arXiv:2307.06887v1 [cs.LG] 13 Jul 2023

Provable Multi-Task Representation Learning by Two-Layer ReLU Neural Networks

Liam Collins[*], Hamed Hassani[†], Mahdi Soltanolkotabi[‡],
Aryan Mokhtari[*], Sanjay Shakkottai[*]

Abstract

Feature learning, i.e. extracting meaningful representations of data, is quintessential to the practical success of neural networks trained with gradient descent, yet it is notoriously difficult to explain how and why it occurs. Recent theoretical studies have shown that shallow neural networks optimized on a single task with gradient-based methods can learn meaningful features, extending our understanding beyond the neural tangent kernel or random feature regime in which negligible feature learning occurs. But in practice, neural networks are increasingly often trained on *many* tasks simultaneously with differing loss functions, and these prior analyses do not generalize to such settings. In the multi-task learning setting, a variety of studies have shown effective feature learning by simple linear models. However, multi-task learning via *nonlinear* models, arguably the most common learning paradigm in practice, remains largely mysterious. In this work, we present the first results proving feature learning occurs in a multi-task setting with a nonlinear model. We show that when the tasks are binary classification problems with labels depending on only r directions within the ambient $d \gg r$-dimensional input space, executing a simple gradient-based multitask learning algorithm on a two-layer ReLU neural network learns the ground-truth r directions. In particular, any downstream task on the r ground-truth coordinates can be solved by learning a linear classifier with sample and neuron complexity independent of the ambient dimension d, while a random feature model requires exponential complexity in d for such a guarantee.

[*]Chandra Family Department of Electrical and Computer Engineering, The University of Texas at Austin, Austin, TX, USA. {liamc@utexas.edu, mokhtari@austin.utexas.edu, sanjay.shakkottai@utexas.edu}.

[†]Department of Electrical and Systems Engineering, University of Pennsylvania, Philadelphia, PA, USA. {hassani@seas.upenn.edu}

[‡]Ming Hsieh Department of Electrical and Computer Engineering, University of Southern California, Los Angeles, CA, USA. {soltanol@usc.edu}

1 Introduction

Recent empirical results have demonstrated huge successes in pre-training large neural networks on many tasks with gradient-based algorithms [Aghajanyan et al., 2021, Zhong et al., 2021, Crawshaw, 2020, Zhang and Yang, 2021, Sanh et al., 2021, Wang et al., 2022, 2023, Daras et al., 2022, Wei et al., 2021, Bubeck et al., 2023]. These works suggest that the generalization capabilities of the pre-trained representation scales with the number of tasks used for pre-training.

At the same time, significant progress has been made in theoretically understanding the dynamics of neural networks trained with gradient-based methods in recent years, especially in regards to proving that shallow neural networks can learn meaningful features when trained with gradient descent and its variants Damian et al. [2022], Shi et al. [2022], Abbe et al. [2022], Allen-Zhu et al. [2019a], Bai and Lee [2019], Li et al. [2020], Daniely and Malach [2020], Barak et al. [2022], Telgarsky [2022], Akiyama and Suzuki [2022], Zhou et al. [2021]. However, these results are limited to single-task settings, so they cannot explain the practical improvements in model performance seen by pre-training on many tasks. While a few studies exist showing the benefit of multi-task pre-training with gradient-based algorithms to representation learning Thekumparampil et al. [2021], Collins et al. [2022, 2021], Saunshi et al. [2021], Sun et al. [2021], Chua et al. [2021], Bullins et al. [2019], Chen et al. [2022], these analyses are limited to linear models and therefore it is not clear whether these results can generalize to even simple non-linear neural networks.

In this work, we aim to bridge this gap by analyzing the dynamics of the parameters of a two-layer ReLU network pre-trained with a generic gradient-based multi-task learning algorithm on many binary classification tasks. Following a long line of works analyzing feature learning dynamics in single-task Damian et al. [2022], Shi et al. [2022], Abbe and Sandon [2020], Abbe et al. [2022], Daniely and Malach [2020], Telgarsky [2022], Maurer et al. [2016], Du et al. [2020] and linear [Thekumparampil et al., 2021, Collins et al., 2022, 2021, Saunshi et al., 2021, Sun et al., 2021, Chua et al., 2021, Bullins et al., 2019, Chen et al., 2022, Tripuraneni et al., 2021] settings, we suppose the existence of a ground-truth low-dimensional subspace that preserves all information in the data relevant to its label. Furthermore, in the multi-task setting of this paper we assume this subspace is shared among all tasks. We ask whether a variant of gradient descent applied to this multi-task setting can learn a representation that projects input data onto the ground-truth subspace. Learning such a representation is a proxy for successful pre-training, since it allows the model to easily be fine-tuned to solve downstream tasks by optimizing a number of last-layer model parameters that scales only with the dimension of this low-dimensional subspace, rather than the potentially very large ambient dimension of the data.

We prove that gradient-based multi-task learning with a two-layer ReLU network with first-layer parameters (the representation) shared among all tasks and last-layer weights (the head) learned uniquely for each task is indeed able to recover the ground-truth subspace. In summary, our main contributions are as follows:

- **Proof of multi-task representation learning with two-layer ReLU network.** We consider binary classification tasks whose labels are functions of only r features of the input data, where r is much smaller than the ambient dimension d. We prove that one gradient descent update of each of the task-specific heads followed by one gradient descent update of the representation removes all spurious features and leads to a dense projection onto the ground-

truth r-dimensional feature space as long as the number of pre-training tasks $T \gg r2^{2r}d^2$.

- **Insight that task specific heads induce a contrastive loss that facilitates representation learning.** Key to our proof is showing that updating task-specific heads prior to the representation induces a contrastive loss function of the representation, which encourages learning the ground-truth features in order to align points likely to share a label on a randomly drawn task.

- **Downstream guarantees.** We show that we can add a random ReLU layer on top of the pre-trained representation, then train a linear layer on top of this random layer with finite samples, to solve *any* downstream task with binary labels that are a function of the r important features. Crucially, we prove that the sample and neuron complexity required for solving the downstream task are independent of the ambient dimension d, making it exponentially smaller in d than the complexity required for solving such tasks with a random feature representation (see Theorems 4.3 and 4.6).

Notations. We uppercase boldface to denote matrices, lowercase boldface to denote vectors, standard typeface to denote scalars. We use $\mathbf{x}_{:r}$ to denote the first r elements of the vector $\mathbf{x}$, and $\mathbf{x}_{r:}$ to denote the remaining elements. We use $\text{Unif}(\mathcal{T})$ to denote the uniform distribution over the set $\mathcal{T}$. We use $\mathbf{0}_d$ to denote the d-dimensional vector of zeros, $\mathbf{I}_d$ to denote the d-by-d-dimensional identity matrix, and $\mathcal{N}(\mathbf{0}_d, \mathbf{I}_d)$ to denote the d-dimensional multivariate normal distribution.

1.1 Related Work

Single-task learning with neural networks. A plethora of works have studied the behavior of gradient-based algorithms for optimizing neural networks on single tasks in recent years. Many of these studies consider the neural tangent kernel (NTK) regime [Jacot et al., 2018, Arora et al., 2019, Du et al., 2019, Allen-Zhu et al., 2019b, Oymak and Soltanolkotabi, 2020, Ji and Telgarsky, 2019, Li and Liang, 2018, Du et al., 2020, Zou et al., 2018, Lee et al., 2019], in which a large initialization and small step size mean that early layer model weights barely change during training, so the algorithm dynamics reduce to those of linear regression on fixed features. We are interested in the feature learning regime of training neural networks, wherein the representation weights change significantly and the dynamics are nonlinear.

Numerous works study feature learning in neural networks, but the vast majority consider optimizing only a single task from a particular class of functions Allen-Zhu et al. [2019a], Bai and Lee [2019], Li et al. [2020], Daniely and Malach [2020], Barak et al. [2022], Telgarsky [2022], Akiyama and Suzuki [2022], Zhou et al. [2021], Abbe et al. [2022]. Ba et al. [2022] study how one step of SGD improves two-layer network performance with respect to a teacher model when compared to random features as long as a large learning rate is used. Similarly, Lewkowycz et al. [2020], Li et al. [2019], Zhu et al. [2022] argue that large learning rate is the key to feature learning, while Yang and Hu [2021] proposes that initializing near zero is critical. We adopt the large learning rate and small initialization in our study. Moreover, Frei et al. [2022] show that ReLU nonlinearity is instrumental to learning robust features with a two-layer model.

A variety of works study whether neural networks can learn low-dimensional representations that generalize to many tasks via training on a single task. Mousavi-Hosseini et al. [2022] show that

stochastic gradient descent on a two-layer ReLU network with arbitrary width can learn a single-index task ($r = 1$). Damian et al. [2022] show that gradient-based methods on two-layer networks can learn r-index polynomials by learning a projection onto the r relevant directions. The closest analysis to ours is due to Shi et al. [2022], who consider a similar data generative model and loss function to show two layer ReLU networks can learn functions of the parity of r inputs. However, unlike our work, their analysis leverages activation noise and the learned features do not allow for learning more general functions of r inputs. Additional works consider single-task feature learning in the mean-field regime with infinitely-wide networks [Chizat and Bach, 2018, Mei et al., 2018, Sirignano and Spiliopoulos, 2020, Nguyen, 2019].

Feature learning in multi-task settings. Other works have considered feature learning in multi-task settings. A popular theme among these works is to assume the existence of a ground-truth low-dimensional data representation shared among all tasks, and to show that the algorithm Thekumparampil et al. [2021], Collins et al. [2022, 2021], Saunshi et al. [2021], Sun et al. [2021], Chua et al. [2021], Bullins et al. [2019], Chen et al. [2022] in consideration recovers this representation. However, all of these of these algorithmic analyses are limited to linear models. A small number of works have studied nonlinear multitask settings. Wu et al. [2020] studies a two-layer ReLU model from statistical perspective and provides a result showing when mutli-task learning on two tasks transfers positively to a third task. Huang et al. [2021] studies the dynamics of FedAvg McMahan et al. [2017], a multi-task learning algorithm used for federated learning, but in the NTK regime, meaning they did not consider feature learning. Deng et al. [2022] shows that FedAvg converges to global optima of the training loss for L-layer ReLU models, but they also do not analyze feature learning. [Kao et al., 2021] empirically noticed a similar phenomenon as us in that adapting task-specific heads to each task induces a contrastive loss, albeit in the context of studying a particular meta-learning algorithm. Finally, Maurer et al. [2016], Tripuraneni et al. [2021], Du et al. [2020], Tripuraneni et al. [2020], Xu and Tewari [2021] give statistical bounds on the downstream task loss after learning a task-specific head on top of a (possibly nonlinear) representation learned by solving an empirical risk minimization problem over a set of pre-training tasks, but do not consider the transferability of representations returned by gradient-based algorithms.

2 Formulation

In section we formally define the multi-task learning problem and the algorithms we analyze in Section 4. We are motivated by classification problems in which the input data for all tasks share a common, small set of features that determine their label, but the mapping from these features to labels differs across tasks. Ultimately, the multi-task learner aims to leverage the large set of available pre-training tasks to learn a representation that recovers the small set of label-relevant features to allow for strong performance on downstream tasks.

2.1 Pre-training tasks and data generating model

To model such settings, we consider pre-training on a set of T binary classification tasks drawn uniformly from a collection of tasks denoted by $\mathcal{T}$. These tasks consist of a distribution on $\mathcal{X} \times \mathcal{Y}$, where the input space is $\mathcal{X} = \mathbb{R}^d$ and the label space is $\mathcal{Y} = \{-1, 1\}$. We assume that the labelling function for each task i is a function of the sign of the first r coordinates of $\mathbf{x}$, denoted by $\mathbf{x}_{:r}$. More

precisely, if we define $f_i^* : \mathcal{X} \to \mathcal{Y}$ as the labelling function of task i, then for any sample $\mathbf{x} \in \mathcal{X}$, where here we assume $\mathbf{x}$ is drawn from the standard Gaussian distribution, we have

$$\mathbf{x} \sim \mathcal{N}(\mathbf{0}_d, \mathbf{I}_d); \quad f_i^*(\mathbf{x}) = \bar{f}_i^*(\text{sign}(\mathbf{x}_{:r})) \in \{-1, 1\}. \tag{1}$$

Here, $\bar{f}_i^* : \{-1, 1\}^r \to \{-1, 1\}$ is the ground-truth labelling function for task i that only depends on the sign of the first r coordinates of $\mathbf{x}$. As a learner, for each task i among the given tasks, we have access to samples pair of the form $(\mathbf{x}, f_i^*(\mathbf{x}))$. This setting is similar to the sparse coding model studied in Shi et al. [2022], except the fact that we assume the input data is defined on $\mathbb{R}^d$, while in Shi et al. [2022] the authors assume the input data is defined over the hypercube in $\mathbb{R}^d$. By assuming that the labelling function is a function of "sign" of the first r coordinates, we make our model more similar to the one in Shi et al. [2022], as they assume the label for each task is a function of the first r coordinates of the integer input data.

Critically, the set of label-relevant coordinates of the input data are shared among all tasks, while the functions $\bar{f}_i^*$ mapping from these coordinates to labels are specific to each task. Thus, the goal of the multi-task learner is to recover the shared label-relevant coordinates, so that it may solve a downstream task with complexity scaling only with the number r of label-relevant coordinates, rather than the much larger ambient dimension of the data d.

This model draws inspiration from realistic classification tasks in which labels are functions of the presence (or lack thereof) of a small number of features in the data. For example, whether or not a brain MRI reveals cancerous tissue depends on the presence of tumor-shaped structures in the image, indicated by a small number of features relative to the dimension of the MRI. While prior work has used such settings to motivate studying the dynamics of neural network training with a similar data model with discrete input vectors in the single-task setting [Shi et al., 2022, Wen and Li, 2021], our model captures more general scenarios in which features appear in the data on a continuous spectrum, while the label is a quantization of these continuous features.

The generative model in (1) implies that, for any task in $\mathcal{T}$, the label of samples are a function of the first r coordinates. However, in order to recover all the r label-relevant coordinates, we must ensure there is sufficient diversity in the labels. To this aim we impose a condition that can be interpreted as task diversity. Specifically, our condition ensures that for any pair of points with different sign patterns on their r label-relevant features, it is equally likely for them to have the same label as it is for them to have different labels on a task drawn from $\mathcal{T}$. Next, we formalize this assumption.

Assumption 2.1. *For any two points $\mathbf{x}, \mathbf{x}' \in \mathcal{X}$ such that $\text{sign}(\mathbf{x}_{:r}) \neq \text{sign}(\mathbf{x}'_{:r})$, the probability that the labels of $\mathbf{x}$ and $\mathbf{x}'$ are the same for a task drawn uniformly from $\mathcal{T}$ satisfies:*

$$\mathbb{P}_{i \sim \text{Unif}(\mathcal{T})} \left[\bar{f}_i^*(\text{sign}(\mathbf{x}_{:r})) = \bar{f}_i^*(\text{sign}(\mathbf{x}'_{:r})) \right] = \tfrac{1}{2}$$

Assumption 2.1 enforces task diversity by ensuring that all pairs of distinct sign inputs are equally likely to have distinct labels as not – i.e. there is no subset of the r coordinates that is more indicative of the inputs sharing a label than the other coordinates. Such an assumption is necessary because, in the extreme case, if all tasks could be written as functions of only the first coordinate, it would be impossible to recover the remaining $r - 1$ coordinates. Thus, we need the collection of possible training tasks to be sufficiently diverse such that all r coordinates are equally relevant to the label, on average across tasks (i.e. one coordinate may be irrelevant to the label for some task, but it must be relevant for others). We note that, strict equality in the assumption is not

necessary, but made for convenience. Indeed, we only sample $T < \infty$ tasks from $\mathcal{T}$ for training, so our empirical task distribution is not guaranteed to satisfy Assumption 2.1 with strict equality. To satisfy the assumption, we can set $\mathcal{T}$ to be the set of all possible labellings of the 2^r inputs on the r-dimensional hypercube, noting that our results do not require T to scale with $\mathcal{T}$.

2.2 Learning model and loss

We consider pre-training a two-layer neural network $f(\cdot; \mathbf{W}, \mathbf{b}, \mathbf{a}) : \mathbb{R}^d \to \mathbb{R}$ with m ReLU-activated neurons in the hidden layer, namely

$$f(\mathbf{x}; \mathbf{W}, \mathbf{b}, \mathbf{a}) := \sum_{j=1}^{m} a_j \sigma(\mathbf{w}_j^\top \mathbf{x} + b_j) \tag{2}$$

where $\sigma(\mathbf{x}) = \max(\mathbf{x}, \mathbf{0})$ element-wise, $\mathbf{w}_j \in \mathbb{R}^d$ and $b_j \in \mathbb{R}$ are the weight vector and bias for the j-th neuron, respectively, and $a_j \in \mathbb{R}$ is the last-layer weight for the j-th neuron. We let $\mathbf{W} = [\mathbf{w}_1, \ldots, \mathbf{w}_m] \in \mathbb{R}^{d \times m}$ denote the matrix of concatenated weight vectors, $\mathbf{b} = [b_1, \ldots, b_m] \in \mathbb{R}^m$ denote the vector of biases, and $\mathbf{a} = [a_1, \ldots, a_m] \in \mathbb{R}^m$ denote the vector of last-layer weights, which we call the head. Solving a task i entails that predictions $f(\mathbf{x}; \mathbf{W}, \mathbf{b}, \mathbf{a})$ align well with the corresponding label $f_i^*(\mathbf{x})$ of $\mathbf{x}$, which we measure using the hinge loss:

$$\ell(f(\mathbf{x}; \mathbf{W}, \mathbf{b}, \mathbf{a}), f_i^*(\mathbf{x})) := \max\left(1 - f_i^*(\mathbf{x}) f(\mathbf{x}; \mathbf{W}, \mathbf{b}, \mathbf{a}), 0\right) \tag{3}$$

The loss for task i as a function of the model parameters $(\mathbf{W}, \mathbf{b}, \mathbf{a})$ is the average point-wise loss over labelled points $(\mathbf{x}, f_i^*(\mathbf{x}))$ drawn from $\mathcal{D}_i$:

$$\mathcal{L}_i(\mathbf{W}, \mathbf{b}, \mathbf{a}) := \mathbb{E}_{(\mathbf{x}, f_i^*(\mathbf{x})) \sim \mathcal{D}_i} \left[\ell(f(\mathbf{x}; \mathbf{W}, \mathbf{b}, \mathbf{a}), f_i^*(\mathbf{x}))\right] \tag{4}$$

Ultimately, the goal of multi-task pre-training is to learn a first-layer representation that generalizes to downstream tasks, in the sense that we can easily train a new classifier on top of the first layer in order to achieve small task-specific loss. To this end, our goal is minimize the following multi-task loss function

$$\min_{\mathbf{W}, \mathbf{b}, \{\mathbf{a}_1, \ldots, \mathbf{a}_T\}} \mathcal{L}(\mathbf{W}, \mathbf{b}, \{\mathbf{a}_i\}_{i=1}^T) := \frac{1}{T} \sum_{i=1}^{T} \mathcal{L}_i(\mathbf{W}, \mathbf{b}, \mathbf{a}_i) + \frac{\lambda_{\mathbf{a}}}{2} \|\mathbf{a}_i\|_2^2 + \frac{\lambda_{\mathbf{w}}}{2} \|\mathbf{W}\|_F^2, \tag{5}$$

where $\lambda_{\mathbf{a}}$ and $\lambda_{\mathbf{w}}$ are regularization parameters. Optimizing $\mathcal{L}$ entails learning task-specific heads on top of a shared representation, a widely used and empirically successful approach to multi-task learning [Zhang and Yang, 2021, Crawshaw, 2020, Ruder, 2017]. Basically, by solving the above problem, we hope to find a set of $(\mathbf{W}, \mathbf{b})$ such that they learn the important features and we can argue that the output of first layer denoted by $[\sigma(\mathbf{w}_1^\top \mathbf{x} + b_1), \ldots, \sigma(\mathbf{w}_m^\top \mathbf{x} + b_m)] \in \mathbb{R}^m$ only contains information about the first r coordinates of $\mathbf{x}$. Next we discuss the optimization algorithms we study.

3 Algorithm

We consider a two-stage learning process: (1) Representation learning, in which we aim to learn effective features using n available tasks, and (2) Downstream evaluation, in which we encounter a new task and aim to efficiently learn an accurate linear classifier on top of the pre-trained features.

Representation learning phase. The multi-task learning algorithm we consider aims to solve the global objective (5) with task-specific heads. To do so, it first executes a symmetric initialization, as in previous studies of feature learning Damian et al. [2022], Shi et al. [2022]. Consider $\mathbf{w}_j^0$ and b_j^0 as the weights and bias of first layer corresponding to the j-th neuron at time 0, respectively, and $a_{i,j}^0$ as the j-th element of the head corresponding to task i at time 0. Further for simplicity assume m is even. Then, for all the weights corresponding to $j \leq \frac{m}{2}$, we set the weights as

$$\mathbf{w}_j^0 \sim \mathcal{N}(\mathbf{0}_d, \nu_{\mathbf{w}}^2 \mathbf{I}_d), \quad a_{i,j}^0 \sim \mathcal{N}(0, \nu_{\mathbf{a}}^2), \quad b_j^0 = 0 \tag{6}$$

where $\nu_{\mathbf{w}}, \nu_{\mathbf{a}} \in \mathbb{R}_{\geq 0}$. Moreover, for the remaining weights that correspond to $j > \frac{m}{2}$, we set the initial weights as $\mathbf{w}_j^0 = \mathbf{w}_{j-m/2}^0$, $b_j^0 = 0$, and $a_{i,j}^0 = -a_{i,j-m/2}^0$ for all tasks $i \in [T]$. Once the random initialization procedure is done, we first optimize the heads, i.e., $\mathbf{a}_1, \ldots, \mathbf{a}_T$. Note that since each $\mathcal{L}_i$ is a strongly convex function with respect to $\mathbf{a}$, this optimization problem can be solved efficiently and the condition number is 1 so that a constant number of iterations can yield a small loss. In fact, we only run one step of gradient descent for each head on the regularized task-specific loss:

$$\mathbf{a}_i^+ = (1 - \eta\lambda_{\mathbf{a}})\mathbf{a}_i^0 - \eta \nabla_{\mathbf{a}} \mathcal{L}_i(\mathbf{W}^0, \mathbf{b}^0, \mathbf{a}_i^0) \quad \forall\, i \in [n] \tag{7}$$

Next, the algorithm updates the model weights with one step of gradient descent on the global loss induced by the task-optimal heads:

$$\mathbf{W}^+ = \mathbf{W}^0 - \frac{\eta}{T} \sum_{i=1}^{T} \nabla_{\mathbf{W}} \mathcal{L}_i(\mathbf{W}^0, \mathbf{b}^0, \mathbf{a}_i^+) \tag{8}$$

Later in Theorem 4.2, we show that one gradient step is sufficient to learn meaningful features. We note that it is standard practice in the theory of feature learning literature to consider only one gradient descent step for the first layer weights [Daniely and Malach, 2020, Abbe et al., 2022, Barak et al., 2022, Damian et al., 2022, Ba et al., 2022]. We do not update the biases during pre-training, as in [Damian et al., 2022]. Note that we only use $\mathbf{W}^+$ for the downstream task as we specify next.

Downstream evaluation phase. After the first phase, we consider learning a prediction function to fit a downstream task. This function is a neural network with two hidden layers where the weights of the first layer are fixed and determined by the output of the first phase, and the parameters of the second hidden layer are set randomly. Intuitively, we add a random second layer on top of the trained first layer so that we can solve a wide range of downstream tasks on the r important coordinates with high probability. Indeed this is necessary for the network g to be able to linearly separate the classes induced any binary function on the r coordinates with high probability, without having to use a very wide first layer. See Section 4 for more details.

In other words, the first hidden layer has m neurons and the weights are a scaled version of $\mathbf{W}^+$ denoted by $\alpha\mathbf{W}^+$ and the bias term is $\mathbf{b}$. Note that here $\alpha > 0$ is a re-scaling factor (see Appendix B for more details on α). The second hidden layer of the classifier has $\hat{m}$ neurons and we denote its weights by $\hat{\mathbf{W}} \coloneqq [\hat{\mathbf{w}}_1, \ldots, \hat{\mathbf{w}}_{\hat{m}}]^\top \in \mathbb{R}^{\hat{m} \times m}$ and its bias by $\hat{\mathbf{b}} \in \mathbb{R}^{\hat{m}}$. Hence, the embedding of these two layers for input $\mathbf{x}$, which we denote by $g(\mathbf{x}) \in \mathbb{R}^{\hat{m}}$ is given by

$$g(\mathbf{x}) = \sigma\left(\hat{\mathbf{W}}\ \sigma\left(\alpha\mathbf{W}^+\mathbf{x} + \mathbf{b}\right) + \hat{\mathbf{b}}\right) \tag{9}$$

Again note that $\mathbf{W}^+$ are fixed and selected in the previous phase; it remains to set $\mathbf{b}$, $\hat{\mathbf{W}}$, and $\hat{\mathbf{b}}$ to create a good embedding for the downstream task. To do so, we randomly set these parameters

according to

$$\mathbf{b} \sim \text{Unif}\left(\left[-\frac{\sqrt{2}\gamma}{\sqrt{m}}, \frac{\sqrt{2}\gamma}{\sqrt{m}}\right]^m\right), \quad \hat{\mathbf{b}} \sim \text{Unif}\left(\left[-\frac{\sqrt{2}\hat{\gamma}}{\sqrt{\hat{m}}}, \frac{\sqrt{2}\hat{\gamma}}{\sqrt{\hat{m}}}\right]^{\hat{m}}\right) \quad \hat{\mathbf{w}}_j \sim \mathcal{N}\left(\mathbf{0}_m, \frac{2}{\hat{m}}\mathbf{I}_m\right),$$

for all $j \in [\hat{m}]$, where $\text{Unif}([a,b]^m)$ denotes the uniform distribution on the box $[a,b]^m \subset \mathbb{R}^m$.

