Transfer Regression with Data-Augmented Ensemble Learning Framework

Linchun Li\textsuperscript{a}, Cheng Ma\textsuperscript{b}

Department of Precision Instrument Tsinghua University Beijing, China

\textsuperscript{a}lile14@mails.tsinghua.edu.cn, \textsuperscript{b}macheng@mail.tsinghua.edu.cn

Abstract. In biological measuring instruments design, it is time-consuming and expensive to acquire sufficient samples for calibration process. Transfer learning has provided an efficient approach for this problem by leveraging the labeled samples from calibration channels, considered as source domain, to annotate the target domain which has few labels under actual working conditions. In this paper, we design a general transfer regression framework improved by data-augmented ensemble learning called DA-TRTs. First, we modify two-stage TrAdaBoost.R2 to TrAdaBoost. RT (TRT) which concentrates more on hard examples during transfer boosting. Second, model stacking framework is introduced to improve the performance of regression, calling TRT as a base learner. Then, we introduce a simple data augmentation routine, based on label smoothing assumption. Data augmentation improves the strength in diversity of base learners and implicitly controls model complexity. Finally, to illustrate the performance of DA-TRTs, we conduct experiments on synthetic datasets.

1. Introduction
Transfer learning, as a branch of machine learning, has been extensively studied to leverage knowledge from related domains (source domains) and then apply it to target domain, which has few labeled samples [1]. The learning performance can be improved and the cost of labeling samples of target domain can be reduced, especially in the scene of biomedical and medical data analysis [2,3]. Many classic algorithms have been developed broadly covering the subfield of transfer learning solutions [4]. Considering homogeneous transfer learning, the transfer categories can be divided into subsections, such as instance-based, feature-based, model-based, and relational-based. First, we focus on instance-based transfer category to solve transfer regression problems. We find a series of work [5-7] dedicated to estimate the distribution ratio of source and target domains, which means to design a metric-based adjustment strategy of sample weight when training the learner. In particular, Dai [5] proposed the TrAdaBoost algorithm, which applies the idea of AdaBoost to transfer learning. TrAdaBoost increases the weights of instances that are conducive to the target classification task, and reduces the weight of instances that are unfavorable to the task, and it proves to be a high-performance and simple transfer learning algorithm according to experimental results. Pardoe et al. proposed two approaches to modify algorithms for a regression setting, which is using source models and using source data directly [8]. Then we try to combine these two points based on ensemble learning framework.

Many scholars have tried to combine transfer learning and ensemble learning to solve real-world problems [9-12]. These works prove that ensemble learning can improve the stability and accuracy of...
transfer learning algorithms. Zhou [13] described the implementation of common ensemble algorithm, such as bagging, boosting, stacking. Model stacking claims that meta models should behave with high accuracy and diversity [14]. Strong ensemble learning frameworks, such as GBDT, Random Forests, XGBoost, show amazing performance in Kaggle, a popular platform for predictive modelling and analytics competitions, and we use them as a comparison. Considering imbalanced distribution of source domain and target domain, SMOTE Bagging was proposed to increase diversity of meta models and control complexity of the full framework using data augmentation [15]. Instead of data augmentation using transform invariance, we focus on generator design of exploratory training cases [15-17]. Especially, Zhang et al. proposed a simple and data-agnostic data augmentation routine, termed mixup, which constructs virtual training examples that is convex combinations of pairs of original examples [20]. Considering a regression setting, this work can be promoted by label smoothing [21].

The main contribution of this paper is to promote an implementation of transfer regression and design a data-augmented ensemble learning framework that transfer regression algorithm is assembled in. The remainder of this paper is organized as follows. In Section II, we review methods related to our work. In Section III, we describe the details of our implementation of the framework. In Section IV, we conduct experiments on synthetic datasets. Section V concludes the whole paper and gives future work directions.

2. Background
In this section, we introduce the methods related to our work respectively.

2.1. Transfer Regression
AdaBoost.R2 is an ensemble learning framework that boosts a weak learner by carefully reweighting training instances according to the predict error in relation to the largest error each iteration [22]. Pardoe et al. applied the idea to transfer learning and proposed two-stage TrAdaBoost.R2 that overcomes the flaw that weights of the target data may be heavily skewed [8].

