Journal of Machine Learning Research 23 (2022) 1-4 Submitted 1/21; Revised 5/22; Published 9/22

Label Alignment for Multiclass Domain Adaptation

Vsevolod Ladtchenko VLADTCHE@UWATERLOO.CA
Department of Statistics
University of Waterloo
200 University Ave W, Waterloo ON N2L 3G1, Canada

Editor:

Abstract

TODO

Keywords: label alignment, domain adaptation

1 Introduction

Domain adaptation is the problem of training a model on one set of data, called the source data, and then applyng the trained model on a second set of data, called the target data. The reason for doing this is because in a supervised setting, we have labels for the source data, but we do not have labels for the target data. So a model trained using supervision from the source labels should extrapolate that knowledge to the target data, which has a different distribution. For example, letters drawn by one group of people have a different distribution than letters drawn by a different group of people. We have labels only for the first group, and after our model learns from the first group, we would like it to generalize to the second group. There is no general way to describe the transformation between these distributions. Previously used methods attempt to learn representations that are invariant to the transformation between distributions, and this works for specific cases, see Section 1 of Imani et al. (2022b).

The method we investigate is a novel approach that relies on a property of the dataset itself. A dataset has the *label alignment* property when the label vector is mostly in the span of the top singular vectors of the data matrix. This means that for a dataset with n samples and d features represented by a matrix of shape (n, d), we can take the Singular Value Decomposition (SVD) of the data matrix, project the label vector on the resulting d singular vectors, and find that a small number $k \ll d$ of singular vectors will contain a majority of the norm of the projection. This label alignment property emerges as a result of columns of the data matrix being correlated to the label vector (see Appendix A of Imani et al. (2022b)). This property also emerges in hidden representations of neural networks, meaning the label vector is in the span of the top few singular vectors of the SVD of a weight matrix of a hidden layer of a neural network Imani et al. (2022a). In particular this happens towards the topmost layers, showing that neural networks transform the input data until it is correlated to the label vector.

In a linear regression setting, we can show that the label alignment property implies a certain structure on the weights (later we will reverse this phenomenon by imposing this structure on the weights to force label alignment with the target domain). If we have a data

matrix $\mathbf{X}$ which has n data points, d features, and thus shape (n, d), and a label vector $\mathbf{y}$ of length n, the usual linear regression problem is to find weights to minimize the square error:

$$\begin{aligned}\mathbf{w}^* &= \arg\min_{\mathbf{w}} MSE(\mathbf{w}) \\ &= \arg\min_{\mathbf{w}} \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2\end{aligned}$$

First, replace $\mathbf{X}$ by its SVD. Then left-multiply by $\mathbf{U}^T$ which is a unitary matrix (a rotation), meaning it does not change the norm, nor its square, and $\mathbf{U}^T\mathbf{U} = \mathbf{I}$.

$$\begin{aligned}&= \arg\min_{\mathbf{w}} \|\mathbf{U}\mathbf{\Sigma}\mathbf{V}^T\mathbf{w} - \mathbf{y}\|^2 \\ &= \arg\min_{\mathbf{w}} \|\mathbf{\Sigma}\mathbf{V}^T\mathbf{w} - \mathbf{U}^T\mathbf{y}\|^2\end{aligned}$$

Since $\mathbf{w}$ is a vector of length d, and $\mathbf{V}$ is a basis for $\mathbb{R}^d$, then $\mathbf{V}^T\mathbf{w}$ is a projection of $\mathbf{w}$ on the basis spanned by $\mathbf{V}$. Since $\mathbf{\Sigma}$ is a diagonal matrix, we multiply the i^{th} element of $\mathbf{V}^T\mathbf{w}$, or $\mathbf{v}_i^T\mathbf{w}$, by σ_i. Similarly, $\mathbf{U}^T\mathbf{y}$ is a projection of $\mathbf{y}$ on the d singular vectors $\mathbf{u}_i$. Altogether, this is vector notation for the following sum:

$$= \sum_{i=1}^{d} (\sigma_i \mathbf{v}_i^T\mathbf{w} - \mathbf{u}_i\mathbf{y})^2$$

It is at this point that we invoke the label alignment property of $\mathbf{y}$. Because $\mathbf{y}$ is spanned mostly by the top k singular vectors $\mathbf{u}_i$, we have $\mathbf{u}_i\mathbf{y} \approx 0$ for $i > k$. This is approximate due to noise. Now we can remove $\mathbf{u}_i\mathbf{y}$ from the above sum for terms $i > k$, yielding:

$$= \sum_{i=1}^{k} (\sigma_i \mathbf{v}_i^T\mathbf{w} - \mathbf{u}_i\mathbf{y})^2 + \sum_{i=k+1}^{d} (\sigma_i \mathbf{v}_i^T\mathbf{w})^2$$

REWRITE: Keep in mind we got this by assuming the linear regression problem, and a dataset that follows the label alignment property. The second term does not involve the labels $\mathbf{y}$. Since we do not have labels for our target dataset, we can compute its SVD as $(\tilde{\mathbf{U}}, \tilde{\mathbf{\Sigma}}, \tilde{\mathbf{V}}^T)$ and derive a similar term:

$$\sum_{i=k+1}^{d} (\tilde{\sigma}_i \tilde{\mathbf{v}}_i^T\mathbf{w})^2$$

This is called the *label alignment regularizer*. It imposes structure on $\mathbf{w}$, which, based on the previous derivation, would again imply the linear regression problem and the label alignment property of the target dataset, except we do not need to know the label this time. [Imani 2] has shown promise in using this method to transfer knowledge from the source domain to the target domain. It only assumes the target domain has the label alignment property (which is fair if it is similar to the source domain), and a specific k' which may differ from k of the source domain.

For some derivations of k on real world datasets, see [Imani 2 Table 1]. The metric used there to derive k is to see how many vectors we need until the norm of the projection is at

least 0.9 (we normalize $\mathbf{y}$ so we can compare between datasets). If we define $\texttt{norm}_k(\mathbf{y})$ as the norm of the first k components of the vector, the metric can be expressed as:

$$\texttt{norm}_k(\mathbf{x}) = \sqrt{\sum_{i=1}^{k} x_i} \qquad k \leq \texttt{length}(\mathbf{x})$$

$$k^*(0.9) = \min_k \left\{ k \;\middle|\; \texttt{norm}_k(\mathbf{U^T y}) > 0.9 \right\}$$

Appendix A.

Appendix B.

References

Ehsan Imani, Wei Hu, and Martha White. Representation alignment in neural networks. *Transactions on Machine Learning Research*, 2022a. ISSN 2835-8856. URL https://openreview.net/forum?id=fLIWMnZ9ij.

Ehsan Imani, Guojun Zhang, Jun Luo, Pascal Poupart, and Yangchen Pan. Label alignment regularization for distribution shift, 2022b.