Dropout: Explicit Forms and Capacity Control

Raman Arora* Peter Bartlett† Poorya Mianjy‡ Nathan Srebro§

March 10, 2020

Abstract

We investigate the capacity control provided by dropout in various machine learning problems. First, we study dropout for matrix completion, where it induces a data-dependent regularizer that, in expectation, equals the weighted trace-norm of the product of the factors. In deep learning, we show that the data-dependent regularizer due to dropout directly controls the Rademacher complexity of the underlying class of deep neural networks. These developments enable us to give concrete generalization error bounds for the dropout algorithm in both matrix completion as well as training deep neural networks. We evaluate our theoretical findings on real-world datasets, including MovieLens, MNIST, and Fashion-MNIST.

1 Introduction

Dropout is a popular algorithmic regularization technique for training deep neural networks that aims at "breaking co-adaptation" among neurons by randomly dropping them at training time (Hinton et al., 2012). Dropout has been shown effective across a wide range of machine learning tasks, from classification (Srivastava et al., 2014; Szegedy et al., 2015) to regression (Toshev & Szegedy, 2014). Notably, dropout is considered an essential component in the design of AlexNet (Krizhevsky et al., 2012), which won the prominent ImageNet challenge in 2012 with a significant margin and helped transform the field of computer vision.

Dropout regularizes the empirical risk by randomly perturbing the model parameters during training. A natural first step toward understanding generalization due to dropout, therefore, is to instantiate the explicit form of the regularizer due to dropout. In linear regression, with dropout applied to the input layer (i.e., on the input features), the explicit regularizer was shown to be akin to a data-dependent ridge penalty Srivastava et al. (2014); Wager et al. (2013); Baldi & Sadowski (2013); Wang & Manning (2013). In factored models dropout yields more exotic forms of regularization. For instance, dropout induces regularizer that behaves similar to nuclear norm regularization in matrix factorization Cavazza et al. (2018), in single hidden-layer linear networks Mianjy et al. (2018), and in deep linear networks Mianjy & Arora (2019). However, none of the works above discuss how the induced regularizer provides capacity control, or equivalently, help us establish generalization bounds for dropout.

In this paper, we provide an answer to this question. We give *explicit forms* of the regularizers induced by dropout for the matrix sensing problem and two-layer neural networks with ReLU activations. Further, we establish *capacity control* due to dropout and give precise generalization bounds. Our key contributions are as follows.

1. In Section 2, we study dropout for matrix completion, wherein, the matrix factors are dropped randomly during training. We show that this algorithmic procedure induces a data-dependent regularizer that behaves similar to the weighted trace-norm which has been shown to yield strong generalization guarantees for matrix completion (Foygel et al., 2011).

*Johns Hopkins University, email: arora@cs.jhu.edu
†University of California, Berkeley, email: bartlett@cs.berkeley.edu
‡Johns Hopkins University, email: mianjy@jhu.edu
§TTI Chicago, email: nati@ttic.edu

2. In Section 3, we study dropout in two-layer ReLU networks. We show that the regularizer induced by dropout is a data-dependent measure that behaves as ℓ_2-path norm Neyshabur et al. (2015a), and establish data-dependent generalization bounds.

3. In Section 5, we present empirical evaluations that confirm our theoretical findings for matrix completion and deep regression on real world datasets including the MovieLens data, as well as the MNIST and Fashion MNIST datasets.

1.1 Related Work

Dropout was first introduced by Hinton et al. (2012) as an effective heuristic for algorithmic regularization, yielding lower test errors on the MNIST and TIMIT datasets. In a subsequent work, Srivastava et al. (2014) reported similar improvements over several tasks in computer vision (on CIFAR-10/100 and ImageNet datasets), speech recognition, text classification and genetics.

Thenceforth, dropout has been widely used in training state-of-the-art systems for several tasks including large-scale visual recognition Szegedy et al. (2015), large vocabulary continuous speech recognition Dahl et al. (2013), image question answering Yang et al. (2016), handwriting recognition Pham et al. (2014), sentiment prediction and question classification Kalchbrenner et al. (2014), dependency parsing Chen & Manning (2014), and brain tumor segmentation Havaei et al. (2017).

Following the empirical success of dropout, there have been several studies in recent years aimed at establishing theoretical underpinnings of why and how dropout helps with generalization. Early work of Baldi & Sadowski (2013) showed that for a single linear unit (and a single sigmoid unit, approximately), dropout amounts to weight decay regularization on the weights. A similar result was shown by McAllester (2013) in a PAC-Bayes setting. For generalized linear models, Wager et al. (2013) established that dropout performs an adaptive regularization which is equivalent to a data-dependent scaling of the weight decay penalty. In their follow-up work, Wager et al. (2014) show that for linear classification, under a generative assumption on the data, dropout improves the convergence rate of the generalization error. In this paper, we focus on predictors represented in a factored form and give generalization bounds for matrix learning problems and single hidden layer ReLU networks.

In a related line of work, Helmbold & Long (2015) study the structural properties of the dropout regularizer in the context of linear classification. They characterize the landscape of the dropout criterion in terms of unique minimizers and establish non-monotonic and non-convex nature of the regularizer. In a follow up work, Helmbold & Long (2017) extend their analysis to dropout in deep ReLU networks and surprisingly find that the nature of regularizer is different from that in linear classification. In particular, they show that unlike weight decay, dropout regularizer in deep networks can grow exponentially with depth and remains invariant to rescaling of inputs, outputs, and network weights. We confirm some of these findings in our theoretical analysis. However, counter to the claims of Helmbold & Long (2017), we argue that dropout does indeed prevent co-adaptation.

In a closely related approach as ours, the works of Zhai & Wang (2018), Gao & Zhou (2016), and Wan et al. (2013) bound the Rademacher complexity of deep neural networks trained using dropout. In particular, Gao & Zhou (2016) show that the Rademacher complexity of the target class decreases polynomially or exponentially, for shallow and deep networks, respectively, albeit they assume additional norm bounds on the weight vectors. Similarly, the works of Wan et al. (2013) and Zhai & Wang (2018) assume that certain norms of the weights are bounded, and show that the Rademacher complexity of the target class decreases with dropout rates. We argue in this paper that dropout alone does not directly control the norms of the weight vectors; therefore, each of the works above fail to capture the practice. We emphasize that none of the previous works provide a generalization guarantee, i.e., a bound on the gap between the population risk and the empirical risk, merely in terms of the value of the explicit regularizer due to dropout. We give a first such result for dropout in the context of matrix completion and for a single hidden layer ReLU network.

