Minimax Lower Bounds for Transfer Learning with Linear and One-hidden Layer Neural Networks

Seyed Mohammadreza Mousavi Kalan, Zalan Fabian, Salman Avestimehr, and Mahdi Soltanolkotabi
Ming Hsieh Department of Electrical Engineering
University of Southern California
California, Los Angeles 90089
mmousavi@usc.edu,zfabian@usc.edu,avestimehr@ee.usc.edu,soltanol@usc.edu

Abstract

Transfer learning has emerged as a powerful technique for improving the performance of machine learning models on new domains where labeled training data may be scarce. In this approach a model trained for a *source* task, where plenty of labeled training data is available, is used as a starting point for training a model on a related *target* task with only few labeled training data. Despite recent empirical success of transfer learning approaches, the benefits and fundamental limits of transfer learning are poorly understood. In this paper we develop a statistical minimax framework to characterize the fundamental limits of transfer learning in the context of regression with linear and one-hidden layer neural network models. Specifically, we derive a lower-bound for the target generalization error achievable by any algorithm as a function of the number of labeled source and target data as well as appropriate notions of similarity between the source and target tasks. Our lowerbound provides new insights into the benefits and limitations of transfer learning. We further corroborate our theoretical finding with various experiments.

1 Introduction

Deep learning approaches have recently enjoyed wide empirical success in many applications spanning natural language processing to object recognition. A major challenge with deep learning techniques however is that training accurate models typically requires lots of labeled data. While for many of the aforementioned tasks labeled data can be collected by using crowd-sourcing, in many other settings such data collection procedures are expensive, time consuming, or impossible due to the sensitive nature of the data. Furthermore, deep learning techniques often are brittle and do not adapt well to changes in the data or the environment. Transfer learning approaches have emerged as a way to mitigate these issues. Roughly speaking, the goal of transfer learning is to borrow knowledge from a *source* domain, where lots of training data is available, to improve the learning process in a related but different *target* domain. Despite recent empirical success the benefits as well as fundamental limitations of transfer learning remains unclear with many open challenges:

What is the best possible accuracy that can be obtained via any transfer learning algorithm? How does this accuracy depend on how similar the source and target domain tasks are? What is a good way to measure similarity/distance between two source and target domains? How does the transfer learning accuracy scale with the number of source and target data? How do the answers to the above questions change for different learning models?

At the heart of answering these questions is the ability to predict the best possible accuracy achievable by any algorithm and characterize how this accuracy scales with how related the source and target data are as well as the number of labeled data in the source and target domains. In this paper we take

34th Conference on Neural Information Processing Systems (NeurIPS 2020), Vancouver, Canada.

a step towards this goal by developing statistical minimax lower bounds for transfer learning focusing on regression problems with linear and one-hidden layer neural network models. Specifically, we derive a minimax lower bound for the generalization error in the target task as a function of the number of labeled training data from source and target tasks. Our lower bound also explicitly captures the impact of the noise in the labels as well as an appropriate notion of *transfer distance* between source and target tasks on the target generalization error. Our analysis reveals that in the regime where the transfer distance between the source and target tasks is large (i.e. the source and target are dissimilar) the best achievable accuracy mainly depends on the number of labeled training data available from the target domain and there is a limited benefit to having access to more training data from the source domain. However, when the transfer distance between the source and target domains are small (i.e. the source and target are similar) both source and target play an important role in improving the target training accuracy. Furthermore, we provide various experiments on real data sets as well as synthetic simulations to empirically investigate the effect of the parameters appearing in our lower bound on the target generalization error.

Related work. There is a vast theoretical literature on the problem of domain adaptation which is closely related to transfer learning (1; 2; 3; 4; 5; 6; 7). The key difference is that in domain adaptation there is no labeled target data while in transfer learning a few labeled target data is available in addition to source data. Most of the existing results in the domain adaptation literature give an upper bound for the target generalization error. For instance, the papers (8; 9) provide an upper bound on the target generalization error in classification problems in terms of quantities such as source generalization error, the optimal joint error of source and target as well as VC-dimension of the hypothesis class. A more recent work (10) generalizes these results to a broad family of loss functions using Rademacher complexity measures. Related, (11) derives a similar upper bound for target generalization error as in (8) but in terms of other quantities. Finally, the recent paper (12) generalizes the results of (8; 10) to multiclass classification using margin loss.

More closely related to this paper, there are a few interesting results that provide lower bounds for the target generalization error. For instance, focusing on domain adaptation the paper (13) provides necessary conditions for successful target learning under a variety of assumptions such as a covariate shift, similarity of unlabeled distributions, and existence of a joint optimal hypothesis. More recently, the paper (14) defines a new discrepancy measure between source and target domains, called *transfer exponent*, and proves a minimax lower bound on the target generalization error under a relaxed covariate-shift assumption and a Bernstein class condition. (15) provides a minimax lower bound for a related multi-task learning setting in sparse linear regression. (11) derives an information theoretic lower bound on the joint optimal error of source and target domains defined in (8). Most of the above results are based on a covariate shift assumption which requires the conditional distributions of the source and target tasks to be equal and the source and target tasks to have the same best classifier. In this paper, however, we consider a more general case in which source and target tasks are allowed to have different optimal classifiers. Furthermore, these results do not specifically study a neural network model. To the extent of our knowledge this is the first paper to develop minimax lower bounds for transfer learning with neural networks.

2 Problem Setup

We now formalize the transfer learning problem considered in this paper. We begin by describing the linear and one-hidden layer neural network transfer learning regression models that we study. We then discuss the minimax approach to deriving transfer learning lower bounds.

2.1 Transfer Learning Models

We consider a transfer learning problem in which there are labeled training data from a source and a target task and the goal is to find a model that has good performance in the target task. Specifically, we assume we have n_S labeled training data from the source domain generated according to a source domain distribution $(\boldsymbol{x}_S, \boldsymbol{y}_S) \sim \mathbb{P}$ with $\boldsymbol{x}_S \in \mathbb{R}^d$ representing the input/feature and $\boldsymbol{y}_S \in \mathbb{R}^k$ the corresponding output/label. Similarly, we assume we have n_T training data from the target domain generated according to $(\boldsymbol{x}_T, \boldsymbol{y}_T) \sim \mathbb{Q}$ with $\boldsymbol{x}_T \in \mathbb{R}^d$ and $\boldsymbol{y}_T \in \mathbb{R}^k$. Furthermore, we assume that the features are distributed as $\boldsymbol{x}_S \sim \mathcal{N}(0, \boldsymbol{\Sigma_S})$, $\boldsymbol{x}_T \sim \mathcal{N}(0, \boldsymbol{\Sigma_T})$ with $\boldsymbol{\Sigma}_S$ and $\boldsymbol{\Sigma}_T \in \mathbb{R}^{d\times d}$ denoting the covariance matrices. We also assume that the labels $\boldsymbol{y}_S/\boldsymbol{y}_T$ are generated from ground truth mappings relating the features to the labels as follows

$$\boldsymbol{y}_S = f(\boldsymbol{\theta}_S; \boldsymbol{x}_S) + \boldsymbol{w}_S \quad \text{and} \quad \boldsymbol{y}_T = f(\boldsymbol{\theta}_T; \boldsymbol{x}_T) + \boldsymbol{w}_T \tag{2.1}$$

where $\boldsymbol{\theta}_S$ and $\boldsymbol{\theta}_T$ are the parameters of the function f and $\boldsymbol{w}_S, \boldsymbol{w}_T \sim \mathcal{N}(0, \sigma^2 \mathbf{I}_k)$ represents source/target label noise. In this paper we focus on the following linear and one-hidden layer neural network models.

Linear model. In this case, we assume that $f(\boldsymbol{\theta}_S; \boldsymbol{x}_S) := f(\boldsymbol{W}_S; \boldsymbol{x}_S) = \boldsymbol{W}_S \boldsymbol{x}_S$ and $f(\boldsymbol{\theta}_T; \boldsymbol{x}_T) := f(\boldsymbol{W}_T; \boldsymbol{x}_T) = \boldsymbol{W}_T \boldsymbol{x}_T$ where $\boldsymbol{W}_S, \boldsymbol{W}_T \in \mathbb{R}^{k \times d}$ are two unknown matrices denoting the source/target parameters. The goal is to use the source and target training data to find a parameter matrix $\widehat{\boldsymbol{W}}_T$ with estimated label $\widehat{\boldsymbol{y}}_T = \widehat{\boldsymbol{W}}_T \boldsymbol{x}_T$ that achieves the smallest risk/generalization error $\mathbb{E}[\|\boldsymbol{y}_T - \widehat{\boldsymbol{y}}_T\|_{\ell_2}^2]$.

One-hidden layer neural network models. We consider two different neural network models where in one the hidden-to-output layer is fixed and in the other the input-to-hidden layer is fixed. Specifically, in the first model, we assume that $f(\boldsymbol{\theta}_S; \boldsymbol{x}_S) := f(\boldsymbol{W}_S; \boldsymbol{x}_S) = \boldsymbol{V} \varphi(\boldsymbol{W}_S \boldsymbol{x}_S)$ and $f(\boldsymbol{\theta}_T; \boldsymbol{x}_T) := f(\boldsymbol{W}_T; \boldsymbol{x}_T) = \boldsymbol{V} \varphi(\boldsymbol{W}_T \boldsymbol{x}_T)$ where $\boldsymbol{W}_S, \boldsymbol{W}_T \in \mathbb{R}^{\ell \times d}$ are two unknown weight matrices, $\boldsymbol{V} \in \mathbb{R}^{k \times \ell}$ is a fixed and known matrix, and φ is the ReLU activation function. Similarly in the second model, we assume that $f(\boldsymbol{\theta}_S; \boldsymbol{x}_S) := f(\boldsymbol{V}_S; \boldsymbol{x}_S) = \boldsymbol{V}_S \varphi(\boldsymbol{W} \boldsymbol{x}_S)$ and $f(\boldsymbol{\theta}_T; \boldsymbol{x}_T) := f(\boldsymbol{V}_T; \boldsymbol{x}_T) = \boldsymbol{V}_T \varphi(\boldsymbol{W} \boldsymbol{x}_T)$ with $\boldsymbol{V}_S, \boldsymbol{V}_T \in \mathbb{R}^{k \times \ell}$ two unknown weight matrices and $\boldsymbol{W} \in \mathbb{R}^{\ell \times d}$ a known matrix. In both cases the goal is to use the source and target training data to find the unknown target parameter weights ($\widehat{\boldsymbol{W}}_T$ or $\widehat{\boldsymbol{V}}_T$) that achieve the smallest risk/generalization error $\mathbb{E}[\|\boldsymbol{y}_T - \widehat{\boldsymbol{y}}_T\|_{\ell_2}^2]$. Here, $\widehat{\boldsymbol{y}}_T = \boldsymbol{V} \varphi(\widehat{\boldsymbol{W}}_T \boldsymbol{x}_T)$ in the first model and $\widehat{\boldsymbol{y}}_T = \widehat{\boldsymbol{V}}_T \varphi(\boldsymbol{W} \boldsymbol{x}_T)$ in the second.

2.2 Minimax Framework for Transfer Learning

We now describe our minimax framework for developing lower bounds for transfer learning. As with most lower bounds, in a minimax framework we need to define a class of transfer learning problems for which the lower bound is derived. Therefore, we define $(\mathbb{P}_{\boldsymbol{\theta}_S}, \mathbb{Q}_{\boldsymbol{\theta}_T})$ as a pair of joint distributions of features and labels over a source and a target task, that is, $(\boldsymbol{x}_S, \boldsymbol{y}_S) \sim \mathbb{P}_{\boldsymbol{\theta}_S}$ and $(\boldsymbol{x}_T, \boldsymbol{y}_T) \sim \mathbb{Q}_{\boldsymbol{\theta}_T}$ with the labels obeying (2.1). In this notation, each pair of a source and target task is parametrized by $\boldsymbol{\theta}_S$ and $\boldsymbol{\theta}_T$. We stress that over the different pairs of source and target tasks, $\boldsymbol{\Sigma}_S, \boldsymbol{\Sigma}_T$, and σ^2 are fixed and only the parameters $\boldsymbol{\theta}_S$ and $\boldsymbol{\theta}_T$ change.

As mentioned earlier, in a transfer learning problem we are interested in using both source and target training data to find an estimate $\widehat{\boldsymbol{\theta}}_T$ of $\boldsymbol{\theta}_T$ with small target generalization error. In a minimax framework, $\boldsymbol{\theta}_T$ is chosen in an adversarial way, and the goal is to find an estimate $\hat{\boldsymbol{\theta}}_T$ that achieves the smallest worst case target generalization risk $\sup \mathbb{E}_{\sim \text{ source and target samples}} \left[\mathbb{E}_{\mathbb{Q}_{\boldsymbol{\theta}_T}} [\|\boldsymbol{y}_T - \widehat{\boldsymbol{y}}_T\|_{\ell_2}^2] \right]$. Here, the supremum is taken over the class of transfer problems under study (possible $(\mathbb{P}_{\boldsymbol{\theta}_S}, \mathbb{Q}_{\boldsymbol{\theta}_T})$ pairs). We are interested in considering classes of transfer learning problems which properly reflect the difficulty of transfer learning. To this aim we need to have an appropriate notion of similarity or *transfer distance* between source and target tasks. To define the appropriate measure of transfer distance we are guided by the following proposition (see Section 7.1 for the proof) which characterizes the target generalization error for linear and one-hidden layer neural network models.