Next, given a set $S_{T+1} \coloneqq \{(\mathbf{x}_l, f^*_{T+1}(\mathbf{x}_l))\}_{\ell=1}^n$ of n i.i.d. samples from a downstream task denoted by $T+1$, we learn a task-specific head $\mathbf{a}$ and a bias term τ by solving the following problem

$$(\mathbf{a}_{T+1}, \tau_{T+1}) \in \underset{\mathbf{a}\in\mathbb{R}^{\hat{m}}, \tau\in\mathbb{R}}{\arg\min} \frac{1}{n}\sum_{l=1}^n \ell(\mathbf{a}^\top g(\mathbf{x}_l) + \tau, f^*_{T+1}(\mathbf{x}_l)) + \frac{\hat{\lambda}_{\mathbf{a}}}{2}\|\mathbf{a}\|_2^2 \tag{10}$$

Once this step is done, we use the learned head, i.e., $\mathbf{a}_{T+1}$, and bias term τ_{T+1} to define the prediction function for task $T+1$. In particular, using the fact that $g(\mathbf{x}) = \sigma\left(\hat{\mathbf{W}}\ \sigma\left(\alpha\mathbf{W}^+\mathbf{x} + \mathbf{b}\right) + \hat{\mathbf{b}}\right)$, our final prediction function is $F(\mathbf{x}\ ;\ \mathbf{a}_{T+1}, \tau_{T+1}, \mathbf{W}^+, \mathbf{b}, \hat{\mathbf{W}}, \hat{\mathbf{b}})$ is given by

$$F(\mathbf{x}\ ;\ \mathbf{a}_{T+1}, \tau, \mathbf{W}^+, \mathbf{b}, \hat{\mathbf{W}}, \hat{\mathbf{b}}) \ :=\ \mathbf{a}_{T+1}^\top \sigma\left(\hat{\mathbf{W}}\ \sigma\left(\alpha\mathbf{W}^+\mathbf{x} + \mathbf{b}\right) + \hat{\mathbf{b}}\right) + \tau_{T+1}$$

For ease of notation, we denote the above function by $F(\mathbf{x})$. Hence, the performance of our prediction function can be evaluated by

$$\mathcal{L}^{(\text{eval})}_{T+1} \coloneqq \mathbb{E}_{(\mathbf{x}, f^*_{T+1}(\mathbf{x}))\sim\mathcal{D}_{T+1}}[\ell(F(\mathbf{x}_l), f^*_{T+1}(\mathbf{x}_l))]. \tag{11}$$

Note that $\mathcal{L}^{(\text{eval})}_{T+1}$ is a random function of $\mathbf{b}, \hat{\mathbf{W}}, \hat{\mathbf{b}}$, and S_{T+1} in addition to the randomness in the representation learning process. As a result, we provide a high probability upper bound on $\mathcal{L}^{(\text{eval})}_{T+1}$ for particular types of downstream tasks in Theorem 4.3.

4 Theoretical Results

Feature learning guarantees. We start by showing that the gradient-based multi-task learning algorithm described in the previous section recovers the ground-truth features. To do this, we first need the following proposition, which shows that the projection of the initial features $\mathbf{W}_0$ onto the subspace spanned by the label-relevant, or ground-truth, features stays roughly the same after one step, while their projection onto the subspace spanned by the spurious features becomes very small.

Here, we let $\Pi_{\|}(\mathbf{W}) \coloneqq \mathbf{W}_{:r}$ denote the projection of the rows of the matrix $\mathbf{W}$ onto the ground-truth subspace (i.e., selecting the first r elements of each row), and $\Pi_{\perp}(\mathbf{W}) \coloneqq \mathbf{W}_{r:}$ denote the projection onto the spurious subspace (the last $d-r$ elements). All proofs are deferred to the Appendix.

Proposition 4.1. *Consider the gradient-based multi-task algorithm described in Section 3 and suppose Assumption 2.1 holds. Further assume $d = \Omega(r^4)$, and let $\eta = \Theta(1)$, $\lambda_{\mathbf{a}} = 1/\eta$, $\lambda_{\mathbf{w}} = (1 + (\eta^2 2^{-r-1})/\pi)/\eta$, and $\nu_{\mathbf{w}} = O(d^{-1.25})$. Then for any $m = o(\exp(d))$, with probability at least* 0.99 *over the random initialization we have*

1. $\frac{1}{\nu_{\mathbf{w}}\sqrt{m}}\|\Pi_{\|}(\mathbf{W}^+) - \Omega(\frac{1}{2^r})\Pi_{\|}(\mathbf{W}^0)\|_2 = O\left(\frac{r^4+\log^4(m)}{2^r d} + \frac{\sqrt{r}d}{\sqrt{T}}\right)$,

2. $\frac{1}{\nu_{\mathbf{w}}\sqrt{m}}\|\Pi_{\perp}(\mathbf{W}^{+})\|_2 = O\left(\frac{r^{3.5}+\log^{3.5}(m)}{2^r d^{1.5}} + \frac{\sqrt{r}d}{\sqrt{T}}\right)$

Proposition 4.1 shows that with high probability over the random initialization, the weights learned by the gradient-based multi-task learning algorithm satisfy two properties, for sufficiently large d and T: (1) the projection of these weights onto the ground-truth subspace is close to a slightly scaled down (by a factor of 2^{-r}) version of their projection at initialization, and (2) their projection onto the spurious subspace is negligible. These two observations, combined with the fact that the neuron weights have independent standard Gaussian initializations, imply that the projection of the neuron weights onto the ground-truth subspace dominates their projection onto the spurious subspace. We formalize this observation in the statement below.

Theorem 4.2. *Consider the setting in Proposition 4.1 and let $d = \Omega(r^4 + \log^4(m))$, $m = \Omega(r)$, and $T = \Omega(r2^{2r}d^2)$. Let $\sigma_r(\mathbf{B})$ denote the r-th singular value of a matrix $\mathbf{B}$. Then with probability at least 0.99 over the random initialization, we have*

$$\frac{\|\Pi_{\perp}(\mathbf{W}^{+})\|_2}{\sigma_r(\Pi_{\|}(\mathbf{W}^{+}))} = O\left(\frac{r^{3.5}+\log^{3.5}(m)}{d^{1.5}} + \frac{2^r\sqrt{r}d}{\sqrt{T}}\right).$$

Theorem 4.2 characterizes the representation learned by multi-task pre-training in an intuitive manner. For $d \gg r^4$ and $T \gg 2^{2r}d^2$, we have $\sigma_r(\Pi_{\|}(\mathbf{W}^{+})) \gg \|\Pi_{\perp}(\mathbf{W}^{+})\|$, meaning that most of the energy in each neuron weight is in the column space of the ground-truth subspace. This is equivalent to saying that applying $\mathbf{W}^{+}$ to an input $\mathbf{x}$ essentially projects it onto the ground-truth subspace spanned by the first r coordinates, as desired.

Downstream performance. Now that we have shown that the learned representation recovers the ground-truth subspace, we use this result to show that the representation generalizes to downstream tasks. We consider downstream tasks with input data sharing the same label-relevant r coordinates as the pre-training tasks, but here the input data is on the hypercube. In particular, input data points $(\mathbf{v}, f^*_{T+1}(\mathbf{v}))$ are generated by the downstream task $T+1$ as follows:

$$\mathbf{v} \sim \mathrm{Unif}(\{-1,1\}^d), \quad f^*_{T+1}(\mathbf{v}) = \bar{f}^*_{T+1}(\mathbf{v}_{:r}) \in \{-1,1\} \tag{12}$$

where, as for pre-training, $\bar{f}^*_{T+1} : \{-1,1\}^r \to \{-1,1\}$. We formally show below that the learned features generalize to *any* such labeling function $\bar{f}^*_{T+1}$.

Theorem 4.3 (End-to-end Guarantee)**.** *Let $\mathbf{W}^{+}$ be the outcome of the multi-task representation learning algorithm described in Section 3, and suppose Assumption 2.1 holds. Consider a downstream task with labeling function $\bar{f}^*_{T+1}$. Construct the two-layer ReLU embedding g using the rescaled $\mathbf{W}^{+}$ for first layer weights as in (9), and train the task-adapted head $(\mathbf{a}_{T+1}, \tau_{T+1})$ using n i.i.d. samples from the downstream task, as in (10). Further, suppose $d = \exp(\tilde{\Omega}(r^5))$, $T = d^3\exp(\tilde{\Omega}(r^5))$, $m = \tilde{\Theta}(r^5)$, and $\hat{m} = \exp\left(\tilde{\Omega}(r^5)\right)$. Then for any $\delta, \delta' \in (0,1)$, with probability at least $0.99 - \delta - \delta'$ over the random initializations, draw of T pre-training tasks, and draw of n samples, we have*

$$\mathcal{L}^{(eval)}_{T+1} = \frac{\exp(\tilde{O}(r^5))}{\sqrt{n}}, \tag{13}$$

where $\tilde{\Omega}(\cdot), \tilde{\Theta}(\cdot)$ and $\tilde{O}(\cdot)$ hide logarithmic factors in r and δ^{-1}.

Theorem 4.3 confirms that the features learned by multi-task pre-training generalize to *any* downstream task mapping from $\{-1,1\}^d$ to $\{-1,1\}$ that has the same representation as the tasks in the training, i.e., its label is a function of the first r coordinates. In particular, if we compose the learned representation with a random ReLU layer, then learn a linear head using $\exp(\tilde{O}(r^5))$ samples from the task, we solve the task with high probability. Critically, the number of samples, pre-training tasks, and neurons needed to ensure the downstream task is solved with high probability are only functions of r and the high-probability constants. In other words, multi-task learning reduces the effective dimension of the downstream task from d to r.

The proof of Theorem 4.3 leverages Proposition 4.1 to show that the embedding g generated by multi-task learning is close to the embedding of a coupled, "purified" two-hidden layer random ReLU network that takes as input *only* the r important coordinates. Then, we apply Theorem 2 from Dirksen et al. [2022] which implies that w.h.p. a random two-hidden layer ReLU network makes two classes of points on the r-dimensional hypercube linearly separable with margin and neuron complexity scaling as functions of the input dimension r.

While Theorem 4.3 ensures generalization to an arbitrary task sharing the ground-truth important coordinates, it can only guarantee generalization to a single such task at once. To ensure that the learned embedding generalizes to a larger set of tasks simultaneously, we have the following corollary.

Corollary 4.4 (Generalization to set of tasks)**.** *Consider a set of possible downstream tasks $\mathcal{T}^{(eval)}$ with $|\mathcal{T}^{(eval)}| = D$. Suppose $d = \exp(\tilde{\Omega}(r^5 \log(D)))$, $T = d^3 \exp(\tilde{\Omega}(r^5 \log(D)))$, $m = \tilde{\Theta}(r^5 \log(D))$, and $\hat{m} = \exp\left(\tilde{\Omega}(r^5 \log(D))\right)$. Construct the two-layer ReLU embedding as in (9). Then with probability at least $0.99 - \delta$, any task $T+1$ in $\mathcal{T}^{(eval)}$ satisfies*

$$\mathcal{L}_{T+1}^{(eval)} = \frac{\exp(\tilde{O}(r^5 \log(D)))}{\sqrt{n}}, \tag{14}$$

where $\tilde{\Omega}(\cdot), \tilde{\Theta}(\cdot)$ and $\tilde{O}(\cdot)$ hide logarithmic factors in r and δ^{-1}.

For example, if $\mathcal{T}^{(\text{eval})}$ is the set of all 2^{2^r} functions from $\{-1,1\}^r$ to $\{-1,1\}$, then Corollary 4.4 shows that multi-task pre-training learns a representation that for all such target functions yields an accurate classifier with $O(\exp(\exp(r)))$ sample and neuron complexity. While this is huge in r, it is still independent of the ambient dimension d.

Lower bound: Random features do not generalize. A consequence of Theorem 4.3 is that multi-task pre-training (albeit with infinite samples) improves the sample and neuron complexity of solving hard downstream tasks by an exponential factor in d. To show this, we consider the well-studied problem of learning sparse parities, which are functions that output the parity of a small subset of input bits. Formally, for $\mathbf{u} \in \{0,1\}^d$ with $\|\mathbf{u}\|_1 = r$, an $(r, \mathbf{u})$-sparse parity task is defined as follows.

Definition 4.5 ($(r, \mathbf{u})$-sparse parity task)**.** *An $(r, \mathbf{u})$-sparse parity task $\mathcal{D}_{\mathbf{u}}$ has data $(\mathbf{v}, f_{\mathbf{u}}^*(\mathbf{v}))$ generated as $\mathbf{v} \sim$ Unif$(\{-1,1\}^d)$ and $f_{\mathbf{u}}^*(\mathbf{v}) = (-1)^{\mathbf{u}^\top \mathbf{v}}$, where $\mathbf{u} \in \{0,1\}^d$ with $\|\mathbf{u}\|_1 = r$.*

In other words, an $(r, \mathbf{u})$-sparse parity task has labels being the parity of the bits selected by the r-sparse vector $\mathbf{u}$. While there is a large literature demonstrating the hardness of learning sparse

parities in various settings Kearns [1998], Abbe and Sandon [2020], Telgarsky [2022], Barak et al. [2022], Malach and Shalev-Shwartz [2022], Kamath et al. [2020], Goel et al. [2019], the most relevant results to our setting show that any data-independent, $\hat{m}$-dimensional embedding of d inputs can admit linear classifiers that solve all $(r, \mathbf{u})$-sparse parity tasks only if the dimension $\hat{m}$ and/or the classification margin is exponentially large in d. In particular, the following result adapts Theorem 5 in Barak et al. [2022], which in turn draws on the works of Kamath et al. [2020] and Malach and Shalev-Shwartz [2022].

Theorem 4.6. *Consider any embedding $\Psi : \{-1,1\}^d \to \mathbb{R}^{\hat{m}}$ such that $\|\Psi(\mathbf{v})\|_2 \leq 1$ for all $\mathbf{v} \in \{-1,1\}^d$. For any $\epsilon > 0$, if $\hat{m}B^2 \leq \epsilon^2 \binom{d}{r}$ then there exists an $(r, \mathbf{u})$-sparse parity task $\mathcal{D}_{\mathbf{u}}$ s.t.*

$$\inf_{\mathbf{a}:\|\mathbf{a}\|_2 \leq B} \mathbb{E}_{(\mathbf{v}, f^*_{\mathbf{u}}(\mathbf{v})) \sim \mathcal{D}_{\mathbf{u}}} \left[\ell \left(\mathbf{a}^\top \Psi(\mathbf{v}), f^*_{\mathbf{u}}(\mathbf{v}) \right) \right] \geq 1 - \epsilon. \tag{15}$$

Theorem 4.6 implies that any random feature model requires a number of neurons and/or samples that is exponentially large in d in order to solve a downstream sparse parity task with a linear classifier. On the other hand, our Theorem 4.2 guarantees that after multi-task pre-training, the output embedding admits a linear classifier that solves the $(r, \mathbf{u})$-sparse parity on the extracted r coordinates, with the number of neurons and samples of the downstream task of the order of $\exp(\text{poly}(r))$.

5 Proof Sketch

In this section we sketch the proof of Proposition 4.1, which is the key feature learning result that enables downstream guarantees. The proof heavily leverages the fact that multi-task pre-training entails updating the first-layer weights *after* fitting a unique head to each task. Surprisingly, we show that solving for the optimal head for each task induces a pseudo-contrastive loss that encourages representations of two points to be similar if and only if they are likely to share a label on a randomly drawn task. Since two points are likely to share a label on a drawn task if and only if they share the same sign pattern for their first r ground-truth coordinates (by Assumption 2.1), the pseudo-contrastive loss inclines the representation to extract the first r latent coordinates.

Step 1: Derive pseudo-contrastive loss after head updates. We first update the task-specific head with one gradient step for each task i given the initial parameters $(\mathbf{W}^0, \mathbf{b}^0, \mathbf{a}_i^0)$. Due to the symmetric initialization, $f(\mathbf{x}; \mathbf{W}^0, \mathbf{b}^0, \mathbf{a}_i^0) = 0$ for all $\mathbf{x} \in \mathbb{R}^d$, so the hinge loss is affine in $\mathbf{a}_i^0$ for all tasks (the $\max(\cdot, 0)$ threshold is inactive). Therefore, using the choice of $\lambda_{\mathbf{a}} = 1/\eta$ and $\mathbf{b}^0 = \mathbf{0}_m$, we have

$$\mathbf{a}_i^+ = (1 - \eta\lambda_{\mathbf{a}})\mathbf{a}_i^0 - \eta \nabla \mathcal{L}_i(\mathbf{W}^0, \mathbf{b}^0, \mathbf{a}_i^0) = \eta \mathbb{E}_{\mathbf{x}}[f_i^*(\mathbf{x})\sigma(\mathbf{W}^0\mathbf{x})]$$

As a result, the updated head for task i, $\mathbf{a}_i^+$, is proportional to the average label-weighted neuron output over the dataset for task i. Now we can insert this value of $\mathbf{a}_i^+$ back into the loss, to obtain $\mathcal{L}_i(\mathbf{W}^0, \mathbf{b}^0, \mathbf{a}_i^+)$. For ease of notation we define $\beta(\mathbf{x}, \mathbf{x}') \coloneqq \frac{1}{n}\sum_{i=1}^n f_i^*(\mathbf{x}) f_i^*(\mathbf{x}')$ for all pairs of inputs $\mathbf{x}, \mathbf{x}'$. Taking the average over all tasks yields

$$\begin{aligned}
\mathcal{L}(\mathbf{W}^0, \mathbf{b}^0, \{\mathbf{a}_i^0\}_i) &= \frac{1}{T}\sum_{i=1}^T \mathbb{E}_{\mathbf{x}} \left[\max \left(1 - \eta f_i^*(\mathbf{x}) \mathbb{E}_{\mathbf{x}'}[f_i^*(\mathbf{x}')\sigma(\mathbf{W}^0\mathbf{x}')]^\top \sigma(\mathbf{W}^0\mathbf{x}), 0 \right) \right] \\
&\approx 1 - \mathbb{E}_{\mathbf{x},\mathbf{x}'} \left[\beta(\mathbf{x}, \mathbf{x}') \sigma(\mathbf{W}^0\mathbf{x}')^\top \sigma(\mathbf{W}^0\mathbf{x}) \right]
\end{aligned} \tag{16}$$

where the approximation holds as $|\eta \mathbb{E}_{\mathbf{x}'}[f_i^*(\mathbf{x}')\sigma(\mathbf{W}^0\mathbf{x}')]^\top \sigma(\mathbf{W}^0\mathbf{x})| < 1$ with high probability over $\mathbf{x}$ due to the initial scaling of $\mathbf{W}^0$. The resulting loss in (16) encourages the first-layer representation that aligns pairs of samples $(\mathbf{x}, \mathbf{x}')$ that typically have the same label among the sampled tasks ($\beta(\mathbf{x}, \mathbf{x}') \approx 1$) and penalizes the representation for aligning pairs of samples that typically do not have the same label ($\beta(\mathbf{x}, \mathbf{x}') \approx -1$). In this way, the loss in (16) behaves similarly to supervised contrastive losses, which have been shown to encourage effective representations in practice Khosla et al. [2020].

To translate these intuitive connections with contrastive learning to feature learning in our setting, we must leverage Assumption 2.1, which implies that, the larger number of pre-training tasks T, the more information that $\beta(\mathbf{x}, \mathbf{x}')$ encodes about the ground-truth features $\mathbf{x}_{:r}$ and $\mathbf{x}'_{:r}$. In particular, pairs of points with the *same* sign patterns among the ground-truth features have the same label, so $\beta(\mathbf{x}, \mathbf{x}') = 1$ almost surely, while pairs of points with *different* sign patterns on the ground-truth features have the same label for only half of the tasks in the universe of tasks $\mathcal{T}$, meaning $\beta(\mathbf{z}, \mathbf{z}') \to 0$ as the number of tasks drawn for training increases. As a result, the loss can now be approximated as:

$$\mathcal{L}(\mathbf{W}^0, \mathbf{b}^0, \{\mathbf{a}_i^0\}_i) \approx -\mathbb{E}_{\mathbf{x},\mathbf{x}'}\left[\chi\{\text{sign}(\mathbf{x}_{:r}) = \text{sign}(\mathbf{x}'_{:r})\}\sigma(\mathbf{W}^0\mathbf{x}')^\top \sigma(\mathbf{W}^0\mathbf{x})\right] \tag{17}$$

where $\chi\{A\}$ is the indicator function for the event A. We next show that taking a gradient descent step with respect to (17) results in $\mathbf{W}^1$ essentially becoming a projection onto the ground-truth first r coordinates.

Step 2: Update neuron weights. The proof of this step requires computing the gradient of $\mathcal{L}(\mathbf{W}^0, \mathbf{b}^0, \{\mathbf{a}_i^0\}_i)$ with respect to $\mathbf{w}_j$ for each neuron $j \in [m]$. Using (17), this gradient can be approximated by $\mathbf{A}(\mathbf{w}_j^0)\mathbf{w}_j^0$, where $\mathbf{A}(\mathbf{w}_j^0) \in \mathbb{R}^{d\times d}$ is defined as:

$$\mathbf{A}(\mathbf{w}_j^0) = -\mathbb{E}_{\mathbf{x},\mathbf{x}'}\left[\chi\{\text{sign}(\mathbf{x}_{:r}) = \text{sign}(\mathbf{x}'_{:r})\}\sigma'((\mathbf{w}_j^0)^\top\mathbf{x})\sigma'((\mathbf{w}_j^0)^\top\mathbf{x}')\mathbf{x}(\mathbf{x}')^\top\right]. \tag{18}$$

where $\sigma'(z) = 1$ if $z > 0$ and $\sigma'(z) = 0$ otherwise. The crucial reason why $\mathbf{A}(\mathbf{w}_j^0)$ has favorable structure is the indicator $\chi\{\text{sign}(\mathbf{x}_{:r}) = \text{sign}(\mathbf{x}'_{:r})\}$ in the RHS of (17). Intuitively, this indicator is encouraging the first-layer weights to align *only* the representations of points with the same sign pattern on the label-relevant coordinates, by ensuring that only these pairs of points appear in the gradient. With this indicator removed, we would have $\mathbf{A}(\mathbf{w}_j^0) = \frac{2}{\pi\|\mathbf{w}_j^0\|_2^2}\mathbf{w}_j^0(\mathbf{w}_j^0)^\top$, meaning the gradient would not put any emphasis on the ground-truth projection. However, the indicator means that $\mathbf{A}(\mathbf{w}_j^0)$ is an average outer product over vectors whose signs agree on the first r coordinates and may disagree on all other $d - r$ coordinates. This disagreement results in cancellation in the average for these $d - r$ coordinates, unlike the first r coordinates, leading to:

$$\mathbf{A}(\mathbf{w}_j^0) \approx 2^{-r-2}\begin{bmatrix} \mathbf{I}_r & \frac{2}{\pi\|\mathbf{w}_{j,r:}^0\|_2^2}\mathbf{w}_{j,:r}^0(\mathbf{w}_{j,r:}^0)^\top \\ \frac{2}{\pi\|\mathbf{w}_{j,r:}^0\|_2^2}\mathbf{w}_{j,r:}^0(\mathbf{w}_{j,:r}^0)^\top & \frac{2}{\pi\|\mathbf{w}_{j,r:}^0\|_2^2}\mathbf{w}_{j,r:}^0(\mathbf{w}_{j,r:}^0)^\top \end{bmatrix} \tag{19}$$

Observe that the minimum eigenvalue of the top left submatrix of $\mathbf{A}(\mathbf{w}_j^0)$ is strictly larger than the maximum eigenvalue of the bottom right submatrix. This shows $\mathbf{w}_j^+$ gains more energy in the ground-truth projection $\mathbf{w}_{j,:r}^0$ than the spurious projection $\mathbf{w}_{j,r:}^0$ by an $\Omega(1)$ factor. Moreover, applying $\mathbf{A}(\mathbf{w}_j^0)$ to $\mathbf{w}_j^0$ does not change the direction of $\mathbf{w}_{j:r}^0$, meaning $\mathbf{w}_{j,:r}^+$ remains isotropic in $\mathbb{R}^r$.

To complete the proof, we use (19) to approximate the updated projections after one gradient update:

$$\mathbf{w}_{j,:r}^{+} \approx \left(1 - \eta\lambda_{\mathbf{w}} + \eta^2 2^{-r-2}\left(1 + \frac{2}{\pi}\right)\right)\mathbf{w}_{j,:r}^{0}, \quad \mathbf{w}_{j,r:}^{+} \approx \left(1 - \eta\lambda_{\mathbf{w}} + \eta^2 2^{-r-2}\frac{2}{\pi}\right)\mathbf{w}_{j,r:}^{0}$$

The coefficient of $\mathbf{w}_{j,:r}^{0}$ in the RHS of the first equation is strictly larger than the coefficient of $\mathbf{w}_{j,r:}^{0}$ in the RHS of the second equation. Thus, with proper regularization, we can approximately eliminate the spurious coordinates while maintaining $\eta\Omega(1)$ energy (and isotropicity) in the ground-truth coordinates with one gradient step.

6 Conclusion

We have taken an initial step towards explaining the practical successes of multi-task representation learning by providing the *first results* showing that multi-task pre-training with a gradient-based algorithm on a non-linear neural network learns generalizable features. Moreover, our analysis reveals that updating the task-specific heads prior to updating the first-layer weights induces a supervised contrastive loss that encourages recovering the features indicative of whether two points share a label. As a result, this work opens up exciting possibilities for exploring the role of adapting the head to each task in order to learn more expressive features.

Acknowledgements

L.C., A.M. and S.S. are supported in part by NSF Grants 2127697, 2019844, 2107037, and 2112471, ARO Grant W911NF2110226, ONR Grant N00014-19-1-2566, the Machine Learning Lab (MLL) at UT Austin, and the Wireless Networking and Communications Group (WNCG) Industrial Affiliates Program. M.S. is supported by the Packard Fellowship in Science and Engineering, a Sloan Research Fellowship in Mathematics, an NSF-CAREER under award 1846369, DARPA FastNICS programs, and NSF-CIF awards 1813877and 2008443. H.H. is supported by the NSF Institute for CORE Emerging Methods in Data Science (EnCORE) as well as The Institute for Learning-enabled Optimization at Scale (TILOS).