There are a bunch of other works that do not fall into any of the categories above, and, in fact, are somewhat unrelated to the focus in this paper. Nonetheless, we discuss them here for completeness. For instance, Gal & Ghahramani (2016) study dropout as Bayesian approximation , Bank & Giryes (2018)

draw insights from frame theory to connect the notion of equiangular tight frames with dropout training in auto-encoders. Also, some recent works have considered variants of dropout. For instance, Mou et al. (2018) consider a variant of dropout, which they call "truthful" dropout, that ensures that the output of the randomly perturbed network is unbiased. However, rather than bound generalization error, Mou et al. (2018) bound the gap between the population risk and the dropout objective, i.e., the empirical risk plus the explicit regularizer. Li et al. (2016) study a yet another variant based on multinomoal sampling (different nodes are dropped with different rates), and establish sub-optimality bounds for stochastic optimization of linear models (for convex Lipschitz loss functions).

Matrix Factorization with Dropout. Our study of dropout is motivated in part by recent works of Cavazza et al. (2018), Mianjy et al. (2018), and Mianjy & Arora (2019). This line of work was initiated by Cavazza et al. (2018), who studied dropout for low-rank matrix factorization without constraining the rank of the factors or adding an explicit regularizer to the objective. They show that dropout in the context of matrix factorization yields an explicit regularizer whose convex envelope is given by nuclear norm. This result is further strengthened by Mianjy et al. (2018) who show that induced regularizer is indeed nuclear norm.

While matrix factorization is not a learning problem per se (for instance, what is training versus test data), in follow-up works by Mianjy et al. (2018) and Mianjy & Arora (2019), the authors show that training deep linear networks with ℓ_2-loss using dropout reduces to the matrix factorization problem if the marginal distribution of the input feature vectors is assumed to be isotropic, i.e., $\mathbb{E}[\mathrm{x}\mathrm{x}^\top] = \mathrm{I}$. We note that this is a strong assumption. If we do not assume isotropy, we show that dropout induces a data-dependent regularizer which amounts to a simple scaling of the parameters and, therefore, does not control capacity in any meaningful way. We revisit this discussion in Section 4.

To summarize, while we are motivated by Cavazza et al. (2018), the problem setup, the nature of statements in this paper, and the tools we use are different from that in Cavazza et al. (2018). Our proofs are simple and quickly verified. We do build closely on the prior work of Mianjy et al. (2018).

However, different from Mianjy et al. (2018), we rigorously argue for dropout in matrix completion by 1) showing that the induced regularizer is equal to weighted trace-norm, which as far as we know, is a novel result, 2) giving strong generalization bounds, and 3) providing extensive experimental evidence that dropout provides state of the art performance on one of the largest datasets in recommendation systems research. Beyond that we rigorously extend our results to two layer ReLU networks, describe the explicit regularizer, bound the Rademacher complexity of the hypothesis class controlled by dropout, show precise generalization bounds, and support them with empirical results.

1.2 Notation and Preliminaries

We denote matrices, vectors, scalar variables and sets by Roman capital letters, Roman small letters, small letters, and script letters, respectively (e.g. X, x, x, and $\mathcal{X}$). For any integer d, we represent the set $\{1, \dots, d\}$ by $[d]$. For any vector $\mathrm{x} \in \mathbb{R}^d$, $\mathrm{diag}(\mathrm{x}) \in \mathbb{R}^{d\times d}$ represents the diagonal matrix with the i^{th} diagonal entry equal to x_i, and $\sqrt{\mathrm{x}}$ is the elementwise squared root of x. Let $\|\mathrm{x}\|$ represent the ℓ_2-norm of vector x, and $\|\mathrm{X}\|$, $\|\mathrm{X}\|_F$, and $\|\mathrm{X}\|_*$ represent the spectral norm, the Frobenius norm, and the nuclear norm of matrix X, respectively. Let $\mathrm{X}^\dagger$ denote the Moore-Penrose pseudo-inverse of X. Given a positive definite matrix C, we denote the Mahalonobis norm as $\|\mathrm{x}\|_\mathrm{C}^2 = \mathrm{x}^\top \mathrm{C}\mathrm{x}$. For a random variable x that takes values in $\mathcal{X}$, given n i.i.d. samples $\{\mathrm{x}_1, \cdots, \mathrm{x}_n\}$, the empirical average of a function $f : \mathcal{X} \to \mathbb{R}$ is denoted by $\widehat{\mathbb{E}}_i[f(\mathrm{x}_i)] := \frac{1}{n}\sum_{i\in[n]} f(\mathrm{x}_i)$. Furthermore, we denote the second moment of x as $\mathrm{C} := \mathbb{E}[\mathrm{x}\mathrm{x}^\top]$. The standard inner product is represented by $\langle \cdot, \cdot \rangle$, for vectors or matrices, where $\langle \mathrm{X}, \mathrm{X}' \rangle = \mathrm{Tr}(\mathrm{X}^\top \mathrm{X}')$.

We are primarily interested in understanding how dropout controls the capacity of the hypothesis class when using dropout for training. To that end, we consider Rademacher complexity, a sample dependent measure of complexity of a hypothesis class that can directly bound the generalization gap (Bartlett & Mendelson, 2002). Formally, let $\mathcal{S} = \{(\mathrm{x}_1, y_1), \dots, (\mathrm{x}_n, y_n)\}$ be a sample of size n. Then, the empirical Rademacher complexity of a function class $\mathcal{F}$ with respect to $\mathcal{S}$, and the expected Rademacher complexity

are defined, respectively, as

$$\mathfrak{R}_{\mathcal{S}}(\mathcal{F}) = \mathbb{E}_{\sigma} \sup_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} \sigma_i f(\mathrm{x}_i), \quad \mathfrak{R}_n(\mathcal{F}) = \mathbb{E}_{\mathrm{x}}[\mathfrak{R}_{\mathcal{S}}(\mathcal{F})],$$

where σ_i are i.i.d. Rademacher random variables.