Proposition 1 *Let $\mathbb{Q}_{\boldsymbol{\theta}_T}$ be the data distribution over the target task with parameter $\boldsymbol{\theta}_T$ according to one of the models defined in Section 2.2. The target generalization error of an estimated model with parameter $\widehat{\boldsymbol{\theta}}_T$ is given by:*

- *Linear model:*

$$\mathbb{E}_{\mathbb{Q}_{\boldsymbol{\theta}_T}} [\|\widehat{\boldsymbol{y}}_T - \boldsymbol{y}_T\|_{\ell_2}^2] = \|\boldsymbol{\Sigma}_T^{\frac{1}{2}} (\widehat{\boldsymbol{W}}_T - \boldsymbol{W}_T)^T\|_F^2 + k\sigma^2 \tag{2.2}$$

- *One-hidden layer neural network model with fixed hidden-to-output layer:*

$$\mathbb{E}_{\mathbb{Q}_{\boldsymbol{\theta}_T}} [\|\widehat{\boldsymbol{y}}_T - \boldsymbol{y}_T\|_{\ell_2}^2] \geq \frac{1}{4} \sigma_{min}^2(\boldsymbol{V}) \|\boldsymbol{\Sigma}_T^{\frac{1}{2}} (\widehat{\boldsymbol{W}}_T - \boldsymbol{W}_T)^T\|_F^2 + k\sigma^2 \tag{2.3}$$

- *One-hidden layer neural network model with fixed input-to-hidden layer:*

$$\mathbb{E}_{\mathbb{Q}_{\boldsymbol{\theta}_T}} [\|\widehat{\boldsymbol{y}}_T - \boldsymbol{y}_T\|_{\ell_2}^2] = \|\widetilde{\boldsymbol{\Sigma}}_T^{\frac{1}{2}} (\widehat{\boldsymbol{V}}_T - \boldsymbol{V}_T)^T\|_F^2 + k\sigma^2 \tag{2.4}$$

Here, $\widetilde{\boldsymbol{\Sigma}}_T := \left[\frac{1}{2} \|\boldsymbol{a}_i\|_{\ell_2} \|\boldsymbol{a}_j\|_{\ell_2} \frac{\sqrt{1-\gamma_{ij}^2} + (\pi - \cos^{-1}(\gamma_{ij}))\gamma_{ij}}{\pi} \right]_{ij}$ where $\boldsymbol{a}_i$ is the ith row of the matrix $\boldsymbol{W} \boldsymbol{\Sigma}_T^{\frac{1}{2}}$ and $\gamma_{ij} := \frac{\boldsymbol{a}_i^T \boldsymbol{a}_j}{\|\boldsymbol{a}_i\|_{\ell_2} \|\boldsymbol{a}_j\|_{\ell_2}}$.

Proposition 1 essentially shows how the generalization error is related to an appropriate distance between the estimated and ground truth parameters. This in turn motivates our notion of transfer distance/similarity between source and target tasks discussed next.

Definition 1 *(Transfer distance) For a source and target task generated according to one of the models in Section 2.2 parametrized by $\boldsymbol{\theta}_S$ and $\boldsymbol{\theta}_T$, we define the transfer distance between these two tasks as follows:*

- *Linear model and one-hidden layer neural network model with fixed hidden-to-output layer:*

$$\rho(\boldsymbol{\theta}_S, \boldsymbol{\theta}_T) = \rho(\boldsymbol{W}_S, \boldsymbol{W}_T) := \|\boldsymbol{\Sigma}_T^{\frac{1}{2}}(\boldsymbol{W}_S - \boldsymbol{W}_T)^T\|_F \tag{2.5}$$

- *One-hidden layer neural network model with fixed input-to-hidden layer:*

$$\rho(\boldsymbol{\theta}_S, \boldsymbol{\theta}_T) = \rho(\boldsymbol{V}_S, \boldsymbol{V}_T) := \|\widetilde{\boldsymbol{\Sigma}}_T^{\frac{1}{2}}(\boldsymbol{V}_S - \boldsymbol{V}_T)^T\|_F \tag{2.6}$$

where $\widetilde{\Sigma}_T$ is defined in Proposition 1.

With the notion of transfer distance in hand we are now ready to formally define the class of pairs of distributions over source and target tasks which we focus on in this paper.

Definition 2 *(Class of pairs of distributions) For a given $\Delta \in \mathbb{R}^+$, $\mathcal{P}_\Delta$ is the class of pairs of distributions over source and target tasks whose transfer distance according to Definition 1 is less than Δ. That is, $\mathcal{P}_\Delta = \{(\mathbb{P}_{\boldsymbol{\theta}_S}, \mathbb{Q}_{\boldsymbol{\theta}_T}) |\ \rho(\boldsymbol{\theta}_S, \boldsymbol{\theta}_T) \le \Delta\}$.*

With these ingredients in place we are now ready to formally state the transfer learning minimax risk.

$$\mathcal{R}_T(\mathcal{P}_\Delta) := \inf_{\widehat{\boldsymbol{\theta}}_T} \sup_{(\mathbb{P}_{\boldsymbol{\theta}_S}, \mathbb{Q}_{\boldsymbol{\theta}_T}) \in \mathcal{P}_\Delta} \mathbb{E}_{S_{\mathbb{P}_{\boldsymbol{\theta}_S}} \sim \mathbb{P}_{\boldsymbol{\theta}_S}^{1:n_S}} \left[\mathbb{E}_{S_{\mathbb{Q}_{\boldsymbol{\theta}_T}} \sim \mathbb{Q}_{\boldsymbol{\theta}_T}^{1:n_T}} \left[\mathbb{E}_{\mathbb{Q}_{\boldsymbol{\theta}_T}} [\|\boldsymbol{y}_T - \widehat{\boldsymbol{y}}_T\|_{\ell_2}^2] \right] \right] \tag{2.7}$$

Here, $S_{\mathbb{P}_{\boldsymbol{\theta}_S}}$ and $S_{\mathbb{Q}_{\boldsymbol{\theta}_T}}$ denote i.i.d. samples $\{(\boldsymbol{x}_S^{(i)}, \boldsymbol{y}_S^{(i)})\}_{i=1}^{n_S}$ and $\{(\boldsymbol{x}_T^{(i)}, \boldsymbol{y}_T^{(i)})\}_{i=1}^{n_T}$ generated from the source and target distributions. We would like to emphasize that $\widehat{y}_T$ as defined in section 2.1, is a function of samples $(S_{\mathbb{P}_{\boldsymbol{\theta}_S}}, S_{\mathbb{Q}_{\boldsymbol{\theta}}})$.

3 Main Results

In this section, we provide a lower bound on the transfer learning minimax risk (2.7) for the three transfer learning models defined in Section 2.1. As with any other quantity related to generalization error this risk naturally depends on the size of the model and how correlated the features are in the target model. The following definition aims to capture the effective number of parameters of the model.

Definition 3 *(Effective dimension) The effective dimension of the three models defined in Section 2.1 are defined as follows:*

- *Linear model: $D := rank(\boldsymbol{\Sigma}_T)k - 1$,*
- *One-hidden layer neural network model with fixed hidden-to-output layer: $D := rank(\boldsymbol{\Sigma}_T)\ell - 1$,*
- *One-hidden layer neural network model with fixed input-to-hidden layer: $D := rank(\widetilde{\boldsymbol{\Sigma}}_T)k - 1$.*

Our results also depend on another quantity which we refer to as the transfer coefficient. Roughly speaking these quantities are meant to capture the relative effectiveness of a source training data from the perspective of the generalization error of the target task and vice versa.

Definition 4 *(Transfer coefficients) Let n_S and n_T be the number of source and target training data. We define the transfer coefficients in the three models defined in Section 2.1 as follows*

- *Linear model: $r_S := \left\| \boldsymbol{\Sigma}_S^{\frac{1}{2}} \boldsymbol{\Sigma}_T^{-\frac{1}{2}} \right\|^2$ and $r_T := 1$.*
- *One-hidden layer neural net with fixed output layer: $r_S := \left\| \boldsymbol{\Sigma}_S^{\frac{1}{2}} \boldsymbol{\Sigma}_T^{-\frac{1}{2}} \right\|^2 \|\boldsymbol{V}\|^2$ and $r_T := \|\boldsymbol{V}\|^2$.*
- *One-hidden layer neural net model with fixed input layer: $r_S := \left\| \widetilde{\boldsymbol{\Sigma}}_S^{\frac{1}{2}} \widetilde{\boldsymbol{\Sigma}}_T^{-\frac{1}{2}} \right\|^2$ and $r_T := 1$. Here, $\widetilde{\boldsymbol{\Sigma}}_S := \left[\frac{1}{2} \|\boldsymbol{c}_i\|_{\ell_2} \|\boldsymbol{c}_j\|_{\ell_2} \frac{\sqrt{1-\widetilde{\gamma}_{ij}^2} + (\pi - \cos^{-1}(\widetilde{\gamma}_{ij}))\widetilde{\gamma}_{ij}}{\pi} \right]_{ij}$ where $\boldsymbol{c}_i$ is the ith row of $\boldsymbol{W}\boldsymbol{\Sigma}_S^{\frac{1}{2}}$ and $\widetilde{\gamma}_{ij} = \frac{\boldsymbol{c}_i^T \boldsymbol{c}_j}{\|\boldsymbol{c}_i\|_{\ell_2} \|\boldsymbol{c}_j\|_{\ell_2}}$ and $\widetilde{\boldsymbol{\Sigma}}_T$ are defined per Proposition 1.*

In the above expressions $\|\cdot\|$ stands for the operator norm. Furthermore, we define the effective number of source and target samples as $r_S n_S$ and $r_T n_T$, respectively.

With these definitions in place we now present our lower bounds on the transfer learning minimax risk of any algorithm for the linear and one-hidden layer neural network models (see sections 6 and 7 for the proof).

Theorem 1 *Consider the three transfer learning models defined in Section 2.1 consisting of n_S and n_T source and target training data generated i.i.d. according to a class of source/target distributions with transfer distance at most Δ per Definition 2. Moreover, let r_S and r_T be the source and target transfer coefficients per Definition 4. Furthermore, assume the effective dimension D per Definition 3 obeys $D \geq 20$. Then, the transfer learning minimax risk (2.7) obeys the following lower bounds:*

- *Linear model: $\mathcal{R}_T(\mathcal{P}_\Delta) \geq B + k\sigma^2$.*
- *One-hidden layer neural network with fixed hidden-to-output layer: $\mathcal{R}_T(\mathcal{P}_\Delta) \geq \frac{1}{4}\sigma_{min}^2(\boldsymbol{V})B + k\sigma^2$.*
- *One-hidden layer neural network model with fixed input-to-hidden layer: $\mathcal{R}_T(\mathcal{P}_\Delta) \geq B + k\sigma^2$.*

Here, $\sigma_{min}(\boldsymbol{V})$ denotes the minimum singular value of $\boldsymbol{V}$ and

$$B := \begin{cases} \frac{\sigma^2 D}{256 r_T n_T}, & \text{if } \Delta \geq \sqrt{\frac{\sigma^2 D \log 2}{r_T n_T}} \\ \frac{1}{100}\Delta^2\left[1 - 0.8\frac{r_T n_T \Delta^2}{\sigma^2 D}\right], & \text{if } \frac{1}{45}\sqrt{\frac{\sigma^2 D}{r_S n_S + r_T n_T}} \leq \Delta < \sqrt{\frac{\sigma^2 D \log 2}{r_T n_T}} \\ \frac{\Delta^2}{1000} + \frac{6}{1000}\frac{D\sigma^2}{r_S n_S + r_T n_T}, & \text{if } \Delta < \frac{1}{45}\sqrt{\frac{\sigma^2 D}{r_S n_S + r_T n_T}} \end{cases} \tag{3.1}$$

Note that, the nature of the lower bound and final conclusions provided by the above theorem are similar for all three models. More specifically, Theorem 1 leads to the following conclusions:

- **Large transfer distance** ($\Delta \geq \sqrt{\frac{D\sigma^2 \log 2}{r_T n_T}}$). When the transfer distance between the source and target tasks is large, source samples are helpful in decreasing the target generalization error until the error reaches $\frac{\sigma^2 D}{256 r_T n_T}$. Beyond this point, by increasing the number of source samples, target generalization error does not decrease further and it becomes dominated by the target samples. In other words, when the distance is large, source samples cannot compensate for target samples.
- **Moderate distance** ($\frac{1}{45}\sqrt{\frac{\sigma^2 D}{r_S n_S + r_T n_T}} \leq \Delta < \sqrt{\frac{\sigma^2 D \log 2}{r_T n_T}}$). The lower bound of this regime suggests that if the distance between the source and target tasks is strictly positive, i.e $\Delta > 0$, even if we have infinitely many source samples, target generalization error still does not go to zero and depends on the number of available target samples. In other words, source samples cannot compensate for the lack of target samples.
- **Small distance** ($\Delta < \frac{1}{45}\sqrt{\frac{\sigma^2 D}{r_S n_S + r_T n_T}}$). In this case, the lower bound on the target generalization error scales with $\frac{1}{r_S n_S + n_T r_T}$ where $r_S n_S$ and $r_T n_T$ are the effective number of source and target samples per Definition 4. Hence, when Δ is small, the target generalization error scales with the reciprocal of the total effective number of source and target samples which means that source samples are indeed helpful in reducing the target generalization error and every source sample is roughly equivalent to $\frac{r_S}{r_T}$ target samples. Furthermore, when the distance of source and target is zero, i.e. $\Delta = 0$, the lower bound reduces to $\frac{6}{1000}\frac{D\sigma^2}{r_S n_S + r_T n_T}$. Conforming with our intuition, in this case the bound resembles a non-transfer learning scenario where a combination of source and target samples are used. Indeed, the lower bound is proportional to the noise level, effective dimension and the total number of samples matching typical statistical learning lower bounds.

4 Experiments and Numerical Results

We demonstrate the validity of our theoretical framework through experiments on real datasets sampled from ImageNet as well as synthetic simulated data. The experiments on ImageNet data allow us to investigate the impact of transfer distance and noise parameters appearing in Theorem 1 on the target generalization error. However, since the source and target tasks are both image classification, they are inherently correlated with each other and we cannot expect a wide range of transfer distances between them. Therefore, we carry out a more in-depth study on simulated data to investigate the effect of the number of source and target samples on the target generalization error in different transfer distance regimes. Full source code to reproduce the results is available at (16).