References

Emmanuel Abbe and Colin Sandon. Poly-time universality and limitations of deep learning. *arXiv preprint arXiv:2001.02992*, 2020.

Emmanuel Abbe, Enric Boix Adsera, and Theodor Misiakiewicz. The merged-staircase property: a necessary and nearly sufficient condition for sgd learning of sparse functions on two-layer neural networks. In *Conference on Learning Theory*, pages 4782–4887. PMLR, 2022.

Armen Aghajanyan, Anchit Gupta, Akshat Shrivastava, Xilun Chen, Luke Zettlemoyer, and Sonal Gupta. Muppet: Massive multi-task representations with pre-finetuning. *arXiv preprint arXiv:2101.11038*, 2021.

Shunta Akiyama and Taiji Suzuki. Excess risk of two-layer relu neural networks in teacher-student settings and its superiority to kernel methods. *arXiv preprint arXiv:2205.14818*, 2022.

Zeyuan Allen-Zhu, Yuanzhi Li, and Yingyu Liang. Learning and generalization in overparameterized neural networks, going beyond two layers. *Advances in neural information processing systems*, 32, 2019a.

Zeyuan Allen-Zhu, Yuanzhi Li, and Zhao Song. A convergence theory for deep learning via over-parameterization. In *International Conference on Machine Learning*, pages 242–252. PMLR, 2019b.

Sanjeev Arora, Simon S Du, Wei Hu, Zhiyuan Li, Russ R Salakhutdinov, and Ruosong Wang. On exact computation with an infinitely wide neural net. *Advances in neural information processing systems*, 32, 2019.

Jimmy Ba, Murat A Erdogdu, Taiji Suzuki, Zhichao Wang, Denny Wu, and Greg Yang. High-dimensional asymptotics of feature learning: How one gradient step improves the representation. *arXiv preprint arXiv:2205.01445*, 2022.

Yu Bai and Jason D Lee. Beyond linearization: On quadratic and higher-order approximation of wide neural networks. *arXiv preprint arXiv:1910.01619*, 2019.

Boaz Barak, Benjamin Edelman, Surbhi Goel, Sham Kakade, Eran Malach, and Cyril Zhang. Hidden progress in deep learning: Sgd learns parities near the computational limit. *Advances in Neural Information Processing Systems*, 35:21750–21764, 2022.

Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, et al. Sparks of artificial general intelligence: Early experiments with gpt-4. *arXiv preprint arXiv:2303.12712*, 2023.

Brian Bullins, Elad Hazan, Adam Kalai, and Roi Livni. Generalize across tasks: Efficient algorithms for linear representation learning. In *Algorithmic Learning Theory*, pages 235–246. PMLR, 2019.

Yifang Chen, Kevin Jamieson, and Simon Du. Active multi-task representation learning. In *International Conference on Machine Learning*, pages 3271–3298. PMLR, 2022.

Lenaic Chizat and Francis Bach. On the global convergence of gradient descent for over-parameterized models using optimal transport. *Advances in neural information processing systems*, 31, 2018.

Kurtland Chua, Qi Lei, and Jason D Lee. How fine-tuning allows for effective meta-learning. *Advances in Neural Information Processing Systems*, 34, 2021.

Liam Collins, Hamed Hassani, Aryan Mokhtari, and Sanjay Shakkottai. Exploiting shared representations for personalized federated learning. In *International Conference on Machine Learning*, pages 2089–2099. PMLR, 2021.

Liam Collins, Aryan Mokhtari, Sewoong Oh, and Sanjay Shakkottai. Maml and anil provably learn representations. *arXiv preprint arXiv:2202.03483*, 2022.

Michael Crawshaw. Multi-task learning with deep neural networks: A survey. *arXiv preprint arXiv:2009.09796*, 2020.

Alexandru Damian, Jason Lee, and Mahdi Soltanolkotabi. Neural networks can learn representations with gradient descent. In *Conference on Learning Theory*, pages 5413–5452. PMLR, 2022.

Amit Daniely and Eran Malach. Learning parities with neural networks. *Advances in Neural Information Processing Systems*, 33:20356–20365, 2020.

Giannis Daras, Negin Raoof, Zoi Gkalitsiou, and Alex Dimakis. Multitasking models are robust to structural failure: A neural model for bilingual cognitive reserve. *Advances in Neural Information Processing Systems*, 35:35130–35142, 2022.

Yuyang Deng, Mohammad Mahdi Kamani, and Mehrdad Mahdavi. Local sgd optimizes overparameterized neural networks in polynomial time. In *International Conference on Artificial Intelligence and Statistics*, pages 6840–6861. PMLR, 2022.

Sjoerd Dirksen, Martin Genzel, Laurent Jacques, and Alexander Stollenwerk. The separation capacity of random neural networks. *The Journal of Machine Learning Research*, 23(1):13924–13970, 2022.

Simon Du, Jason Lee, Haochuan Li, Liwei Wang, and Xiyu Zhai. Gradient descent finds global minima of deep neural networks. In *International conference on machine learning*, pages 1675–1685. PMLR, 2019.

Simon Shaolei Du, Wei Hu, Sham M Kakade, Jason D Lee, and Qi Lei. Few-shot learning via learning the representation, provably. In *International Conference on Learning Representations*, 2020.

Spencer Frei, Niladri S Chatterji, and Peter L Bartlett. Random feature amplification: Feature learning and generalization in neural networks. *arXiv preprint arXiv:2202.07626*, 2022.

Surbhi Goel, Sushrut Karmalkar, and Adam Klivans. Time/accuracy tradeoffs for learning a relu with respect to gaussian marginals. *Advances in neural information processing systems*, 32, 2019.

Baihe Huang, Xiaoxiao Li, Zhao Song, and Xin Yang. Fl-ntk: A neural tangent kernel-based framework for federated learning convergence analysis. *arXiv preprint arXiv:2105.05001*, 2021.

Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: Convergence and generalization in neural networks. *Advances in neural information processing systems*, 31, 2018.

Ziwei Ji and Matus Telgarsky. Polylogarithmic width suffices for gradient descent to achieve arbitrarily small test error with shallow relu networks. *arXiv preprint arXiv:1909.12292*, 2019.

Pritish Kamath, Omar Montasser, and Nathan Srebro. Approximate is good enough: Probabilistic variants of dimensional and margin complexity. In *Conference on Learning Theory*, pages 2236–2262. PMLR, 2020.

Chia-Hsiang Kao, Wei-Chen Chiu, and Pin-Yu Chen. Maml is a noisy contrastive learner in classification. *arXiv preprint arXiv:2106.15367*, 2021.

Michael Kearns. Efficient noise-tolerant learning from statistical queries. *Journal of the ACM (JACM)*, 45(6):983–1006, 1998.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. *Advances in neural information processing systems*, 33:18661–18673, 2020.

Jaehoon Lee, Lechao Xiao, Samuel Schoenholz, Yasaman Bahri, Roman Novak, Jascha Sohl-Dickstein, and Jeffrey Pennington. Wide neural networks of any depth evolve as linear models under gradient descent. *Advances in neural information processing systems*, 32, 2019.

Aitor Lewkowycz, Yasaman Bahri, Ethan Dyer, Jascha Sohl-Dickstein, and Guy Gur-Ari. The large learning rate phase of deep learning: the catapult mechanism. *arXiv preprint arXiv:2003.02218*, 2020.

Yuanzhi Li and Yingyu Liang. Learning overparameterized neural networks via stochastic gradient descent on structured data. *Advances in neural information processing systems*, 31, 2018.

Yuanzhi Li, Colin Wei, and Tengyu Ma. Towards explaining the regularization effect of initial large learning rate in training neural networks. *Advances in Neural Information Processing Systems*, 32, 2019.

Yuanzhi Li, Tengyu Ma, and Hongyang R Zhang. Learning over-parametrized two-layer neural networks beyond ntk. In *Conference on learning theory*, pages 2613–2682. PMLR, 2020.

Roi Livni, 2017. URL `https://www.cs.princeton.edu/~rlivni/cos511/lectures/lect13.pdf`.

Eran Malach and Shai Shalev-Shwartz. When hardness of approximation meets hardness of learning. *Journal of Machine Learning Research*, 23(91):1–24, 2022.

Andreas Maurer, Massimiliano Pontil, and Bernardino Romera-Paredes. The benefit of multitask representation learning. *Journal of Machine Learning Research*, 17(81):1–32, 2016.

Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial Intelligence and Statistics*, pages 1273–1282. PMLR, 2017.

Song Mei, Andrea Montanari, and Phan-Minh Nguyen. A mean field view of the landscape of two-layer neural networks. *Proceedings of the National Academy of Sciences*, 115(33):E7665–E7671, 2018.

Alireza Mousavi-Hosseini, Sejun Park, Manuela Girotti, Ioannis Mitliagkas, and Murat A Erdogdu. Neural networks efficiently learn low-dimensional representations with sgd. *arXiv preprint arXiv:2209.14863*, 2022.

Phan-Minh Nguyen. Mean field limit of the learning dynamics of multilayer neural networks. *arXiv preprint arXiv:1902.02880*, 2019.

Samet Oymak and Mahdi Soltanolkotabi. Toward moderate overparameterization: Global convergence guarantees for training shallow neural networks. *IEEE Journal on Selected Areas in Information Theory*, 1(1):84–105, 2020.

Sebastian Ruder. An overview of multi-task learning in deep neural networks. *arXiv preprint arXiv:1706.05098*, 2017.

Victor Sanh, Albert Webson, Colin Raffel, Stephen H Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, et al. Multitask prompted training enables zero-shot task generalization. *arXiv preprint arXiv:2110.08207*, 2021.

Nikunj Saunshi, Arushi Gupta, and Wei Hu. A representation learning perspective on the importance of train-validation splitting in meta-learning. In *International Conference on Machine Learning*, pages 9333–9343. PMLR, 2021.

Zhenmei Shi, Junyi Wei, and Yingyu Liang. A theoretical analysis on feature learning in neural networks: Emergence from inputs and advantage over fixed features. *arXiv preprint arXiv:2206.01717*, 2022.

Justin Sirignano and Konstantinos Spiliopoulos. Mean field analysis of neural networks: A law of large numbers. *SIAM Journal on Applied Mathematics*, 80(2):725–752, 2020.

Yue Sun, Adhyyan Narang, Ibrahim Gulluk, Samet Oymak, and Maryam Fazel. Towards sample-efficient overparameterized meta-learning. *Advances in Neural Information Processing Systems*, 34:28156–28168, 2021.

Matus Telgarsky. Feature selection with gradient descent on two-layer networks in low-rotation regimes. *arXiv preprint arXiv:2208.02789*, 2022.

Kiran Koshy Thekumparampil, Prateek Jain, Praneeth Netrapalli, and Sewoong Oh. Sample efficient linear meta-learning by alternating minimization. *arXiv preprint arXiv:2105.08306*, 2021.

Nilesh Tripuraneni, Michael Jordan, and Chi Jin. On the theory of transfer learning: The importance of task diversity. *Advances in Neural Information Processing Systems*, 33:7852–7862, 2020.

Nilesh Tripuraneni, Chi Jin, and Michael Jordan. Provable meta-learning of linear representations. In *International Conference on Machine Learning*, pages 10434–10443. PMLR, 2021.

Roman Vershynin. *High-dimensional probability: An introduction with applications in data science*, volume 47. Cambridge university press, 2018.

Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Anjana Arunkumar, Arjun Ashok, Arut Selvan Dhanasekaran, Atharva Naik, David Stap, et al. Benchmarking generalization via in-context instructions on 1,600+ language tasks. *arXiv preprint arXiv:2204.07705*, 2022.

Zhen Wang, Rameswar Panda, Leonid Karlinsky, Rogerio Feris, Huan Sun, and Yoon Kim. Multitask prompt tuning enables parameter-efficient transfer learning. *arXiv preprint arXiv:2303.02861*, 2023.

Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*, 2021.

Zixin Wen and Yuanzhi Li. Toward understanding the feature learning process of self-supervised contrastive learning. In *International Conference on Machine Learning*, pages 11112–11122. PMLR, 2021.

Sen Wu, Hongyang R Zhang, and Christopher Ré. Understanding and improving information transfer in multi-task learning. *arXiv preprint arXiv:2005.00944*, 2020.

Ziping Xu and Ambuj Tewari. Representation learning beyond linear prediction functions. *Advances in Neural Information Processing Systems*, 34, 2021.

Greg Yang and Edward J Hu. Tensor programs iv: Feature learning in infinite-width neural networks. In *International Conference on Machine Learning*, pages 11727–11737. PMLR, 2021.

Yu Zhang and Qiang Yang. A survey on multi-task learning. *IEEE Transactions on Knowledge and Data Engineering*, 34(12):5586–5609, 2021.

Ruiqi Zhong, Kristy Lee, Zheng Zhang, and Dan Klein. Adapting language models for zero-shot learning by meta-tuning on dataset and prompt collections. *arXiv preprint arXiv:2104.04670*, 2021.

Mo Zhou, Rong Ge, and Chi Jin. A local convergence theory for mildly over-parameterized two-layer neural network. In *Conference on Learning Theory*, pages 4577–4632. PMLR, 2021.

Libin Zhu, Chaoyue Liu, Adityanarayanan Radhakrishnan, and Mikhail Belkin. Quadratic models for understanding neural network dynamics. *arXiv preprint arXiv:2205.11787*, 2022.

Difan Zou, Yuan Cao, Dongruo Zhou, and Quanquan Gu. Stochastic gradient descent optimizes over-parameterized deep relu networks. *arXiv preprint arXiv:1811.08888*, 2018.

A Proofs of Proposition 4.1 and Theorem 4.2

In this section we prove Proposition 4.1 and Theorem 4.2. Throughout, we will abuse notation by reusing c, c', c'' and C as absolute constants independent of all other parameters. The notations $O(\cdot)$, $\Theta(\cdot)$, and $\Omega(\cdot)$ describe scalings up to absolute constants independent of all other parameters. We start by establishing that each neuron is in general position with high probability at initialization.

Lemma A.1 (Initialization I)**.** *Define the set*

$$\begin{aligned}\mathcal{G}_{\mathbf{w}} \coloneqq \big\{\mathbf{w} \in \mathbb{R}^d : \|\mathbf{w}_{:r}\|_2 &\le c(\sqrt{r} + \sqrt{\log(m)}),\\ c(\sqrt{d-r} - \sqrt{\log(m)}) &\le \|\mathbf{w}_{r:}\|_2 \le c(\sqrt{d-r} + \sqrt{\log(m)}),\\ c(\sqrt{d} - \sqrt{\log(m)}) &\le \|\mathbf{w}\|_2 \le c(\sqrt{d} + \sqrt{\log(m)})\big\}\end{aligned} \tag{20}$$

for an absolute constant c with probability at least 0.999, $\mathbf{w}_j \in \mathcal{G}_{\mathbf{w}}$ for all $j \in [m]$.

Proof. Since each $\mathbf{w}_j \sim \mathcal{N}(\mathbf{0}_d, \nu_{\mathbf{w}}^2 \mathbf{I}_d)$, each $\|\mathbf{w}_{:r}\|_2$, $\|\mathbf{w}_{r:}\|_2$, and $\|\mathbf{w}\|_2$ are sub-gaussian with parameter $\nu_{\mathbf{w}}$, i.e.

$$\mathbb{P}_{\mathbf{w}_{j,:r}}[|\|\mathbf{w}_{j,:r}\|_2 - \sqrt{r}| \le t] \le e^{-c't^2/\nu_{\mathbf{w}}^2} \tag{21}$$

Choosing $t = c''\nu_{\mathbf{w}}\sqrt{\log(m)}$ and union bounding over all $j \in [m]$ completes the proof. □

Lemma A.2 (Initialization II)**.** *Suppose $m > r$. For an absolute constant c,*

$$\sigma_{\min}(\mathbf{W}^0) \ge \nu_{\mathbf{w}}\sqrt{m}\left(1 - c\frac{\sqrt{r}}{\sqrt{m}}\right) \tag{22}$$

With probability at least 0.999.

Proof. The result follows by the fact that each row of $\mathbf{W}^0$ is drawn independently from $\mathcal{N}(\mathbf{0}_d, \nu_{\mathbf{w}}^2\mathbf{I}_d)$ and applying a standard sub-Gaussian matrix concentration inequality (e.g. Equation 4.22 in Vershynin [2018]). □

Lemma A.3. *Let $\lambda_{\mathbf{a}} = \frac{1}{\eta}$ and $\mathbf{x} \sim \mathcal{N}(\mathbf{0}_d, \mathbf{I}_d)$ On the first iteration of the multi-task learning algorithm described in Section 3, the locally updated head for task i is:*

$$\mathbf{a}_i^+ = \eta\mathbb{E}_{\mathbf{x}}[f_i^*(\mathbf{x})\sigma(\mathbf{W}^0\mathbf{x})] \tag{23}$$

for all tasks $i \in [T]$.

Proof. Due to the symmetric initialization, $\mathbf{a}_0^\top\sigma(\mathbf{W}^0\mathbf{x} + \mathbf{b}^0) = 0$ for all $\mathbf{x}$. Using also that $\mathbf{b}^0 = \mathbf{0}_m$, we have that $\max(1 - f_i^*(\mathbf{x})\mathbf{a}_0^\top\sigma(\mathbf{W}^0\mathbf{x} + \mathbf{b}^0), 0) = 1 - f_i^*(\mathbf{x})\mathbf{a}_0^\top\sigma(\mathbf{W}^0\mathbf{x})$ for all $\mathbf{x}$. Therefore

$$\begin{aligned}\mathbf{a}_i^+ &= \mathbf{a}^0 - \eta\nabla\mathcal{L}_i(\mathbf{W}^0, \mathbf{b}^0, \mathbf{a}^0)\\ &= (1 - \eta\lambda_{\mathbf{a}})\mathbf{a}^0 - \nabla_{\mathbf{a}}\mathbb{E}_{\mathbf{x}}[\max(1 - f_i^*(\mathbf{M}\mathbf{x})\mathbf{a}_0^\top\sigma(\mathbf{W}^0\mathbf{x} + \mathbf{b}^0), 0)]\\ &= (1 - \eta\lambda_{\mathbf{a}})\mathbf{a}_0 + \eta\mathbb{E}_{\mathbf{x}}[f_i^*(\mathbf{x})\sigma(\mathbf{W}^0\mathbf{x})] \qquad (24)\\ &= \eta\mathbb{E}_{\mathbf{x}}[\bar{f}_i^*(\mathbf{x})\sigma(\mathbf{W}^0\mathbf{x} + \mathbf{b}^0)] \qquad (25)\end{aligned}$$

where (24) follows by the symmetric initialization and (25) follows by choice of $\lambda_{\mathbf{a}}$. □

Lemma A.4. *The loss after updating the heads one the first iteration is given by*

$$\begin{aligned}
&\mathcal{L}(\mathbf{W}^0, \mathbf{b}^0, \{\mathbf{a}_i^+\}_{i=1}^T) \\
&= 1 - \mathbb{E}_{\mathbf{x},\mathbf{x}'}\left[\chi\{\text{sign}(\mathbf{x}_{:r}) = \text{sign}(\mathbf{x}'_{:r})\}\sigma(\mathbf{W}^0\mathbf{x}')^\top\sigma(\mathbf{W}^0\mathbf{x})\right] \\
&\quad - \mathbb{E}_{\mathbf{x},\mathbf{x}'}\left[\chi\{\text{sign}(\mathbf{x}_{:r}) \neq \text{sign}(\mathbf{x}'_{:r})\}\left(\frac{1}{T}\sum_{i=1}^T f_i^*(\mathbf{x})f_i^*(\mathbf{x}')\right)\sigma(\mathbf{W}^0\mathbf{x}')^\top\sigma(\mathbf{W}^0\mathbf{x})\right] \\
&\quad + \frac{1}{T}\sum_{i=1}^T \mathbb{E}_{\mathbf{x},\mathbf{x}'}\Big[\chi\left\{\eta f_i^*(\mathbf{x})\mathbb{E}_{\mathbf{x}''}[f_i^*(\mathbf{x}'')\sigma(\mathbf{W}^0\mathbf{x}'')]^\top\sigma(\mathbf{W}^0\mathbf{x}) > 1\right\} \\
&\qquad\qquad \times\left(1 - f_i^*(\mathbf{x})f_i^*(\mathbf{x}')\sigma(\mathbf{W}^0\mathbf{x}')^\top\sigma(\mathbf{W}^0\mathbf{x})\right)\Big]
\end{aligned}$$

Proof. Using the computation of each $\mathbf{a}_i^+$ from Lemma A.4, and the fact that $\mathbf{b}^0 = \mathbf{0}$ by choice of initialization,

$$\begin{aligned}
&\mathcal{L}(\mathbf{W}^0, \mathbf{b}^0, \{\mathbf{a}_i^+\}_{i=1}^T) \\
&= \frac{1}{T}\sum_{i=1}^T \mathcal{L}_i(\mathbf{W}^0, \mathbf{b}^0, \{\mathbf{a}_i^+\}_{i=1}^T) \\
&= \frac{1}{T}\sum_{i=1}^T \mathbb{E}_{\mathbf{x}}\left[\max\left(1 - \eta f_i^*(\mathbf{x})\mathbb{E}_{\mathbf{x}'}[f_i^*(\mathbf{x}')\sigma(\mathbf{W}^0\mathbf{x}')]^\top\sigma(\mathbf{W}^0\mathbf{x}), 0\right)\right] \\
&= \frac{1}{T}\sum_{i=1}^T \mathbb{E}_{\mathbf{x},\mathbf{x}'}\Big[\chi\left\{\eta f_i^*(\mathbf{x})\mathbb{E}_{\mathbf{x}''}[f_i^*(\mathbf{x}'')\sigma(\mathbf{W}^0\mathbf{x}'')]^\top\sigma(\mathbf{W}^0\mathbf{x}) < 1\right\} \\
&\qquad\qquad \times\left(1 - \eta f_i^*(\mathbf{x})f_i^*(\mathbf{x}')\sigma(\mathbf{W}^0\mathbf{x}')^\top\sigma(\mathbf{W}^0\mathbf{x})\right)\Big] \\
&= 1 - \eta\,\mathbb{E}_{\mathbf{x},\mathbf{x}'}\left[\left(\frac{1}{T}\sum_{i=1}^T f_i^*(\mathbf{x})f_i^*(\mathbf{x}')\right)\sigma(\mathbf{W}^0\mathbf{x}')^\top\sigma(\mathbf{W}^0\mathbf{x})\right] \\
&\quad + \frac{1}{T}\sum_{i=1}^T \mathbb{E}_{\mathbf{x},\mathbf{x}'}\Big[\chi\left\{\eta f_i^*(\mathbf{x})\mathbb{E}_{\mathbf{x}''}[f_i^*(\mathbf{x}'')\sigma(\mathbf{W}^0\mathbf{x}'')]^\top\sigma(\mathbf{W}^0\mathbf{x}) > 1\right\} \\
&\qquad\qquad \times\left(1 - \eta f_i^*(\mathbf{x})f_i^*(\mathbf{x}')\sigma(\mathbf{W}^0\mathbf{x}')^\top\sigma(\mathbf{W}^0\mathbf{x})\right)\Big]
\end{aligned}$$

Next, using the fact that $\text{sign}(\mathbf{x}_{:r}) = \text{sign}(\mathbf{x}'_{:r})$ implies $f_i^*(\mathbf{x}) = f_i^*(\mathbf{x}')$ for all $i \in [n]$, we have

$$
\begin{aligned}
&= 1 - \eta\, \mathbb{E}_{\mathbf{x},\mathbf{x}'}\left[\chi\{\text{sign}(\mathbf{x}_{:r}) = \text{sign}(\mathbf{x}'_{:r})\}\left(\frac{1}{T}\sum_{i=1}^{T} f_i^*(\mathbf{x}) f_i^*(\mathbf{x}')\right)\sigma(\mathbf{W}^0\mathbf{x}')^\top\sigma(\mathbf{W}^0\mathbf{x})\right] \\
&\quad - \eta\, \mathbb{E}_{\mathbf{x},\mathbf{x}'}\left[\chi\{\text{sign}(\mathbf{x}_{:r}) \neq \text{sign}(\mathbf{x}'_{:r})\}\left(\frac{1}{T}\sum_{i=1}^{T} f_i^*(\mathbf{x}) f_i^*(\mathbf{x}')\right)\sigma(\mathbf{W}^0\mathbf{x}')^\top\sigma(\mathbf{W}^0\mathbf{x})\right] \\
&\quad + \frac{1}{T}\sum_{i=1}^{T}\mathbb{E}_{\mathbf{x},\mathbf{x}'}\Big[\chi\left\{\eta f_i^*(\mathbf{x})\mathbb{E}_{\mathbf{x}''}[f_i^*(\mathbf{x}'')\sigma(\mathbf{W}^0\mathbf{x}'')]^\top\sigma(\mathbf{W}^0\mathbf{x}) > 1\right\} \\
&\qquad\qquad \times\left(1 - \eta f_i^*(\mathbf{x}) f_i^*(\mathbf{x}')\sigma(\mathbf{W}^0\mathbf{x}')^\top\sigma(\mathbf{W}^0\mathbf{x})\right)\Big] \\
&= 1 - \eta\, \mathbb{E}_{\mathbf{x},\mathbf{x}'}\Big[\chi\{\text{sign}(\mathbf{x}_{:r}) = \text{sign}(\mathbf{x}'_{:r})\}\sigma(\mathbf{W}^0\mathbf{x}')^\top\sigma(\mathbf{W}^0\mathbf{x})\Big] \\
&\quad - \eta\, \mathbb{E}_{\mathbf{x},\mathbf{x}'}\left[\chi\{\text{sign}(\mathbf{x}_{:r}) \neq \text{sign}(\mathbf{x}'_{:r})\}\left(\frac{1}{T}\sum_{i=1}^{T} f_i^*(\mathbf{x}) f_i^*(\mathbf{x}')\right)\sigma(\mathbf{W}^0\mathbf{x}')^\top\sigma(\mathbf{W}^0\mathbf{x})\right] \\
&\quad + \frac{1}{T}\sum_{i=1}^{T}\mathbb{E}_{\mathbf{x},\mathbf{x}'}\Big[\chi\left\{\eta f_i^*(\mathbf{x})\mathbb{E}_{\mathbf{x}''}[f_i^*(\mathbf{x}'')\sigma(\mathbf{W}^0\mathbf{x}'')]^\top\sigma(\mathbf{W}^0\mathbf{x}) > 1\right\} \\
&\qquad\qquad \times\left(1 - \eta f_i^*(\mathbf{x}) f_i^*(\mathbf{x}')\sigma(\mathbf{W}^0\mathbf{x}')^\top\sigma(\mathbf{W}^0\mathbf{x})\right)\Big]
\end{aligned}
$$