2 Matrix Sensing

We begin with understanding dropout for matrix sensing, a problem which arguably is an important instance of a matrix learning problem with lots of applications, and is well understood from a theoretical perspective. The problem setup is the following. Let $\mathrm{M}_* \in \mathbb{R}^{d_2 \times d_0}$ be a matrix with rank $r_* := \mathrm{Rank}(\mathrm{M}_*)$. Let $\mathrm{A}^{(1)}, \ldots, \mathrm{A}^{(n)}$ be a set of measurement matrices of the same size as M_*. The goal of matrix sensing is to recover the matrix M_* from n observations of the form $y_i = \langle \mathrm{M}_*, \mathrm{A}^{(i)} \rangle$ such that $n \ll d_2 d_0$.

A natural approach is to represent the matrix in terms of factors and solve the following *empirical risk* minimization problem:

$$\min_{\mathrm{U},\mathrm{V}} \widehat{L}(\mathrm{U}, \mathrm{V}) := \widehat{\mathbb{E}}_i (y_i - \langle \mathrm{U}\mathrm{V}^\top, \mathrm{A}^{(i)} \rangle)^2 \tag{1}$$

where $\mathrm{U} = [\mathrm{u}_1, \ldots, \mathrm{u}_{d_1}] \in \mathbb{R}^{d_2 \times d_1}, \mathrm{V} = [\mathrm{v}_1, \ldots, \mathrm{v}_{d_1}] \in \mathbb{R}^{d_0 \times d_1}$. When the number of factors is unconstrained, i.e., when $d_1 \gg r_*$, there exist many "bad" empirical minimizers, i.e., those with a large *true risk* $L(\mathrm{U}, \mathrm{V}) := \mathbb{E}(y - \langle \mathrm{U}\mathrm{V}^\top, \mathrm{A} \rangle)^2$. Interestingly, Li et al. (2018) showed recently that under a restricted isometry property (RIP), despite the existence of such poor ERM solutions, gradient descent with proper initialization is *implicitly* biased towards finding solutions with minimum nuclear norm – this is an important result which was first conjectured and empirically verified by Gunasekar et al. (2017). We do not make an RIP assumption here. Further, we argue that for the most part, modern machine learning systems employ *explicit* regularization techniques. In fact, as we show in the experimental section, the *implicit* bias due to (stochastic) gradient descent does not prevent it from blatant overfitting in the matrix completion problem.

We propose solving the ERM problem (1) using dropout, where at training time, corresponding columns of U and V are dropped uniformly at random. As opposed to an *implicit* effect of gradient descent, dropout *explicitly* regularizes the empirical objective. It is then natural to ask, in the case of matrix sensing, if dropout also biases the ERM towards certain low norm solutions. To answer this question, we begin with the observation that dropout can be viewed as an instance of SGD on the following objective Cavazza et al. (2018); Mianjy et al. (2018):

$$\widehat{L}_{\mathrm{drop}}(\mathrm{U}, \mathrm{V}) = \widehat{\mathbb{E}}_j \mathbb{E}_{\mathrm{B}} (y_j - \langle \mathrm{U}\mathrm{B}\mathrm{V}^\top, \mathrm{A}^{(j)} \rangle)^2, \tag{2}$$

where $\mathrm{B} \in \mathbb{R}^{d_1 \times d_1}$ is a diagonal matrix whose diagonal elements are Bernoulli random variables distributed as $\mathrm{B}_{ii} \sim \frac{1}{1-p}\mathrm{Ber}(1-p)$. It is easy to show that for $p \in [0, 1)$:

$$\widehat{L}_{\mathrm{drop}}(\mathrm{U}, \mathrm{V}) = \widehat{L}(\mathrm{U}, \mathrm{V}) + \frac{p}{1-p} \widehat{R}(\mathrm{U}, \mathrm{V}), \tag{3}$$

where $\widehat{R}(\mathrm{U}, \mathrm{V}) := \sum_{i=1}^{d_1} \widehat{\mathbb{E}}_j (\mathrm{u}_i^\top \mathrm{A}^{(j)} \mathrm{v}_i)^2$ is a data-dependent term that captures the *explicit* regularizer due to dropout. A similar result was shown by Cavazza et al. (2018) and Mianjy et al. (2018), but we provide a proof for completeness (see Proposition 2 in the Appendix).

We show that the *explicit* regularizer concentrates around its expected value w.r.t. the data distribution (see Lemma 2 in the Appendix). Furthermore, given that we seek a minimum of $\widehat{L}_{\mathrm{drop}}$, it suffices to consider the factors with the minimal value of the regularizer among all that yield the same empirical loss. This motivates studying the the following distribution-dependent *induced* regularizer:

$$\Theta(\mathrm{M}) := \min_{\mathrm{U}\mathrm{V}^\top = \mathrm{M}} R(\mathrm{U}, \mathrm{V}), \quad \text{where } R(\mathrm{U}, \mathrm{V}) := \mathbb{E}_{\mathrm{A}}[\widehat{R}(\mathrm{U}, \mathrm{V})].$$

For a wide range of random measurements, $\Theta(\cdot)$ turns out to be a "suitable" regularizer. Here, we instantiate two important examples (see Proposition 3 in the Appendix).

Gaussian Measurements. For all $j \in [n]$, let $\mathrm{A}^{(j)}$ be standard Gaussian matrices. In this case, it is easy to see that $\mathrm{L}(\mathrm{U}, \mathrm{V}) = \|\mathrm{M}_* - \mathrm{U}\mathrm{V}^\top\|_F^2$ and we recover the matrix factorization problem. Furthermore, we know from Cavazza et al. (2018); Mianjy & Arora (2019) that dropout regularizer acts as trace-norm regularization, i.e., $\Theta(\mathrm{M}) = \frac{1}{d_1}\|\mathrm{M}\|_*^2$.

Matrix Completion. For all $j \in [n]$, let $\mathrm{A}^{(j)}$ be an indicator matrix whose (i,k)-th element is selected randomly with probability $p(i)q(k)$, where $p(i)$ and $q(k)$ denote the probability of choosing the i-th row and the k-th column, respectively. Then

$$\Theta(\mathrm{M}) = \frac{1}{d_1}\|\operatorname{diag}(\sqrt{p})\mathrm{U}\mathrm{V}^\top \operatorname{diag}(\sqrt{q})\|_*^2$$

is the *weighted trace-norm* studied by Srebro & Salakhutdinov (2010) and Foygel et al. (2011).