2.2. Ensemble Learning
According to SMOTE Bagging, majority class and minority class from the bootstrap process can be balanced by SMOTE before the learner [15]. A percentage value b\% is introduced to control the number of new generated instances, as well as two parameters in SMOTE: k-nearest neighbors and N-the total number of over-sampling from minority class in the resampling rate formula:

\[
\text{resampling rate} = \begin{cases} 
N = \left( \frac{N_{\text{majority}}}{N_{\text{minority}}} \right) \cdot (1 - b\%) \text{ by SMOTE}(k, N) 
\end{cases}
\]

(1)

2.3. Data Augmentation
SMOTE algorithm makes convex combinations of the raw inputs between nearest neighbors of the same class. In contrast, mixup makes convex combination of randomly drawn raw inputs pairs from the training set [30]. The routine can be implemented easily in the formula below where \(\lambda \sim \text{Beta}(a, a)\), for \(a \in (0, \infty)\). \((x_i, y_i)\) and \((x_j, y_j)\) are two feature-target vectors from training data at random.

\[
\mu(\bar{x}, \bar{y}) = \frac{1}{n} \sum_{k=1}^{n} E_k \{ \bar{\delta} (\bar{x} = \lambda \cdot x_i + (1 - \lambda) \cdot x_j, \bar{y} = \lambda \cdot y_i + (1 - \lambda) \cdot y_j) \}\]

(2)

3. Implementation of the framework
In this section, we promote the related work. The data distribution for transfer regression is a low dimensional manifold in contrast to image classification. We try to improve the transfer booster for
regression and use data augmentation by mixing source data and target data to control the complexity of the assembled base learners.

3.1. Transfer Booster (TRT)
When implementing transfer regression between a large amount of source data and little auxiliary target data, we divide the source and generated training samples into three kinds: hard samples, easy samples and outliers, according to the error in boosting iteration, as showed in figure 1.

Figure 1. hard samples, easy samples, outliers in transfer booster

In instance-based transfer learning, we need a method of mapping an error into an adjusted error as the metric of attention. AdaBoost.R2 expresses each error in relation to the largest error. The iteration is repeated until a preset number is constructed or error rate is higher than 0.5. It is not applicable to some datasets and it is possible that the performance on tails (hard samples or extreme events) may be improved without the limit of early stop. We modify AdaBoost.R2 into AdaBoost.RT in the two-stage framework by introducing the down threshold $\Phi_{\text{down}}$ of absolute relative error (ARE). Considering the outlier, we introduce the up threshold $\Phi_{\text{up}}$ to evaluate the several recent iterations and crop the training dataset by taking the sample weight of outliers into zero.

3.2. Data-augmented Model Stacking
Considering the datasets with distribution in a low dimensional manifold, we make several assumptions in the data augmentation process. First, similar data has the same label. Second, data under the same cluster has the same label. Third, data in the same manifold structure has the same label. We combine mixup and label smoothing for a setting of regression and get the generated samples following the routine below where $\sum W = 1$ ($w_{\text{neighbor}} \sim N(\lambda, 1)$ $w_{\text{sample}} \sim N(1 - \lambda, 1)$ before normalization process).

$$\text{Sample}^{*} = W \cdot \left[ \begin{array}{c} \text{neighbor} \\ \text{Sample} \end{array} \right]^{T} \left( \text{label} \rightarrow \text{S}\text{neighbor by KNN} \right)$$

(3)

We get different bootstrap by setting $\lambda \in (0, 1)$ increased by a linear gradient. We apply the two-stage TrAdaBoost. RT algorithm to the several bootstraps to get different sub-models for stacking. The base learner for TRT can be different if needed, and we simply to set decision tree regressor in contrast to the ensemble learning algorithms of trees.

When stacking, we compute the estimation of correlation between sub-models on the auxiliary target dataset in the formula below. Then we normalize the reciprocal of the sum to get the optimized weight vector of sub-models.

$$C\text{OR}E_{ij} = \frac{\sum_{x\in T}(f_{i}(x) - y(x))(f_{j}(x) - y(x))}{|T|}$$

(4)

3.3. Framework of DA-TRTs
The framework of DA-TRTs is showed below.
4. Experimental Study

In the following, we apply DA-TRTs into one synthetic dataset.