□

Lemma A.5. *Under Assumption 2.1, for all* $(\mathbf{v}, \mathbf{v}') \in \{-1,1\}^r \times \{-1,1\}^r$ *such that* $\mathbf{v} \neq \pm\mathbf{v}'$,

$$
\left|\frac{1}{T}\sum_{i=1}^{T}\chi\{\bar{f}_i^*(\mathbf{v}) \neq \bar{f}_i^*(\mathbf{v}')\} - \frac{1}{2}\right| = c\frac{\sqrt{r}}{\sqrt{T}} \tag{26}
$$

with probability at least 0.999 *over the random draw of* T *tasks from* $\mathcal{T}$*, for an absolute constant* c*.*

Proof. Under Assumption 2.1, each $\chi\{\bar{f}_i^*(\mathbf{v}) \neq \bar{f}_i^*(\mathbf{v}')\}$ is an i.i.d. Bernoulli random variable with parameter $\frac{1}{2}$. Thus, for any particular $(\mathbf{v}, \mathbf{v}') \in \{-1,1\}^r \times \{-1,1\}^r$ such that $\mathbf{v} \neq \pm\mathbf{v}'$, we have

$$
\mathbb{P}\left[\left|\frac{1}{T}\sum_{i=1}^{T}\chi\{\bar{f}_i^*(\mathbf{v}) \neq \bar{f}_i^*(\mathbf{v}')\} - \frac{1}{2}\right| \geq t\right] \leq \exp\left(-\frac{Tt^2}{2}\right) \tag{27}
$$

for any $t \geq 0$. Union bounding over all $O\left(2^{2r}\right)$ pairs $(\mathbf{v}, \mathbf{v}') \in \{-1,1\}^r \times \{-1,1\}^r$ and choosing $t = \frac{\sqrt{c' \log(2^{2r})}}{\sqrt{n}}$ completes the proof. □

Lemma A.6. *For any* $j \in [m]$ *such that* $\mathbf{w}_j^0 \in \mathcal{G}_{\mathbf{w}}$*, then*

$$
\nabla_{\mathbf{w}_j}\mathcal{L}(\mathbf{W}^0, \mathbf{b}^0, \{\mathbf{a}_i^0\}_{i\in[T]}) = -\eta^2 2^{-r}\mathbf{A}(\mathbf{w}_j^0)\mathbf{w}_j^0 + \eta\mathbf{e} \tag{28}
$$

where $\|\mathbf{e}\|_2 \leq O(\nu_{\mathbf{w}} d e^{-cd} + \nu_{\mathbf{w}} d\frac{\sqrt{r}}{\sqrt{T}})$ *and* $\mathbf{A}(\mathbf{w}_j^0) \coloneqq \begin{bmatrix}\mathbf{A}_{||,||}(\mathbf{w}_j^0) & \mathbf{A}_{||,\perp}(\mathbf{w}_j^0)\\ \mathbf{A}_{\perp,||}(\mathbf{w}_j^0) & \mathbf{A}_{\perp,\perp}(\mathbf{w}_j^0)\end{bmatrix}$ *for matrices* $\mathbf{A}_{\parallel,\parallel}(\mathbf{w}_j^0) \in \mathbb{R}^{r\times r}$*,* $\mathbf{A}_{\parallel,\perp}(\mathbf{w}_j^0) \in \mathbb{R}^{r\times(d-r)}$*,* $\mathbf{A}_{\perp,\parallel}(\mathbf{w}_j^0) \in \mathbb{R}^{r\times(d-r)}$*,* $\mathbf{A}_{\perp,\perp}(\mathbf{w}_j^0) \in \mathbb{R}^{(d-r)\times(d-r)}$*. Let* $\mathbf{u} \sim$ *Unif*$(\{-1,1\}^r)$ *be a random variable drawn uniformly from the Rademacher hypercube in* r *dimensions, and* $\mathbf{x}$ *and*

$\mathbf{x}'$ *be drawn independently from* $\mathcal{N}(\mathbf{0}_d, \mathbf{I}_d)$*, then for any* $\mathbf{w} \in \mathbb{R}^d$ *the matrices* $\mathbf{A}_{||,||}(\mathbf{w})$*,* $\mathbf{A}_{||,\perp}(\mathbf{w})$*,* $\mathbf{A}_{\perp,||}(\mathbf{w})$*, and* $\mathbf{A}_{\perp,\perp}(\mathbf{w})$ *are defined as*

$$\mathbf{A}_{||,||}(\mathbf{w}) = \mathbb{E}_{\mathbf{u}}\left[\mathbb{E}_{\mathbf{x},\mathbf{x}'}\left[\sigma'(\mathbf{w}^\top\mathbf{x})\sigma'(\mathbf{w}^\top\mathbf{x}')\mathbf{x}_{:r}(\mathbf{x}'_{:r})^\top | \operatorname{sign}(\mathbf{x}_{:r}) = \operatorname{sign}(\mathbf{x}'_{:r}) = \mathbf{u}\right]\right] \tag{29}$$

$$\mathbf{A}_{||,\perp}(\mathbf{w}) = \mathbb{E}_{\mathbf{u}}\left[\mathbb{E}_{\mathbf{x},\mathbf{x}'}\left[\sigma'(\mathbf{w}^\top\mathbf{x})\sigma'(\mathbf{w}^\top\mathbf{x})\mathbf{x}_{:r}(\mathbf{x}'_{r:})^\top | \operatorname{sign}(\mathbf{x}_{:r}) = \operatorname{sign}(\mathbf{x}'_{:r}) = \mathbf{u}\right]\right] \tag{30}$$

$$\mathbf{A}_{\perp,||}(\mathbf{w}) = \mathbb{E}_{\mathbf{u}}\left[\mathbb{E}_{\mathbf{x},\mathbf{x}'}\left[\sigma'(\mathbf{w}^\top\mathbf{x})\sigma'(\mathbf{w}^\top\mathbf{x}')\mathbf{x}_{r:}(\mathbf{x}'_{:r})^\top | \operatorname{sign}(\mathbf{x}_{:r}) = \operatorname{sign}(\mathbf{x}'_{:r}) = \mathbf{u}\right]\right] \tag{31}$$

$$\mathbf{A}_{\perp,\perp}(\mathbf{w}) = \mathbb{E}_{\mathbf{u}}\left[\mathbb{E}_{,\mathbf{x},\mathbf{x}'}\left[\sigma'(\mathbf{w}^\top\mathbf{x})\sigma'(\mathbf{w}^\top\mathbf{x}')\mathbf{x}_{r:}(\mathbf{x}'_{r:})^\top | \operatorname{sign}(\mathbf{x}_{:r}) = \operatorname{sign}(\mathbf{x}'_{:r}) = \mathbf{u}\right]\right] \tag{32}$$

Proof. Using Lemma A.4, we have

$$\begin{aligned}
&\mathcal{L}(\mathbf{W}^0, \mathbf{b}^0, \{\mathbf{a}_i^+\}_{i\in[T]}) \\
&= 1 - \eta\,\mathbb{E}_{\mathbf{x},\mathbf{x}'}\left[\chi\{\operatorname{sign}(\mathbf{x}_{:r}) = \operatorname{sign}(\mathbf{x}'_{:r})\}\sigma(\mathbf{W}^0\mathbf{x}')^\top\sigma(\mathbf{W}^0\mathbf{x})\right] \\
&\quad - \eta\,\mathbb{E}_{\mathbf{x},\mathbf{x}'}\left[\chi\{\operatorname{sign}(\mathbf{x}_{:r}) \neq \operatorname{sign}(\mathbf{x}'_{:r})\}\left(\frac{1}{T}\sum_{i=1}^T f_i^*(\mathbf{x})f_i^*(\mathbf{x}')\right)\sigma(\mathbf{W}^0\mathbf{x}')^\top\sigma(\mathbf{W}^0\mathbf{x})\right] \\
&\quad + \frac{1}{T}\sum_{i=1}^T \mathbb{E}_{\mathbf{x},\mathbf{x}'}\Big[\chi\left\{\eta f_i^*(\mathbf{x})\mathbb{E}_{\mathbf{x}''}[f_i^*(\mathbf{x}'')\sigma(\mathbf{W}^0\mathbf{x}'')]^\top\sigma(\mathbf{W}^0\mathbf{x}) > 1\right\} \\
&\qquad\qquad \times\left(1 - \eta f_i^*(\mathbf{x})f_i^*(\mathbf{x}')\sigma(\mathbf{W}^0\mathbf{x}')^\top\sigma(\mathbf{W}^0\mathbf{x})\right)\Big]
\end{aligned}$$

Let $\mathbf{u} \sim \operatorname{Unif}(\{-1,1\}^r)$ be uniformly drawn from the r-dimensional hypercube. Taking the gradient

of both terms with respect to $\mathbf{w}_j$ yields

$$
\begin{aligned}
&\nabla_{\mathbf{w}_j}\mathcal{L}(\mathbf{W}^0, \mathbf{b}^0, \{\mathbf{a}_i^+\}_{i\in[T]}) \\
&= -\eta\mathbb{E}_{\mathbf{x},\mathbf{x}'}\Big[\chi\{\mathrm{sign}(\mathbf{x}_{:r}) = \mathrm{sign}(\mathbf{x}'_{:r})\}\sigma((\mathbf{w}_j^0)^\top\mathbf{x}')\sigma'((\mathbf{w}_j^0)^\top\mathbf{x})\mathbf{x}\Big] \\
&\quad - \eta\,\mathbb{E}_{\mathbf{x},\mathbf{x}'}\left[\chi\{\mathrm{sign}(\mathbf{x}_{:r}) \neq \mathrm{sign}(\mathbf{x}'_{:r})\}\left(\frac{1}{T}\sum_{i=1}^T f_i^*(\mathbf{x})f_i^*(\mathbf{x}')\right)\sigma(\mathbf{W}^0\mathbf{x}')^\top\sigma'((\mathbf{w}_j^0)^\top\mathbf{x})\mathbf{x}\right] \\
&\quad - \eta\mathbb{E}_{\mathbf{x},\mathbf{x}'}\Big[\chi\{\eta f_i^*(\mathbf{x})\mathbb{E}_{\mathbf{x}''}[f_i^*(\mathbf{x}'')\sigma((\mathbf{w}_j^0)^\top\mathbf{x}'')]^\top\sigma((\mathbf{w}_j^0)^\top\mathbf{x}) > 1\}\sigma((\mathbf{w}_j^0)^\top\mathbf{x}')\sigma'((\mathbf{w}_j^0)^\top\mathbf{x})\mathbf{x}\Big] \\
&= \eta 2^{-r}\mathbb{E}_{\mathbf{u}}\Big[\mathbb{E}_{\mathbf{x},\mathbf{x}'}\Big[\sigma'((\mathbf{w}_j^0)^\top\mathbf{x}')\sigma'((\mathbf{w}_j^0)^\top\mathbf{x})\mathbf{x}(\mathbf{x}')^\top\,|\,\mathrm{sign}(\mathbf{x}_{:r}) = \mathrm{sign}(\mathbf{x}'_{:r}) = \mathbf{u}\Big]\Big]\mathbf{w}_j^0 \\
&\quad - \eta\,\mathbb{E}_{\mathbf{x},\mathbf{x}'}\left[\chi\{\mathrm{sign}(\mathbf{x}_{:r}) \neq \mathrm{sign}(\mathbf{x}'_{:r})\}\left(\frac{1}{T}\sum_{i=1}^T f_i^*(\mathbf{x})f_i^*(\mathbf{x}')\right)\right. \\
&\qquad\qquad \left.\times\,\sigma'((\mathbf{w}_j^0)^\top\mathbf{x}')^\top\sigma'((\mathbf{w}_j^0)^\top\mathbf{x})\mathbf{x}(\mathbf{x}')^\top\right]\mathbf{w}_j^0 \\
&\quad - \eta\mathbb{E}_{\mathbf{x},\mathbf{x}'}\Big[\chi\{\eta f_i^*(\mathbf{x})\mathbb{E}_{\mathbf{x}''}[f_i^*(\mathbf{x}'')\sigma(\mathbf{W}^0\mathbf{x}'')]^\top\sigma(\mathbf{W}^0\mathbf{x}) > 1\} \\
&\qquad\qquad \times\,\chi\{\mathrm{sign}(\mathbf{x}_{:r}) = \mathrm{sign}(\mathbf{x}'_{:r})\}\sigma'((\mathbf{w}_j^0)^\top\mathbf{x}')\sigma'((\mathbf{w}_j^0)^\top\mathbf{x})\mathbf{x}(\mathbf{x}')^\top\Big]\mathbf{w}_j^0 \qquad (33) \\
&= \eta 2^{-r}\mathbf{A}(\mathbf{w}_j^0)\mathbf{w}_j^0 \\
&\quad - \eta\,\mathbb{E}_{\mathbf{x},\mathbf{x}'}\left[\chi\{\mathrm{sign}(\mathbf{x}_{:r}) \neq \mathrm{sign}(\mathbf{x}'_{:r})\}\left(\frac{1}{T}\sum_{i=1}^T f_i^*(\mathbf{x})f_i^*(\mathbf{x}')\right)\right. \\
&\qquad\qquad \left.\times\,\sigma'((\mathbf{w}_j^0)^\top\mathbf{x}')^\top\sigma'((\mathbf{w}_j^0)^\top\mathbf{x})\mathbf{x}(\mathbf{x}')^\top\right]\mathbf{w}_j^0 \\
&\quad - \eta\mathbb{E}_{\mathbf{x},\mathbf{x}'}\Big[\chi\{\eta f_i^*(\mathbf{x})\mathbb{E}_{\mathbf{x}''}[f_i^*(\mathbf{x}'')\sigma(\mathbf{W}^0\mathbf{x}'')]^\top\sigma(\mathbf{W}^0\mathbf{x}) > 1\} \\
&\qquad\qquad \times\,\chi\{\mathrm{sign}(\mathbf{x}_{:r}) = \mathrm{sign}(\mathbf{x}'_{:r})\}\sigma'((\mathbf{w}_j^0)^\top\mathbf{x}')\sigma'((\mathbf{w}_j^0)^\top\mathbf{x})\mathbf{x}(\mathbf{x}')^\top\Big]\mathbf{w}_j^0 \qquad (34)
\end{aligned}
$$

where we have used $\sigma(x) = \sigma'(x)x$ in (33). It is easy to check that the first term in (33) is equal to $\eta 2^{-r}\mathbf{A}(\mathbf{w}_j^0)\mathbf{w}_j^0$ by definition of $\mathbf{A}(\mathbf{w}_j^0)$. Now we show that the remaining terms are small. The norm of the second term in (34) can be bounded using Jensen's Inequality (which implies

$\|\mathbb{E}_{\mathbf{x}}[\mathbf{x}]\|_2 \le \mathbb{E}_{\mathbf{x}}[\|\mathbf{x}\|_2]$ for any random vector $\mathbf{x}$) and the Cauchy-Schwarz Inequality as:

$$
\begin{aligned}
&\left\| \eta\, \mathbb{E}_{\mathbf{x},\mathbf{x}'}\left[\chi\{\mathrm{sign}(\mathbf{x}_{:r}) \neq \mathrm{sign}(\mathbf{x}'_{:r})\}\left(\tfrac{1}{T}\sum_{i=1}^{T} f_i^*(\mathbf{x}) f_i^*(\mathbf{x}')\right)\sigma'((\mathbf{w}_j^0)^\top \mathbf{x}')\sigma'((\mathbf{w}_j^0)^\top \mathbf{x})\mathbf{x}(\mathbf{x}')^\top\right]\mathbf{w}_j^0\right\|_2 \\
&\le \eta\Bigg(\mathbb{E}_{\mathbf{x},\mathbf{x}'}\left[\left\|\chi\{\mathrm{sign}(\mathbf{x}_{:r}) \neq \mathrm{sign}(\mathbf{x}'_{:r})\}\left(\frac{1}{T}\sum_{i=1}^{T} f_i^*(\mathbf{x}) f_i^*(\mathbf{x}')\right)\sigma'((\mathbf{w}_j^0)^\top \mathbf{x}')\sigma'((\mathbf{w}_j^0)^\top \mathbf{x})\mathbf{x}\right\|_2^2\right] \\
&\qquad \mathbb{E}_{\mathbf{x},\mathbf{x}'}\left[((\mathbf{x}')^\top \mathbf{w}_j^0)^2\right]\Bigg)^{1/2} \\
&= \eta\|\mathbf{w}_j^0\|_2 \\
&\quad \times \mathbb{E}_{\mathbf{x},\mathbf{x}'}\left[\left\|\chi\{\mathrm{sign}(\mathbf{x}_{:r}) \neq \mathrm{sign}(\mathbf{x}'_{:r})\}\left(\frac{1}{T}\sum_{i=1}^{T} f_i^*(\mathbf{x}) f_i^*(\mathbf{x}')\right)\sigma'((\mathbf{w}_j^0)^\top \mathbf{x}')\sigma'((\mathbf{w}_j^0)^\top \mathbf{x})\mathbf{x}\right\|_2^2\right]^{1/2} \\
&\le \eta\|\mathbf{w}_j^0\|_2\, \mathbb{E}_{\mathbf{x},\mathbf{x}'}\left[\left\|\chi\{\mathrm{sign}(\mathbf{x}_{:r}) \neq \mathrm{sign}(\mathbf{x}'_{:r})\}\sigma'((\mathbf{w}_j^0)^\top \mathbf{x}')\sigma'((\mathbf{w}_j^0)^\top \mathbf{x})\mathbf{x}\right\|_2^4\right]^{1/4} \\
&\qquad \times \mathbb{E}_{\mathbf{x},\mathbf{x}'}\left[\left(\frac{1}{T}\sum_{i=1}^{T} f_i^*(\mathbf{x}) f_i^*(\mathbf{x}')\right)^4\right]^{1/4} \\
&\le \eta\|\mathbf{w}_j^0\|_2\, \mathbb{E}_{\mathbf{x},\mathbf{x}'}\left[\left\|\chi\{\mathrm{sign}(\mathbf{x}_{:r}) \neq \mathrm{sign}(\mathbf{x}'_{:r})\}\sigma'((\mathbf{w}_j^0)^\top \mathbf{x}')\sigma'((\mathbf{w}_j^0)^\top \mathbf{x})\mathbf{x}\right\|_2^4\right]^{1/4} c\sqrt{\frac{r}{T}} \\
&\le \eta\|\mathbf{w}_j^0\|_2\, \mathbb{E}_{\mathbf{x},\mathbf{x}'}\left[\chi\{\mathrm{sign}(\mathbf{x}_{:r}) \neq \mathrm{sign}(\mathbf{x}'_{:r})\}\sigma'((\mathbf{w}_j^0)^\top \mathbf{x}')\sigma'((\mathbf{w}_j^0)^\top \mathbf{x})\right]^{1/8} \mathbb{E}_{\mathbf{x}}\left[\|\mathbf{x}\|_2^8\right]^{1/8} c\sqrt{\frac{r}{T}} \qquad (35) \\
&\le \eta\|\mathbf{w}_j^0\|_2\, \mathbb{E}_{\mathbf{x}}\left[\|\mathbf{x}\|_2^8\right]^{1/8} c\sqrt{\frac{r}{T}} \\
&= \eta\|\mathbf{w}_j^0\|_2 (d(d+2)(d+4)(d+6))^{1/8} c\sqrt{\frac{r}{T}} \\
&\le c'\eta\sqrt{\frac{r(d^2 + d\log(m))}{T}} \\
&\le c''\nu_{\mathbf{w}} d\sqrt{\frac{r}{T}} \qquad (36)
\end{aligned}
$$

where (35) follows with probability at least 0.999 by Lemma A.5 and (36) follows deterministically since $\mathbf{w}_j^0 \in \mathcal{G}_{\mathbf{w}}$ and $\eta = \Theta(1)$.

The norm of the third term in (33) can similarly be bounded as:

$$
\begin{aligned}
&\Big\|\eta\mathbb{E}_{\mathbf{x},\mathbf{x}'}\Big[\chi\{\eta f_i^*(\mathbf{x})\mathbb{E}_{\mathbf{x}''}[f_i^*(\mathbf{x}'')\sigma(\mathbf{W}^0\mathbf{x}''+\mathbf{b}^0)]^\top\sigma(\mathbf{W}^0\mathbf{x}+\mathbf{b}^0)>1\}\\
&\qquad\qquad\times\chi\{\mathrm{sign}(\mathbf{x}_{:r})=\mathrm{sign}(\mathbf{x}'_{:r})\}\sigma'(\mathbf{w}_j^0\mathbf{x}')\sigma'((\mathbf{w}_j^0)^\top\mathbf{x})\mathbf{x}(\mathbf{x}')^\top\Big]\mathbf{w}_j^0\Big\|_2\\
&\le\eta\mathbb{E}_{\mathbf{x}'}[\chi\{\eta f_i^*(\mathbf{x})\mathbb{E}_{\mathbf{x}''}[f_i^*(\mathbf{x}'')\sigma(\mathbf{W}^0\mathbf{x}'')]^\top\sigma(\mathbf{W}^0\mathbf{x})>1\}]^{1/2}\\
&\quad\times\mathbb{E}_{\mathbf{x},\mathbf{x}'}\Big[\chi\{\mathrm{sign}(\mathbf{x}_{:r})=\mathrm{sign}(\mathbf{x}'_{:r})\}\sigma'(\mathbf{w}_j^0\mathbf{x}')\sigma'((\mathbf{w}_j^0)^\top\mathbf{x})\|\mathbf{x}\|_2^2((\mathbf{x}')^\top\mathbf{w}_j^0)^2\Big]^{1/2}\\
&\le\eta\mathbb{P}_{\mathbf{x}}\big[\eta f_i^*(\mathbf{x})\mathbb{E}_{\mathbf{x}''}[f_i^*(\mathbf{x}'')\sigma(\mathbf{W}^0\mathbf{x}''+\mathbf{b}^0)]^\top\sigma(\mathbf{W}^0\mathbf{x})>1\big]^{1/2}\\
&\qquad\times\mathbb{E}_{\mathbf{x},\mathbf{x}'}\Big[\chi\{\mathrm{sign}(\mathbf{x}_{:r})=\mathrm{sign}(\mathbf{x}'_{:r})\}\sigma'(\mathbf{w}_j^0\mathbf{x}')\sigma'((\mathbf{w}_j^0)^\top\mathbf{x})\|\mathbf{x}\|_2^4\Big]^{1/4}\mathbb{E}_{\mathbf{x},\mathbf{x}'}\Big[((\mathbf{x}')^\top\mathbf{w}_j^0)^4\Big]^{1/4}\\
&\le\eta\mathbb{P}_{\mathbf{x}}\big[\eta f_i^*(\mathbf{x})\mathbb{E}_{\mathbf{x}''}[f_i^*(\mathbf{x}'')\sigma(\mathbf{W}^0\mathbf{x}'')]^\top\sigma(\mathbf{W}^0\mathbf{x})>1\big]^{1/2}\\
&\qquad\times\mathbb{E}_{\mathbf{x},\mathbf{x}'}\Big[\chi\{\mathrm{sign}(\mathbf{x}_{:r})=\mathrm{sign}(\mathbf{x}'_{:r})\}\sigma'(\mathbf{w}_j^0\mathbf{x}')\sigma'((\mathbf{w}_j^0)^\top\mathbf{x})\|\mathbf{x}\|_2^4\Big]^{1/4}\mathbb{E}_{\mathbf{x},\mathbf{x}'}\Big[((\mathbf{x}')^\top\mathbf{w}_j^0)^4\Big]^{1/4}\\
&=c\,\nu_{\mathbf{w}}\,d\,\mathbb{P}_{\mathbf{x}}\big[\eta f_i^*(\mathbf{x})\mathbb{E}_{\mathbf{x}''}[f_i^*(\mathbf{x}'')\sigma(\mathbf{W}^0\mathbf{x}'')]^\top\sigma(\mathbf{W}^0\mathbf{x})>1\big] \qquad (37)
\end{aligned}
$$

where the last line follows since $\eta=\Theta(1)$. Conditioned on $\mathbf{w}_j^0\in\mathcal{G}_{\mathbf{w}}$ for all $j\in[m]$, we have $\|\mathbf{W}^0\|_2\le c\nu_{\mathbf{w}}(\sqrt{d}+\sqrt{m})$. Thus, we have

$$
\begin{aligned}
&\mathbb{P}_{\mathbf{x}}\big[\eta f_i^*(\mathbf{x})\mathbb{E}_{\mathbf{x}''}[f_i^*(\mathbf{x}'')\sigma(\mathbf{W}^0\mathbf{x}'')]^\top\sigma(\mathbf{W}^0\mathbf{x})>1\big]\\
&\le\mathbb{P}_{\mathbf{x}}\left[\|\mathbb{E}_{\mathbf{x}''}[\sigma(\mathbf{W}^0\mathbf{x}'')]\|_2\|\sigma(\mathbf{W}^0\mathbf{x})\|_2>\frac{1}{\eta}\right]\\
&\le\mathbb{P}_{\mathbf{x}}\left[\mathbb{E}_{\mathbf{x}''}[\|\sigma(\mathbf{W}^0\mathbf{x}'')\|_2]\|\sigma(\mathbf{W}^0\mathbf{x})\|_2>\frac{1}{\eta}\right]\\
&\le\mathbb{P}_{\mathbf{x}}\left[c\|\mathbf{W}^0\|_2^2\mathbb{E}_{\mathbf{x}''}[\|\mathbf{x}''\|_2]\|\mathbf{x}\|_2>\frac{1}{\eta}\right]\\
&\le\mathbb{P}_{\mathbf{x}}\left[\|\mathbf{x}\|_2>\frac{1}{c\eta\|\mathbf{W}^0\|_2^2\sqrt{d}}\right]\\
&\le\mathbb{P}_{\mathbf{x}}\left[\|\mathbf{x}\|_2>\frac{1}{c'\eta\nu_{\mathbf{w}}^2(d+m)\sqrt{d}}\right]\\
&\le\mathbb{P}_{\mathbf{x}}\left[\|\mathbf{x}\|_2>\frac{d}{c''\eta}\right]\\
&\le\exp\left(-c'''d^2\right) \qquad (38)
\end{aligned}
$$

using $\nu_{\mathbf{w}}=O(d^{-1.25})$, $m=O(d)$ and $\eta=\Theta(1)$. Combining (38) with (37) and (33) completes the proof. □

Next we control the matrices $\mathbf{A}_{||,||}(\mathbf{w})$, $\mathbf{A}_{||,\perp}(\mathbf{w})$, $\mathbf{A}_{\perp,||}(\mathbf{w})$, and $\mathbf{A}_{\perp,\perp}(\mathbf{w})$.