These observations are specifically important because they connect dropout, an algorithmic heuristic in deep learning, to strong complexity measures that are empirically effective as well as theoretically well understood. To illustrate, here we give a generalization bound for matrix completion using dropout in terms of the value of the *explicit* regularizer at the minimizer.

Theorem 1. Assume that $d_2 \geq d_0$ and $\|\mathrm{M}_*\| \leq 1$. Furthermore, assume that $\min_{i,k} p(i)q(k) \geq \frac{\log(d_2)}{n\sqrt{d_2 d_0}}$. Let (U, V) be a minimizer of the dropout ERM objective in equation (3). Let α be such that $R(\mathrm{U}, \mathrm{V}) \leq \alpha/d_1$. Then, for any $\delta \in (0,1)$, the following generalization bounds holds with probability at least $1-\delta$ over a sample of size n:

$$L(g(\mathrm{U}\mathrm{V}^\top)) \leq \widehat{L}(\mathrm{U}, \mathrm{V}) + 8\sqrt{\frac{2\alpha d_2 \log(d_2) + \frac{1}{4}\log(2/\delta)}{n}}$$

where $g(\mathrm{M})$ thresholds M at ± 1, i.e. $g(\mathrm{M})(i,j) = \max\{-1, \min\{1, \mathrm{M}(i,j)\}\}$, and $L(g(\mathrm{U}\mathrm{V}^\top)) := \mathbb{E}(y - \langle g(\mathrm{U}\mathrm{V}^\top), \mathrm{A}\rangle)^2$ is the *true risk* of $g(\mathrm{U}\mathrm{V}^\top)$.

The proof of Theorem 1 follows from standard generalization bounds for ℓ_2 loss (Mohri et al., 2018) based on the Rademacher complexity (Bartlett & Mendelson, 2002) of the class of functions with weighted trace-norm bounded by $\sqrt{\alpha}$, i.e. $\mathcal{M}_\alpha := \{\mathrm{M} : \ \|\operatorname{diag}(\sqrt{\mathrm{p}})\mathrm{M}\operatorname{diag}(\sqrt{\mathrm{q}})\|_*^2 \leq \alpha\}$. The non-degeneracy condition $\min_{i,j} p(i)q(j) \geq \frac{\log(d_2)}{n\sqrt{d_2 d_0}}$ is required to obtain a bound on the Rademacher complexity of $\mathcal{M}_\alpha$, as established by Foygel et al. (2011).

We note that for large enough sample size, $\widehat{R}(\mathrm{U}, \mathrm{V}) \approx R(\mathrm{U}, \mathrm{V}) \approx \Theta(\mathrm{U}\mathrm{V}^\top) = \frac{1}{d_1}\|\operatorname{diag}(\sqrt{p})\mathrm{U}\mathrm{V}^\top \operatorname{diag}(\sqrt{\mathrm{q}})\|_*^2$, where the second approximation is due the fact that the pair (U, V) is a minimizer. That is, compared to the weighted trace-norm, the value of the explicit regularizer at the minimizer roughly scales as $1/d_1$. Hence the assumption $\widehat{R}(\mathrm{U}, \mathrm{V}) \leq \alpha/d_1$ in the statement of the corollary.

In practice, for models that are trained with dropout, the training error $\widehat{L}(\mathrm{U}, \mathrm{V})$ is negligible (see Figure 1 for experiments on the MovieLens dataset). Moreover, given that the sample size is large enough, the third term can be made arbitrarily small. Having said that, the second term, which is $\tilde{O}(\sqrt{\alpha d_2/n})$, dominates the right hand side of generalization error bound in Theorem 9. In Appendix, we also give optimistic generalization bounds that decay as $\tilde{O}(ad_2/n)$.

Finally, the required sample size heavily depends on the value of the explicit regularizer (i.e., α/d_1) at a minimizer, and hence, on the dropout rate p. In particular, increasing the dropout rate increases the regularization parameter $\lambda := \frac{p}{1-p}$, thereby intensifying the penalty due to the explicit regularizer. Intuitively, a larger dropout rate p results in a smaller α, thereby a tighter generalization gap can be guaranteed. We show through experiments that that is indeed the case in practice.

3 Non-linear Networks

Next, we focus on neural networks with a single hidden layer. Let $\mathcal{X} \subseteq \mathbb{R}^{d_0}$ and $\mathcal{Y} \subseteq [-1,1]^{d_2}$ denote the input and output spaces, respectively. Let $\mathcal{D}$ denote the joint probability distribution on $\mathcal{X} \times \mathcal{Y}$. Given n

examples $\{(\mathrm{x}_i, \mathrm{y}_i)\}_{i=1}^n \sim \mathcal{D}^n$ drawn i.i.d. from the joint distribution and a loss function $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$, the goal of learning is to find a hypothesis $f_{\mathrm{w}} : \mathcal{X} \to \mathcal{Y}$, parameterized by w, that has a small *population risk* $L(f_{\mathrm{w}}) := \mathbb{E}_{\mathcal{D}}[\ell(f_{\mathrm{w}}(\mathrm{x}), \mathrm{y})]$.

We focus on the squared ℓ_2 loss, i.e., $\ell(\mathrm{y}, \mathrm{y}') = \|\mathrm{y} - \mathrm{y}'\|^2$, and study the generalization properties of the dropout algorithm for minimizing the *empirical risk* $\widehat{L}(f_{\mathrm{w}}) := \widehat{\mathbb{E}}_i[\|\mathrm{y}_i - f_{\mathrm{w}}(\mathrm{x}_i)\|^2]$. We consider the hypothesis class associated with feed-forward neural networks with 2 layers, i.e., functions of the form $f_{\mathrm{w}}(\mathrm{x}) = \mathrm{U}\sigma(\mathrm{V}^\top \mathrm{x})$, where $\mathrm{U} = [\mathrm{u}_1, \ldots, \mathrm{u}_{d_1}] \in \mathbb{R}^{d_2 \times d_1}, \mathrm{V} = [\mathrm{v}_1, \ldots, \mathrm{v}_{d_1}] \in \mathbb{R}^{d_0 \times d_1}$ are the weight matrices. The parameter w is the collection of weight matrices $\{\mathrm{U}, \mathrm{V}\}$ and $\sigma : \mathbb{R} \to \mathbb{R}$ is the ReLU activation function applied entrywise to an input vector.