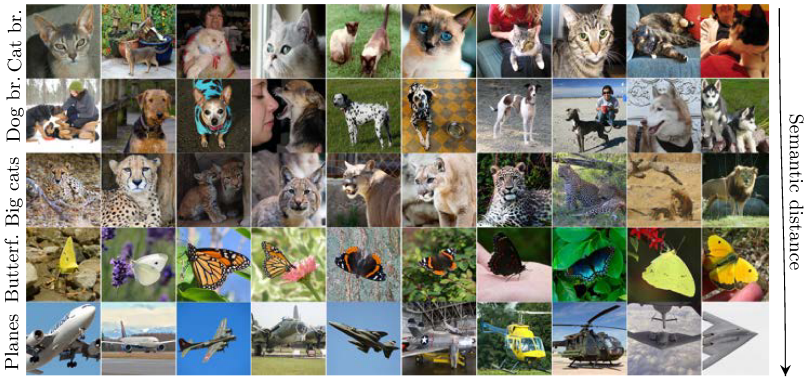


Figure 1: Sample images from the source/target datasets derived from ImageNet. Transfer distance increases from top to bottom.

Source / target task	$\rho(source, target)$	Validation loss	Noise level (σ)
cat breeds / *dog breeds*	11.62	0.2194	0.2095
cat breeds / *big cats*	12.35	0.1682	0.1834
cat breeds / *butterflies*	13.48	0.1367	0.1653
cat breeds / *planes*	16.41	0.1450	0.1703

Table 1: Transfer distance and noise level for various source-target pairs.

4.1 ImageNet Experiments

Here we verify our theoretical formulation on a subset of ImageNet, a well-known image classification dataset and show that our main theorem conforms with practical transfer learning scenarios.

Sample datasets. We create five datasets by sub-sampling 2000 images in five classes from ImageNet (400 examples per class). As depicted in Figure 1, we deliberately compile datasets covering a spectrum of semantic distances from each other in order to study the utility/effect of transfer distance on transfer learning. The picked datasets are as follows: *cat breeds*, *big cats*, *dog breeds*, *butterflies*, *planes*. For details of the classes in each dataset please refer to (16). We pass the images through a VGG16 network pretrained on ImageNet with the fully connected top classifier removed and use the extracted features instead of the actual images. We set aside 10% of the dataset as test set. Furthermore, 10% of the remaining data is used for validation and 90% for training. In the following we fix identifying *cat breeds* as the source task and the four other datasets as target tasks.

Training. We trained a one-hidden layer neural network for each dataset. To facilitate fair comparison between the trained models in weight-space, we fixed a random initialization of the hidden-to-output layer shared between all networks and we only trained over the input-to-hidden layer (in accordance with the theoretical formulation). Moreover, we used the same initialization of input-to-hidden weights. We trained a separate model on each of the five datasets on MSE loss with one-hot encoded labels. We use an Adam optimizer with a learning rate of 0.0001 and train for 100 epochs or until the network reaches 99.5% accuracy whichever occurs first. The target noise levels are calculated based on the average loss of the trained ground truth models on the target validation set (note that this average loss equals $k\sigma^2 = 5\sigma^2$).

Results. First, we calculate the transfer distance from Definition 1 between the model trained on the source task (*cat breeds*) and the other four models trained on target tasks by fitting a ground truth model to each task using complete training data. Our results depicted in Table 1 demonstrate that the introduced metric strongly correlates with perceived semantic distance. The closest tasks, *cat breeds* and *dog breeds*, are both characterized by pets with similar features, and with humans frequently appearing in the images. Images in the second closest pair, *cat breeds* and *big cats*, include animals with similar features, but *big cats* have more natural scenes and less humans compared with *dog breeds*, resulting in slightly higher distance from the source task. As expected, *cat breeds*-*butterflies* distance is significantly higher than in case of the previous two targets, but they share some characteristics such as the presence of natural backgrounds. The largest distance is between *cat breeds* and *planes*, which is clearly the furthest task semantically as well.

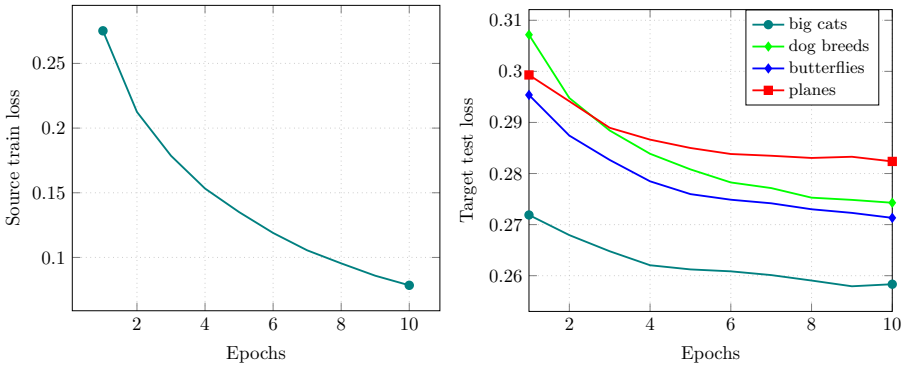


Figure 2: Train and test loss of a one-hidden layer network trained on *cat breeds* dataset.

Our next set of experiments focuses on checking whether the transfer distance is indicative of transfer target risk/generalization error. To this aim we use a very simple transfer learning approach where we use only source data to train a one-hidden layer network as described before and measure its performance on the target tasks. Note that the network has never seen examples of the target dataset. Figure 2 depicts how train and test loss evolved over the training process. We stop after 10 epochs when validation losses on target tasks have more or less stabilized. The results closely match our expectations from Theorem 1. Based on Table 1 the noise level of ground truth models for *big cats*, *butterflies* and *planes* are about the same and therefore their test loss follows the same ordering as their distances from the source task (see Table 1). Moreover, even though *dog breeds* has the lowest distance from the source task, it is also the noisiest. The lower bound in Theorem 1 includes an additive noise term, and therefore the change in ordering between *dog breeds* and *butterflies* is justified by our theory and demonstrates the effect of the target task noise level on generalization.

Theoretical lower bounds for the ImageNet experiments. In order to plot the theoretical lower bounds, first we estimate the parameters appearing in the bounds. Then using those parameters we depict the lower bounds in Figure 3. In Figure 3 each plot consists of two lower bounds, namely a crude bound (presented in Theorem 1) and a more precise bound presented in the proofs.

4.2 Numerical Results

In this section we perform synthetic numerical simulations in order to carefully cover all regimes of transfer distance from our main theorem, and show how the target generalization error depends on the number of source and target samples in different regimes.

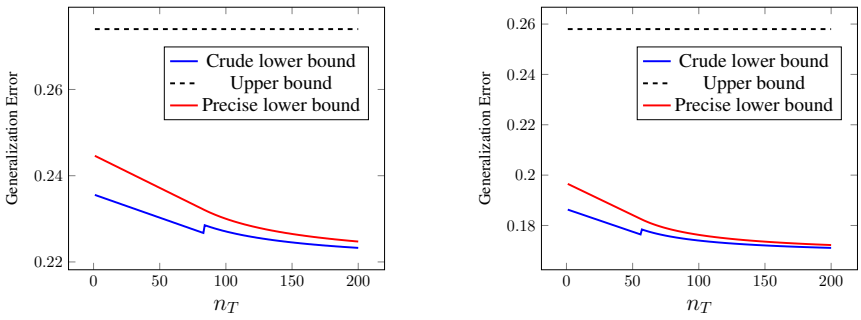
Experimental setup 1. First, we generate data according to the linear model with parameters $d = 200, k = 30, \sigma = 1, \mathbf{\Sigma}_S = 2 \cdot I_d, \mathbf{\Sigma}_T = I_d$. Then we generate the source parameter matrix $\boldsymbol{W}_S \in \mathbb{R}^{k \times d}$ with elements sampled from $\mathcal{N}(0, 10)$. Furthermore, we generate two target parameter matrices $\boldsymbol{W}_{T_1}$ and $\boldsymbol{W}_{T_2} \in \mathbb{R}^{k \times d}$ for tasks T_1 and T_2 such that $\boldsymbol{W}_{T_1} = \boldsymbol{W}_S + \boldsymbol{M}_1$ and $\boldsymbol{W}_{T_2} = \boldsymbol{W}_S + \boldsymbol{M}_2$ where the elements of $\boldsymbol{M}_1, \boldsymbol{M}_2$ are sampled from $\mathcal{N}(0, 10^{-3})$ and $\mathcal{N}(0, 3.6 \times 10^5)$ respectively. Similarly for the one-hidden layer neural network model when the the output layer is fixed, we set the parameters $k = 1, \ell = 30, d = 200, \sigma = 1, \mathbf{\Sigma}_S = 2 \cdot I_d, \mathbf{\Sigma}_T = I_d$ and $V = \mathbf{1}_{k \times \ell}$. We also use the same $\boldsymbol{W}_S, \boldsymbol{W}_{T_1}, \boldsymbol{W}_{T_2}$ as in the linear model. We note that the transfer distance between the source task to target task T_1 is small but the transfer distance between the source task to target task T_2 is large ($\rho(\boldsymbol{W}_S, \boldsymbol{W}_{T_1}) = .0183$ and $\rho(\boldsymbol{W}_S, \boldsymbol{W}_{T_2}) = 116.694$).

Training approach 1. We test the performance of a simple transfer learning approach. Given n_S source samples and n_T target samples, we estimate $\widehat{\boldsymbol{W}}_T$ by minimizing the weighted empirical risk

$$\min_{\boldsymbol{W}} \frac{1}{2n_T} \sum_{i=1}^{n_T} \left\| f(\boldsymbol{W}; \boldsymbol{x}_T^{(i)}) - \boldsymbol{y}_T^{(i)} \right\|_{\ell_2}^2 + \frac{\lambda}{2n_S} \sum_{j=1}^{n_S} \left\| f(\boldsymbol{W}; \boldsymbol{x}_S^{(j)}) - \boldsymbol{y}_S^{(j)} \right\|_{\ell_2}^2 \tag{4.1}$$

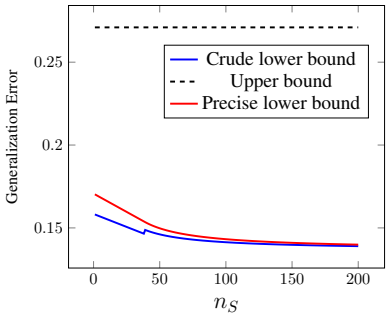
We then evaluate the generalization error by testing the estimated model $\widehat{\boldsymbol{W}}_T$ on 200 unseen test data points generated by the target model. All reported plots are the average of 10 trials.

Results 1. Figure 4 (a) depicts the target generalization error for target tasks T_1 and T_2 for the linear model for different n_S values with $\lambda = 1$ and $n_T = 50$. Figure 4 (b) depicts the target generalization error for tasks T_1 and target T_2 for the linear model for different n_T values with the number of source

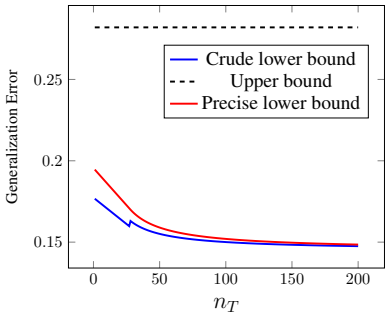


(a) cat breeds / dog breeds

(b) cat breeds / big cats

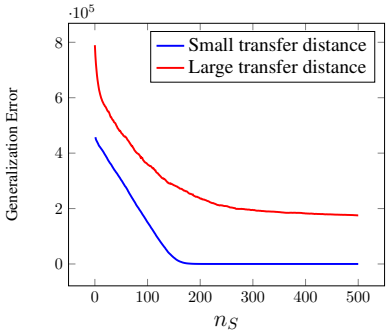


(c) cat breeds / butterflies

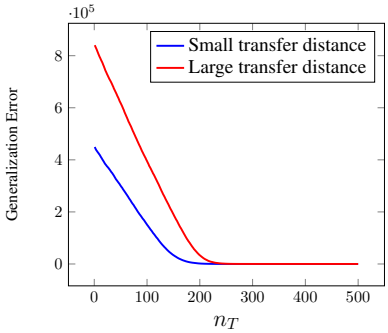


(d) cat breeds / planes

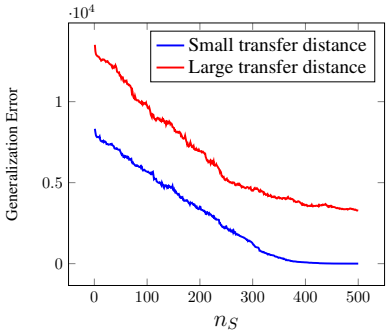
Figure 3: Theoretical lower bounds and experimental upper bounds.



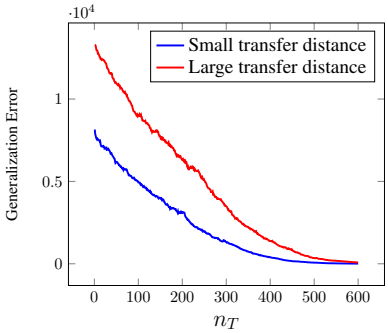
(a) Linear model with $n_T = 50$



(b) Linear model with $n_S = 50$

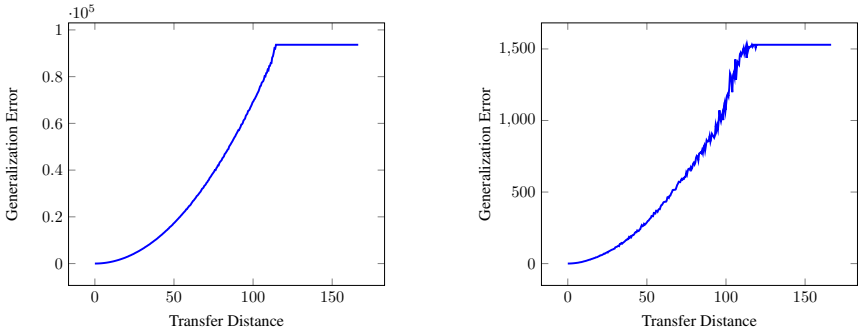


(c) Neural network model with $n_T = 50$



(d) Neural network model with $n_S = 50$

Figure 4: Target generalization error for a linear model ((a) and (b)) and a neural network model with fixed hidden-to-output layer ((c) and (d)).