Lemma A.7. *For any $\mathbf{w}\in\mathcal{G}_{\mathbf{w}}$, we have*

$$
\big\|\mathbf{A}_{\|,\|}(\mathbf{w})-\tfrac{1}{4}\mathbf{I}_r\big\|_2\le c\frac{r^3+\log^3(m)}{d} \qquad (39)
$$

for an absolute constant c.

Proof. Recall that $\mathbf{u}$ is drawn uniformly from $\{-1,1\}^r$. From Lemma A.6, we have

$$
\begin{aligned}
&\mathbf{A}_{\parallel,\parallel}(\mathbf{w}) \\
&= \mathbb{E}_{\mathbf{u}}\left[\mathbb{E}_{\mathbf{x},\mathbf{x}'}\left[\sigma'(\mathbf{w}^\top\mathbf{x}')\sigma'(\mathbf{w}^\top\mathbf{x})\mathbf{x}_{:r}(\mathbf{x}'_{:r})^\top \,|\, \operatorname{sign}(\mathbf{x}'_{:r}) = \operatorname{sign}(\mathbf{x}_{:r}) = \mathbf{u}\right]\right] \\
&= \mathbb{E}_{\mathbf{u}}\left[\mathbb{E}_{\mathbf{x}_{:r},\mathbf{x}'_{:r}}\left[\mathbb{E}_{\mathbf{x}_{r:},\mathbf{x}'_{r:}}\left[\sigma'(\mathbf{w}^\top\mathbf{x}')\sigma'(\mathbf{w}^\top\mathbf{x})\right]\mathbf{x}_{:r}(\mathbf{x}'_{:r})^\top \,|\, \operatorname{sign}(\mathbf{x}'_{:r}) = \operatorname{sign}(\mathbf{x}_{:r}) = \mathbf{u}\right]\right] \\
&= \mathbb{E}_{\mathbf{u}}\Big[\mathbb{E}_{\mathbf{x}_{:r},\mathbf{x}'_{:r}}\Big[\mathbf{x}_{:r}(\mathbf{x}'_{:r})^\top \mathbb{P}_{\mathbf{x}_{r:}}\left[\mathbf{w}_{r:}^\top\mathbf{x}_{r:} > -\mathbf{w}_{:r}^\top\mathbf{x}_{:r}\right] \\
&\qquad\qquad \times \mathbb{P}_{\mathbf{x}_{r:}'}\left[\mathbf{w}_{r:}^\top\mathbf{x}'_{r:} > -\mathbf{w}_{:r}^\top\mathbf{x}'_{:r}\right] \,|\, \operatorname{sign}(\mathbf{x}'_{:r}) = \operatorname{sign}(\mathbf{x}_{:r}) = \mathbf{u}\Big]\Big] \\
&= \mathbb{E}_{\mathbf{u},\mathbf{x}_{:r},\mathbf{x}'_{:r}}\Bigg[\mathbf{x}_{:r}(\mathbf{x}'_{:r})^\top\left(\frac{1}{2} + \frac{1}{2}\operatorname{erf}\left(\frac{\mathbf{w}_{:r}^\top\mathbf{x}_{:r}}{\sqrt{2}\|\mathbf{w}_{r:}\|_2}\right)\right) \\
&\qquad\qquad \left(\frac{1}{2} + \frac{1}{2}\operatorname{erf}\left(\frac{\mathbf{w}^\top\mathbf{x}'_{:r}}{\sqrt{2}\|\mathbf{w}_{r:}\|_2}\right)\right) \,|\, \operatorname{sign}(\mathbf{x}'_{:r}) = \operatorname{sign}(\mathbf{x}_{:r}) = \mathbf{u}\Bigg] \qquad (40) \\
&= \frac{1}{4}\mathbb{E}_{\mathbf{u},\mathbf{x}_{:r},\mathbf{x}'_{:r}}\left[\mathbf{x}_{:r}(\mathbf{x}'_{:r})^\top \,|\, \operatorname{sign}(\mathbf{x}'_{:r}) = \operatorname{sign}(\mathbf{x}_{:r}) = \mathbf{u}\right] \\
&\quad + \frac{1}{2}\mathbb{E}_{\mathbf{u},\mathbf{x}_{:r},\mathbf{x}'_{:r}}\left[\mathbf{x}_{:r}(\mathbf{x}'_{:r})^\top \operatorname{erf}\left(\frac{\mathbf{w}_{:r}^\top\mathbf{x}_{:r}}{\sqrt{2}\|\mathbf{w}_{r:}\|_2}\right) \,|\, \operatorname{sign}(\mathbf{x}'_{:r}) = \operatorname{sign}(\mathbf{x}_{:r}) = \mathbf{u}\right] \\
&\quad + \frac{1}{4}\mathbb{E}_{\mathbf{u},\mathbf{x}_{:r},\mathbf{x}'_{:r}}\left[\mathbf{x}_{:r}(\mathbf{x}'_{:r})^\top \operatorname{erf}\left(\frac{\mathbf{w}_{:r}^\top\mathbf{x}_{:r}}{\sqrt{2}\|\mathbf{w}_{r:}\|_2}\right)\operatorname{erf}\left(\frac{\mathbf{w}_{:r}^\top\mathbf{x}'_{:r}}{\sqrt{2}\|\mathbf{w}_{r:}\|_2}\right) \,|\, \operatorname{sign}(\mathbf{x}'_{:r}) = \operatorname{sign}(\mathbf{x}_{:r}) = \mathbf{u}\right] \qquad (41)
\end{aligned}
$$

where (40) follows using the Gaussian CDF. For the first term in (41), we can re-write $\mathbf{x}_{:r}$ conditioned on $\operatorname{sign}(\mathbf{x}_{:r}) = \mathbf{u}$ as $\operatorname{diag}(\mathbf{u})|\mathbf{x}_{:r}|$, where $|\cdot|$ denotes element-wise absolute value, to obtain

$$
\begin{aligned}
&\frac{1}{4}\mathbb{E}_{\mathbf{u},\mathbf{x}_{:r},\mathbf{x}'_{:r}}\left[\mathbf{x}_{:r}(\mathbf{x}'_{:r})^\top \,|\, \operatorname{sign}(\mathbf{x}'_{:r}) = \operatorname{sign}(\mathbf{x}_{:r}) = \mathbf{u}\right] \\
&= \frac{1}{4}\mathbb{E}_{\mathbf{u},\mathbf{x}_{:r},\mathbf{x}'_{:r}}\left[\operatorname{diag}(\mathbf{u})|\mathbf{x}_{:r}||\mathbf{x}'_{:r}|^\top \operatorname{diag}(\mathbf{u})\right] \\
&= \frac{1}{4}\mathbb{E}_{\mathbf{u}}\left[\operatorname{diag}(\mathbf{u})\mathbb{E}_{\mathbf{x}_{:r},\mathbf{x}'_{:r}}\left[|\mathbf{x}_{:r}||\mathbf{x}'_{:r}|^\top\right]\operatorname{diag}(\mathbf{u})\right] \\
&= \frac{1}{4}\mathbb{E}_{\mathbf{u}}\left[\operatorname{diag}(\mathbf{u})\left(\frac{2}{\pi}\mathbf{1}_r\mathbf{1}_r^\top + \left(1 - \frac{2}{\pi}\right)\mathbf{I}_r\right)\operatorname{diag}(\mathbf{u})\right] \\
&= \frac{1}{4}\left(\frac{2}{\pi}\mathbb{E}_{\mathbf{u}}\left[\mathbf{u}\mathbf{u}^\top\right] + \left(1 - \frac{2}{\pi}\right)\mathbf{I}_r\right) \\
&= \frac{1}{4}\mathbf{I}_r \qquad (42)
\end{aligned}
$$

For the second term in (41), we have

$$\begin{aligned}
&\frac{1}{2}\mathbb{E}_{\mathbf{u},\mathbf{x}_{:r},\mathbf{x}'_{:r}}\left[\mathbf{x}_{:r}(\mathbf{x}'_{:r})^\top \operatorname{erf}\left(\frac{\mathbf{w}_{:r}^\top \mathbf{x}_{:r}}{\sqrt{2}\|\mathbf{w}_{r:}\|_2}\right) | \operatorname{sign}(\mathbf{x}'_{:r}) = \operatorname{sign}(\mathbf{x}_{:r}) = \mathbf{u}\right] \\
&= \frac{1}{2}\mathbb{E}_{\mathbf{u},\mathbf{x}_{:r},\mathbf{x}'_{:r}}\left[\operatorname{diag}(\mathbf{u})|\mathbf{x}_{:r}||\mathbf{x}'_{:r}|^\top \operatorname{diag}(\mathbf{u}) \operatorname{erf}\left(\frac{\mathbf{w}_{:r}^\top \operatorname{diag}(\mathbf{u})|\mathbf{x}_{:r}|}{\sqrt{2}\|\mathbf{w}_{r:}\|_2}\right)\right] &(43)\\
&= \frac{1}{2}\mathbb{E}_{\mathbf{u},\mathbf{x}_{:r},\mathbf{x}'_{:r}}\left[\operatorname{diag}(-\mathbf{u})|\mathbf{x}_{:r}||\mathbf{x}'_{:r}|^\top \operatorname{diag}(-\mathbf{u}) \operatorname{erf}\left(\frac{\mathbf{w}_{:r}^\top \operatorname{diag}(-\mathbf{u})|\mathbf{x}_{:r}|}{\sqrt{2}\|\mathbf{w}_{r:}\|_2}\right)\right] &(44)\\
&= -\frac{1}{2}\mathbb{E}_{\mathbf{u},\mathbf{x}_{:r},\mathbf{x}'_{:r}}\left[\operatorname{diag}(\mathbf{u})|\mathbf{x}_{:r}||\mathbf{x}'_{:r}|^\top \operatorname{diag}(\mathbf{u}) \operatorname{erf}\left(\frac{\mathbf{w}_{:r}^\top \operatorname{diag}(\mathbf{u})|\mathbf{x}_{:r}|}{\sqrt{2}\|\mathbf{w}_{r:}\|_2}\right)\right] &(45)\\
&= \mathbf{0} &(46)
\end{aligned}$$

where (44) follows from the fact that $\mathbf{u}$ and $-\mathbf{u}$ have the same distribution, (45) follows since erf() is an odd function, and (46) follows since $x = -x \iff x = 0$.

The final term in (41) is

$$\begin{aligned}
&\mathbb{E}_{\mathbf{u},\mathbf{x}_{:r},\mathbf{x}'_{:r}}\left[\mathbf{x}_{:r}(\mathbf{x}'_{:r})^\top \operatorname{erf}\left(\frac{\mathbf{w}_{:r}^\top \mathbf{x}_{:r}}{\sqrt{2}\|\mathbf{w}_{r:}\|_2}\right) \operatorname{erf}\left(\frac{\mathbf{w}_{:r}^\top \mathbf{x}'_{:r}}{\sqrt{2}\|\mathbf{w}_{r:}\|_2}\right) | \operatorname{sign}(\mathbf{x}'_{:r}) = \operatorname{sign}(\mathbf{x}_{:r}) = \mathbf{u}\right] \\
&= \mathbb{E}_{\mathbf{u}}\left[\mathbb{E}_{\mathbf{x}_{:r}}\left[\mathbf{x}_{:r} \operatorname{erf}\left(\frac{\mathbf{w}_{:r}^\top \mathbf{x}_{:r}}{\sqrt{2}\|\mathbf{w}_{r:}\|_2}\right) \middle| \operatorname{sign}(\mathbf{x}_{:r}) = \mathbf{u}\right]\right. \\
&\qquad \left.\times \mathbb{E}_{\mathbf{x}'_{:r}}\left[(\mathbf{x}'_{:r})^\top \operatorname{erf}\left(\frac{\mathbf{w}_{:r}^\top \mathbf{x}'_{:r}}{\sqrt{2}\|\mathbf{w}_{r:}\|_2}\right) \middle| \operatorname{sign}(\mathbf{x}'_{:r}) = \mathbf{u}\right]\right] &(47)
\end{aligned}$$

Again to remove the conditioning, we can equivalently write $\mathbb{E}_{\mathbf{x}_{:r}}\left[\mathbf{x}_{:r} \operatorname{erf}\left(\frac{\mathbf{w}_{:r}^\top \mathbf{x}_{:r}}{\sqrt{2}\|\mathbf{w}_{r:}\|_2}\right) \middle| \operatorname{sign}(\mathbf{x}_{:r}) = \mathbf{u}\right]$ as

$$\mathbb{E}_{\mathbf{x}_{:r}}\left[\mathbf{x}_{:r} \operatorname{erf}\left(\frac{\mathbf{w}_{:r}^\top \mathbf{x}_{:r}}{\sqrt{2}\|\mathbf{w}_{r:}\|_2}\right) \middle| \operatorname{sign}(\mathbf{x}_{:r}) = \mathbf{u}\right] = \mathbb{E}_{\mathbf{x}_{:r}}\left[\operatorname{diag}(\mathbf{u})|\mathbf{x}_{:r}| \operatorname{erf}\left(\frac{\mathbf{w}_{:r}^\top \operatorname{diag}(\mathbf{u})|\mathbf{x}_{:r}|}{\sqrt{2}\|\mathbf{w}_{r:}\|_2}\right)\right] \tag{48}$$

where $|\mathbf{x}_{:r}| \in \mathbb{R}^r$ denotes the element-wise absolute value of $\mathbf{x}_{:r}$. For all $\mathbf{u} \in \{-1, 1\}^r$, we have

$$\begin{aligned}
&\left\|\mathbb{E}_{\mathbf{x}_{:r}}\left[\operatorname{diag}(\mathbf{u})|\mathbf{x}_{:r}| \operatorname{erf}\left(\frac{\mathbf{w}_{:r}^\top \operatorname{diag}(\mathbf{u})|\mathbf{x}_{:r}|}{\sqrt{2}\|\mathbf{w}_{r:}\|_2}\right)\right]\right\|_2 \\
&\le \mathbb{E}_{\mathbf{x}_{:r}}\left[\left\|\operatorname{diag}(\mathbf{u})|\mathbf{x}_{:r}| \operatorname{erf}\left(\frac{\mathbf{w}_{:r}^\top \operatorname{diag}(\mathbf{u})|\mathbf{x}_{:r}|}{\sqrt{2}\|\mathbf{w}_{r:}\|_2}\right)\right\|_2\right] \\
&\le \mathbb{E}_{\mathbf{x}_{:r}}\left[\|\operatorname{diag}(\mathbf{u})|\mathbf{x}_{:r}|\|_2 \left|\operatorname{erf}\left(\frac{\mathbf{w}_{:r}^\top \operatorname{diag}(\mathbf{u})|\mathbf{x}_{:r}|}{\sqrt{2}\|\mathbf{w}_{r:}\|_2}\right)\right|\right] \\
&\le \mathbb{E}_{\mathbf{x}_{:r}}\left[\|\operatorname{diag}(\mathbf{u})|\mathbf{x}_{:r}|\|_2 \left|\frac{\mathbf{w}_{:r}^\top \operatorname{diag}(\mathbf{u})|\mathbf{x}_{:r}|}{\|\mathbf{w}_{r:}\|_2}\right|\right] &(49)\\
&\le \|\operatorname{diag}(\mathbf{u})\|_2 \, \mathbb{E}_{\mathbf{x}_{:r}}\left[\|\mathbf{x}_{:r}\|_2^2\right] \frac{\|\mathbf{w}_{:r}^\top \operatorname{diag}(\mathbf{u})\|_2}{\|\mathbf{w}_{r:}\|_2} \\
&\le \frac{cr(\sqrt{r} + \sqrt{\log(m)})}{\sqrt{d}} &(50)
\end{aligned}$$

for an absolute constant c, where (49) follows since $|\operatorname{erf}(x)| \le \sqrt{2}|x|$, and (50) follows since $\mathbf{w} \in \mathcal{G}_{\mathbf{w}}$ and $m = O(d)$, thus $\|\mathbf{w}_{r:}\|_2 = \Omega(\sqrt{d})$. Therefore, using (47),

$$\left\| \mathbb{E}_{\mathbf{u},\mathbf{x}_{:r},\mathbf{x}'_{:r}} \left[\mathbf{x}_{:r}(\mathbf{x}'_{:r})^\top \operatorname{erf}\left(\frac{\mathbf{w}_{:r}^\top \mathbf{x}_{:r}}{\sqrt{2}\|\mathbf{w}_{r:}\|_2} \right) \operatorname{erf}\left(\frac{\mathbf{w}_{:r}^\top \mathbf{x}'_{:r}}{\sqrt{2}\|\mathbf{w}_{r:}\|_2} \right) | \operatorname{sign}(\mathbf{x}'_{:r}) = \operatorname{sign}(\mathbf{x}_{:r}) = \mathbf{u} \right] \right\|_2 \le \frac{c' r^3 + c' \log^3(m)}{d} \tag{51}$$

completing the proof. □

Lemma A.8. *For any $\mathbf{w} \in \mathcal{G}_{\mathbf{w}}$, we have*

$$\left\| \mathbf{A}_{\perp,\perp}(\mathbf{w}) - \left(1 - \frac{\|\mathbf{w}_{:r}\|_2^2}{\|\mathbf{w}_{r:}\|_2^2}\right) \frac{\mathbf{w}_{r:}\mathbf{w}_{r:}^\top}{2\pi\|\mathbf{w}_{r:}\|_2^2} \right\|_2 \le c \frac{r^4 + \log^4(m)}{d^2} \tag{52}$$

for an absolute constant c.

Proof. We have

$$\begin{aligned}
\mathbf{A}_{\perp,\perp}(\mathbf{w}) &= \mathbb{E}_{\mathbf{u}} \left[\mathbb{E}_{\mathbf{x},\mathbf{x}'} \left[\mathbf{x}_{r:}(\mathbf{x}'_{r:})^\top \sigma'(\mathbf{w}^\top \mathbf{x}') \sigma'(\mathbf{w}^\top \mathbf{x}) | \operatorname{sign}(\mathbf{x}'_{:r}) = \operatorname{sign}(\mathbf{x}_{:r}) = \mathbf{u} \right] \right] \\
&= \mathbb{E}_{\mathbf{u},\mathbf{x}_{:r},\mathbf{x}'_{:r}} \left[\mathbb{E}_{\mathbf{x}_{r:},\mathbf{x}'_{r:}} \left[\mathbf{x}_{r:}(\mathbf{x}'_{r:})^\top \sigma'(\mathbf{w}^\top \mathbf{x}') \sigma'(\mathbf{w}^\top \mathbf{x}) \right] | \operatorname{sign}(\mathbf{x}'_{:r}) = \operatorname{sign}(\mathbf{x}_{:r}) = \mathbf{u} \right] \\
&= \mathbb{E}_{\mathbf{u},\mathbf{x}_{:r},\mathbf{x}'_{:r}} \left[\mathbb{E}_{\mathbf{x}_{r:}} \left[\mathbf{x}_{r:} \sigma'(\mathbf{w}^\top \mathbf{x}) \right] \mathbb{E}_{\mathbf{x}'_{r:}} \left[(\mathbf{x}'_{r:})^\top \sigma'(\mathbf{w}^\top \mathbf{x}') \right] | \operatorname{sign}(\mathbf{x}'_{:r}) = \operatorname{sign}(\mathbf{x}_{:r}) = \mathbf{u} \right] \qquad (53)
\end{aligned}$$

Next, we have

$$\begin{aligned}
\mathbb{E}_{\mathbf{x}_{r:}} \left[\mathbf{x}_{r:} \sigma'(\mathbf{w}^\top \mathbf{x}) \right] &= \mathbb{E}_{\mathbf{x}_{r:}} \left[\mathbf{x}_{r:} \sigma'(\mathbf{w}_{:r}^\top \mathbf{x}_{:r} + \mathbf{w}_{r:}^\top \mathbf{x}_{r:}) \right] \\
&= \mathbb{E}_{\mathbf{x}_{r:}} \left[\mathbf{x}_{r:} | \mathbf{w}_{:r}^\top \mathbf{x}_{:r} + \mathbf{w}_{r:}^\top \mathbf{x}_{r:} > 0 \right] \mathbb{P}_{\mathbf{x}_{r:}} [\mathbf{w}_{:r}^\top \mathbf{x}_{:r} + \mathbf{w}_{r:}^\top \mathbf{x}_{r:} > 0] \\
&= \mathbb{E}_{\mathbf{x}_{r:}} \left[\mathbf{x}_{r:} \middle| \mathbf{w}_{:r}^\top \mathbf{x}_{:r} > |\mathbf{w}_{r:}^\top \mathbf{x}_{r:}| \right] \mathbb{P}_{\mathbf{x}_{r:}} [\mathbf{w}_{:r}^\top \mathbf{x}_{:r} > |\mathbf{w}_{r:}^\top \mathbf{x}_{r:}|] \qquad (54)
\end{aligned}$$

where the last line follows since if $\mathbf{w}_{r:}^\top \mathbf{x}_{r:} < 0$, then $-\mathbf{w}_{r:}^\top \mathbf{x}_{r:} = |\mathbf{w}_{r:}^\top \mathbf{x}_{r:}|$ so $\mathbb{E}_{\mathbf{x}_{r:}} \left[\mathbf{x}_{r:} | \mathbf{w}_{:r}^\top \mathbf{x}_{:r} + \mathbf{w}_{r:}^\top \mathbf{x}_{r:} > 0 \right] = \mathbb{E}_{\mathbf{x}_{r:}} \left[\mathbf{x}_{r:} \middle| \mathbf{w}_{:r}^\top \mathbf{x}_{:r} > |\mathbf{w}_{r:}^\top \mathbf{x}_{r:}| \right]$ and $\mathbb{P}_{\mathbf{x}_{r:}} [\mathbf{w}_{:r}^\top \mathbf{x}_{:r} + \mathbf{w}_{r:}^\top \mathbf{x}_{r:} > 0] = \mathbb{P}_{\mathbf{x}_{r:}} [\mathbf{w}_{:r}^\top \mathbf{x}_{:r} > |\mathbf{w}_{r:}^\top \mathbf{x}_{r:}|]$, and if $\mathbf{w}_{r:}^\top \mathbf{x}_{r:} > 0$, then by the law of total expectation,

$$\begin{aligned}
&\mathbb{E}_{\mathbf{x}_{r:}} \left[\mathbf{x}_{r:} | \mathbf{w}_{:r}^\top \mathbf{x}_{:r} + \mathbf{w}_{r:}^\top \mathbf{x}_{r:} > 0 \right] \\
&= \mathbb{E}_{\mathbf{x}_{r:}} \left[\mathbf{x}_{r:} | \mathbf{w}_{:r}^\top \mathbf{x}_{:r} > \mathbf{w}_{r:}^\top \mathbf{x}_{r:} \right] \mathbb{P}_{\mathbf{x}_{r:}} \left[\mathbf{w}_{:r}^\top \mathbf{x}_{:r} > \mathbf{w}_{r:}^\top \mathbf{x}_{r:} | \mathbf{w}_{:r}^\top \mathbf{x}_{:r} > -\mathbf{w}_{r:}^\top \mathbf{x}_{r:} \right] \\
&\quad + \mathbb{E}_{\mathbf{x}_{r:}} \left[\mathbf{x}_{r:} | \mathbf{w}_{r:}^\top \mathbf{x}_{r:} > \mathbf{w}_{:r}^\top \mathbf{x}_{:r} > -\mathbf{w}_{r:}^\top \mathbf{x}_{r:} \right] \\
&\qquad \times \mathbb{P}_{\mathbf{x}_{r:}} \left[\mathbf{w}_{r:}^\top \mathbf{x}_{r:} > \mathbf{w}_{:r}^\top \mathbf{x}_{:r} > -\mathbf{w}_{r:}^\top \mathbf{x}_{r:} | \mathbf{w}_{:r}^\top \mathbf{x}_{:r} > -\mathbf{w}_{r:}^\top \mathbf{x}_{r:} \right] \\
&= \mathbb{E}_{\mathbf{x}_{r:}} \left[\mathbf{x}_{r:} | \mathbf{w}_{:r}^\top \mathbf{x}_{:r} > \mathbf{w}_{r:}^\top \mathbf{x}_{r:} \right] \mathbb{P}_{\mathbf{x}_{r:}} \left[\mathbf{w}_{:r}^\top \mathbf{x}_{:r} > \mathbf{w}_{r:}^\top \mathbf{x}_{r:} | \mathbf{w}_{:r}^\top \mathbf{x}_{:r} > -\mathbf{w}_{r:}^\top \mathbf{x}_{r:} \right] \\
&= \mathbb{E}_{\mathbf{x}_{r:}} \left[\mathbf{x}_{r:} | \mathbf{w}_{:r}^\top \mathbf{x}_{:r} > |\mathbf{w}_{r:}^\top \mathbf{x}_{r:}| \right] \mathbb{P}_{\mathbf{x}_{r:}} \left[\mathbf{w}_{:r}^\top \mathbf{x}_{:r} > \mathbf{w}_{r:}^\top \mathbf{x}_{r:} | \mathbf{w}_{:r}^\top \mathbf{x}_{:r} > -\mathbf{w}_{r:}^\top \mathbf{x}_{r:} \right] \qquad (55) \\
&= \mathbb{E}_{\mathbf{x}_{r:}} \left[\mathbf{x}_{r:} | \mathbf{w}_{:r}^\top \mathbf{x}_{:r} > |\mathbf{w}_{r:}^\top \mathbf{x}_{r:}| \right] \frac{\mathbb{P}_{\mathbf{x}_{r:}} \left[\mathbf{w}_{:r}^\top \mathbf{x}_{:r} > |\mathbf{w}_{r:}^\top \mathbf{x}_{r:}| \right]}{\mathbb{P}_{\mathbf{x}_{r:}} [\mathbf{w}_{:r}^\top \mathbf{x}_{:r} > -\mathbf{w}_{r:}^\top \mathbf{x}_{r:}]}
\end{aligned}$$

where (55) follows since $\mathbf{w}_{r:}^\top \mathbf{x}_{r:} = |\mathbf{w}_{r:}^\top \mathbf{x}_{r:}|$. Now we return to (54).