As in Section 2, we view dropout as an instance of stochastic gradient descent on the following *dropout objective*:

$$\widehat{L}_{\mathrm{drop}}(\mathrm{w}) := \widehat{\mathbb{E}}_i \mathbb{E}_{\mathrm{B}} \|\mathrm{y}_i - \mathrm{UB}\sigma(\mathrm{V}^\top \mathrm{x}_i)\|^2, \tag{4}$$

where B is a diagonal random matrix with diagonal elements distributed identically and independently as $\mathrm{B}_{ii} \sim \frac{1}{1-p}\mathrm{Bern}(1-p)$, $i \in [d_1]$, for some *dropout rate* p. We seek to understand the *explicit regularizer* due to dropout:

$$\widehat{R}(\mathrm{w}) := \widehat{L}_{\mathrm{drop}}(\mathrm{w}) - \widehat{L}(\mathrm{w}). \tag{5}$$

We denote the output of the i-th hidden node on an input vector x by $a_i(\mathrm{x}) \in \mathbb{R}$; for example, $a_2(\mathrm{x}) = \sigma(\mathrm{v}_2^\top \mathrm{x})$. Similarly, the vector $\mathrm{a}(\mathrm{x}) \in \mathbb{R}^{d_1}$ denotes the activation of the hidden layer on input x. Using this notation, we can rewrite the objective in (4) as $\widehat{L}_{\mathrm{drop}}(\mathrm{w}) := \mathbb{E}_i \mathbb{E}_{\mathrm{B}} \|\mathrm{y}_i - \mathrm{UBa}(\mathrm{x}_i)\|^2$. It is then easy to show that the regularizer due to dropout in (5) is given as (see Proposition 4 in Appendix):

$$\widehat{R}(\mathrm{w}) = \frac{p}{1-p} \sum_{j=1}^{d_1} \|\mathrm{u}_j\|^2 \widehat{a}_j^2, \text{ where } \widehat{a}_j = \sqrt{\widehat{\mathbb{E}}_i a_j(\mathrm{x}_i)^2}.$$

The explicit regularizer $\widehat{R}(\mathrm{w})$ is a summation over hidden nodes, of the product of the squared norm of the outgoing weights with the empirical second moment of the output of the corresponding neuron. We should view it as a data-dependent variant of the ℓ_2 path-norm of the network, studied recently by Neyshabur et al. (2015b) and shown to yield capacity control in deep learning. Indeed, if we consider ReLU activations and input distributions that are symmetric and isotropic Mianjy et al. (2018), the expected regularizer is equal to the sum over all paths from input to output of the product of the squares of weights along the paths, i.e.,

$$R(\mathrm{w}) := \mathbb{E}[\widehat{R}(\mathrm{w})] = \frac{1}{2} \sum_{i_0, i_1, i_2 = 1}^{d_0, d_1, d_2} \mathrm{U}(i_2, i_1)^2 \mathrm{V}(i_0, i_1)^2,$$

which is precisely the squared ℓ_2 path-norm of the network. We refer the reader to Proposition 5 in the Appendix for a formal statement and proof.

3.1 Generalization Bounds

To understand the generalization properties of dropout, we focus on the following distribution-dependent class

$$\mathcal{F}_\alpha := \{f_{\mathrm{w}} : \mathrm{x} \mapsto \mathrm{u}^\top \sigma(\mathrm{V}^\top \mathrm{x}),\ \sum_{i=1}^{d_1} |u_i| a_i \le \alpha\}.$$

where $\mathrm{u} \in \mathbb{R}^{d_1}$ is the top layer weight vector, u_i denotes the i-th entry of u, and $a_i^2 := \mathbb{E}_{\mathrm{x}}[\widehat{a}_i^2] = \mathbb{E}_{\mathrm{x}}[a_i(\mathrm{x})^2]$ is the expected squared neural activation for the i-th hidden node. For simplicity, we focus on networks with one output neuron $d_2 = 1$; extension to multiple output neurons is rather straightforward.

We argue that networks that are trained with dropout belong to the class $\mathcal{F}_\alpha$, for a small value of α. In particular, by Cauchy-Schwartz inequality, it is easy to to see that $\sum_{i=1}^{d_1} |u_i| a_i \le \sqrt{d_1 R(\mathrm{w})}$. Thus, for a fixed width, dropout implicitly controls the function class $\mathcal{F}_\alpha$. More importantly, this inequality is loose

if a small subset of hidden nodes $\mathcal{J} \subset [d_1]$ *"co-adapt"* in a way that for all $j \in [d_1] \setminus \mathcal{J}$, the other hidden nodes are almost inactive, i.e. $u_j a_j \approx 0$. In other words, by minimizing the expected regularizer, dropout is biased towards networks where the gap between $R(\mathrm{w})$ and $(\sum_{i=1}^{d_1} |u_i|a_i)^2/d_1$ is small, which in turn happens if $|u_i|a_i \approx |u_j|a_j, \forall i,j \in [d_1]$. In this sense, dropout breaks *"co-adaptation"* between neurons by promoting solutions with nearly equal contribution from hidden neurons.

As we mentioned in the introduction, a bound on the dropout regularizer is not sufficient to guarantee a bound on a norm-based complexity measures that are common in the deep learning literature (see, e.g. Golowich et al. (2018) and the references therein), whereas a norm bound on the weight vector would imply a bound on the explicit regularizer due to dropout. Formally, we show the following.

Proposition 1. *For any $C > 0$, there exists a distribution on the unit Euclidean sphere, and a network $f_{\mathrm{w}} : \mathrm{x} \mapsto \sigma(\mathrm{w}^\top \mathrm{x})$, such that $R(\mathrm{w}) = \sqrt{\mathbb{E}\sigma(\mathrm{w}^\top \mathrm{x})^2} \leq 1$, while $\|\mathrm{w}\| > C$.*

In other words, even though we connect the dropout regularizer to path-norm, the data-dependent nature of the regularizer prevents us from leveraging that connection in data-independent manner (i.e., for all distributions). At the same time, making strong distributional assumptions (as in Proposition 5) would be impractical. Instead, we argue for the following milder condition on the input distribution which we show as sufficient to ensure generalization.

Assumption 1 (β-retentive)**.** The marginal input distribution is β-retentive for some $\beta \in (0, 1/2]$, if for any non-zero vector $\mathrm{v} \in \mathbb{R}^d$, it holds that $\mathbb{E}\sigma(\mathrm{v}^\top \mathrm{x})^2 \geq \beta \mathbb{E}(\mathrm{v}^\top \mathrm{x})^2$.