(a) Linear model with $n_S = 300$ and $n_T = 20$

(b) Neural network model with $n_S = 300$ and $n_T = 20$

Figure 5: Target generalization error for a linear model (a) and a neural network model with fixed hidden-to-output layer (b).

samples fixed at $n_S = 50$. Here, we set $\lambda = 1$ for target task T_1, where the transfer distance from source is small, and $\lambda = .001$ for target task T_2, where the transfer distance from source is large. Figures 4 (c) and 4 (d) have the same settings as in Figures 4 (a) and 4 (b) but we use a one-hidden layer neural network model with fixed hidden-to-output weights in lieu of the linear model.

Figures 4 (a) and (c) clearly demonstrate that when the transfer distance between the source and target tasks is large, increasing the number of source samples is not helpful beyond a certain point. In particular, the target generalization error starts to saturate and does not decrease further. Stated differently, in this case the source samples cannot compensate for the target samples. This observation conforms with our main theoretical result. Indeed, when the transfer distance Δ is large, B is lower bounded by $\frac{\sigma^2 D}{256 r_T n_T}$ which is independent of the number of source samples n_S. Furthermore, these figures also demonstrate that when the transfer distance is small, increasing the number of source samples is helpful and results in lower target generalization error. This also matches our theoretical results as when the transfer distance Δ is small, the target generalization error is proportional to $\frac{D\sigma^2}{r_S n_S + r_T n_T}$.

Figures 4 (b) and (d) indicate that regardless of the transfer distance between the source and target tasks the target generalization error steadily decreases as the number of target samples increases. This is a good match with our theoretical results as n_T appears in the denominator of our lower bound in all three regimes.

To further investigate the effect of transfer distance between the source and target on the target generalization error we consider another set of experiments below.

Experimental setup 2. For the linear model, we use the parameters $d = 50, k = 30, \sigma = 0.3, \mathbf{\Sigma}_S = 2 \cdot I_d$, and $\mathbf{\Sigma}_T = I_d$. We generate the target parameter $\boldsymbol{W}_T \in \mathbb{R}^{k \times d}$ with entries generated i.i.d. $\mathcal{N}(0, 10)$. To create different transfer distances between the source and target data we then generate the source parameter $\boldsymbol{W}_S \in \mathbb{R}^{k \times d}$ as $\boldsymbol{W}_S = \boldsymbol{W}_T + i \cdot \boldsymbol{M}$ where the elements of the matrix $\boldsymbol{M}$ are sampled from $\mathcal{N}(0, 10^{-4})$ and i varies between 1 and 140000 in increments of 400. Similarly for the one-hidden layer neural network model when the the output layer is fixed, we pick parameter values $k = 1, \ell = 30, d = 50, \sigma = 0.3, \mathbf{\Sigma}_S = 2 \cdot \boldsymbol{I}_d$, and $\mathbf{\Sigma}_T = \boldsymbol{I}_d$ and set all of the entries of V equal to one. Furthermore, we use the same source and target parameters $\boldsymbol{W}_S$ and $\boldsymbol{W}_T$ as in the linear model.

Training approach 2. Given $n_S = 300$ and $n_T = 20$ source and target samples we minimize the weighted empirical risk (4.1). In this experiment we pick $\lambda \in \{0, \frac{1}{4}, \frac{1}{2}, \frac{3}{4}, 1\}$ that minimizes a validation set consisting of 50 data points created from the same distribution as the target task. Finally we test the estimated model on 200 unseen target test data points. The reported numbers are based on an average of 20 trials .

Results 2. Fig. 5 depicts the target generalization error as a function of the transfer distance between the source and target in the linear and neural network models. This figure clearly shows that when the transfer distance is small, the generalization error has a quadratic growth. However, as the distance increases the error saturates which matches the behavior of Δ predicted by our lower bounds.

Broader Impact

While our work is theoretical/foundations in nature let us discuss a few ways in which it may have broader impacts. In this paper, we characterize a lower bound for transfer learning in the context of linear models and one-hidden layer neural networks. More specifically, we provide a lower bound for target generalization error in terms of the number of source and target tasks and an appropriately defined transfer distance between the source and target tasks. Given the amount of effort dedicated to data collection, curation, and storage, a precise understanding of the amount of data needed may help utilize a variety of resources more effectively. Moreover, our results may guide practitioners to when there is no hope of knowledge transfer from one domain to another. This may help avoid unwarranted generalizations from one situation/environment to unrelated instances. On the other hand, it is worth emphasizing that this paper focuses on shallow linear/neural network models and does not capture more realistic Deep Neural Network (DNN) models typically used in practice. Therefore, one has to be cautious in over-interpreting the results of this paper for general DNN models.

5 Acknowledgments and Disclosure of Funding

This material is based upon work supported by Defense Advanced Research Projects Agency (DARPA) under Contract No. FA8750-19-2-1005. The views, opinions, and/or findings expressed are those of the author(s) and should not be interpreted as representing the official views or policies of the Department of Defense or the U.S. Government.

M. Soltanolkotabi is also supported by the Packard Fellowship in Science and Engineering, a Sloan Research Fellowship in Mathematics, an NSF-CAREER under award $\#1846369$, the Air Force Office of Scientific Research Young Investigator Program (AFOSR-YIP) under award $\#\text{FA}9550-18-1-0078$, DARPA FastNICS program, and NSF-CIF awards $\#1813877$ and $\#2008443$.

References

[1] J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. Wortman, "Learning bounds for domain adaptation," in *Advances in neural information processing systems*, 2008, pp. 129–136.

[2] K. You, X. Wang, M. Long, and M. Jordan, "Towards accurate model selection in deep unsupervised domain adaptation," in *International Conference on Machine Learning*, 2019, pp. 7124–7133.

[3] X. Chen, S. Wang, M. Long, and J. Wang, "Transferability vs. discriminability: Batch spectral penalization for adversarial domain adaptation," in *International Conference on Machine Learning*, 2019, pp. 1081–1090.

[4] Y. Wu, E. Winston, D. Kaushik, and Z. Lipton, "Domain adaptation with asymmetrically-relaxed distribution alignment," *arXiv preprint arXiv:1903.01689*, 2019.

[5] K. Azizzadenesheli, A. Liu, F. Yang, and A. Anandkumar, "Regularized learning for domain adaptation under label shifts," *arXiv preprint arXiv:1903.09734*, 2019.

[6] J. Shen, Y. Qu, W. Zhang, and Y. Yu, "Wasserstein distance guided representation learning for domain adaptation," in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[7] M. Long, H. Zhu, J. Wang, and M. I. Jordan, "Unsupervised domain adaptation with residual transfer networks," in *Advances in neural information processing systems*, 2016, pp. 136–144.

[8] S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. W. Vaughan, "A theory of learning from different domains," *Machine learning*, vol. 79, no. 1-2, pp. 151–175, 2010.

[9] S. Ben-David, J. Blitzer, K. Crammer, and F. Pereira, "Analysis of representations for domain adaptation," in *Advances in neural information processing systems*, 2007, pp. 137–144.

[10] Y. Mansour, M. Mohri, and A. Rostamizadeh, "Domain adaptation: Learning bounds and algorithms," *arXiv preprint arXiv:0902.3430*, 2009.

[11] H. Zhao, R. T. d. Combes, K. Zhang, and G. J. Gordon, "On learning invariant representation for domain adaptation," *arXiv preprint arXiv:1901.09453*, 2019.

[12] Y. Zhang, T. Liu, M. Long, and M. I. Jordan, "Bridging theory and algorithm for domain adaptation," *arXiv preprint arXiv:1904.05801*, 2019.

[13] S. Ben-David, T. Lu, T. Luu, and D. Pál, "Impossibility theorems for domain adaptation," in *International Conference on Artificial Intelligence and Statistics*, 2010, pp. 129–136.

[14] S. Hanneke and S. Kpotufe, "On the value of target data in transfer learning," in *Advances in Neural Information Processing Systems*, 2019, pp. 9867–9877.

[15] K. Lounici, M. Pontil, S. Van De Geer, A. B. Tsybakov *et al.*, "Oracle inequalities and optimal inference under group sparsity," *The annals of statistics*, vol. 39, no. 4, pp. 2164–2204, 2011.

[16] https://github.com/MathFLDS/TransferLowerbounds.

[17] M. J. Wainwright, *High-dimensional statistics: A non-asymptotic viewpoint*. Cambridge University Press, 2019, vol. 48.

[18] A. Daniely, R. Frostig, and Y. Singer, "Toward deeper understanding of neural networks: The power of initialization and a dual view on expressivity," in *Advances In Neural Information Processing Systems*, 2016, pp. 2253–2261.

6 Proof outline and proof of Theorem 1 in the linear model

In this section we present a sketch of the proof of Theorem 1 for the linear model. The proof for the neural network models follow a similar approach and appear in Sections 7.5 and 7.7.

Note that by Proposition 1, the generalization error is given by

$$\mathbb{E}_{\mathbb{Q}_{\boldsymbol{\theta}_T}}[\|\widehat{\boldsymbol{y}}_T - \boldsymbol{y}_T\|_{\ell_2}^2] = \|\boldsymbol{\Sigma}_T^{\frac{1}{2}}(\widehat{\boldsymbol{W}}_T - \boldsymbol{W}_T)^T\|_F^2 + k\sigma^2.$$

Therefore, in order to find a minimax lower bound on the target generalization error, it suffices to find a lower bound for the following quantity

$$\begin{aligned}&\mathcal{R}_T(\mathcal{P}_\Delta; \phi \circ \rho)\\ &\quad := \inf_{\widehat{\boldsymbol{W}}_T} \sup_{(\mathbb{P}_{\boldsymbol{W}_S}, \mathbb{Q}_{\boldsymbol{W}_T}) \in \mathcal{P}_\Delta} \mathbb{E}_{S_{\mathbb{P}_{\boldsymbol{W}_S}} \sim \mathbb{P}_{\boldsymbol{W}_S}^{1:n_{\mathbb{P}}}} \Big[\mathbb{E}_{S_{\mathbb{Q}_{\boldsymbol{W}_T}} \sim \mathbb{Q}_{\boldsymbol{W}_T}^{1:n_{\mathbb{Q}}}} \Big[\phi(\rho(\widehat{\boldsymbol{W}}_T(S_{\mathbb{P}_{\boldsymbol{W}_S}}, S_{\mathbb{Q}_{\boldsymbol{W}_T}}), \boldsymbol{W}_T))\Big]\Big]\end{aligned} \tag{6.1}$$

where $\phi(x) = x^2$ for $x \in \mathbb{R}$ and ρ is per Definition 1. By using well-known techniques from the statistical minimax literature we reduce the problem of finding a lower bound to a hypothesis testing problem (e.g. see (17, Chapter 15)). Since we are estimating the target parameter, i.e. $\boldsymbol{W}_T$, to apply this framework we need to pick N pairs of distributions from the set $\mathcal{P}_\Delta$ such that their target parameters are 2δ-separated by the transfer distance per Definition 1. To be more precise, we pick N arbitrary pairs of distributions from $\mathcal{P}_\Delta$:

$$(\mathbb{P}_{\boldsymbol{W}_S^{(1)}}, \mathbb{Q}_{\boldsymbol{W}_T^{(1)}}), ..., (\mathbb{P}_{\boldsymbol{W}_S^{(N)}}, \mathbb{Q}_{\boldsymbol{W}_T^{(N)}})$$

such that

$$\rho(\boldsymbol{W}_T^{(i)}, \boldsymbol{W}_T^{(j)}) \geq 2\delta \;\text{ for each } i \neq j \in [N] \times [N] \;\; (2\delta\text{-separated set})$$

and

$$\rho(\boldsymbol{W}_S^{(i)}, \boldsymbol{W}_T^{(i)}) \leq \Delta \;\text{ for each } i \in [N] \;\; (\text{as they belong to } \mathcal{P}_\Delta)$$

With these N distribution pairs in place we can follow a proof similar to that of (17, Proposition 15.1) to reduce the minimax problem to a hypothesis test problem. In particular, consider the following N-array hypothesis testing problem:

- J is the uniform distribution over the index set $[N] := \{1, 2, ..., N\}$
- Given $J = i$, generate n_S i.i.d. samples from $\mathbb{P}_{\boldsymbol{W}_S^{(i)}}$ and n_T i.i.d. samples from $\mathbb{Q}_{\boldsymbol{W}_T^{(i)}}$.

Here the goal is to find the true index using $n_S + n_T$ available samples by a testing function ψ from the samples to the indices.

