aided Civil and Infrastructure Engineering*, *34*(11), 935–950.

Moehle, J. (2014). *Seismic design of reinforced concrete buildings*. New York, NY: McGraw Hill Professional.

Mu, H. Q., & Yuen, K. V. (2015). Novel outlier-resistant extended Kalman filter for robust online structural identification. *Journal of Engineering Mechanics*, *141*(1), 04014100.

Pal, M., & Deswal, S. (2011). Support vector regression based shear strength modelling of deep beams. *Computers & Structures*, *89*(13-14), 1430–1439.

Pan, S. J., & Yang, Q. (2009). A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, *22*(10), 1345–1359.

Pardoe, D., & Stone, P. (2010). Boosting for regression transfer. In J. Fürnkranz & T. Joachims (Eds.), *Proceedings of the 27th International Conference on Machine Learning*, Madison, WI: Omni Press.

Quionero-Candela, J., Sugiyama, M., Schwaighofer, A., & Lawrence, N. D. (2009). *Dataset shift in machine learning*. Cambridge, MA: The MIT Press.

Rafiei, M. H., & Adeli, H. (2016). A novel machine learning model for estimation of sale prices of real estate units. *Journal of Construction Engineering and Management*, *142*(2), 04015066.

Rafiei, M. H., & Adeli, H. (2018). Novel machine learning model for construction cost estimation taking into account economic variables and indices. *Journal of Construction Engineering and Management*, *144*(12), 04018106.

Rafiei, M. H., Khushefati, W. H., Demirboga, R., & Adeli, H. (2017). Supervised deep restricted Boltzmann machine for estimation of concrete. *ACI Materials Journal*, *114*(2).237–244.

Reich, Y. (1997). Machine learning techniques for civil engineering problems. *Computer-Aided Civil and Infrastructure Engineering*, *12*(4), 295–310.

Rettinger, A., Zinkevich, M., & Bowling, M. (2006). Boosting expert ensembles for rapid concept recall. In A. Cohn (Eds.), *Proceedings of the National Conference on Artificial Intelligence* (Vol. 21, pp. 464). Menlo Park, CA, Cambridge, MA; London: AAAI Press.; 1999.

Rousseeuw, P. J., & Leroy, A. M. (1987). *Robust regression and outlier detection*. New York: Wiley.

Salaken, S. M., Khosravi, A., Nguyen, T., & Nahavandi, S. (2019). Seeded transfer learning for regression problems with deep learning. *Expert Systems with Applications*, *115*, 565–577.

Sugiyama, M., Nakajima, S., Kashima, H., Buenau, P. V., & Kawanabe, M. (2008). Direct importance estimation with model selection and its application to covariate shift adaptation. In J. C. Platt, D. Koller & S. T. Roweis (Eds.), *Advances in neural information processing systems* (pp. 1433–1440). Red Hook, NY: Curran Associates Inc.

Suykens, J. A., De Brabanter, J., Lukas, L., & Vandewalle, J. (2002). Weighted least squares support vector machines: Robustness and sparse approximation. *Neurocomputing*, *48*(1–4), 85–105.

Suykens, J., Van Gestel, T., De Brabanter, J., De Moor, B., & Vandewalle, J. (2002). *Least squares support vector machines*. Singapore: World Scientific.

Tan, B., Zhong, E., Xiang, E. W., & Yang, Q. (2014). Multi-transfer: Transfer learning with multiple views and multiple sources. *Statistical Analysis and Data Mining: The ASA Data Science Journal*. *4*(7), 282–293.

Weiss, K., Khoshgoftaar, T. M., & Wang, D. (2016). A survey of transfer learning. *Journal of Big Data*, *3*(1), 9.

Yuen, K. V., & Mu, H. Q. (2012). A novel probabilistic method for robust parametric identification and outlier detection. *Probabilistic Engineering Mechanics*, *30*, 48–59.

Yuen, K. V., & Ortiz, G. A. (2017). Outlier detection and robust regression for correlated data. *Computer Methods in Applied Mechanics and Engineering*, *313*, 632–646.

Yuen, K. V., Ortiz, G. A., & Huang, K. (2016). Novel nonparametric modeling of seismic attenuation and directivity relationship. *Computer Methods in Applied Mechanics and Engineering*, *311*, 537–555.

How to cite this article: Luo H, Paal SG. Reducing the effect of sample bias for small data sets with double-weighted support vector transfer regression. *Comput Aided Civ Inf*. 2021;36:248–263. https://doi.org/10.1111/mice.12617