Intuitively, what the assumption implies is that the variance (aka, the information or signal in the data) in the pre-activation at any node in the network is not quashed considerably due to the non-linearity. In fact, no reasonable training algorithm should learn weights where β is small. However, we steer clear from algorithmic aspects of dropout training, and make the assumption above for every weight vector as we will need it when carrying out a union bound.

We now present the first main result of this section, which bounds the Rademacher complexity of $\mathcal{F}_\alpha$ in terms of α, the retentiveness coefficient β, and the Mahalanobis norm of the data with respect to the pseudo-inverse of the second moment, i.e. $\|\mathrm{X}\|_{\mathrm{C}^\dagger}^2 = \sum_{i=1}^n \mathrm{x}_i^\top \mathrm{C}^\dagger \mathrm{x}_i$.

Theorem 2. *For any sample $S = \{(\mathrm{x}_i, \mathrm{y}_i)\}_{i=1}^n$ of size n,*

$$\mathfrak{R}_{\mathcal{S}}(\mathcal{F}_\alpha) \leq \frac{2\alpha \|\mathrm{X}\|_{\mathrm{C}^\dagger}}{n\sqrt{\beta}}.$$

Furthermore, it holds for the expected Rademacher complexity that $\mathfrak{R}_n(\mathcal{F}_\alpha) \leq 2\alpha\sqrt{\frac{\mathrm{Rank(C)}}{\beta n}}$.

First, note that the bound depends on the quantity $\|\mathrm{X}\|_{\mathrm{C}^\dagger}$ which can be in the same order as $\|\mathrm{X}\|_F$ with both scaling as $\asymp \sqrt{nd_0}$; the latter is more common in the literature Neyshabur et al. (2018); Bartlett et al. (2017); Neyshabur et al. (2017); Golowich et al. (2018); Neyshabur et al. (2015b). This is unfortunately unavoidable, unless one makes stronger distributional assumptions.

Second, as we discussed earlier, the dropout regularizer directly controls the value of α, thereby controlling the Rademacher complexity in Theorem 2. This bound also gives us a bound on the Rademacher complexity of the networks trained using dropout. To see that, consider the following class of networks with bounded explicit regularizer, i.e., $\mathcal{H}_r := \{h_{\mathrm{w}} : \mathrm{x} \mapsto \mathrm{u}^\top \sigma(\mathrm{V}^\top \mathrm{x}),\ R(\mathrm{u}, \mathrm{V}) \leq r\}$. Then, Theorem 2 yields $\mathfrak{R}_{\mathcal{S}}(\mathcal{H}_r) \leq \frac{2\sqrt{d_1 r}\|\mathrm{X}\|_{\mathrm{C}^\dagger}}{n\sqrt{\beta}}$. In fact, we can show that this bound is tight up to $1/\sqrt{\beta}$ by a reduction to the linear case. Formally, we show the following.

Theorem 3 (Lowerbound on $\mathfrak{R}_{\mathcal{S}}$)**.** *There is a constant c such that for any scalar $r > 0$, $\mathfrak{R}_{\mathcal{S}}(\mathcal{H}_r) \geq \frac{c\sqrt{d_1 r}\|\mathrm{X}\|_{\mathrm{C}^\dagger}}{n}$.*

Moreover, it is easy to give a generalization bound based on Theorem 2 that depends only on the distribution dependent quantities α and β. Let $g_{\mathrm{w}}(\cdot) := \max\{-1, \min\{1, f_{\mathrm{w}}(\cdot)\}\}$ project the network output f_{w} onto the range $[-1, 1]$. We have the following generalization gurantees for g_{w}.

	plain SGD		dropout			
width	last iterate	best iterate	$p = 0.1$	$p = 0.2$	$p = 0.3$	$p = 0.4$
$d_1 = 30$	0.8041	0.7938	0.7805	0.785	0.7991	0.8186
$d_1 = 70$	0.8315	0.7897	0.7899	0.7771	0.7763	0.7833
$d_1 = 110$	0.8431	0.7873	0.7988	0.7813	0.7742	0.7743
$d_1 = 150$	0.8472	0.7858	0.8042	0.7852	0.7756	0.7722
$d_1 = 190$	0.8473	0.7844	0.8069	0.7879	0.7772	0.772

Table 1: MovieLens dataset: Test RMSE of plain SGD as well as the dropout algorithm with various dropout rates for various factorization sizes. The grey cells shows the best performance(s) in each row.

Corollary 1. *For any* $\mathrm{w} \in \mathcal{F}_\alpha$*, for any* $\delta \in (0,1)$*, the following generalization bound holds with probability at least* $1-\delta$ *over a sample* $\mathcal{S}$ *of size* n

$$L_{\mathcal{D}}(g_{\mathrm{w}}) \leq \widehat{L}_{\mathcal{S}}(g_{\mathrm{w}}) + \frac{16\alpha\|\mathrm{X}\|_{\mathrm{C}^\dagger}}{\sqrt{\beta}n} + 12\sqrt{\frac{\log(2/\delta)}{2n}}$$

β-independent Bounds. Geometrically, β-retentiveness requires that for any hyperplane passing through the origin, both halfspaces contribute significantly to the second moment of the data in the direction of the normal vector. It is not clear, however, if β can be estimated efficiently, given a dataset. Nonetheless, when $\mathcal{X} \subseteq \mathbb{R}_+^{d_0}$, which is the case for image datasets, a simple *symmetrization* technique, described below, allows us to give bounds that are β-independent; note that the bound still depend on the sample as $\|\mathrm{X}\|_{\mathrm{C}^\dagger}$. Here is the randomized symmetrization we propose. *Given a training sample* $\mathcal{S} = \{(\mathrm{x}_i, y_i),\ i \in [n]\}$*, consider the following randomized perturbation,* $\mathcal{S}' = \{(\zeta_i \mathrm{x}_i, y_i),\ i \in [n]\}$*, where* ζ_i*'s are i.i.d. Rademacher random variables.* We give a generalization bound (w.r.t. the original data distribution) for the hypothesis class with bounded regularizer w.r.t. the perturbed data distribution.