4.1. Synthetic Datasets

We consider Friedman #1, a well-known regression problem [24]. We get the instance \((X, y)\) in source dataset following the formula below where \(x_i \sim \text{random}[0, 1], a_i \sim \mathcal{N}(1, 0.1d), b_i \sim \mathcal{N}(1, 0.1d), c_i \sim \mathcal{N}(0, 0.05d)\).

\[
\begin{align*}
    y &= a_1 \cdot 10 \sin \left( \pi (b_1 x_1 + c_1) \cdot (b_2 x_2 + c_2) \right) + \\
    &\quad a_2 \cdot 20(b_3 x_3 + c_3 - 0.5)^2 + a_3 \cdot 10(b_4 x_4 + c_4) + \\
    &\quad a_4 \cdot 5(b_5 x_5 + c_5) + \mathcal{N}(0, 1) 
\end{align*}
\]  

We set \(a_i = 1, b_i = 1, c_i = 0\) for target data and test dataset. The parameter \(d\) is used to control the similarity of source and target data. We perform the experiment with the setting: \(d = 1, 2, 5\), target training dataset of size 100, target testing dataset of size 10000, and source dataset of size 1000. Decision tree regressor is chosen as the base learner and we compare the performance on test dataset using average RMSE results with ensemble learning framework, such as AdaBoost.R2, Random Forests, GBDT, XGBoost. The experiments on other datasets are following this.

\[
RMSE = \sqrt{\frac{\sum_{i=1}^{N} (y_{\text{train}} - y_{\text{model}})^2}{N}}  
\]  

4.2. Results and Analysis

The average RMSE results of different algorithms are in Table I below.

| Algorithm     | Synthetic datasets with different S-T similarities |
|---------------|-----------------------------------------------|
|               | \(d=1\)          | \(d=2\)          | \(d=5\)          |
| DA-TRTs       | 1.8554           | 2.0930           | 3.6255           |
| Decision tree | 3.2829           | 4.5059           | 11.2192          |
| AdaBoost.R2   | 1.8902           | 2.2139           | 4.2617           |
| GBDT          | 2.7543           | 2.8962           | 4.0616           |
| Random Forests| 2.3375           | 2.7591           | 4.2076           |
| XGBoost       | 1.8625           | 2.6426           | 4.0644           |

We find that the performance of DA-TRTs has a higher ranking over other algorithms. And the result in Table I shows that with the S-T similarity decreasing, DA-TRTs can maintain the effect of transfer regression.
5. Conclusion and Future Work
In this paper, we propose a transfer regression algorithm based on data-augmented ensemble learning framework. The primary contribution is that we try to combine instance-based transfer and model-based transfer in the framework and control the complexity sub-models by data augmentation. The experiments on several datasets show that it is effective and this work can be expanded in a number of areas.

Future work is needed to better understand the boosting strategy and feature fusion of stacking. Also, we will explore the influences of data augmentation on different transfer problems.

Acknowledgments
The study was supported granularity by the National Special Scientific Instrument and Equipment Development Projects, China (Grant No. 1YQ03013406).