Let E and F be random variables such that $E|\{J = i\} \sim \mathbb{P}_{\boldsymbol{W}_S^{(i)}}$ and $F|\{J = i\} \sim \mathbb{Q}_{\boldsymbol{W}_T^{(i)}}$. Furthermore, let $Z_{\mathbb{P}}$ and $Z_{\mathbb{Q}}$ consist of n_S independent copies of random variable E and n_T independent copies of random variable F, respectively. In this setting, by slightly modifying the (17, Proposition 15.1) we can conclude that

$$\mathcal{R}_T(\mathcal{P}_\Delta; \phi \circ \rho) \geq \phi(\delta) \frac{1}{N} \sum_{i=1}^{N} \text{Prob}(\psi(Z_{\mathbb{P}}, Z_{\mathbb{Q}}) \neq i)$$

where

$$\psi(Z_{\mathbb{P}}, Z_{\mathbb{Q}}) := \arg\min_{n \in [N]} \rho(\widehat{\boldsymbol{W}}_T, \boldsymbol{W}_T^n).$$

Furthermore, by using Fano's inequality we can conclude that

$$\begin{aligned}\mathcal{R}_T(\mathcal{P}_\Delta; \phi \circ \rho) &\geq \phi(\delta) \frac{1}{N} \sum_{i=1}^{N} \mathbb{P}\Big\{\psi(Z_{\mathbb{P}}, Z_{\mathbb{Q}}) \neq i\Big\}\\ &\geq \phi(\delta)\left(1 - \frac{I(J; (Z_{\mathbb{P}}, Z_{\mathbb{Q}})) + \log 2}{\log N}\right)\\ &\geq \phi(\delta)\left(1 - \frac{I(J; Z_{\mathbb{P}}) + I(J; Z_{\mathbb{Q}}) + \log 2}{\log N}\right)\\ &\geq \phi(\delta)\left(1 - \frac{n_S I(J; E) + n_T I(J; F) + \log 2}{\log N}\right).\end{aligned} \tag{6.2}$$

Here the third inequality is due to the fact that given $J = i$, $Z_{\mathbb{P}}$ and $Z_{\mathbb{Q}}$ are independent. To continue further, note that we can bound the mutual information by the following KL-divergences

$$I(J;E) \le \frac{1}{N^2}\sum_{i,j} D_{KL}(\mathbb{P}_{\boldsymbol{W}_S^{(i)}} \| \mathbb{P}_{\boldsymbol{W}_S^{(j)}})$$
$$I(J;F) \le \frac{1}{N^2}\sum_{i,t} D_{KL}(\mathbb{Q}_{\boldsymbol{W}_T^{(i)}} \| \mathbb{Q}_{\boldsymbol{W}_T^{(j)}}). \tag{6.3}$$

In the next lemma, proven in Section 7.2, we explicitly calculate the above KL-divergences.

Lemma 1 *Suppose that $\mathbb{P}_{\boldsymbol{W}_S^{(i)}}$ and $\mathbb{P}_{\boldsymbol{W}_S^{(j)}}$ are the joint distributions of features and labels in a source task and $\mathbb{Q}_{\boldsymbol{W}_T^{(i)}}$ and $\mathbb{Q}_{\boldsymbol{W}_T^{(j)}}$ are joint distributions of features and labels in a target task as defined in Section 2.1 for the linear model. Then $D_{KL}(\mathbb{P}_{\boldsymbol{W}_S^{(i)}} \| \mathbb{P}_{\boldsymbol{W}_S^{(j)}}) = \frac{\|\boldsymbol{\Sigma}_S^{\frac{1}{2}}(\boldsymbol{W}_S^{(i)} - \boldsymbol{W}_S^{(j)})^T\|_F^2}{2\sigma^2}$ and $D_{KL}(\mathbb{Q}_{\boldsymbol{W}_T^{(i)}} \| \mathbb{Q}_{\boldsymbol{W}_T^{(j)}}) = \frac{\|\boldsymbol{\Sigma}_T^{\frac{1}{2}}(\boldsymbol{W}_T^{(i)} - \boldsymbol{W}_T^{(j)})^T\|_F^2}{2\sigma^2}$.*

In the following two lemmas, we use local packing techniques to further simplify (6.2) using (6.3) and find minimax lower bounds in different transfer distance regimes. We defer the proof of these lemmas to Sections 7.3 and 7.4.

Lemma 2 *Assume $\Delta \ge \sqrt{\frac{\sigma^2 D \log 2}{r_T n_T}}$, where n_T is the number of target samples and D and r_T are defined per Definitions 3 and 4. Then we have the following lowerbound*

$$\mathcal{R}_T(\mathcal{P}_\Delta; \phi \circ \rho) \ge \frac{\sigma^2 D}{256 r_T n_T}. \tag{6.4}$$

Furthermore, if $\Delta < \sqrt{\frac{\sigma^2 D \log 2}{r_T n_T}}$ then

$$\mathcal{R}_T(\mathcal{P}_\Delta; \phi \circ \rho) \ge \frac{1}{100}\Delta^2\left(1 - 0.8\frac{r_T n_T \Delta^2}{\sigma^2 D}\right). \tag{6.5}$$

Lemma 3 *Assume we have access to n_S source samples as well as n_T target samples and the transfer distance obeys $\Delta \le \frac{1}{45}\sqrt{\frac{\sigma^2 D}{r_S n_S + r_T n_T}}$, where D, r_S, and r_T are per Definitions 3 and 4. Then,*

$$\mathcal{R}_T(\mathcal{P}_\Delta; \phi \circ \rho) \ge \frac{\Delta^2}{1000} + \frac{6}{1000}\frac{D\sigma^2}{r_S n_S + r_T n_T}. \tag{6.6}$$

The proof of the lower bound in Theorem 1 is complete by combining Lemmas 2 and 3.

7 Appendix A

7.1 Calculating the Generalization Errors (Proof of Proposition 1)

- **Linear model:**

By expanding the expression we get

$$\begin{aligned}
\mathbb{E}_{\mathbb{Q}_{\boldsymbol{\theta}_T}}[\|\widehat{\boldsymbol{y}}_T - \boldsymbol{y}_T\|_{\ell_2}^2] &= \mathbb{E}[\|\widehat{\boldsymbol{W}}_T\boldsymbol{x}_T - \boldsymbol{W}_T\boldsymbol{x}_T - w_T\|_{\ell_2}^2] \\
&= \mathbb{E}[\|\widehat{\boldsymbol{W}}_T\boldsymbol{x}_T - \boldsymbol{W}_T\boldsymbol{x}_T\|_{\ell_2}^2] + k\sigma^2 \\
&= \mathbb{E}[\boldsymbol{x}_T^T(\boldsymbol{W}_T - \widehat{\boldsymbol{W}_T})^T(\boldsymbol{W}_T - \widehat{\boldsymbol{W}_T})\boldsymbol{x}_T] + k\sigma^2 \\
&= \mathbb{E}[\text{trace}(\boldsymbol{x}_T^T(\boldsymbol{W}_T - \widehat{\boldsymbol{W}_T})^T(\boldsymbol{W}_T - \widehat{\boldsymbol{W}_T})\boldsymbol{x}_T)] + k\sigma^2 \\
&= \mathbb{E}[\text{trace}((\boldsymbol{W}_T - \widehat{\boldsymbol{W}_T})^T(\boldsymbol{W}_T - \widehat{\boldsymbol{W}_T})\boldsymbol{x}_T\boldsymbol{x}_T^T)] + k\sigma^2 \\
&= \text{trace}((\boldsymbol{W}_T - \widehat{\boldsymbol{W}_T})^T(\boldsymbol{W}_T - \widehat{\boldsymbol{W}_T})\mathbb{E}[\boldsymbol{x}_T\boldsymbol{x}_T^T]) + k\sigma^2 \\
&= \text{trace}((\boldsymbol{W}_T - \widehat{\boldsymbol{W}_T})^T(\boldsymbol{W}_T - \widehat{\boldsymbol{W}_T})\boldsymbol{\Sigma}_T) + k\sigma^2 \\
&= \|\boldsymbol{\Sigma}_T^{\frac{1}{2}}(\boldsymbol{W}_T - \widehat{\boldsymbol{W}_T})^T\|_F^2 + k\sigma^2
\end{aligned} \tag{7.1}$$

• **One-hidden layer neural network model with fixed hidden-to-output layer:**

By expanding the expression we obtain

$$\begin{aligned}\mathbb{E}_{\mathbb{Q}_{\boldsymbol{\theta}_T}}[\|\widehat{\boldsymbol{y}}_T - \boldsymbol{y}_T\|_{\ell_2}^2] &= \mathbb{E}[\left\|\boldsymbol{V}\varphi(\widehat{\boldsymbol{W}}_T\boldsymbol{x}_T) - \boldsymbol{V}\varphi(\boldsymbol{W}_T\boldsymbol{x}_T)\right\|_{\ell_2}^2] + k\sigma^2 \\ &\geq \sigma_{\min}^2(\boldsymbol{V})\mathbb{E}[\left\|\varphi(\widehat{\boldsymbol{W}}_T\boldsymbol{x}_T) - \varphi(\boldsymbol{W}_T\boldsymbol{x}_T)\right\|_{\ell_2}^2] + k\sigma^2 \end{aligned} \tag{7.2}$$

Let $\boldsymbol{A} = \widehat{\boldsymbol{W}}_T\boldsymbol{\Sigma}_T^{\frac{1}{2}}$, $\boldsymbol{B} = \boldsymbol{W}_T\boldsymbol{\Sigma}_T^{\frac{1}{2}}$, and $\boldsymbol{x} = \boldsymbol{\Sigma}_T^{\frac{-1}{2}}\boldsymbol{x}_T$. So $\boldsymbol{x} \sim \mathcal{N}(0, I_d)$. Moreover, let $\boldsymbol{A} = \begin{bmatrix} \boldsymbol{\alpha}_1^T \\ \vdots \\ \boldsymbol{\alpha}_\ell^T \end{bmatrix}$ and $\boldsymbol{B} = \begin{bmatrix} \boldsymbol{\beta}_1^T \\ \vdots \\ \boldsymbol{\beta}_\ell^T \end{bmatrix}$. Since $\mathbb{E}[\left\|\varphi(\widehat{\boldsymbol{W}}_T\boldsymbol{x}_T) - \varphi(\boldsymbol{W}_T\boldsymbol{x}_T)\right\|_{\ell_2}^2] = \sum_{i=1}^{\ell}\mathbb{E}[|\varphi(\boldsymbol{\alpha}_i^T\boldsymbol{x}) - \varphi(\boldsymbol{\beta}_i^T\boldsymbol{x})|^2]$, it suffices to find a lower bound for the following expression

$$\mathbb{E}[|\varphi(\boldsymbol{a}^T\boldsymbol{x}) - \varphi(\boldsymbol{b}^T\boldsymbol{x})|^2]$$

where a and b are two arbitrary vectors in $\mathbb{R}^d$, φ is the ReLU activation function, and $\boldsymbol{x} \sim \mathcal{N}(0, I_d)$. We have

$$\mathbb{E}[|\varphi(\boldsymbol{a}^T\boldsymbol{x}) - \varphi(\boldsymbol{b}^T\boldsymbol{x})|^2] = \mathbb{E}[|\varphi(\boldsymbol{a}^T\boldsymbol{x})|^2] + \mathbb{E}[|\varphi(\boldsymbol{b}^T\boldsymbol{x})|^2] - 2\mathbb{E}[\varphi(\boldsymbol{a}^T\boldsymbol{x})\varphi(\boldsymbol{b}^T\boldsymbol{x})]. \tag{7.3}$$

Now we calculate each term appearing on the right hand side.

Since $\boldsymbol{a}^T\boldsymbol{x} \sim N(0, \|\boldsymbol{a}\|_{\ell_2}^2)$, we have

$$\begin{aligned}\mathbb{E}[|\varphi(\boldsymbol{a}^T\boldsymbol{x})|^2] &= \mathbb{E}[|\mathrm{ReLU}(\boldsymbol{a}^T\boldsymbol{x})|^2] \\ &= \int_0^{+\infty} \frac{t^2}{\sqrt{2\pi}\,\|\boldsymbol{a}\|_{\ell_2}} e^{\frac{-t^2}{2\|\boldsymbol{a}\|_{\ell_2}^2}} dt \\ &= \frac{\|\boldsymbol{a}\|_{\ell_2}^2}{2}.\end{aligned}$$

Similarly, $\mathbb{E}[|\varphi(\boldsymbol{b}^T\boldsymbol{x})|^2] = \frac{\|\boldsymbol{b}\|_{\ell_2}^2}{2}$. To calculate the cross term note that $\boldsymbol{a}^T\boldsymbol{x}$ and $\boldsymbol{b}^T\boldsymbol{x}$ are jointly Gaussian with zero mean and covariance matrix equal to

$$\begin{bmatrix} \|\boldsymbol{a}\|_{\ell_2}^2 & \boldsymbol{a}^T\boldsymbol{b} \\ \boldsymbol{a}^T\boldsymbol{b} & \|\boldsymbol{b}\|_{\ell_2}^2 \end{bmatrix}.$$

Therefore, we have (e.g. see (18))

$$\begin{aligned}2\mathbb{E}[\varphi(\boldsymbol{a}^T\boldsymbol{x})\varphi(\boldsymbol{b}^T\boldsymbol{x})] &= 2\mathbb{E}[\mathrm{ReLU}(\boldsymbol{a}^T\boldsymbol{x})\mathrm{ReLU}(\boldsymbol{b}^T\boldsymbol{x})] \\ &= \|\boldsymbol{a}\|_{\ell_2}\|\boldsymbol{b}\|_{\ell_2}\frac{\sqrt{1-\gamma^2} + (\pi - \cos^{-1}(\gamma))\gamma}{\pi}\end{aligned} \tag{7.4}$$

where $\gamma := \frac{\boldsymbol{a}^T\boldsymbol{b}}{\|\boldsymbol{a}\|_{\ell_2}\|\boldsymbol{b}\|_{\ell_2}}$.