Note that

$$\mathbb{E}_{\mathbf{x}_{r:}}\left[\mathbf{x}_{r:}\middle|\mathbf{w}_{:r}^\top \mathbf{x}_{:r} > |\mathbf{w}_{r:}^\top \mathbf{x}_{r:}|\right] = \mathbb{E}_{\mathbf{x}_{r:}}\left[\mathbf{x}_{r:} - 2\sigma\left(-\frac{\mathbf{w}_{r:}^\top}{\|\mathbf{w}_{r:}\|_2}\mathbf{x}_{r:}\right)\frac{\mathbf{w}_{r:}}{\|\mathbf{w}_{r:}\|_2}\middle||\mathbf{w}_{:r}^\top \mathbf{x}_{:r}| > |\mathbf{w}_{r:}^\top \mathbf{x}_{r:}|\right]$$

by the symmetry of the Gaussian distribution (and the fact that

$$\mathbf{x}_{r:} - 2\sigma\left(-\frac{\mathbf{w}_{r:}^\top}{\|\mathbf{w}_{r:}\|_2}\mathbf{x}_{r:}\right)\frac{\mathbf{w}_{r:}}{\|\mathbf{w}_{r:}\|_2}$$

is the flip of $\mathbf{x}_{r:}$ across the hyperplane with normal vector $\frac{\mathbf{w}_{r:}^\top}{\|\mathbf{w}_{r:}\|_2}$ when $\mathbf{w}_{r:}^\top \mathbf{x}_{r:} < -|\mathbf{w}_{r:}^\top \mathbf{x}_{r:}|$). Using this, we obtain

$$\begin{aligned}
&\mathbb{E}_{\mathbf{x}_{r:}}\left[\mathbf{x}_{r:}\sigma'(\mathbf{w}^\top \mathbf{x})\right] \\
&= \mathbb{E}_{\mathbf{x}_{r:}}\left[\mathbf{x}_{r:} - 2\sigma\left(-\frac{\mathbf{w}_{r:}^\top}{\|\mathbf{w}_{r:}\|_2}\mathbf{x}_{r:}\right)\frac{\mathbf{w}_{r:}}{\|\mathbf{w}_{r:}\|_2}\middle||\mathbf{w}_{r:}^\top \mathbf{x}_{r:}| > |\mathbf{w}_{:r}^\top \mathbf{x}_{:r}|\right]\mathbb{P}_{\mathbf{x}_{r:}}\left[\mathbf{w}_{r:}^\top \mathbf{x}_{r:} > |\mathbf{w}_{:r}^\top \mathbf{x}_{:r}|\right] \\
&= 2\mathbb{E}_{\mathbf{x}_{r:}}\left[\sigma\left(-\frac{\mathbf{w}_{r:}^\top \mathbf{x}_{r:}}{\|\mathbf{w}_{r:}\|_2}\right)\middle||\mathbf{w}_{r:}^\top \mathbf{x}_{r:}| > |\mathbf{w}_{:r}^\top \mathbf{x}_{:r}|\right]\mathbb{P}_{\mathbf{x}_{r:}}\left[\mathbf{w}_{r:}^\top \mathbf{x}_{r:} > |\mathbf{w}_{:r}^\top \mathbf{x}_{:r}|\right]\frac{\mathbf{w}_{r:}}{\|\mathbf{w}_{r:}\|_2} \\
&= \mathbb{E}_{\mathbf{x}_{r:}}\left[\sigma\left(-\frac{\mathbf{w}_{r:}^\top \mathbf{x}_{r:}}{\|\mathbf{w}_{r:}\|_2}\right)\middle|\mathbf{w}_{r:}^\top \mathbf{x}_{r:} < -|\mathbf{w}_{:r}^\top \mathbf{x}_{:r}|\right]\mathbb{P}_{\mathbf{x}_{r:}}\left[\mathbf{w}_{r:}^\top \mathbf{x}_{r:} > |\mathbf{w}_{:r}^\top \mathbf{x}_{:r}|\right]\frac{\mathbf{w}_{r:}}{\|\mathbf{w}_{r:}\|_2} \\
&= \frac{\frac{1}{\sqrt{2\pi}}\exp(-(\mathbf{w}_{:r}^\top \mathbf{x}_{:r})^2/2)}{\mathbb{P}_{\mathbf{x}_{r:}}[\mathbf{w}_{r:}^\top \mathbf{x}_{r:} < -|\mathbf{w}_{:r}^\top \mathbf{x}_{:r}|]}\mathbb{P}_{\mathbf{x}_{r:}}\left[\mathbf{w}_{r:}^\top \mathbf{x}_{r:} > |\mathbf{w}_{:r}^\top \mathbf{x}_{:r}|\right]\frac{\mathbf{w}_{r:}}{\|\mathbf{w}_{r:}\|_2} && (56) \\
&= \frac{1}{\sqrt{2\pi}}\exp(-(\mathbf{w}_{:r}^\top \mathbf{x}_{:r})^2/2)\frac{\mathbf{w}_{r:}}{\|\mathbf{w}_{r:}\|_2} && (57)
\end{aligned}$$

where (56) follows by the definition of the inverse Mills ratio. Repeating the same argument for $\mathbf{x}'_{r:}$ and using (53) yields

$$\begin{aligned}
\mathbf{A}_{\perp,\perp}(\mathbf{w}) = \mathbb{E}_{\mathbf{u},\mathbf{x}_{:r},\mathbf{x}'_{:r}}&\left[\exp\left(-\frac{(\mathbf{x}_{:r}^\top \mathbf{w}_{:r})^2 + ((\mathbf{x}'_{:r})^\top \mathbf{w}_{:r})^2}{2\|\mathbf{w}_{r:}\|_2^2}\right)\middle|\operatorname{sign}(\mathbf{x}'_{:r}) = \operatorname{sign}(\mathbf{x}_{:r}) = \mathbf{u}\right] \\
&\times \frac{1}{2\pi\|\mathbf{w}_{r:}\|_2^2}\mathbf{w}_{r:}\mathbf{w}_{r:}^\top && (58)
\end{aligned}$$

We analyze the scalar term in the top line. We have

$$\begin{aligned}
&\mathbb{E}_{\mathbf{u}}\left[\mathbb{E}_{\mathbf{x}_{:r},\mathbf{x}'_{:r}}\left[\exp\left(-\frac{(\mathbf{x}_{:r}^\top \mathbf{w}_{:r})^2 + ((\mathbf{x}'_{:r})^\top \mathbf{w}_{:r})^2}{2\|\mathbf{w}_{r:}\|_2^2}\right)\middle|\operatorname{sign}(\mathbf{x}'_{:r}) = \operatorname{sign}(\mathbf{x}_{:r}) = \mathbf{u}\right]\right] \\
&= \mathbb{E}_{\mathbf{u}}\left[\mathbb{E}_{\mathbf{x}_{:r}}\left[\exp\left(-\frac{(\mathbf{x}_{:r}^\top \mathbf{w}_{:r})^2}{2\|\mathbf{w}_{r:}\|_2^2}\right)\middle|\operatorname{sign}(\mathbf{x}_{:r}) = \mathbf{u}\right]\right. \\
&\qquad \left.\times\, \mathbb{E}_{\mathbf{x}'_{:r}}\left[\exp\left(-\frac{((\mathbf{x}'_{:r})^\top \mathbf{w}_{:r})^2}{2\|\mathbf{w}_{r:}\|_2^2}\right)\middle|\operatorname{sign}(\mathbf{x}'_{:r}) = \mathbf{u}\right]\right] \\
&= \mathbb{E}_{\mathbf{u}}\left[\mathbb{E}_{\mathbf{x}_{:r}}\left[\exp\left(-\frac{(\mathbf{x}_{:r}^\top \mathbf{w}_{:r})^2}{2\|\mathbf{w}_{r:}\|_2^2}\right)\middle|\operatorname{sign}(\mathbf{x}_{:r}) = \mathbf{u}\right]^2\right] && (59)
\end{aligned}$$

where, using the Taylor expansion of $\exp(-x^2)$ and re-writing $\mathbf{x}_{:r}$ conditioned on $\text{sign}(\mathbf{x}_{:r}) = \mathbf{u}$ as $\text{diag}(\mathbf{u})|\mathbf{x}_{:r}|$,

$$\begin{aligned}
&\mathbb{E}_{\mathbf{x}_{:r}}\left[\exp\left(-\frac{(\mathbf{x}_{:r}^\top \mathbf{w}_{:r})^2}{2\|\mathbf{w}_{r:}\|_2^2}\right)\middle|\,\text{sign}(\mathbf{x}_{:r}) = \mathbf{u}\right] \\
&= \mathbb{E}_{\mathbf{x}_{:r}}\left[\exp\left(-\frac{(|\mathbf{x}_{:r}|^\top \text{diag}(\mathbf{u})\mathbf{w}_{:r})^2}{2\|\mathbf{w}_{r:}\|_2^2}\right)\right] \\
&= 1 - \frac{1}{2\|\mathbf{w}_{r:}\|_2^2}\mathbb{E}_{\mathbf{x}_{:r}}\left[(|\mathbf{x}_{:r}|^\top \text{diag}(\mathbf{u})\mathbf{w}_{:r})^2\right] + O\left(\frac{r^4 + \log^4(m)}{d^2}\right) \qquad (60)
\end{aligned}$$

where (60) follows since $\|\mathbf{w}_{r:}\|_2^2 = \Omega((\sqrt{d} - \sqrt{\log(m)})^4) = \Omega(d^2)$ and $\mathbb{E}_{\mathbf{x}_{:r}}[(|\mathbf{x}_{:r}|^\top \text{diag}(\mathbf{u})\mathbf{w}_{:r})^4] \leq \mathbb{E}_{\mathbf{x}_{:r}}[\|\mathbf{x}_{:r}\|_2^4]\|\mathbf{w}_{:r}\|_2^4 = O(r^4 + \log^4(m))$ since $\mathbf{w} \in \mathcal{G}_{\mathbf{w}}$. To compute the second term in (60), note that

$$\begin{aligned}
\mathbb{E}_{\mathbf{x}_{:r}}\left[(|\mathbf{x}_{:r}|^\top \text{diag}(\mathbf{u})\mathbf{w}_{:r})^2\right] &= \mathbf{w}_{:r}^\top \text{diag}(\mathbf{u})\mathbb{E}_{\mathbf{x}_{:r}}\left[|\mathbf{x}_{:r}||\mathbf{x}_{:r}|^\top\right]\text{diag}(\mathbf{u})\mathbf{w}_{:r} \\
&= \mathbf{w}_{:r}^\top \text{diag}(\mathbf{u})\left(\frac{2}{\pi}\mathbf{1}_{d-r}\mathbf{1}_{d-r}^\top + \left(1 - \frac{2}{\pi}\right)\mathbf{I}_{d-r}\right)\text{diag}(\mathbf{u})\mathbf{w}_{:r} \\
&= \mathbf{w}_{:r}^\top\left(\frac{2}{\pi}\mathbf{u}\mathbf{u}^\top + \left(1 - \frac{2}{\pi}\right)\mathbf{I}_{d-r}\right)\mathbf{w}_{:r}
\end{aligned}$$

therefore

$$
\begin{aligned}
&\mathbb{E}_{\mathbf{u}}\left[\mathbb{E}_{\mathbf{x}_{:r}}\left[\exp\left(-\frac{(\mathbf{x}_{:r}^\top \mathbf{w}_{:r})^2}{2\|\mathbf{w}_{r:}\|_2^2}\right)\,\middle|\,\operatorname{sign}(\mathbf{x}_{:r})=\mathbf{u}\right]^2\right] \\
&= \mathbb{E}_{\mathbf{u}}\left[\left(1-\frac{1}{2\|\mathbf{w}_{r:}\|_2^2}\mathbf{w}_{:r}^\top\left(\frac{2}{\pi}\mathbf{u}\mathbf{u}^\top+\left(1-\frac{2}{\pi}\right)\mathbf{I}_{d-r}\right)\mathbf{w}_{:r}+O\left(\frac{r^4+\log^4(m)}{d^2}\right)\right)^2\right] \\
&= 1-\frac{1}{\|\mathbf{w}_{r:}\|_2^2}\mathbf{w}_{:r}^\top\left(\frac{2}{\pi}\mathbb{E}_{\mathbf{u}}[\mathbf{u}\mathbf{u}^\top]+\left(1-\frac{2}{\pi}\right)\mathbf{I}_{d-r}\right)\mathbf{w}_{:r} \\
&\quad+\frac{1}{4\|\mathbf{w}_{r:}\|_2^4}\mathbf{w}_{:r}^\top\mathbb{E}_{\mathbf{u}}\left[\left(\frac{2}{\pi}\mathbf{u}\mathbf{u}^\top+\left(1-\frac{2}{\pi}\right)\mathbf{I}_{d-r}\right)\mathbf{w}_{:r}\mathbf{w}_{:r}^\top\left(\frac{2}{\pi}\mathbf{u}\mathbf{u}^\top+\left(1-\frac{2}{\pi}\right)\mathbf{I}_{d-r}\right)\right]\mathbf{w}_{:r} \\
&\quad+O\left(\frac{r^4+\log^4(m)}{d^2}\right) \\
&= 1-\frac{\|\mathbf{w}_{:r}\|_2^2}{\|\mathbf{w}_{r:}\|_2^2}+\frac{\left((1-\frac{2}{\pi})^2+\frac{4}{\pi}\left(1-\frac{2}{\pi}\right)\right)\|\mathbf{w}_{r:}\|_2^4}{4\|\mathbf{w}_{r:}\|_2^4} \\
&\quad+\frac{4}{\pi^2}\mathbf{w}_{:r}^\top\mathbb{E}_{\mathbf{u}}\left[\mathbf{u}\mathbf{u}^\top\mathbf{w}_{:r}\mathbf{w}_{:r}^\top\mathbf{u}\mathbf{u}^\top\right]\mathbf{w}_{:r}+O\left(\frac{r^4+\log^4(m)}{d^2}\right) \\
&= 1-\frac{\|\mathbf{w}_{:r}\|_2^2}{\|\mathbf{w}_{r:}\|_2^2}+\frac{\left((1-\frac{2}{\pi})^2+\frac{4}{\pi}\left(1-\frac{2}{\pi}\right)\right)\|\mathbf{w}_{r:}\|_2^4}{4\|\mathbf{w}_{r:}\|_2^4} \\
&\quad+\frac{1}{\pi^2\|\mathbf{w}_{r:}\|_2^4}\mathbf{w}_{:r}^\top\mathbb{E}_{\mathbf{u}}\left[\mathbf{u}\mathbf{u}^\top(\mathbf{u}^\top\mathbf{w}_{:r})^2\right]\mathbf{w}_{:r}+O\left(\frac{r^4+\log^4(m)}{d^2}\right) \\
&= 1-\frac{\|\mathbf{w}_{:r}\|_2^2}{\|\mathbf{w}_{r:}\|_2^2}+\frac{\left((1-\frac{2}{\pi})^2+\frac{4}{\pi}\left(1-\frac{2}{\pi}\right)\right)\|\mathbf{w}_{r:}\|_2^4}{4\|\mathbf{w}_{r:}\|_2^4} \\
&\quad+\frac{1}{\pi^2\|\mathbf{w}_{r:}\|_2^4}\mathbf{w}_{:r}^\top\left(\|\mathbf{w}_{:r}\|_2^2\mathbf{I}_r+2\mathbf{w}_{:r}\mathbf{w}_{:r}^\top-\operatorname{diag}(\mathbf{w}_{:r})^2\right)\mathbf{w}_{:r}+O\left(\frac{r^4+\log^4(m)}{d^2}\right) \\
&= 1-\frac{\|\mathbf{w}_{:r}\|_2^2}{\|\mathbf{w}_{r:}\|_2^2}+\frac{\left((1-\frac{2}{\pi})^2+\frac{4}{\pi}\left(1-\frac{2}{\pi}\right)\right)\|\mathbf{w}_{r:}\|_2^4}{4\|\mathbf{w}_{r:}\|_2^4}+\frac{3\|\mathbf{w}_{:r}\|_2^4-\|\mathbf{w}_{:r}\|_4^4}{\pi^2\|\mathbf{w}_{r:}\|_2^4}+O\left(\frac{r^4+\log^4(m)}{d^2}\right) \\
&= 1-\frac{\|\mathbf{w}_{:r}\|_2^2}{\|\mathbf{w}_{r:}\|_2^2}+O\left(\frac{r^4+\log^4(m)}{d^2}\right)
\end{aligned}
$$

for an absolute constant c, where we have used $\mathbf{w}\in\mathcal{G}_{\mathbf{w}}$. Combining this with (58) and (59) completes the proof. □

Lemma A.9. *For any $\mathbf{w}\in\mathcal{G}_{\mathbf{w}}$, we have*

$$
\left\|\mathbf{A}_{\|,\perp}(\mathbf{w})-\frac{\mathbf{w}_{:r}\mathbf{w}_{:r}^\top}{2\pi\|\mathbf{w}_{r:}\|_2^2}\right\|_2\le c\frac{r^{3.5}+\log^{3.5}(m)}{d^{1.5}} \tag{61}
$$

Proof. Arguing similarly to the previous two lemmas, we obtain

$$\begin{aligned}
\mathbf{A}_{\parallel,\perp}(\mathbf{w}) &= \mathbb{E}_{\mathbf{u},\mathbf{x},\mathbf{x}'}\left[\mathbf{x}_{:r}(\mathbf{x}'_{r:})^\top \sigma'(\mathbf{w}^\top\mathbf{x}')\sigma'(\mathbf{w}^\top\mathbf{x})|\,\mathrm{sign}(\mathbf{x}'_{:r}) = \mathrm{sign}(\mathbf{x}_{:r}) = \mathbf{u}\right] \\
&= \mathbb{E}_{\mathbf{u},\mathbf{x},\mathbf{x}'_{:r}}\left[\mathbf{x}_{:r}\sigma'(\mathbf{w}^\top\mathbf{x})\,\mathbb{E}_{\mathbf{x}'_{r:}}\left[(\mathbf{x}'_{r:})^\top\sigma'(\mathbf{w}^\top\mathbf{x}')\right]|\,\mathrm{sign}(\mathbf{x}'_{:r}) = \mathrm{sign}(\mathbf{x}_{:r}) = \mathbf{u}\right] \\
&= \mathbb{E}_{\mathbf{u},\mathbf{x},\mathbf{x}'_{:r}}\left[\mathbf{x}_{:r}\sigma'(\mathbf{w}^\top\mathbf{x})\exp\left(-\frac{((\mathbf{x}'_{:r})^\top\mathbf{w}_{:r})^2}{2\|\mathbf{w}_{r:}\|_2^2}\right)|\,\mathrm{sign}(\mathbf{x}'_{:r}) = \mathrm{sign}(\mathbf{x}_{:r}) = \mathbf{u}\right] \\
&\quad\times\frac{\mathbf{w}_{:r}^\top}{\sqrt{2\pi}\|\mathbf{w}_{r:}\|_2} \qquad (62)
\end{aligned}$$

where (62) follows by the same calculation as in (57). Next, since the only term that depends on $\mathbf{x}_{r:}$ is $\sigma'(\mathbf{w}^\top\mathbf{x})$, we have

$$\begin{aligned}
&= \mathbb{E}_{\mathbf{u},\mathbf{x}_{:r},\mathbf{x}'_{:r}}\left[\mathbf{x}_{:r}\mathbb{E}_{\mathbf{x}_{r:}}[\sigma'(\mathbf{w}^\top\mathbf{x})]\exp\left(-\frac{((\mathbf{x}'_{:r})^\top\mathbf{w}_{:r})^2}{2\|\mathbf{w}_{r:}\|_2^2}\right)|\,\mathrm{sign}(\mathbf{x}'_{:r}) = \mathrm{sign}(\mathbf{x}_{:r}) = \mathbf{u}\right] \\
&\quad\times\frac{\mathbf{w}_{:r}^\top}{\sqrt{2\pi}\|\mathbf{w}_{r:}\|_2} \\
&= \mathbb{E}_{\mathbf{u},\mathbf{x}_{:r},\mathbf{x}'_{:r}}\left[\mathbf{x}_{:r}\mathbb{P}_{\mathbf{x}_{r:}}[\mathbf{w}_{r:}^\top\mathbf{x}_{r:} > -\mathbf{w}_{:r}^\top\mathbf{x}_{:r}]\exp\left(-\frac{((\mathbf{x}'_{:r})^\top\mathbf{w}_{:r})^2}{2\|\mathbf{w}_{r:}\|_2^2}\right)|\,\mathrm{sign}(\mathbf{x}'_{:r}) = \mathrm{sign}(\mathbf{x}_{:r}) = \mathbf{u}\right] \\
&\quad\times\frac{\mathbf{w}_{:r}^\top}{\sqrt{2\pi}\|\mathbf{w}_{r:}\|_2} \\
&= \mathbb{E}_{\mathbf{u},\mathbf{x},\mathbf{x}'_{:r}}\left[\mathbf{x}_{:r}\left(\frac{1}{2}+\frac{1}{2}\,\mathrm{erf}\left(\frac{\mathbf{x}_{:r}^\top\mathbf{w}_{:r}}{\sqrt{2}\|\mathbf{w}_{r:}\|_2}\right)\right)\exp\left(-\frac{((\mathbf{x}'_{:r})^\top\mathbf{w}_{:r})^2}{2\|\mathbf{w}_{r:}\|_2^2}\right)|\,\mathrm{sign}(\mathbf{x}'_{:r}) = \mathrm{sign}(\mathbf{x}_{:r}) = \mathbf{u}\right] \\
&\quad\times\frac{\mathbf{w}_{:r}^\top}{\sqrt{2\pi}\|\mathbf{w}_{r:}\|_2} \qquad (63)
\end{aligned}$$

where (63) follows by the Gaussian CDF. Note that

$$\begin{aligned}
&\mathbb{E}_{\mathbf{u},\mathbf{x},\mathbf{x}'_{:r}}\left[\mathbf{x}_{:r}\exp\left(-\frac{((\mathbf{x}'_{:r})^\top\mathbf{w}_{:r})^2}{2\|\mathbf{w}_{r:}\|_2^2}\right)|\,\mathrm{sign}(\mathbf{x}'_{:r}) = \mathrm{sign}(\mathbf{x}_{:r}) = \mathbf{u}\right] \\
&= \mathbb{E}_{\mathbf{u},\mathbf{x}_{:r},\mathbf{x}'_{:r}}\left[\mathrm{diag}(\mathbf{u})|\mathbf{x}_{:r}|\exp\left(-\frac{(|\mathbf{x}'_{:r}|^\top\mathrm{diag}(\mathbf{u})\mathbf{w}_{:r})^2}{2\|\mathbf{w}_{r:}\|_2^2}\right)\right] \\
&= \mathbb{E}_{\mathbf{x}_{:r},\mathbf{x}'_{:r}}\left[\mathbb{E}_{\mathbf{u}}\left[\mathrm{diag}(\mathbf{u})|\mathbf{x}_{:r}|\left(1-\frac{(|\mathbf{x}'_{:r}|^\top\mathrm{diag}(\mathbf{u})\mathbf{w}_{:r})^2}{2\|\mathbf{w}_{r:}\|_2^2}+\frac{(|\mathbf{x}'_{:r}|^\top\mathrm{diag}(\mathbf{u})\mathbf{w}_{:r})^4}{2\|\mathbf{w}_{r:}\|_2^4}-\dots\right)\right]\right] \qquad (64) \\
&= \mathbf{0} \qquad (65)
\end{aligned}$$

where (65) follows since each term in (64) is an odd power of $\mathbf{u}$. Thus, from (63) we have

$$\begin{aligned}
\mathbf{A}_{\parallel,\perp}(\mathbf{w}) &= \mathbb{E}_{\mathbf{x},\mathbf{x}'_{:r}}\left[\mathbf{x}_{:r}\,\mathrm{erf}\left(\frac{\mathbf{x}_{:r}^\top\mathbf{w}_{:r}}{\sqrt{2}\|\mathbf{w}_{r:}\|_2}\right)\exp\left(-\frac{((\mathbf{x}'_{:r})^\top\mathbf{w}_{:r})^2}{2\|\mathbf{w}_{r:}\|_2^2}\right)|\,\mathrm{sign}(\mathbf{x}'_{:r}) = \mathrm{sign}(\mathbf{x}_{:r}) = \mathbf{u}\right] \\
&\quad\times\frac{\mathbf{w}_{:r}^\top}{2\sqrt{2\pi}\|\mathbf{w}_{r:}\|_2} \qquad (66)
\end{aligned}$$