References
[1] S. J. Pan and Q. Yang, “A Survey on Transfer Learning,” IEEE Transactions on Knowledge and Data Engineering, vol. 22, no. 10, pp. 1345–1359, Oct. 2010.
[2] K. M. Su, W. D. Hairston, and K. A. Robbins, “Adaptive Thresholding and Reweighting to Improve Domain Transfer Learning for Unbalanced Data with Applications to EEG Imbalance,” in 2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA), 2016, pp. 320–325.
[3] D. S. Kermany et al., “Identifying Medical Diagnoses and Treatable Diseases by Image-Based Deep Learning,” Cell, vol. 172, no. 5, pp. 1122-1131.e9, Feb. 2018.
[4] K. Weiss, T. M. Khoshgoftaar, and D. Wang, “A survey of transfer learning,” J Big Data, vol. 3, no. 1, p. 9, Dec. 2016.
[5] W. Dai, Q. Yang, G.-R. Xue, and Y. Yu, “Boosting for Transfer Learning,” in Proceedings of the 24th International Conference on Machine Learning, New York, NY, USA, 2007, pp. 193–200.
[6] B. Tan, Y. Song, E. Zhong, and Q. Yang, “Transitive Transfer Learning,” in Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, New York, NY, USA, 2015, pp. 1155–1164.
[7] B. Tan, Y. Zhang, S. J. Pan, and Q. Yang, “Distant domain transfer learning,” in Thirty-First AAAI Conference on Artificial Intelligence, 2017.
[8] D. Pardoe and P. Stone, “Boosting for Regression Transfer,” in Proceedings of the 27th International Conference on International Conference on Machine Learning, USA, 2010, pp. 863–870.
[9] S. Roy, H. S. Bhatt, and Y. Narahari, “An Iterative Transfer Learning Based Ensemble Technique for Automatic Short Answer Grading,” arXiv:1609.04909 [cs], Sep. 2016.
[10] J. Xiao, R. Wang, G. Teng, and Y. Hu, “A Transfer Learning Based Classifier Ensemble Model for Customer Credit Scoring,” in Proceedings of the 2014 Seventh International Joint Conference on Computational Sciences and Optimization, Washington, DC, USA, 2014, pp. 64–68.
[11] H. S. Bhatt, R. Singh, M. Vatsa, and N. K. Ratha, “Improving Cross-Resolution Face Matching Using Ensemble-Based Co-Transfer Learning,” IEEE Transactions on Image Processing, vol. 23, no. 12, pp. 5654–5669, Dec. 2014.
[12] S. Mei, “SVM ensemble based transfer learning for large-scale membrane proteins discrimination,” Journal of Theoretical Biology, vol. 340, pp. 105–110, Jan. 2014.
[13] Z.-H. Zhou, “Ensemble Learning,” in Encyclopedia of Biometrics, Springer, Boston, MA, 2009, pp. 270–273.
[14] M. P. Sesmero, A. I. Ledezma, and A. Sanchis, “Generating ensembles of heterogeneous classifiers using Stacked Generalization,” WIREs Data Mining Knowl Discov, vol. 5, no. 1, pp. 21–34, Jan. 2015.
[15] S. Wang and X. Yao, “Diversity analysis on imbalanced data sets by using ensemble models,” in 2009 IEEE Symposium on Computational Intelligence and Data Mining, 2009, pp. 324–331.

[16] A. Antoniou, A. Storkey, and H. Edwards, “Data Augmentation Generative Adversarial Networks,” arXiv:1711.04340 [cs, stat], Nov. 2017.

[17] T. Tran, T. Pham, G. Carneiro, L. Palmer, and I. Reid, “A Bayesian Data Augmentation Approach for Learning Deep Models,” arXiv:1710.10564 [cs], Oct. 2017.

[18] T. White, “Sampling Generative Networks,” arXiv:1609.04468 [cs, stat], Sep. 2016.

[19] A. Sinha, H. Namkoong, and J. Duchi, “Certifiable Distributional Robustness with Principled Adversarial Training,” arXiv:1710.10571 [cs, stat], Oct. 2017.

[20] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, “mixup: Beyond Empirical Risk Minimization,” arXiv:1710.09412 [cs, stat], Oct. 2017.

[21] G. Pereyra, G. Tucker, J. Chorowski, Ł. Kaiser, and G. Hinton, “Regularizing Neural Networks by Penalizing Confident Output Distributions,” arXiv:1701.06548 [cs], Jan. 2017.

[22] H. Drucker, “Improving Regressors Using Boosting Techniques,” in Proceedings of the Fourteenth International Conference on Machine Learning, San Francisco, CA, USA, 1997, pp. 107–115.

[23] T. DeVries and G. W. Taylor, “Dataset Augmentation in Feature Space,” arXiv:1702.05538 [cs, stat], Feb. 2017.

[24] J. H. Friedman, “Multivariate Adaptive Regression Splines,” The Annals of Statistics, vol. 19, no. 1, pp. 1–67, 1991.