Plugging these results in (7.3), we can conlude that

$$\begin{aligned}\mathbb{E}[|\varphi(\boldsymbol{a}^T\boldsymbol{x}) - \varphi(\boldsymbol{b}^T\boldsymbol{x})|^2] &= \frac{\|\boldsymbol{a}\|_{\ell_2}^2}{2} + \frac{\|\boldsymbol{b}\|_{\ell_2}^2}{2} - \|\boldsymbol{a}\|_{\ell_2}\|\boldsymbol{b}\|_{\ell_2}\frac{\sqrt{1-\gamma^2} + (\pi - \cos^{-1}(\gamma))\gamma}{\pi} \\ &= \frac{1}{2}\|\boldsymbol{a} - \boldsymbol{b}\|_{\ell_2}^2 - \|\boldsymbol{a}\|_{\ell_2}\|\boldsymbol{b}\|_{\ell_2}\frac{\sqrt{1-\gamma^2} - \gamma\cos^{-1}(\gamma)}{\pi}.\end{aligned} \tag{7.5}$$

We are interested in finding a universal constant $0 < c < \frac{1}{2}$ such that $\mathbb{E}[|\varphi(\boldsymbol{a}^T\boldsymbol{x}) - \varphi(\boldsymbol{b}^T\boldsymbol{x})|^2] \geq c\,\|\boldsymbol{a}-\boldsymbol{b}\|_{\ell_2}^2$. Using (7.5) and dividing by $\|\boldsymbol{a}\|_{\ell_2}\|\boldsymbol{b}\|_{\ell_2}$ this is equivalent to finding $0 < c < \frac{1}{2}$ such that

$$\Big(\frac{1}{2} - c\Big)\frac{\|\boldsymbol{a}\|_{\ell_2}^2 + \|\boldsymbol{b}\|_{\ell_2}^2 - 2\boldsymbol{a}^T\boldsymbol{b}}{\|\boldsymbol{a}\|_{\ell_2}\|\boldsymbol{b}\|_{\ell_2}} + \frac{\gamma\cos^{-1}(\gamma) - \sqrt{1-\gamma^2}}{\pi} \geq 0$$

Next note that by the AM-GM inequality we have

$$\begin{aligned}
(\frac{1}{2} - c)\frac{\|\boldsymbol{a}\|_{\ell_2}^2 + \|\boldsymbol{b}\|_{\ell_2}^2 - 2\boldsymbol{a}^T\boldsymbol{b}}{\|\boldsymbol{a}\|_{\ell_2}\|\boldsymbol{b}\|_{\ell_2}} &+ \frac{\gamma\cos^{-1}(\gamma) - \sqrt{1-\gamma^2}}{\pi} \\
&\geq (\frac{1}{2} - c)\frac{2\|\boldsymbol{a}\|_{\ell_2}\|\boldsymbol{b}\|_{\ell_2} - 2\boldsymbol{a}^T\boldsymbol{b}}{\|\boldsymbol{a}\|_{\ell_2}\|\boldsymbol{b}\|_{\ell_2}} + \frac{\gamma\cos^{-1}(\gamma) - \sqrt{1-\gamma^2}}{\pi} \\
&= (\frac{1}{2} - c)(2 - 2\gamma) + \frac{\gamma\cos^{-1}(\gamma) - \sqrt{1-\gamma^2}}{\pi} \\
&= (1-\gamma)\Big[(1-2c) + \frac{1}{\pi}\cdot\frac{\gamma\cos^{-1}(\gamma) - \sqrt{1-\gamma^2}}{1-\gamma}\Big].
\end{aligned}$$

Therefore, it suffices to find $0 < c < \frac{1}{2}$ such that the R.H.S. of the above is positive. It is easy to verify that $h(\gamma) := \frac{\gamma\cos^{-1}(\gamma) - \sqrt{1-\gamma^2}}{1-\gamma} \geq \frac{-\pi}{2}$ for $-1 \leq \gamma < 1$. This in turn implies that the R.H.S. above is positive with $c = \frac{1}{4}$.

In the case when $\|\boldsymbol{a}\|_{\ell_2} = 0$ or $\|\boldsymbol{b}\|_{\ell_2} = 0$ (let us assume $\|\boldsymbol{b}\|_{\ell_2} = 0$), (7.3) reduces to

$$\begin{aligned}
\mathbb{E}[|\varphi(\boldsymbol{a}^T\boldsymbol{x}) - \varphi(\boldsymbol{b}^T\boldsymbol{x})|^2] &= \mathbb{E}[|\varphi(\boldsymbol{a}^T\boldsymbol{x})|^2] \\
&= \frac{\|\boldsymbol{a}\|_{\ell_2}^2}{2} \\
&\geq \frac{1}{2}\|\boldsymbol{a}-\boldsymbol{b}\|_{\ell_2}^2 \\
&\geq \frac{1}{4}\|\boldsymbol{a}-\boldsymbol{b}\|_{\ell_2}^2.
\end{aligned}$$

Plugging the latter into (7.2) we arrive at

$$\begin{aligned}
\mathbb{E}_{\mathbb{Q}_{\theta_T}}[\|\widehat{\boldsymbol{y}}_T - \boldsymbol{y}_T\|_{\ell_2}^2] &\geq \sigma_{\min}^2(\boldsymbol{V})\mathbb{E}[\big\|\varphi(\widehat{\boldsymbol{W}}_T\boldsymbol{x}_T) - \varphi(\boldsymbol{W}_T\boldsymbol{x}_T)\big\|_{\ell_2}^2] + k\sigma^2 \\
&\geq \frac{1}{4}\sigma_{\min}^2(\boldsymbol{V})\|\boldsymbol{\Sigma}_T^{\frac{1}{2}}(\widehat{\boldsymbol{W}}_T - \boldsymbol{W}_T)^T\|_F^2 + k\sigma^2,
\end{aligned}$$

concluding the proof.

- **One-hidden layer neural network model with fixed input-to-hidden layer:**

By expanding the expression we get

$$\mathbb{E}_{\mathbb{Q}_{\boldsymbol{\theta}_T}}[\|\widehat{\boldsymbol{y}}_T - \boldsymbol{y}_T\|_{\ell_2}^2] = \mathbb{E}[\big\|\widehat{\boldsymbol{V}}_T\varphi(\boldsymbol{W}\boldsymbol{x}_T) - \boldsymbol{V}_T\varphi(\boldsymbol{W}\boldsymbol{x}_T)\big\|_{\ell_2}^2] + k\sigma^2.$$

If we denote $\mathbb{E}[\varphi(\boldsymbol{W}\boldsymbol{x}_T)\varphi(\boldsymbol{W}\boldsymbol{x}_T)^T] = \widetilde{\boldsymbol{\Sigma}}_T$, then similar to (7.1) we obtain

$$\mathbb{E}_{\mathbb{Q}_{\boldsymbol{\theta}_T}}[\|\widehat{\boldsymbol{y}}_T - \boldsymbol{y}_T\|_{\ell_2}^2] = \|\widetilde{\boldsymbol{\Sigma}}_T^{\frac{1}{2}}(\widehat{\boldsymbol{V}}_T - \boldsymbol{V}_T)^T\|_F^2 + k\sigma^2. \tag{7.6}$$

Therefore, it suffices to calculate $\widetilde{\boldsymbol{\Sigma}}_T$. Let $\boldsymbol{W}\boldsymbol{\Sigma}_T^{\frac{1}{2}} = \begin{bmatrix} \boldsymbol{a}_1^T \\ \vdots \\ \boldsymbol{a}_\ell^T \end{bmatrix}$ and $\boldsymbol{x} = \boldsymbol{\Sigma}_T^{\frac{-1}{2}}\boldsymbol{x}_T$ (so $\boldsymbol{x} \sim \mathcal{N}(0, I_d)$). By (7.4) we obtain that

$$\widetilde{\boldsymbol{\Sigma}}_T = \Big[\frac{1}{2}\|\boldsymbol{a}_i\|_{\ell_2}\|\boldsymbol{a}_j\|_{\ell_2}\frac{\sqrt{1-\gamma_{ij}^2} + (\pi - \cos^{-1}(\gamma_{ij}))\gamma_{ij}}{\pi}\Big]_{ij} \tag{7.7}$$

where $\gamma_{ij} := \frac{\boldsymbol{a}_i^T\boldsymbol{a}_j}{\|\boldsymbol{a}_i\|_{\ell_2}\|\boldsymbol{a}_j\|_{\ell_2}}$.

7.2 Calculating KL-Divergences for the Linear Model (Proof of Lemma 1)

First we compute the KL-divergence between the distributions $\mathbb{P}_{\boldsymbol{W}_S^{(i)}}(\boldsymbol{x}_S, \boldsymbol{y}_S)$ and $\mathbb{P}_{\boldsymbol{W}_S^{(j)}}(\boldsymbol{x}_S, \boldsymbol{y}_S)$:

$$\begin{aligned} D_{KL}(\mathbb{P}_{\boldsymbol{W}_S^{(i)}}(\boldsymbol{x}_S, \boldsymbol{y}_S), \mathbb{P}_{\boldsymbol{W}_S^{(j)}}(\boldsymbol{x}_S, \boldsymbol{y}_S)) =& D_{KL}(\mathbb{P}_{\boldsymbol{W}_S^{(i)}}(\boldsymbol{x}_S), \mathbb{P}_{\boldsymbol{W}_S^{(j)}}(\boldsymbol{x}_S)) \\ &+ \mathbb{E}[D_{KL}(\mathbb{P}_{\boldsymbol{W}_S^{(i)}}(\boldsymbol{y}_S|\boldsymbol{x}_S), \mathbb{P}_{\boldsymbol{W}_S^{(j)}}(\boldsymbol{y}_S|\boldsymbol{x}_S))]. \end{aligned}$$

The marginal distributions $\mathbb{P}_{\boldsymbol{W}_S^{(i)}}(\boldsymbol{x}_S)$ and $\mathbb{P}_{\boldsymbol{W}_S^{(j)}}(\boldsymbol{x}_S)$ are equal so their KL-divergence is zero. The conditional distributions $\mathbb{P}_{\boldsymbol{W}_S^{(i)}}(\boldsymbol{y}_S|\boldsymbol{x}_S)$ and $\mathbb{P}_{\boldsymbol{W}_S^{(j)}}(\boldsymbol{y}_S|\boldsymbol{x}_S)$ are normally distributed with covariance matrix $\sigma^2\mathbf{I}_k$ and with mean respectively equal to $\boldsymbol{W}_S^{(i)}\boldsymbol{x}_S$ and $\boldsymbol{W}_S^{(j)}\boldsymbol{x}$. Therefore,

$$D_{KL}(\mathbb{P}_{\boldsymbol{W}_S^{(i)}}(\boldsymbol{y}_S|\boldsymbol{x}_S), \mathbb{P}_{\boldsymbol{W}_S^{(j)}}(\boldsymbol{y}_S|\boldsymbol{x}_S)) = \frac{\left\|\boldsymbol{W}_S^{(i)}\boldsymbol{x}_S - \boldsymbol{W}_S^{(j)}\boldsymbol{x}_S\right\|_{\ell_2}^2}{2\sigma^2}.$$

This in turn implies that

$$D_{KL}(\mathbb{P}_{\boldsymbol{W}_S^{(i)}}(\boldsymbol{x}_S, \boldsymbol{y}_S), \mathbb{P}_{\boldsymbol{W}_S^{(j)}}(\boldsymbol{x}_S, \boldsymbol{y}_S)) = \frac{\mathbb{E}[\left\|\boldsymbol{W}_S^{(i)}\boldsymbol{x}_S - \boldsymbol{W}_S^{(j)}\boldsymbol{x}_S\right\|_{\ell_2}^2]}{2\sigma^2} = \frac{||\boldsymbol{\Sigma}_S^{\frac{1}{2}}(\boldsymbol{W}_S^{(i)} - \boldsymbol{W}_S^{(j)})^T||_F^2}{2\sigma^2},$$

where the last equality follows similarly to the proof of Proposition 1 in the linear case.

A similar calculation also yields

$$D_{KL}(\mathbb{Q}_{\boldsymbol{W}_T^{(i)}}(\boldsymbol{x}_T, \boldsymbol{y}_T), \mathbb{Q}_{\boldsymbol{W}_T^{(j)}}(\boldsymbol{x}_T, \boldsymbol{y}_T)) = \frac{||\boldsymbol{\Sigma}_T^{\frac{1}{2}}(\boldsymbol{W}_T^{(i)} - \boldsymbol{W}_T^{(j)})^T||_F^2}{2\sigma^2}.$$

7.3 Lower Bound for Minimax Risk When $\Delta \geq \sqrt{\frac{\sigma^2 D \log 2}{r_T n_T}}$ and $\Delta < \sqrt{\frac{\sigma^2 D \log 2}{r_T n_T}}$ (Proof of Lemma 2)

Consider the set

$$\left\{\eta : \eta = \boldsymbol{\Sigma}_T^{\frac{1}{2}}\boldsymbol{W}_T^T \text{for some } \mathbf{W}_T \in \mathbb{R}^{k\times d} \text{ and } ||\eta||_F \leq 4\delta\right\}$$

where $\delta > 0$ is a value to be determined later in the proof. Furthermore, let $\{\eta^1, ..., \eta^N\}$ be a 2δ-packing of this set in the F-norm. Since dim(range($\boldsymbol{\Sigma}_T^{\frac{1}{2}}\boldsymbol{W}_T^T$))= rk in which $\boldsymbol{W}_T$ is regarded as an input, this set sits in a space of dimension rk, where r =rank($\boldsymbol{\Sigma}_T$). Hence we can find such a packing with $\log N \geq rk \log 2$ elements.