Next we take the Taylor expansions of $\mathrm{erf}(x)$ and $\exp(-x^2)$ to obtain

$$\begin{aligned}
&\mathbf{A}_{\|,\perp}(\mathbf{w}) \\
&= \mathbb{E}_{\mathbf{u},\mathbf{x},\mathbf{x}'_{:r}}\Bigg[\mathbf{x}_{:r}\left(\frac{\mathbf{x}_{:r}^\top \mathbf{w}_{:r}}{\|\mathbf{w}_{r:}\|_2} - \frac{(\mathbf{x}_{:r}^\top \mathbf{w}_{:r})^3}{6\|\mathbf{w}_{r:}\|_2^3} + \dots\right) \\
&\qquad \times \left(1 - \frac{((\mathbf{x}'_{:r})^\top \mathbf{w}_{:r})^2}{2\|\mathbf{w}_{r:}\|_2^2} + \dots\right) | \operatorname{sign}(\mathbf{x}'_{:r}) = \operatorname{sign}(\mathbf{x}_{:r}) = \mathbf{u}\Bigg] \frac{\mathbf{w}_{:r}^\top}{2\pi\|\mathbf{w}_{r:}\|_2} \\
&= \mathbb{E}_{\mathbf{u},\mathbf{x},\mathbf{x}'_{:r}}\left[\mathbf{x}_{:r}\mathbf{x}_{:r}^\top | \operatorname{sign}(\mathbf{x}'_{:r}) = \operatorname{sign}(\mathbf{x}_{:r}) = \mathbf{u}\right] \frac{\mathbf{w}_{:r}\mathbf{w}_{:r}^\top}{2\pi\|\mathbf{w}_{r:}\|_2^2} + \mathbf{E} \\
&= \frac{\mathbf{w}_{:r}\mathbf{w}_{:r}^\top}{2\pi\|\mathbf{w}_{r:}\|_2^2} + \mathbf{E}
\end{aligned}$$

where $\|\mathbf{E}\|_2 = O\left(\frac{r^{3.5}+\log^{3.5}(m)}{d^{1.5}}\right)$ since $\|\mathbf{w}_{r:}\|_2 = \Omega(\sqrt{d})$ and $\|\mathbf{w}_{:r}\|_2 = O(\sqrt{r} + \sqrt{\log(m)})$ as $\mathbf{w} \in \mathcal{G}_{\mathbf{w}}$, and $\mathbb{E}_{\mathbf{u},\mathbf{x},\mathbf{x}'_{:r}}\left[\mathbf{x}_{:r}\mathbf{x}_{:r}^\top | \operatorname{sign}(\mathbf{x}'_{:r}) = \operatorname{sign}(\mathbf{x}_{:r}) = \mathbf{u}\right] = \mathbf{I}_r$ due to the same argument as in (42). □

Lemma A.10. *Set $\eta = \Theta(1)$ and $\lambda_{\mathbf{w}} = \frac{2^{-r-2}\pi+\eta}{\eta 2^{-r-2}\pi}$. We have, for any $\mathbf{w}_j^0 \in \mathcal{G}_{\mathbf{w}}$, there are absolute constants c, c' such that*

$$\begin{aligned}
\left\|\mathbf{w}_{j,:r}^0 - \frac{\eta^2}{2^{r+2}}\mathbf{w}_{j,:r}^0\right\|_2 &\le c\nu_{\mathbf{w}}\left(\frac{r^4+\log^4(m)}{2^r d} + de^{-c'd} + d\frac{\sqrt{r}}{\sqrt{T}}\right) \\
\left\|\mathbf{w}_{j,r:}^0\right\|_2 &\le c\nu_{\mathbf{w}}\left(\frac{r^{3.5}+\log^{3.5}(m)}{2^r d^{1.5}} + de^{-c'd} + d\frac{\sqrt{r}}{\sqrt{T}}\right)
\end{aligned}$$

Proof. Due to the computation of the gradient in Lemma A.6, we have

$$\begin{aligned}
\mathbf{w}_j^+ &= (1-\eta\lambda_{\mathbf{w}})\mathbf{w}_j^0 - \eta\nabla_{\mathbf{w}_j}\mathcal{L}(\mathbf{W}^0, \mathbf{b}^0, \{\mathbf{a}_i^+\}_{i\in[n]}) \\
&= (1-\eta\lambda_{\mathbf{w}})\mathbf{w}_j^0 + \eta^2 2^{-r}\mathbf{A}(\mathbf{w}_j^0)\mathbf{w}_j^0 + \eta\tilde{\mathbf{e}} \\
\mathbf{w}_{j,:r}^+ &= (1-\eta\lambda_{\mathbf{w}})\mathbf{w}_{j,:r}^0 + \eta^2 2^{-r}\mathbf{A}_{\|,\|}(\mathbf{w}_j^0)\mathbf{w}_{j,:r}^0 + \eta^2 2^{-r}\mathbf{A}_{\|,\perp}(\mathbf{w}_j^0)\mathbf{w}_{j,r:}^0 + \eta\tilde{\mathbf{e}}_{:r} \\
\mathbf{w}_{j,r:}^+ &= (1-\eta\lambda_{\mathbf{w}})\mathbf{w}_{j,r:}^0 + \eta^2 2^{-r}\mathbf{A}_{\perp,\|}(\mathbf{w}_j^0)\mathbf{w}_{j,:r}^0 + \eta^2 2^{-r}\mathbf{A}_{\perp,\perp}(\mathbf{w}_j^0)\mathbf{w}_{j,r:}^0 + \eta\tilde{\mathbf{e}}_{r:}
\end{aligned}$$

where $\|\tilde{\mathbf{e}}\|_2 \le O(\nu_{\mathbf{w}} de^{-cd} + \nu_{\mathbf{w}} d\frac{\sqrt{r}}{\sqrt{T}})$.

Next,

$$\begin{aligned}
\mathbf{w}_{j,:r}^+ &= (1-\eta\lambda_{\mathbf{w}})\mathbf{w}_{j,:r}^0 + \eta^2 2^{-r}\mathbf{w}_{j,:r}^0 + \frac{\eta^2 2^{-r}}{2\pi\|\mathbf{w}_{j,r:}^0\|_2^2}\mathbf{w}_{j,:r}^0(\mathbf{w}_{j,r:}^0)^\top\mathbf{w}_{j,r:}^0 \\
&\quad + \eta^2 2^{-r}\left(\mathbf{A}_{\|,\|}(\mathbf{w}_j^0) - \tfrac{1}{4}\mathbf{I}_r\right)\mathbf{w}_{j,:r}^0 \\
&\quad + \eta^2 2^{-r}\left(\mathbf{A}_{\|,\perp}(\mathbf{w}_j^0) - \frac{1}{2\pi\|\mathbf{w}_{j,r:}^0\|_2^2}\mathbf{w}_{j,:r}^0(\mathbf{w}_{j,r:}^0)^\top\right)\mathbf{w}_{j,r:}^0 + \eta\tilde{\mathbf{e}}_{:r} \\
&= \left(1 - \eta\lambda_{\mathbf{w}} + \eta^2 2^{-r-2}\left(1+\frac{2}{\pi}\right)\right)\mathbf{w}_{j,:r}^0 + \eta^2 2^{-r}\mathbf{e}_{\|} + \eta\tilde{\mathbf{e}}_{:r} \qquad (67)
\end{aligned}$$

where

$$\begin{aligned}
\|\mathbf{e}_{\|}\|_2 &= \left\| \left(\mathbf{A}_{\|,\|}(\mathbf{w}_j^0) - \tfrac{1}{4}\mathbf{I}_r\right)\mathbf{w}_{j,:r}^0 + \left(\mathbf{A}_{\|,\perp}(\mathbf{w}_j^0) - \frac{1}{2\pi\|\mathbf{w}_{j,r:}^0\|_2^2}\mathbf{w}_{j,:r}^0(\mathbf{w}_{j,r:}^0)^\top\right)\mathbf{w}_{j,r:}^0 \right\|_2 \\
&\le \left\| \left(\mathbf{A}_{\|,\|}(\mathbf{w}_j^0) - \tfrac{1}{4}\mathbf{I}_r\right)\mathbf{w}_{j,:r}^0 \right\|_2 \\
&\quad + \left\| \left(\mathbf{A}_{\|,\perp}(\mathbf{w}_j^0) - \frac{1}{2\pi\|\mathbf{w}_{j,r:}^0\|_2^2}\mathbf{w}_{j,:r}^0(\mathbf{w}_{j,r:}^0)^\top\right)\mathbf{w}_{j,r:}^0 \right\|_2 \\
&\le \left\|\mathbf{A}_{\|,\|}(\mathbf{w}_j^0) - \tfrac{1}{4}\mathbf{I}_r\right\|_2 \left\|\mathbf{w}_{j,:r}^0\right\|_2 \\
&\quad + \left\| \mathbf{A}_{\|,\perp}(\mathbf{w}_j^0) - \frac{1}{2\pi\|\mathbf{w}_{j,r:}^0\|_2^2}\mathbf{w}_{j,:r}^0(\mathbf{w}_{j,r:}^0)^\top \right\|_2 \left\|\mathbf{w}_{j,r:}^0\right\|_2 \\
&= O\left(\nu_{\mathbf{w}}\frac{r^4+\log^4(m)}{d}\right) \qquad (68)
\end{aligned}$$

where (68) follows by Lemmas A.7 and A.9 and the fact that $\mathbf{w} \in \mathcal{G}_{\mathbf{w}}$.

Similarly,

$$\begin{aligned}
\mathbf{w}_{j,r:}^+ &= (1-\eta\lambda_{\mathbf{w}})\mathbf{w}_{j,r:}^0 + \eta^2 2^{-r}\frac{\mathbf{w}_{j,r:}^0(\mathbf{w}_{j,:r}^0)^\top}{2\pi\|\mathbf{w}_{j,r:}^0\|_2^2}\mathbf{w}_{j,:r}^0 \\
&\quad + \left(1 - \frac{\|\mathbf{w}_{j,:r}^0\|_2^2}{\|\mathbf{w}_{j,r:}^0\|_2^2}\right)\eta^2 2^{-r}\frac{\mathbf{w}_{j,r:}^0(\mathbf{w}_{j,r:}^0)^\top}{2\pi\|\mathbf{w}_{j,r:}^0\|_2^2}\mathbf{w}_{j,r:}^0 \\
&\quad + \eta^2 2^{-r}\left(\mathbf{A}_{\perp,\|}(\mathbf{w}_j^0) - \frac{\mathbf{w}_{j,r:}^0(\mathbf{w}_{j,:r}^0)^\top}{2\pi\|\mathbf{w}_{j,r:}^0\|_2^2}\right)\mathbf{w}_{j,:r}^0 \\
&\quad + \eta^2 2^{-r}\left(\mathbf{A}_{\perp,\perp}(\mathbf{w}_j^0) - \left(1 - \frac{\|\mathbf{w}_{j,:r}^0\|_2^2}{\|\mathbf{w}_{j,r:}^0\|_2^2}\right)\frac{\mathbf{w}_{j,r:}^0(\mathbf{w}_{j,r:}^0)^\top}{2\pi\|\mathbf{w}_{j,r:}^0\|_2^2}\right)\mathbf{w}_{j,r:}^0 + \eta\tilde{\mathbf{e}}_{r:} \\
&= (1-\eta\lambda_{\mathbf{w}})\mathbf{w}_{j,r:}^0 + \eta^2 2^{-r}\frac{\mathbf{w}_{j,r:}^0(\mathbf{w}_{j,r:}^0)^\top}{2\pi\|\mathbf{w}_{j,r:}^0\|_2^2}\mathbf{w}_{j,r:}^0 \\
&\quad + \eta^2 2^{-r}\left(\mathbf{A}_{\perp,\|}(\mathbf{w}_j^0) - \frac{\mathbf{w}_{j,r:}^0(\mathbf{w}_{j,:r}^0)^\top}{2\pi\|\mathbf{w}_{j,r:}^0\|_2^2}\right)\mathbf{w}_{j,:r}^0 \\
&\quad + \eta^2 2^{-r}\left(\mathbf{A}_{\perp,\perp}(\mathbf{w}_j^0) - \left(1 - \frac{\|\mathbf{w}_{j,:r}^0\|_2^2}{\|\mathbf{w}_{j,r:}^0\|_2^2}\right)\frac{\mathbf{w}_{j,r:}^0(\mathbf{w}_{j,r:}^0)^\top}{2\pi\|\mathbf{w}_{j,r:}^0\|_2^2}\right)\mathbf{w}_{j,r:}^0 + \eta\tilde{\mathbf{e}}_{r:} \\
&= \left(1 - \eta\lambda_{\mathbf{w}} + \eta^2 2^{-r-2}\frac{2}{\pi}\right)\mathbf{w}_{j,r:}^0 + \eta^2 2^{-r}\mathbf{e}_\perp + \eta\tilde{\mathbf{e}}_{r:} \qquad (69)
\end{aligned}$$

where

$$\begin{aligned}
\|\mathbf{e}_\perp\| &= \left\| \left(\mathbf{A}_{\perp,\|}(\mathbf{w}_j^0) - \frac{\mathbf{w}_{j,r:}^0 (\mathbf{w}_{j,:r}^0)^\top}{2\pi \|\mathbf{w}_{j,r:}^0\|_2^2} \right) \mathbf{w}_{j,:r}^0 \right. \\
&\quad \left. + \left(\mathbf{A}_{\perp,\perp}(\mathbf{w}_j^0) - \left(1 - \frac{\|\mathbf{w}_{j,:r}^0\|_2^2}{\|\mathbf{w}_{j,r:}^0\|_2^2}\right) \frac{\mathbf{w}_{j,r:}^0 (\mathbf{w}_{j,r:}^0)^\top}{2\pi \|\mathbf{w}_{j,r:}^0\|_2^2} \right) \mathbf{w}_{j,r:}^0 \right\|_2 \\
&\le \left\| \mathbf{A}_{\perp,\|}(\mathbf{w}_j^0) - \frac{\mathbf{w}_{j,r:}^0 (\mathbf{w}_{j,:r}^0)^\top}{2\pi \|\mathbf{w}_{j,r:}^0\|_2^2} \right\|_2 \|\mathbf{w}_{j,:r}^0\|_2 \\
&\quad + \left\| \mathbf{A}_{\perp,\perp}(\mathbf{w}_j^0) - \left(1 - \frac{\|\mathbf{w}_{j,:r}^0\|_2^2}{\|\mathbf{w}_{j,r:}^0\|_2^2}\right) \frac{\mathbf{w}_{j,r:}^0 (\mathbf{w}_{j,r:}^0)^\top}{2\pi \|\mathbf{w}_{j,r:}^0\|_2^2} \right\|_2 \|\mathbf{w}_{j,r:}^0\|_2 \\
&= O\left(\nu_{\mathbf{w}} \frac{r^{3.5} + \log^{3.5}(m)}{d^{1.5}} \right) \qquad (70)
\end{aligned}$$

where (70) follows by Lemmas A.8 and A.9 and the fact that $\mathbf{w} \in \mathcal{G}_{\mathbf{w}}$. Applying the choice of $\lambda_{\mathbf{w}}$ completes the proof. □

Finally we are ready to prove Proposition 4.1 and Theorem 4.2. For convenience, we restate the statements here.

Proposition A.11. *Consider the gradient-based multi-task algorithm described in Section 3 and suppose Assumption 2.1 holds. Further assume $d = \Omega(r^4)$, and let $\eta = \Theta(1)$, $\lambda_{\mathbf{a}} = 1/\eta$, $\lambda_{\mathbf{w}} = (1 + (\eta^2 2^{-r-1})/\pi)/\eta$, and $\nu_{\mathbf{w}} = O(d^{-1.25})$. Then for any $m = O(d)$, with probability at least 0.99 over the random initialization we have*

1. $\frac{1}{\nu_{\mathbf{w}}\sqrt{m}} \|\Pi_\|(\mathbf{W}^+) - \Omega(\frac{1}{2^r})\Pi_\|(\mathbf{W}^0)\|_2 = O\left(\frac{r^4 + \log^4(m)}{2^r d} + \frac{\sqrt{r}d}{\sqrt{T}}\right)$,
2. $\frac{1}{\nu_{\mathbf{w}}\sqrt{m}} \|\Pi_\perp(\mathbf{W}^+)\|_2 = O\left(\frac{r^{3.5} + \log^{3.5}(m)}{2^r d^{1.5}} + \frac{\sqrt{r}d}{\sqrt{T}}\right)$

Proof. By Lemma A.1, $\mathbf{w}_j^0 \in \mathcal{G}_{\mathbf{w}}$ for all $j \in [m]$ with probability at least 0.99. Conditioned on this event, we can directly apply Lemma A.10 along with the fact that $\|\mathbf{B}\|_2 \le \|\mathbf{B}\|_F \sqrt{m} \max_{j\in[m]} \|\mathbf{b}_j\|_2$ for any matrix $\mathbf{B} \in \mathbb{R}^{m\times d}$, where $\mathbf{b}_j$ is the j-th row of $\mathbf{B}$. We also use that $\frac{r^4 + \log^4(m)}{2^r d} \gg d e^{-c'd}$ and $\frac{r^{3.5} + \log^{3.5}(m)}{2^r d^{1.5}} \gg d e^{-c'd}$ as d is sufficiently large. □

Theorem A.12. *Consider the setting in Proposition 4.1 and let $d = \Omega(r^4 + \log^4(m))$, $m = \Omega(r)$, and $T = \Omega(r 2^{2r} d^2)$. Let $\sigma_r(\mathbf{B})$ denote the r-th singular value of a matrix $\mathbf{B}$. Then with probability at least 0.99 over the random initialization, we have*

$$\frac{\|\Pi_\perp(\mathbf{W}^+)\|_2}{\sigma_r(\Pi_\|(\mathbf{W}^+))} = O\left(\frac{r^{3.5} + \log^{3.5}(m)}{d^{1.5}} + \frac{2^r \sqrt{r} d}{\sqrt{T}} \right).$$

Proof. By Lemmas A.1 and A.2 and a union bound, we have that $\mathbf{w}_j^0 \in \mathcal{G}_{\mathbf{w}}$ for all $j \in [m]$, and $\nu_{\mathbf{w}}\sqrt{m}\left(1 - c\frac{\sqrt{r}}{\sqrt{m}}\right)$ with probability at least 0.99. Conditioned on these events, we can invoke Lemma

A.10 to obtain

$$\begin{aligned}\sigma_r(\Pi_{\|}(\mathbf{W}^+)) &\geq \eta 2^{-r-2}\sigma_r(\Pi_{\|}(\mathbf{W}^0)) - \nu_{\mathbf{w}}\sqrt{\bar{m}}O\left(\frac{r^4+\log^4(m)}{2^r d} + \frac{\sqrt{r}d}{\sqrt{T}}\right) \\ &\geq \eta 2^{-r-2}\nu_{\mathbf{w}}\sqrt{\bar{m}} - \nu_{\mathbf{w}}\sqrt{\bar{m}}O\left(\frac{\sqrt{r}}{2^r\sqrt{\bar{m}}} - \frac{r^4+\log^4(m)}{2^r d} + \frac{\sqrt{r}d}{\sqrt{T}}\right) \\ &= \Omega(2^{-r}\nu_{\mathbf{w}}\sqrt{\bar{m}}) \end{aligned} \tag{71}$$

Likewise, conditioned again on the event that $\mathbf{w}_j^0 \in \mathcal{G}_{\mathbf{w}}$ for all $j \in [m]$,

$$\|\Pi_{\perp}(\mathbf{W}^+)\|_2 = \nu_{\mathbf{w}}\sqrt{\bar{m}}O\left(\frac{r^{3.5}+\log^{3.5}(m)}{2^r d^{1.5}} + \frac{\sqrt{r}d}{\sqrt{T}}\right). \tag{72}$$

Combining (71) and (72) completes the proof. □

B Proof of Downstream Guarantees.

In this section we prove Theorem 4.3 and Corollary 4.4. Given a function $\bar{f}^* : \{-\frac{1}{\sqrt{r}}, \frac{1}{\sqrt{r}}\}^r \to \{-1, 1\}$, we refer to the sets $\mathcal{V}^+ \coloneqq \{\mathbf{v}_{:r} \in \{-\frac{1}{\sqrt{r}}, \frac{1}{\sqrt{r}}\}^r : \bar{f}^* = 1\}$ and $\mathcal{V}^- \coloneqq \{\mathbf{v}_{:r} \in \{-\frac{1}{\sqrt{r}}, \frac{1}{\sqrt{r}}\}^r : \bar{f}^* = -1\}$ as the inverse sets of $\bar{f}^*$. Note that solving the task described by $\bar{f}^*$ entails finding a classifier that separates the inverse sets of $\bar{f}^*$. As in A, we will abuse notation by reusing c, c', c'' and C as absolute constants independent of all other parameters. The notations $O(\cdot)$, $\Theta(\cdot)$, and $\Omega(\cdot)$ describe scalings up to absolute constants independent of all other parameters.

Proof summary. Informally, the proof of Theorem 4.3 follows six steps:

1. Construct a two-layer ReLU network embedding using the weights output by the during multi-task pretraining algorithm for the first-layer weights, then randomly sampling first-layer biases and second layer weights and biases (calling this the "learned" embedding),
2. Construct a nearby two-layer ReLU network embedding that is "purified" in the sense that it is only a function of the the r label-relevant features of the input,
3. Show that the "purified" network linearly separates the pair of inverse sets corresponding to any binary function on the r-dimensional hypercube with high probability as long as the number of neurons in each layer is larger than some function of r,
4. Prove that the outputs of the learned embedding are very close to the outputs of the "purified" embedding, meaning the learned embedding has the same linear separation capability with only slightly smaller margin,
5. Prove that the learned embedding linearly separates the inverse sets with lower bounded margin, and
6. Apply a standard generalization result for linear classification showing that the empirically-optimal head achieves loss close to the minimal loss (zero).

Step 1: Construct downstream embedding. We start by stating the construction of the downstream classifier as described first in Section 3. Let $\mathbf{W}^+$ be the model weights resulting from one step of the multitask representation learning algorithm. For simplicity, here we discard the last half of the weights and consider a network with $\bar{m} \coloneqq m/2$ neurons in the first layer (this means that all neurons are independent, which was originally not the case due to the symmetric initialization, and thereby alleviates technical issues but is not necessary).

Let $\bar{\mathbf{W}}^+ \in \mathbb{R}^{\bar{m} \times d}$ denote the submatrix of $\mathbf{W}^+$ obtained by taking its top $\bar{m}$ rows. Similarly, let $\bar{\mathbf{W}}^0 \in \mathbb{R}^{\bar{m} \times d}$ denote the submatrix of $\mathbf{W}^0$ obtained by taking its top $\bar{m}$ rows.

The downstream classifier is a linear head composed with a two-layer ReLU network embedding with $\bar{m}$ neurons in the first layer and $\hat{m}$ neurons in the second layer. For now, we focus on the embedding itself, excluding the linear classification head. The weights of the first layer of the embedding are equal to the weights in $\bar{\mathbf{W}}^+$ up to rescaling. The biases of the first layer and weights and biases of

the second layer are contained in $\mathbf{b}$, $\hat{\mathbf{W}} := [\hat{\mathbf{w}}_1, \ldots, \hat{\mathbf{w}}_{\hat{m}}]^\top$ and $\hat{\mathbf{b}}$, respectively, where

$$\mathbf{b} \sim \text{Unif}\left(\left[-\frac{\sqrt{2}\gamma}{\sqrt{\bar{m}}}, \frac{\sqrt{2}\gamma}{\sqrt{\bar{m}}}\right]^m\right)$$
$$\hat{\mathbf{w}}_j \sim \mathcal{N}\left(\mathbf{0}_d, \frac{2}{\hat{m}}\mathbf{I}_d\right) \quad \forall\, j = 1, \ldots, \hat{m}$$
$$\hat{\mathbf{b}} \sim \text{Unif}\left(\left[-\frac{\sqrt{2}\hat{\gamma}}{\sqrt{\hat{m}}}, \frac{\sqrt{2}\hat{\gamma}}{\sqrt{\hat{m}}}\right]^{\hat{m}}\right)$$

for some $\gamma, \hat{\gamma} > 0$. The full embedding is given by:

$$g(\mathbf{v}; \alpha\bar{\mathbf{W}}^+, \mathbf{b}, \hat{\mathbf{W}}, \hat{\mathbf{b}}) := \sigma(\hat{\mathbf{W}}\sigma(\alpha\bar{\mathbf{W}}^+\mathbf{v} + \bar{\mathbf{b}}) + \hat{\mathbf{b}}) \quad \forall\, \mathbf{v} \in \{-1, 1\}^d \tag{73}$$

where $\alpha := \frac{2^{r+2.5}}{\sqrt{\bar{m}}\eta^2}$ is a rescaling factor.

For ease of notation we denote $g(\mathbf{v}) := g(\mathbf{v}; \alpha\bar{\mathbf{W}}^+, \mathbf{b}, \hat{\mathbf{W}}, \hat{\mathbf{b}})$.

Step 2: Construct purified downstream embedding. Next, the "purified" embedding also has $\bar{m}$ neurons in the first layer and $\hat{m}$ neurons in the second layer, and is also parameterized by $\mathbf{b}$, $\mathbf{W}$ and $\mathbf{b}$, for the first layer biases and second layer weights and biases, respectively, but has a different construction of the first layer weights. In particular, the first-layer weights are equal to the first r coordinates of the corresponding weights in $\mathbf{W}^0$, up to rescaling, and zero on all other coordinates. Formally, this embedding is given by

$$g(\mathbf{v}; [\bar{\mathbf{W}}^0_{:r}, \mathbf{0}_{\bar{m}\times(d-r)},], \mathbf{b}, \hat{\mathbf{W}}, \hat{\mathbf{b}}) := \sigma(\hat{\mathbf{W}}\sigma(\hat{\alpha}\bar{\mathbf{W}}^0_{:r}\mathbf{v}_{:r} + \mathbf{b}) + \hat{\mathbf{b}}) \quad \forall\, \mathbf{v} \in \{-1, 1\}^d \tag{74}$$

where $\hat{\alpha} = \frac{2}{\nu_{\mathbf{w}}\sqrt{\bar{m}}}$.