Corollary 2. *Given an i.i.d. sample* $\mathcal{S} = \{(\mathrm{x}_i, \mathrm{y}_i)\}_{i=1}^n$*, let*

$$\mathcal{F}'_\alpha := \{f_{\mathrm{w}} : \mathrm{x} \mapsto u^\top \sigma(V^\top \mathrm{x}),\ \sum_{i=1}^{d_1} |u_i| a'_i \leq \alpha\},$$

where $a_i'^2 := \mathbb{E}_{\mathrm{x},\zeta}[a_i(\zeta \mathrm{x})^2]$*. For any* $\mathrm{w} \in \mathcal{F}'_\alpha$*, for any* $\delta \in (0,1)$*, the following generalization bound holds with probability at least* $1-\delta$ *over a sample of size* n *and the randomization in symmetrization*

$$L_{\mathcal{D}}(g_{\mathrm{w}}) \leq 2\widehat{L}_{\mathcal{S}'}(g_{\mathrm{w}}) + \frac{46\alpha\|\mathrm{X}\|_{\mathrm{C}^\dagger}}{n} + 24\sqrt{\frac{\log(2/\delta)}{2n}}.$$

Note that the population risk of the clipped predictor $g_{\mathrm{w}}(\cdot) := \max\{-1, \min\{1, f_{\mathrm{w}}(\cdot)\}\}$ is bounded in terms of the empirical risk on $\mathcal{S}'$. Finally, we verify in Section 5 that symmetrization of the training set, on MNIST and FashionMNIST datasets, does not have an effect on the performance of the trained models.

4 Role of Parametrization

In this section, we argue that parametrization plays an important role in determining the nature of the inductive bias.

We begin by considering matrix sensing in non-factorized form, which entails minimizing $\widehat{L}(\mathrm{M}) := \widehat{\mathbb{E}}_i(y_i - \langle \mathrm{vec}\,(\mathrm{M}), \mathrm{vec}\big(\mathrm{A}^{(i)}\big)\rangle)^2$, where $\mathrm{vec}\,(\mathrm{M})$ denotes the column vectorization of M. Then, the expected explicit regularizer due to dropout equals $R(\mathrm{M}) = \frac{p}{1-p}\|\,\mathrm{vec}\,(\mathrm{M})\,\|^2_{\mathrm{diag(C)}}$, where $\mathrm{C} = \mathbb{E}[\mathrm{vec}\,(\mathrm{A})\,\mathrm{vec}\,(\mathrm{A})^\top]$ is the second moment of the measurement matrices. For instance, with Gaussian measurements, the second moment equals the identity matrix, in which case, the regularizer reduces to the Frobenius norm of the parameters

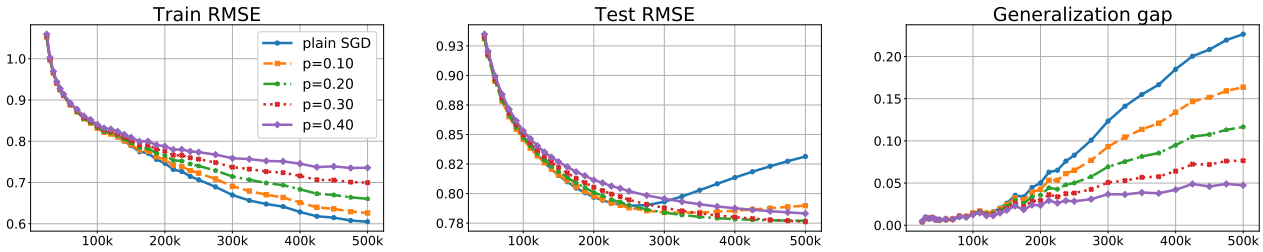

Figure 1: MovieLens dataset: the training error (**left**), the test error (**middle**), and the generalization gap (**right**) for plain SGD and dropout with $p \in \{0.1, 0.2, 0.3, 0.4\}$ as a function of the number of iterations. The factorization size is $d_1 = 70$.

$R(\mathrm{M}) = \frac{p}{1-p}\|\mathrm{M}\|_F^2$. While such a ridge penalty yields a useful inductive bias in linear regression, it is not *"rich"* enough to capture the kind of inductive bias that provides rank control in matrix sensing.

However, simply representing the hypotheses in a factored form alone is not sufficient in terms of imparting a rich inductive bias to the learning problem. Recall that in linear regression, dropout, when applied on the input features, yields ridge regularization. However, if we were to represent the linear predictor in terms of a deep linear network, then we argue that the effect of dropout is markedly different. Consider a deep linear network, $f_{\mathrm{w}} : \mathrm{x} \mapsto \mathrm{W}_k \cdots \mathrm{W}_1 \mathrm{x}$ with a single output neuron. In this case, Mianjy & Arora (2019) show that $\nu\|f\|_{\widehat{\mathrm{C}}}^2 = \min_{f_{\mathrm{w}}=f} \widehat{R}(\mathrm{w})$, where ν is a regularization parameter independent of the parameters w. Consequently, in deep linear networks with a single output neuron, dropout reduces to solving

$$\min_{\mathrm{u}\in\mathbb{R}^{d_0}} \widehat{\mathbb{E}}_i (y_i - \mathrm{u}^\top \mathrm{x}_i)^2 + \nu\|\mathrm{u}\|_{\widehat{\mathrm{C}}}^2.$$

All the minimizers of the above problem are solutions to the system of linear equations $(1 + \frac{\nu}{n})\mathrm{X}\mathrm{X}^\top \mathrm{u} = \mathrm{X}\mathrm{y}$, where $\mathrm{X} = [\mathrm{x}_1, \ldots, \mathrm{x}_n] \in \mathbb{R}^{d_0 \times n}, \mathrm{y} = [y_1; \ldots; y_n] \in \mathbb{R}^n$ are the design matrix and the response vector, respectively. In other words, the dropout regularizer manifests itself merely as a scaling of the parameters which does not seem to be useful in any meaningful way.