Therefore, we have a collection of matrices of the form $\eta^j = \boldsymbol{\Sigma}_T^{\frac{1}{2}}(\boldsymbol{W}_T^{(i)})^T$ for some $\boldsymbol{W}_T^{(i)} \in \mathbb{R}^{k\times d}$ such that

$$\left\|\boldsymbol{\Sigma}_T^{\frac{1}{2}}(\boldsymbol{W}_T^{(i)})^T\right\|_F \leq 4\delta \text{ for each } i \in [N]$$

and

$$2\delta \leq ||\boldsymbol{\Sigma}_T^{\frac{1}{2}}(\boldsymbol{W}_T^{(i)} - \boldsymbol{W}_T^{(j)})^T||_F \leq 8\delta \text{ for each } i \neq j \in [N] \times [N].$$

So by Lemma 1 we get

$$D_{KL}(\mathbb{Q}_{\boldsymbol{W}_T^{(i)}}, \mathbb{Q}_{\boldsymbol{W}_T^{(j)}}) \leq \frac{32\delta^2}{\sigma^2} \text{ for each } i \neq j \in [N] \times [N].$$

Then set $\delta \leq \frac{\Delta}{8}$. We can choose $\mathbb{P}_{\boldsymbol{W}_S^{(1)}} = ... = \mathbb{P}_{\boldsymbol{W}_S^{(N)}}$ with $\boldsymbol{W}_S^{(1)} = ... = \boldsymbol{W}_S^{(N)}$ and $||\boldsymbol{\Sigma}_T^{\frac{1}{2}}(\boldsymbol{W}_S^{(1)})^T||_F = 4\delta$. So they satisfy the condition

$$\rho(\boldsymbol{W}_S^{(i)}, \boldsymbol{W}_T^{(i)}) \leq 8\delta \leq \Delta \text{ for each } i \in [N].$$

Figure 6 illustrates this configuration. So having samples from $\mathbb{P}_{\boldsymbol{W}_S^{(i)}}$ does not contain any informa-

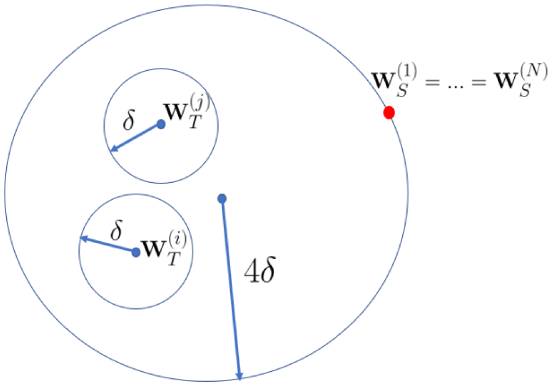


Figure 6: Configuration of the parameters of source and target distributions in Lemma 2.

tion about the true index in $[N]$ in the hypothesis testing problem and E is independent of J, hence we get

$$I(J;E) = 0.$$

Therefore, using (6.2) and (6.3) we get

$$\begin{aligned}\mathcal{R}_T(\mathcal{P}_\Delta;\phi\circ\rho) &\geq \phi(\delta)\left(1-\frac{n_T\frac{32\delta^2}{\sigma^2}+\log 2}{rk\log 2}\right)\\ &= \delta^2\left(1-\frac{n_T\frac{32\delta^2}{\sigma^2}+\log 2}{rk\log 2}\right)\end{aligned}$$

for any $0 \leq \delta \leq \frac{\Delta}{8}$. We need to check the boundary and stationary points to solve the above optimization problem.

If $\Delta \geq \sqrt{\frac{\sigma^2(rk-1)\log 2}{n_T}} = \sqrt{\frac{\sigma^2 D\log 2}{n_T}}$ holds, then

$$\mathcal{R}_T(\mathcal{P}_\Delta;\phi\circ\rho) \geq \frac{\sigma^2(rk-1)^2\log 2}{128 n_T rk}.$$

Since $D = rk - 1 \geq 20$, we have $\frac{\log 2}{rk} \geq \frac{1}{2(rk-1)}$, so

$$\mathcal{R}_T(\mathcal{P}_\Delta;\phi\circ\rho) \geq \frac{\sigma^2 D}{256 n_T} \tag{7.8}$$

and if $\Delta < \sqrt{\frac{\sigma^2(rk-1)\log 2}{n_T}} = \sqrt{\frac{\sigma^2 D\log 2}{n_T}}$ then

$$\begin{aligned}\mathcal{R}_T(\mathcal{P}_\Delta;\phi\circ\rho) &\geq (\frac{\Delta}{8})^2[1-\frac{\frac{n_T\Delta^2}{2\sigma^2}+\log 2}{rk\log 2}]\\ &\geq (\frac{\Delta}{8})^2[1-\frac{\frac{n_T\Delta^2}{2\sigma^2}+\log 2}{D\log 2}].\end{aligned}$$

Since $D \geq 20$ we get

$$\mathcal{R}_T(\mathcal{P}_\Delta;\phi\circ\rho) \geq \frac{1}{100}\Delta^2[1-0.8\frac{n_T\Delta^2}{\sigma^2 D}]. \tag{7.9}$$

7.4 Lower Bound for Minimax Risk When $\Delta \leq \frac{1}{45}\sqrt{\frac{\sigma^2 D}{r_S n_S + r_T n_T}}$ (Proof of Lemma 3)

Let $\delta' = \Delta + \underbrace{u\Delta}_{=\delta}$, for $u > 0$ to be determined, Consider the set

$$\{\eta : \eta = \mathbf{\Sigma}_T^{\frac{1}{2}}\boldsymbol{W}_S^T \text{ for some } \mathbf{W}_S \in \mathbb{R}^{k\times d} \text{ and } ||\eta||_F \leq 4\delta'\}$$

and let $\{\eta^1, ..., \eta^N\}$ be a $2\delta'$-packing in the F-norm and consider each η^i as a single point. Since dim(range($\Sigma_T^{\frac{1}{2}} \boldsymbol{W}_T^T$))= rk in which $\boldsymbol{W}_T$ is regarded as an input, this set sits in a space of dimension rk where r =rank(Σ_T). Therefore, we can find such a packing with $\log N \geq rk \log 2$ elements.

Hence, we have a collection of matrices of the form $\eta^i = \Sigma_T^{\frac{1}{2}} (\boldsymbol{W}_S^{(i)})^T$ for some $\boldsymbol{W}_S^{(i)} \in \mathbb{R}^{k \times d}$ such that

$$\|\Sigma_T^{\frac{1}{2}} (\boldsymbol{W}_S^{(i)})^T\|_F \leq 4\delta' \text{ for each } i \in [N]$$

$$2\delta' \leq \|\Sigma_T^{\frac{1}{2}} (\boldsymbol{W}_S^{(i)} - \boldsymbol{W}_S^{(j)})^T\|_F \leq 8\delta' \text{ for each } i \neq j \in [N] \times [N].$$

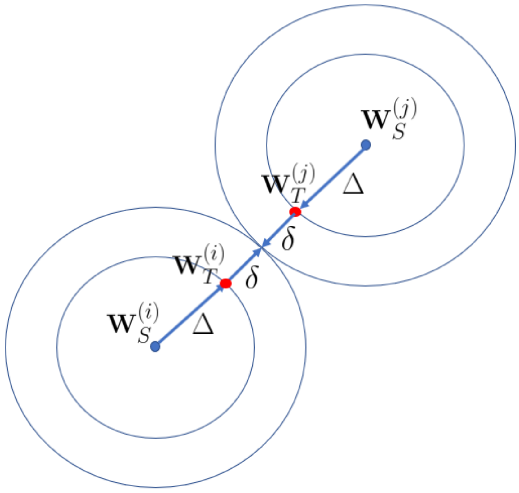


Figure 7: Configuration of the parameters of source and target distributions in Lemma 3.

We choose each $\boldsymbol{W}_T^{(i)}$ such that $\rho(\boldsymbol{W}_T^{(i)}, \boldsymbol{M}_S^{(i)}) = \|\Sigma_T^{\frac{1}{2}} (\boldsymbol{W}_T^{(i)} - \boldsymbol{W}_S^{(i)})^T\|_F = \Delta$. So

$$\rho(\boldsymbol{W}_T^{(i)}, \boldsymbol{W}_T^{(j)}) \geq 2\delta \text{ for each } i \neq j \in [N] \times [N].$$

Moreover,

$$\begin{aligned} \rho(\boldsymbol{W}_T^{(i)}, \boldsymbol{W}_T^{(j)}) &\leq \rho(\boldsymbol{W}_T^{(i)}, \boldsymbol{W}_S^{(i)}) + \rho(\boldsymbol{W}_S^{(i)}, \boldsymbol{M}_S^{(j)}) + \rho(\boldsymbol{W}_S^{(j)}, \boldsymbol{W}_T^{(j)}) \\ &\leq 2\Delta + 8(\Delta + u\Delta). \end{aligned}$$

Figure 7 illustrates this configuration. By Lemma 1 we have

$$D_{KL}(\mathbb{Q}_{\boldsymbol{W}_T^{(i)}}, \mathbb{Q}_{\boldsymbol{W}_T^{(j)}}) \leq \frac{2\Delta^2(5 + 4u)^2}{\sigma^2} \text{ for each } i \neq j \in [N] \times [N].$$

Also we have

$$\begin{aligned} \|\Sigma_S^{\frac{1}{2}} (\boldsymbol{W}_S^{(i)} - \boldsymbol{W}_S^{(j)})^T\|_F &= \|\Sigma_S^{\frac{1}{2}} \Sigma_T^{-\frac{1}{2}} \Sigma_T^{\frac{1}{2}} (\boldsymbol{W}_S^{(i)} - \boldsymbol{W}_S^{(j)})^T\|_F \\ &\leq 8 \left\| \Sigma_S^{\frac{1}{2}} \Sigma_T^{-\frac{1}{2}} \right\| \delta' \\ &= 8 \left\| \Sigma_S^{\frac{1}{2}} \Sigma_T^{-\frac{1}{2}} \right\| (\Delta + u\Delta) \end{aligned}$$

Hence, by Lemma 1 we have

$$D_{KL}(\mathbb{P}_{\boldsymbol{W}_S^{(i)}}, \mathbb{P}_{\boldsymbol{W}_S^{(j)}}) \leq \frac{32 \left\| \Sigma_S^{\frac{1}{2}} \Sigma_T^{-\frac{1}{2}} \right\|^2 \Delta^2 (u+1)^2}{\sigma^2} \text{ for each } i \neq j \in [N] \times [N].$$

Therefore, using (6.2) and (6.3) we arrive at

$$\mathcal{R}_T(\mathcal{P};\phi\circ\rho) \geq \phi(u\Delta)[1-\frac{n_S\frac{32\left\|\Sigma_S^{\frac{1}{2}}\Sigma_T^{-\frac{1}{2}}\right\|^2\Delta^2(u+1)^2}{\sigma^2}+n_T\frac{2\Delta^2(5+4u)^2}{\sigma^2}+\log 2}{rk\log 2}]$$
$$= (u\Delta)^2[1-\frac{n_Sr_S\frac{32\Delta^2(u+1)^2}{\sigma^2}+n_Tr_T\frac{2\Delta^2(5+4u)^2}{\sigma^2}+\log 2}{rk\log 2}].$$

The above inequality holds for every $u\geq 0$. Maximizing the expression above over u we can conclude if $\Delta\leq\sqrt{\frac{\sigma^2(rk-1)\log 2}{32n_Sr_S+50n_Tr_T}}$ then

$$\mathcal{R}_T(\mathcal{P};\phi\circ\rho) \geq (u\Delta)^2[1-\frac{n_Sr_S\frac{32\Delta^2(u+1)^2}{\sigma^2}+n_Tr_T\frac{2\Delta^2(5+4u)^2}{\sigma^2}+\log 2}{rk\log 2}]$$

where $u=\frac{3\Delta(4n_Sr_S+n_Tr_T)+\sqrt{\Delta^2[16(n_Sr_S)^2+25(n_Tr_T)^2+32n_Sr_Sn_Tr_T]+4(n_Sr_S+n_Tr_T)(rk-1)\sigma^2\log 2}}{16\Delta(n_{\mathbb{P}}r_{\mathbb{P}}+n_{\mathbb{Q}})}$.

Now, we need to simplify the above expressions. First note that

$$(u\Delta)\geq(\frac{3\Delta}{16}+\frac{\sqrt{\Delta^2+4\frac{D\sigma^2\log 2}{n_Sr_S+n_Tr_T}}}{16}),$$

so

$$(u\Delta)^2\geq\frac{\Delta^2+2.7\frac{D\sigma^2}{n_Sr_S+n_Tr_T}}{256}. \tag{7.10}$$

Moreover,

$$1-\frac{n_Sr_S\frac{32\Delta^2(u+1)^2}{\sigma^2}+n_T\frac{2\Delta^2(5+4u)^2}{\sigma^2}+\log 2}{rk\log 2}\geq 1-\frac{[n_Sr_S+n_Tr_T]\frac{32\Delta^2(\frac{5}{4}+u)^2}{\sigma^2}+\log 2}{D\log 2}$$

and

$$\Delta(\frac{5}{4}+u)\leq 2\Delta+\frac{1}{16}\sqrt{25\Delta^2+\frac{4\log 2D\sigma^2}{n_Sr_S+n_Tr_T}}.$$

Since $\Delta\leq\frac{1}{45}\sqrt{\frac{\sigma^2D}{n_Sr_S+n_Tr_T}}$,

$$\begin{aligned}\Delta^2(\frac{5}{4}+u)^2 &\leq (4+(\frac{5}{16})^2)\Delta^2+\frac{1}{4}\Delta\sqrt{25\Delta^2+\frac{4\log 2D\sigma^2}{n_Sr_S+n_Tr_T}}\\ &\leq (4+(\frac{5}{16})^2)\frac{1}{45^2}\frac{D\sigma^2}{n_Sr_S+n_Tr_T}+\frac{1}{4\times 45^2}\sqrt{25^2+45^2\times 4\log 2}\frac{D\sigma^2}{n_Sr_S+n_Tr_T}\\ &\leq 0.012\frac{D\sigma^2}{n_Sr_S+n_Tr_T}.\end{aligned}$$

Hence,

$$\begin{aligned}1-\frac{[n_Sr_S+n_Tr_T]\frac{32\Delta^2(\frac{5}{4}+u)^2}{\sigma^2}+\log 2}{D\log 2} &\geq 1-0.56-\frac{1}{D}\\ &\geq 0.39.\end{aligned}$$

Therefore, we arrive at

$$\mathcal{R}_T(\mathcal{P};\phi\circ\rho)\geq\frac{\Delta^2}{1000}+\frac{6}{1000}\frac{D\sigma^2}{n_Sr_S+n_Tr_T}. \tag{7.11}$$

7.5 Proof of Theorem 1 (One-hidden layer neural network with fixed hidden-to-output layer)

By Proposition 1, the generalization error is bounded from below as

$$\mathbb{E}_{\mathbb{Q}_{\boldsymbol{\theta}_T}}[\|\widehat{\boldsymbol{y}}_T - \boldsymbol{y}_T\|_{\ell_2}^2] \geq \frac{1}{4}\sigma_{\min}^2(\boldsymbol{V})\|\boldsymbol{\Sigma}_T^{\frac{1}{2}}(\widehat{\boldsymbol{W}}_T - \boldsymbol{W}_T)^T\|_F^2 + k\sigma^2.$$

Therefore, it suffices to find a lower bound for the following quantity:

$$\begin{aligned}&\mathcal{R}_T(\mathcal{P}_\Delta; \phi \circ \rho)\\ &\quad := \inf_{\widehat{\boldsymbol{W}}_T} \sup_{(\mathbb{P}_{\boldsymbol{W}_S}, \mathbb{Q}_{\boldsymbol{W}_T}) \in \mathcal{P}_\Delta} \mathbb{E}_{S_{\mathbb{P}_{\boldsymbol{W}_S}} \sim \mathbb{P}_{\boldsymbol{W}_S}^{1:n_{\mathbb{P}}}} \Big[\mathbb{E}_{S_{\mathbb{Q}_{\boldsymbol{W}_T}} \sim \mathbb{Q}_{\boldsymbol{W}_T}^{1:n_{\mathbb{Q}}}} \Big[\phi(\rho(\widehat{\boldsymbol{W}}_T(S_{\mathbb{P}_{\boldsymbol{W}_S}}, S_{\mathbb{Q}_{\boldsymbol{W}_T}}), \boldsymbol{W}_T))\Big]\Big]\end{aligned}$$

where $\phi(x) = x^2$ for $x \in \mathbb{R}$ and ρ is defined per Definition 1. The rest of the proof is similar to the linear case as the corresponding transfer distance metrics are the same. We only need to upper bound the corresponding KL-divergences in this case. We do so by the following lemma.