For ease of notation we denote $\tilde{g}(\mathbf{v}_{:r}) := g([\mathbf{v}_{:r}, \mathbf{0}_{d-r}]; [\hat{\alpha}\bar{\mathbf{W}}^0_{:r}, \mathbf{0}_{\bar{m}\times(d-r)},], \mathbf{b}, \hat{\mathbf{W}}, \hat{\mathbf{b}})$.

Step 3: Purified embedding linearly separates two classes. We start by showing that for any function $\bar{f}^* : \{-1, 1\}^r \to \{-1, 1\}$, with high probability $\tilde{g}$ linearly separates the pair of inverse sets $\mathcal{V}^+ := \{\mathbf{v} \in \{-1, 1\}^r : \bar{f}^* = 1\}$ and $\mathcal{V}^- := \{\mathbf{v} \in \{-1, 1\}^r : \bar{f}^* = -1\}$ with lower bounded margin by adapting a result of Dirksen et al. [2022]. Note that we have not optimized the dependence of $\bar{m}$ on log() and the dependence of $\hat{m}$ on log() factors in the exponent.

Lemma B.1 (Adapted from Theorem 1 in Dirksen et al. [2022]). *Let $\delta \in (0, 0.05]$, $\gamma = \Theta(\log(r))$, $\hat{\gamma} = \Theta(r^{2.5}\log^4(r))$, $\bar{m} = \Omega(r^5\log^8(r)\log(1/\delta))$, and $\hat{m} = \exp(\Omega(\bar{m}))$. Consider any function $\bar{f}^* : \{-\frac{1}{\sqrt{r}}, \frac{1}{\sqrt{r}}\}^r \to \{-1, 1\}$. With probability at least $1 - \delta$, $\tilde{g}$ makes the classes $\mathcal{V}^+ := \{\mathbf{v}_{:r} \in \{-\frac{1}{\sqrt{r}}, \frac{1}{\sqrt{r}}\}^r : \bar{f}^* = 1\}$ and $\mathcal{V}^- := \{\mathbf{v}_{:r} \in \{-\frac{1}{\sqrt{r}}, \frac{1}{\sqrt{r}}\}^r : \bar{f}^* = -1\}$ linearly separable with margin $\mu = \exp\left(-O(r^5\log^6(r)\log(\log(r)/\delta))\right)$, i.e. there exists a vector $\mathbf{a} \in \mathbb{R}^{\hat{m}}$ with $\|\mathbf{a}\|_2 = 1$ and bias $\tau \in \mathbb{R}$ such that for all $\mathbf{v}_{:r} \in \left\{-\frac{1}{\sqrt{r}}, \frac{1}{\sqrt{r}}\right\}^r$,*

$$\bar{f}^*(\mathbf{v}) = 1 \implies \mathbf{a}^\top\tilde{g}(\mathbf{v}_{:r}) + \tau > \mu$$
$$\bar{f}^*(\mathbf{v}) = -1 \implies \mathbf{a}^\top\tilde{g}(\mathbf{v}_{:r}) + \tau < -\mu$$

Proof. The above statement is a special case of Theorem 1 in Dirksen et al. [2022] with positive and negative sets $\mathcal{V}^+ \coloneqq \{\mathbf{v}_{:r} \in \{-\frac{1}{\sqrt{r}}, \frac{1}{\sqrt{r}}\}^r : \bar{f}^* = 1\}$ and $\mathcal{V}^- \coloneqq \{\mathbf{v}_{:r} \in \{-\frac{1}{\sqrt{r}}, \frac{1}{\sqrt{r}}\}^r : \bar{f}^* = -1\}$, respectively. Note that the distance between these sets is $\frac{2}{\sqrt{r}}$, and we have used $|\mathcal{V}^+||\mathcal{V}^-| \le 2^{2r}$. □

Lemma B.2. *Let $\gamma = \Theta(\sqrt{r}\log(r))$, $\hat{\gamma} = \Theta(r^3\log^4(r))$, and $\bar{m}$, $\hat{m}$ satisfy the same conditions as in Lemma B.1, for any $\delta \in (0, 0.05]$. Consider any function $\bar{f}^* : \{-1,1\}^r \to \{-1,1\}$. With probability at least $1-\delta$, $\tilde{g}$ makes the classes $\mathcal{V}^+ \coloneqq \{\mathbf{v}_{:r} \in \{-1,1\}^r : \bar{f}^* = 1\}$ and $\mathcal{V}^- \coloneqq \{\mathbf{v}_{:r} \in \{-1,1\}^r : \bar{f}^* = -1\}$ linearly separable with margin $\mu = \exp(-O(r^5\log^6(r)\log(\log(r)/\delta)))$.*

Proof. The result follows from Lemma B.1 and the fact that $g([\sqrt{r}\mathbf{v}_{:r}, \mathbf{0}_{d-r}]; [\bar{\mathbf{W}}^0_{:r}, \mathbf{0}_{\bar{m}\times(d-r)},], \sqrt{r}\mathbf{b}, \hat{\mathbf{W}}, \sqrt{r}\hat{\mathbf{b}}) = \sqrt{r}\tilde{g}(\mathbf{v}_{:r})$, thus if $\bar{\mathcal{V}}^+ \coloneqq \{\mathbf{v}_{:r} \in \{-\frac{1}{\sqrt{r}}, \frac{1}{\sqrt{r}}\}^r : \bar{f}^* = 1\}$ and $\bar{\mathcal{V}}^- \coloneqq \{\mathbf{v}_{:r} \in \{-\frac{1}{\sqrt{r}}, \frac{1}{\sqrt{r}}\}^r : \bar{f}^* = -1\}$ are linearly separated with margin μ' by $\tilde{g}$, then after rescaling the biases, $\mathcal{V}^+ \coloneqq \{\mathbf{v}_{:r} \in \{-\frac{1}{\sqrt{r}}, \frac{1}{\sqrt{r}}\}^r : \bar{f}^* = 1\}$ and $\mathcal{V}^- \coloneqq \{\mathbf{v}_{:r} \in \{-\frac{1}{\sqrt{r}}, \frac{1}{\sqrt{r}}\}^r : \bar{f}^* = -1\}$ are linearly separated with margin $\sqrt{r}\mu'$. □

Step 4: Downstream embedding is close to purified embedding. Next we compare the outputs of the "purified" embedding $\tilde{g}$ to those of the learned embedding g.

Lemma B.3. *Under the same conditions as Lemma B.2 hold. Additionally, suppose $\eta = \Theta(1)$, $\lambda_{\mathbf{a}} = 1/\eta$, $\lambda_{\mathbf{w}} = (1 + (\eta^2 2^{-r-1})/\pi)/\eta$, and Assumption 2.1 holds, then with probability at least 0.99, for all $\mathbf{v} \in \{-1,1\}^d$,*

$$\|\tilde{g}(\mathbf{v}_{:r}) - g(\mathbf{v})\|_2 = O\left(\frac{r^{3.5} + \log^{3.5}(\bar{m})}{2^r\sqrt{d}} + \frac{\sqrt{r}d^{1.5}}{\sqrt{T}}\right).$$

Proof. For any $\mathbf{v} \in \{-1,1\}^d$, we have

$$
\begin{aligned}
&\|\tilde{g}(\mathbf{v}_{:r}) - g(\mathbf{v})\|_2 \\
&= \left\| \sigma\left(\hat{\mathbf{W}}\sigma\left(\frac{2}{\sqrt{\bar{m}}\nu_{\mathbf{w}}}\bar{\mathbf{W}}^0_{:r}\mathbf{v}_{:r} + \mathbf{b}\right) + \hat{\mathbf{b}}\right) - \sigma\left(\hat{\mathbf{W}}\sigma\left(\frac{2}{\sqrt{\bar{m}}\nu_{\mathbf{w}}\eta^2 2^{-r-2}}\bar{\mathbf{W}}^+\mathbf{v} + \mathbf{b}\right) + \hat{\mathbf{b}}\right)\right\|_2 \\
&\leq \left\| \hat{\mathbf{W}}\sigma\left(\frac{2}{\sqrt{\bar{m}}\nu_{\mathbf{w}}}\bar{\mathbf{W}}^0\mathbf{v} + \mathbf{b}\right) - \hat{\mathbf{W}}\sigma(\frac{2}{\sqrt{\bar{m}}\nu_{\mathbf{w}}\eta^2 2^{-r-2}}\bar{\mathbf{W}}^+\mathbf{v} + \mathbf{b})\right\|_2 && (75) \\
&\leq \|\hat{\mathbf{W}}\|_2 \left\| \frac{2}{\sqrt{\bar{m}}\nu_{\mathbf{w}}}\bar{\mathbf{W}}^0_{:r}\mathbf{v}_{:r} - \frac{2}{\sqrt{\bar{m}}\nu_{\mathbf{w}}\eta^2 2^{-r-2}}\bar{\mathbf{W}}^+\mathbf{v}\right\|_2 && (76) \\
&= \frac{2}{\sqrt{\bar{m}}\nu_{\mathbf{w}}}\|\hat{\mathbf{W}}\|_2 \left\| \bar{\mathbf{W}}^0_{:r}\mathbf{v}_{:r} - \frac{1}{\eta^2 2^{-r-2}}\bar{\mathbf{W}}^+_{:r}\mathbf{v}_{:r} - \frac{1}{\eta^2 2^{-r-2}}\bar{\mathbf{W}}^+_{r:}\mathbf{v}_{r:}\right\|_2 && (77) \\
&\leq \frac{2}{\sqrt{\bar{m}}\nu_{\mathbf{w}}}\|\hat{\mathbf{W}}\|_2 \left\| \bar{\mathbf{W}}^0_{:r}\mathbf{v}_{:r} - \frac{1}{\eta^2 2^{-r-2}}\bar{\mathbf{W}}^+_{:r}\mathbf{v}_{:r}\right\|_2 + \frac{2}{\sqrt{\bar{m}}\nu_{\mathbf{w}}}\|\hat{\mathbf{W}}\|_2 \left\| \frac{1}{\eta^2 2^{-r-2}}\bar{\mathbf{W}}^+_{r:}\mathbf{v}_{r:}\right\|_2 && (78) \\
&\leq \frac{2}{\sqrt{\bar{m}}\nu_{\mathbf{w}}}\|\hat{\mathbf{W}}\|_2 \left\| \bar{\mathbf{W}}^0_{:r} - \frac{1}{\eta^2 2^{-r-2}}\bar{\mathbf{W}}^+_{:r}\right\|_2 \|\mathbf{v}_{:r}\|_2 + \frac{2}{\sqrt{\bar{m}}\nu_{\mathbf{w}}\eta^2 2^{-r-2}}\|\hat{\mathbf{W}}\|_2 \left\|\bar{\mathbf{W}}^+_{r:}\right\|_2 \|\mathbf{v}_{r:}\|_2 && (79) \\
&= \frac{2\sqrt{r}}{\sqrt{\bar{m}}\nu_{\mathbf{w}}}\|\hat{\mathbf{W}}\|_2 \left\| \bar{\mathbf{W}}^0_{:r} - \frac{1}{\eta^2 2^{-r-2}}\bar{\mathbf{W}}^+_{:r}\right\|_2 + \frac{2\sqrt{d-r}}{\sqrt{\bar{m}}\nu_{\mathbf{w}}\eta^2 2^{-r-2}}\|\hat{\mathbf{W}}\|_2 \left\|\bar{\mathbf{W}}^+_{r:}\right\|_2 \\
&= O\left(\frac{r^{4.5} + \sqrt{r}\log^4(\bar{m})}{2^r d} + \frac{rd}{\sqrt{T}}\right)\|\hat{\mathbf{W}}\|_2 + O\left(\frac{r^{3.5} + \log^{3.5}(\bar{m})}{2^r\sqrt{d}} + \frac{\sqrt{r}d^{1.5}}{\sqrt{T}}\right)\|\hat{\mathbf{W}}\|_2 && (80) \\
&= O\left(\frac{r^{3.5} + \log^{3.5}(\bar{m})}{2^r\sqrt{d}} + \frac{\sqrt{r}d^{1.5}}{\sqrt{T}}\right) && (81)
\end{aligned}
$$

where (75) and (76) follow since $\sigma()$ is 1-Lipschitz and by the Cauchy-Schwarz Inequality, (79) follows by the Cauchy-Schwarz Inequality, (80) follows by Proposition A.11 with probability at least 0.999, and (81) follows with probability at least 0.99 (since with high probability, $\|\hat{\mathbf{W}}\|_2 \leq O\left(1 + \frac{\bar{m}}{\hat{m}}\right) = O(1)$). ☐

Step 5: Downstream embedding linearly separates two classes. Now we reason that $\bar{f}^*$ linearly separates the two classes with high probability.

Lemma B.4. *Suppose the same conditions as Lemma B.3 hold. Additionally, suppose* $d = \Omega(\log^7(\bar{m})\exp(cr^5\log^6(r)\log(\log(r)/\delta)))$ *and* $T = \Omega(d^3\exp(cr^5\log^6(r)\log(\log(r)/\delta)))$ *for an absolute constant c. Consider any function* $\bar{f}^* : \{-1,1\}^r \to \{-1,1\}$. *With probability at least* $0.99 - \delta$, *g makes the classes* $\mathcal{V}^+ \coloneqq \{\mathbf{v} \in \{-1,1\}^r : \bar{f}^* = 1\}$ *and* $\mathcal{V}^- \coloneqq \{\mathbf{v} \in \{-1,1\}^r : \bar{f}^* = -1\}$ *linearly separable with margin*

$$\mu = \exp(-O(r^5\log^6(r)\log(\log(r)/\delta)))$$

i.e. there exists a vector $\mathbf{a} \in \mathbb{R}^{\hat{m}}$ *with* $\|\mathbf{a}\|_2 = 1$ *and bias* $\tau \in \mathbb{R}$ *such that for all* $\mathbf{v} \in \{-1,1\}^d$,

$$
\begin{aligned}
\bar{f}^*(\mathbf{v}) = 1 &\implies \mathbf{a}^\top g(\mathbf{v}) + \tau > \mu \\
\bar{f}^*(\mathbf{v}) = -1 &\implies \mathbf{a}^\top g(\mathbf{v}) + \tau < -\mu
\end{aligned}
$$

Proof. The proof follows from Lemmas B.2 and B.3. In particular, for any point $\mathbf{v} \in \{-1,1\}^d$, from Lemma B.2 we have with probability at least $1 - \delta$, for absolute constants c, c', there exists $\mathbf{a} \in \mathbb{R}^{\hat{m}}$

with $\|\mathbf{a}\|_2 = 1$ and $\tau \in \mathbb{R}$ such that

$$\begin{aligned}
\bar{f}^*(\mathbf{v}) = 1 &\implies \mathbf{a}^\top \tilde{g}(\mathbf{v}_{:r}) + \tau > \exp(-cr^5 \log^6(r) \log(\log(r)/\delta)) \\
&\implies \mathbf{a}^\top g(\mathbf{v}) + \tau > \exp(-cr^5 \log^6(r) \log(\log(r)/\delta)) \\
&\qquad\qquad - c' \left(\frac{r^{3.5} + \log^{3.5}(\bar{m})}{2^r \sqrt{d}} + \frac{\sqrt{r} d^{1.5}}{\sqrt{T}} \right) \qquad (82) \\
&\implies \mathbf{a}^\top g(\mathbf{v}_{:r}) + \tau > \exp(-O(r^5 \log^6(r) \log(\log(r)/\delta))) \qquad (83)
\end{aligned}$$

where (82) follows with probability at least $0.99 - \delta$ by Lemma B.3 and a union bound, and (83) follows since d and T are sufficiently large. Repeating the same argument for the case $\bar{f}^*(\mathbf{v}) = -1$ with the same $\mathbf{a}, \tau$ completes the proof. □

Step 6: Complexity of learning the linear head. Now that we have shown that for any binary function on the r-dimensional hypercube, g makes its inverse sets linearly separable with high probability, we complete the proof by bounding the sample complexity of finding a linear separator. For convenience, we restate the theorem here in full detail. Recall that in the previous proofs we have used $\bar{m} = m/2$ in place of m; here we give the statements in terms of m.

Theorem B.5 (End-to-end Guarantee)**.** *Consider a downstream task with labeling function $\bar{f}^*_{T+1}$. Construct the two-layer ReLU embedding $g : \{-1,1\}^d \to \mathbb{R}^{\hat{m}}$ using the rescaled $\mathbf{W}^+$ for first layer weights as in* (73)*, and train the task-adapted head $(\mathbf{a}_{T+1}, \tau_{T+1})$ using n i.i.d. samples from the downstream task, as in* (10)*, with regularization parameter $\hat{\lambda}_{\mathbf{a}} = \exp(-cr^5 \log^6(r) \log(\log(r)/\delta))$ for an absolute constant c. Further, suppose Assumption 2.1 holds, $m = \Omega(r^5 \log^8(r) \log(1/\delta))$, $\hat{m} = \exp(\Omega(m))$, $d = \Omega(\log^7(m) \exp(cr^5 \log^6(r) \log(\log(r)/\delta)))$, and $T = \Omega(d^3 \exp(cr^5 \log^6(r) \log(\log(r)/\delta)))$, and set $\gamma = \Theta(\sqrt{r} \log(r))$, $\hat{\gamma} = \Theta(r^3 \log^4(r))$, $\eta = \Theta(1)$, $\lambda_{\mathbf{a}} = 1/\eta$ and $\lambda_{\mathbf{w}} = (1 - (\eta^2 2^{-r-1})/\pi)/\eta$.*

Then for any $\delta \in (0, 0.05]$, with probability at least $0.99 - \delta$ over the random initializations, draw of T pretraining tasks, and draw of n samples, we have

$$\mathcal{L}^{(eval)}_{T+1} = \frac{\exp(-O(r^5 \log^6(r) \log(\log(r)/\delta)))}{\sqrt{n}}. \qquad (84)$$

Proof. We know that there is an absolute constant c such that with probability at least 0.999, $\max(\|\hat{\mathbf{W}}\|_2, \|\hat{\alpha} \bar{\mathbf{W}}^0\|_2) \le c$. Thus, by Lemma B.3, for any $\mathbf{v} \in \{-1,1\}^d$, $\|g(\mathbf{v})\|_2 \le 2\|\tilde{g}(\mathbf{v}_{:r})\|_2 \le c'(\gamma + \hat{\gamma}) \le c'' r^{2.5} \log^4(r)$ for absolute constants c', c''.

Let $(\mathbf{a}^*_{T+1}, \tau^*_{T+1})$ be the linear head that separates the inverse sets of $\bar{f}^*_{T+1}$ with margin $\mu := \exp(-O(r^5 \log^6(r) \log(\log(r)/\delta)))$, whose existence is guaranteed with high probability by Lemma B.4. We have $|\tau^*_{T+1}| \le \max_{\mathbf{v} \in \{\pm 1\}^d} |(\mathbf{a}^*)^\top g(\mathbf{v})| \le c'' r^{2.5} \log^4(r)$. Thus, $\sqrt{\|\mathbf{a}^*_{T+1}\|_2^2 + (\tau^*_{T+1})^2} \le c''' r^{1.25} \log^2(r) =: B$. Since $(\mathbf{a}^*_{T+1}, \tau^*_{T+1})$ linearly separates the two inverse sets with margin μ, $(\mathbf{a}^*_{T+1}/\mu, \tau^*_{T+1}/\mu)$ linearly separates them with margin 1. Define $\mathcal{H} := \{(\mathbf{a}, \tau) : \mathbf{a} \in \mathbb{R}^{\hat{m}}, \tau \in$

$\mathbb{R}, \sqrt{\|\mathbf{a}^*_{T+1}\|_2^2 + (\tau^*_{T+1})^2} \le B/\mu\}$.

$$\begin{aligned}\mathcal{L}^*_{T+1} &\coloneqq \min_{(\mathbf{a},\tau)\in\mathcal{H}} \frac{1}{2^d} \sum_{\mathbf{v}\in\{\pm1\}^d} \ell(f^*_{T+1}(\mathbf{v}_{:r}), \mathbf{a}^\top g(\mathbf{v}) + \tau) \\ &\le \frac{1}{2^d} \sum_{\mathbf{v}\in\{\pm1\}^d} \ell((\mathbf{a}^*_{T+1})^\top g(\mathbf{v})/\mu + \tau^*_{T+1}/\mu, \bar{f}^*_{T+1}(\mathbf{v}_{:r})) \\ &= 0 \end{aligned} \tag{85}$$

where ℓ is the hinge loss, as usual. Recall that

$$(\mathbf{a}_{T+1}, \tau_{T+1}) \in \underset{\mathbf{a}\in\mathbb{R}^{\hat{m}},\tau\in\mathbb{R}}{\arg\min} \frac{1}{n} \sum_{l=1}^{n} \ell(\mathbf{a}^\top g(\mathbf{v}_l) + \tau, \bar{f}^*_{T+1}(\mathbf{v}_{l,:r})) + \frac{\hat{\lambda}_{\mathbf{a}}}{2}(\|\mathbf{a}\|_2^2 + \tau^2) \tag{86}$$

for a set of n random samples $\{\mathbf{v}_l, f^*_{T+1}(\mathbf{v}_l)\}_{l=1}^n$, where each $\mathbf{v}_l$ is drawn independently uniformly from $\{-1,1\}^d$. Equivalently, for $\hat{\lambda}_{\mathbf{a}} = \mu/B$ we have

$$(\mathbf{a}_{T+1}, \tau_{T+1}) \in \underset{(\mathbf{a},\tau)\in\mathcal{H}}{\arg\min} \frac{1}{n} \sum_{l=1}^{n} \ell(\mathbf{a}^\top g(\mathbf{v}_l) + \tau, \bar{f}^*_{T+1}(\mathbf{v}_{l,:r})) \tag{87}$$

Thus, by applying a standard generalization bound for 1-Lipschitz loss functions Livni [2017], we obtain, for an absolute constant C,

$$\begin{aligned}\mathcal{L}^{(\text{eval})}_{T+1} &:= \frac{1}{2^d} \sum_{\mathbf{v}\in\{\pm1\}^d} \ell(\mathbf{a}^\top_{T+1} g(\mathbf{v}) + \tau_{T+1}, \bar{f}^*_{T+1}(\mathbf{v}_{:r})) \\ &\le \mathcal{L}^*_{T+1} + C\frac{B\sqrt{\log(1/\delta')}}{\mu\sqrt{n}} \\ &= C\frac{B\sqrt{\log(1/\delta')}}{\mu\sqrt{n}} \\ &= O\left(\frac{\exp(O(r^5\log^6(r)\log(\log(r)/\delta)))}{\sqrt{n}}\sqrt{\log(1/\delta')}\right). \end{aligned} \tag{88}$$

with probability at least $0.99 - \delta - \delta'$. Setting $\delta' = \delta^{\text{new}}/2$ and $\delta = \delta^{\text{new}}/2$ for some $\delta^{\text{new}} \in (0, 0.05]$ completes the proof. □

To conclude, we state and prove the full version of Corollary 4.4.

Corollary B.6 (Generalization to set of tasks)**.** *Consider a set of possible downstream tasks $\mathcal{T}^{(eval)}$ with cardinality $|\mathcal{T}^{(eval)}| = D$. Construct the two-layer ReLU embedding $g : \{-1,1\}^d \to \mathbb{R}^{\hat{m}}$ using the rescaled $\mathbf{W}^+$ for first layer weights as in* (73)*, and train the task-adapted head $(\mathbf{a}_{T+1}, \tau_{T+1})$ using n i.i.d. samples from the downstream task, as in* (10)*, with regularization parameter $\hat{\lambda}_{\mathbf{a}} = \exp(-cr^5\log^6(r)\log(\log(r)/\delta))$ for an absolute constant c. Further, suppose Assumption 2.1 holds, $m = \Omega(r^5\log^8(r)\log(1/\delta))$, $\hat{m} = \exp(\Omega(m))$, $d = \Omega(\log^7(m)\exp(cr^5\log^6(r)\log(\log(r)/\delta)))$, and $T = \Omega(d^3\exp(cr^5\log^6(r)\log(\log(r)/\delta)))$, and set $\gamma = \Theta(\sqrt{r}\log(r))$, $\hat{\gamma} = \Theta(r^3\log^4(r))$, $\eta = \Theta(1)$, $\lambda_{\mathbf{a}} = 1/\eta$ and $\lambda_{\mathbf{w}} = (1 - (\eta^2 2^{-r-1})/\pi)/\eta$.*

Then with probability at least $0.99 - \delta$*, any task* $T+1$ *in* $\mathcal{T}^{(eval)}$ *satisfies*

$$\mathcal{L}_{T+1}^{(eval)} = \frac{\exp(O(r^5 \log^6(r) \log(D \log(r)/\delta)))}{\sqrt{n}}, \tag{89}$$

Proof. Note that in the proof of Theorem B.5, δ upper bounds the probability of a bad event occurring that depends on the choice of downstream task (while 0.01 upper bounds the probability of a bad event occurring that is independent of the downstream task, namely the event that any $\mathbf{w}_j^0 \notin \mathcal{G}_{\mathbf{w}}$). Thus, setting setting $\delta^{\text{new}} = \delta/D$ applying a union bound over all tasks implies

$$\mathcal{L}_{T+1}^{(\text{eval})} = \frac{\exp(O(r^5 \log^6(r) \log(\log(r)/\delta^{\text{new}})))}{\sqrt{n}} = \frac{\exp(O(r^5 \log^6(r) \log(D \log(r)/\delta)))}{\sqrt{n}}$$

for all tasks $T+1 \in \mathcal{T}^{\text{eval}}$ with probability at least $0.99 - \delta^{\text{new}} D = 0.99 - \delta$, as desired. □