Lemma 4 *Suppose that $\mathbb{P}_{\boldsymbol{W}_S^{(i)}}$ and $\mathbb{P}_{\boldsymbol{W}_S^{(j)}}$ are the joint distributions of features and labels in a source task and $\mathbb{Q}_{\boldsymbol{W}_T^{(i)}}$ and $\mathbb{Q}_{\boldsymbol{W}_T^{(j)}}$ are joint distributions of features and labels in a target task as defined in Section 2.1 in the one-hidden layer neural network with fixed hidden-to-output layer model. Then $D_{KL}(\mathbb{P}_{\boldsymbol{W}_S^{(i)}} \| \mathbb{P}_{\boldsymbol{W}_S^{(j)}}) \leq \frac{\|\boldsymbol{V}\|^2 \|\boldsymbol{\Sigma}_S^{\frac{1}{2}}(\boldsymbol{W}_S^{(i)} - \boldsymbol{W}_S^{(j)})^T\|_F^2}{2\sigma^2}$ and $D_{KL}(\mathbb{Q}_{\boldsymbol{W}_T^{(i)}} \| \mathbb{Q}_{\boldsymbol{W}_T^{(j)}}) \leq \frac{\|\boldsymbol{V}\|^2 \|\boldsymbol{\Sigma}_S^{\frac{1}{2}}(\boldsymbol{W}_T^{(i)} - \boldsymbol{W}_T^{(j)})^T\|_F^2}{2\sigma^2}$.*

Furthermore, we also note that since in this case $\boldsymbol{W}_S, \boldsymbol{W}_T \in \mathbb{R}^{\ell \times d}$, the definition of D is slightly different from that in the linear case. In this case $D = \text{rank}(\boldsymbol{\Sigma}_T)\ell - 1$.

7.6 Bounding the KL-Divergences in the Neural Network Model (Proof of Lemma 4)

First we compute the KL-divergence between the distributions $\mathbb{P}_{\boldsymbol{W}_S^{(i)}}(\boldsymbol{x}_S, \boldsymbol{y}_S)$ and $\mathbb{P}_{\boldsymbol{W}_S^{(j)}}(\boldsymbol{x}_S, \boldsymbol{y}_S)$

$$\begin{aligned}D_{KL}(\mathbb{P}_{\boldsymbol{W}_S^{(i)}}(\boldsymbol{x}_S, \boldsymbol{y}_S), \mathbb{P}_{\boldsymbol{W}_S^{(j)}}(\boldsymbol{x}_S, \boldsymbol{y}_S)) =& D_{KL}(\mathbb{P}_{\boldsymbol{W}_S^{(i)}}(\boldsymbol{x}_S), \mathbb{P}_{\boldsymbol{W}_S^{(j)}}(\boldsymbol{x}_S))\\ &+ \mathbb{E}[D_{KL}(\mathbb{P}_{\boldsymbol{W}_S^{(i)}}(\boldsymbol{y}_S|\boldsymbol{x}_S), \mathbb{P}_{\boldsymbol{W}_S^{(j)}}(\boldsymbol{y}_S|\boldsymbol{x}_S))].\end{aligned}$$

The marginal distributions $\mathbb{P}_{\boldsymbol{M}_S^{(i)}}(\boldsymbol{x}_S)$ and $\mathbb{P}_{\boldsymbol{M}_S^{(j)}}(\boldsymbol{x}_S)$ are equal so their KL-divergence is zero. The conditional distributions $\mathbb{P}_{\boldsymbol{M}_S^{(i)}}(\boldsymbol{y}_S|\boldsymbol{x}_S)$ and $\mathbb{P}_{\boldsymbol{M}_S^{(j)}}(\boldsymbol{y}_S|\boldsymbol{x}_S)$ are normally distributed with covariance matrix $\sigma^2\mathbf{I}_k$ and with mean respectively equal to $\boldsymbol{V}\varphi(\boldsymbol{W}_S^{(i)}\boldsymbol{x}_S)$ and $\boldsymbol{V}\varphi(\boldsymbol{W}_S^{(j)}\boldsymbol{x}_S)$. Therefore, we obtain

$$D_{KL}(\mathbb{P}_{\boldsymbol{W}_S^{(i)}}(\boldsymbol{y}_S|\boldsymbol{x}_S), \mathbb{P}_{\boldsymbol{W}_S^{(j)}}(\boldsymbol{y}_S|\boldsymbol{x}_S)) = \frac{\left\|\boldsymbol{V}\varphi(\boldsymbol{W}_S^{(i)}\boldsymbol{x}_S) - \boldsymbol{V}\varphi(\boldsymbol{W}_S^{(j)}\boldsymbol{x}_S)\right\|_{\ell_2}^2}{2\sigma^2}.$$

Then we have

$$\begin{aligned}D_{KL}(\mathbb{P}_{\boldsymbol{W}_S^{(i)}}(\boldsymbol{x}_S, \boldsymbol{y}_S), \mathbb{P}_{\boldsymbol{W}_S^{(j)}}(\boldsymbol{x}_S, \boldsymbol{y}_S)) &= \frac{\mathbb{E}\left\|\boldsymbol{V}\varphi(\boldsymbol{W}_S^{(i)}\boldsymbol{x}_S) - \boldsymbol{V}\varphi(\boldsymbol{W}_S^{(j)}\boldsymbol{x}_S)\right\|_{\ell_2}^2}{2\sigma^2}\\ &\leq \frac{\|\boldsymbol{V}\|^2 \|\boldsymbol{\Sigma}_S^{\frac{1}{2}}(\boldsymbol{W}_S^{(i)} - \boldsymbol{W}_S^{(j)})^T\|_F^2}{2\sigma^2}.\end{aligned}$$

Since ReLU is a Lipschitz function. Similarly we get

$$\begin{aligned}D_{KL}(\mathbb{Q}_{\boldsymbol{W}_T^{(i)}}(\boldsymbol{x}_T, \boldsymbol{y}_T), \mathbb{Q}_{\boldsymbol{W}_T^{(j)}}(\boldsymbol{x}_T, \boldsymbol{y}_T)) &= \frac{\mathbb{E}\|\boldsymbol{V}\varphi(\boldsymbol{W}_T^{(i)}\boldsymbol{x}_T) - \boldsymbol{V}\varphi(\boldsymbol{W}_T^{(j)}\boldsymbol{x}_T)\|_F^2}{2\sigma^2}\\ &\leq \frac{\|\boldsymbol{V}\|^2 \|\boldsymbol{\Sigma}_T^{\frac{1}{2}}(\boldsymbol{W}_T^{(i)} - \boldsymbol{W}_T^{(j)})^T\|_F^2}{2\sigma^2}.\end{aligned}$$

7.7 Proof of Theorem 1 (One-hidden layer neural network model with fixed input-to-hidden layer)

By Proposition 1, the generalization error is $\mathbb{E}_{\mathbb{Q}_{\boldsymbol{\theta}_T}}[\|\widehat{\boldsymbol{y}}_T - \boldsymbol{y}_T\|_{\ell_2}^2] = \|\widetilde{\boldsymbol{\Sigma}}_T^{\frac{1}{2}}(\widehat{\boldsymbol{V}}_T - \boldsymbol{V}_T)^T\|_F^2 + k\sigma^2$ so it suffices to find a lower bound for the following quantity

$$\mathcal{R}_T(\mathcal{P}_\Delta; \phi \circ \rho) := \inf_{\widehat{\boldsymbol{V}}_T} \sup_{(\mathbb{P}_{\boldsymbol{V}_S}, \mathbb{Q}_{\boldsymbol{V}_T}) \in \mathcal{P}_\Delta} \mathbb{E}_{S_{\mathbb{P}_{\boldsymbol{V}_S}} \sim \mathbb{P}_{\boldsymbol{V}_S}^{1:n_{\mathbb{P}}}} \Big[\mathbb{E}_{S_{\mathbb{Q}_{\boldsymbol{V}_T}} \sim \mathbb{Q}_{\boldsymbol{V}_T}^{1:n_{\mathbb{Q}}}} \big[\phi(\rho(\widehat{\boldsymbol{V}}_T(S_{\mathbb{P}_{\boldsymbol{V}_S}}, S_{\mathbb{Q}_{\boldsymbol{V}_T}}), \boldsymbol{V}_T)) \big] \Big]$$

Where $\phi(x) = x^2$ for $x \in \mathbb{R}$ and ρ is defined per Definition 1. Inherently, this case is the same as the linear model except that the distribution of the features has changed which was calculated in (7.7). The rest of the proof is similar to the linear case with the difference that $\boldsymbol{\Sigma}_S, \boldsymbol{\Sigma}_T$ should be replaced by $\widetilde{\boldsymbol{\Sigma}}_S, \widetilde{\boldsymbol{\Sigma}}_T$.

8 Appendix B

In this section we apply our theorem on DomainNet clipart and DomainNet sketch datasets to find a lower bound for target generalization error.

DomainNet dataset. First we pass the images of DomainNet-Clipart and DomainNet-Sketch through a ResNet-101 network pretrained on ImageNet with the fully connected top classifier removed and use the extracted features instead of the actual images.

Training. We trained a one-hidden layer neural network with 2048 input neurons, 40000 hidden neurons, and 345 output neurons for DomainNet-Clipart and DomainNet-Sketch dataset. We tuned the number of hidden neurons in order to get the best performance on the test dataset. We used MSE loss with one-hot encoded labels for training the networks. the trained network on DomainNet-Clipart has 98.4% train accuracy and 65.7% test accuracy and and the one trained on DomainNet-Sketch has 98.9% train accuracy and 51.2% test accuracy. Then we used the trained weights to estimate/calculate the parameters appearing in our lower bound. The noise levels are calculated based on the average loss of the trained ground truth models on the test dataset (note that this average loss equals $k\sigma^2 = 345\sigma^2$). Moreover, we estimated/calculated r_s, r_T, and the transfer distance, i.e. Δ as reported in Table 2.

Δ	$k\sigma^2$	r_S	r_T	$\sigma_{\min}^2(\boldsymbol{V})$
202.15	.54	2.247	2.58	.249

Table 2: Estimated parameters of Theorem 1 for DomainNet clipart and Sketch datasets.

Lower bound. Using the values of Table 2, we get that $\Delta \geq \sqrt{\frac{D\sigma^2 \log 2}{r_T n_T}}$. Theorem 1 gives a lower bound for the generalization error which is illustrated in Figure 8. In Figure 9 we plot an upper bound for this dataset. The upper bound is obtained by empirical risk minimzation simply over target samples. In Table 3 we provide some exact values of upper and lower bound.

Number of target samples	Lower bound	Upper bound
5000	0.5424	1.0076
8150	0.5415	0.8573
15500	0.5408	0.7518
22150	0.5405	0.7266

Table 3: Some examples of exact values of lower and upper bound.

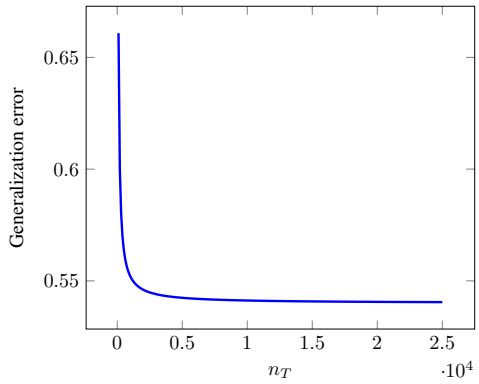


Figure 8: Theoretical lower bound on target generalization error as a function of the number of target samples for DomainNet dataset.

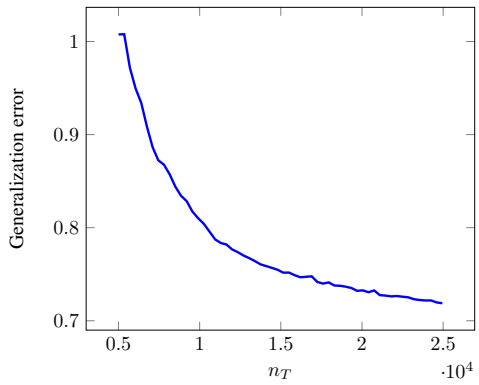


Figure 9: Upper bound on target generalization error as a function of the number of target samples for DomainNet dataset. The upper bound is obtained by simply minimizing the empirical risk over target samples.