What we argue above may at first seem to contradict the results of Section 2 on matrix sensing, which is arguably an instance of regression with a two-layer linear network. Note though that casting matrix sensing in a factored form as a linear regression problem requires us to use a convolutional structure. This is easy to check since

$$\begin{aligned}\langle \mathrm{U}\mathrm{V}^\top, \mathrm{A}\rangle &= \langle \mathrm{vec}\left(\mathrm{U}^\top\right), \mathrm{vec}\left(\mathrm{V}^\top \mathrm{A}^\top\right)\rangle \\ &= \langle \mathrm{vec}\left(\mathrm{U}^\top\right), (\mathrm{I}_{d_2} \otimes \mathrm{V}^\top)\,\mathrm{vec}\left(\mathrm{A}^\top\right)\rangle,\end{aligned}$$

where $\otimes$ is the Kronecker product, and we used the fact that $\mathrm{vec}\,(\mathrm{AB}) = (\mathrm{I} \otimes \mathrm{A})\,\mathrm{vec}\,(\mathrm{B})$ for any pair of matrices A, B. The expression $(\mathrm{I} \otimes \mathrm{V}^\top)$ represents a fully connected convolutional layer with d_1 filters specified by columns of V. The convolutional structure in addition to dropout is what imparts the problem of matrix sensing the nuclear norm regularization. For nonlinear networks, however, a simple feed-forward structure suffices as we saw in Section 3.

5 Experimental Results

In this section, we report our empirical findings on real world datasets. All results are averaged over 50 independent runs with random initialization.

5.1 Matrix Completion

We evaluate dropout on the MovieLens dataset Harper & Konstan (2016), a publicly available collaborative filtering dataset that contains 10M ratings for 11K movies by 72K users of the online movie recommender ser-

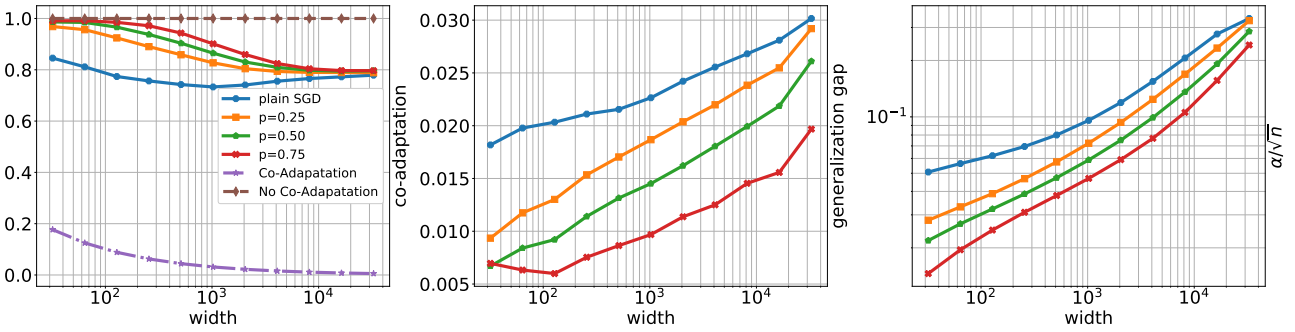

Figure 2: (**left**) *"co-adaptation"*; (**middle**) generalization gap; and (**right**) $\alpha/\sqrt{n}$ as a function of the width of networks trained with dropout on MNIST. In left figure, the dashed brown and dotted purple lines represent minimal and maximal co-adaptations, respectively.

vice MovieLens.

We initialize the factors using the standard He initialization scheme. We train the model for 100 epochs over the training data, where we use a fixed learning rate of $\texttt{lr} = 1$, and a batch size of 2000. We report the results for plain SGD ($p = 0.0$) as well as the dropout algorithm with $p \in \{0.1, 0.2, 0.3, 0.4\}$.

Figure 1 shows the progress in terms of the training and test error as well as the gap between them as a function of the number of iterations for $d_1 = 70$. It can be seen that plain SGD is the fastest in minimizing the empirical risk. The dropout rate clearly determines the trade-off between the approximation error and the estimation error: as the dropout rate p increases, the algorithm favors less complex solutions that suffer larger empirical error (left figure) but enjoy smaller generalization gap (right figure). The best trade-off here seems to be achieved by a moderate dropout rate of $p = 0.3$. We observe similar behaviour for different factorization sizes; please see the Appendix for additional plots with factorization sizes $d_1 \in \{30, 110, 150, 190\}$.

It is remarkable, how even in the "simple" problem of matrix completion, plain SGD lacks a proper inductive bias. As seen in the middle plot, without *explicit* regularization, in particular, without early stopping or dropout, SGD starts overfitting. We further illustrate this in Table 1, where we compare the test root-mean-squared-error (RMSE) of plain SGD with the dropout algorithm, for various factorization sizes. To show the superiority of dropout over SGD with early stopping, we give SGD the advantage of having access to the *test set* (and not a separate validation set), and report the best iterate in the third column. Even with this impractical privilege, dropout performs better (> 0.01 difference in test RMSE).

5.2 Neural Networks

We train 2-layer neural networks with and without dropout, on MNIST dataset of handwritten digits and Fashion MNIST dataset of Zalando's article images, each of which contains 60K training examples and 10K test examples, where each example is a 28×28 grayscale image associated with a label from 10 classes. We extract two classes $\{4, 7\}$ and label them as $\{-1, +1\}$ [1]. The learning rate in all experiments is set to $\texttt{lr} = 1e-3$. We train the models for 30 epochs over the training set. We run the experiments *both with and without symmetrization*. Here we only report the results with symmetrization, and on the MNIST dataset. For experiments without symmetrization, and experiments on FashionMNIST, please see the Appendix. We remark that *under the above experimental setting, the trained networks achieve* 100% *training accuracy.*

For any node $i \in [d_1]$, we define its *flow* as $\psi_i := |u_i|a_i$ (respectively $\psi_i := |u_i|a'_i$ for symmetrized data), which measures the overall contribution of a node to the output of the network. Co-adaptation occurs when a small subset of nodes dominate the overall function of the network. We argue that $\phi(\mathrm{w}) = \frac{\|\psi\|_1}{\sqrt{d_1}\|\psi\|_2}$ is a suitable measure of co-adaptation (or lack thereof) in a network parameterized by w. In case of high co-adaptation, only a few nodes have a high flow, which implies $\phi(\mathrm{w}) \approx \frac{1}{\sqrt{d_1}}$. At the other end of the

[1] We observe similar results across other choices of target classes.

spectrum, all nodes are equally active, in which case $\phi(\mathrm{w}) \approx 1$. Figure 2 (left) illustrates this measure as a function of the network width for several dropout rates $p \in \{0, 0.25, 0.5, 0.75\}$. In particular, we observe that a higher dropout rate corresponds to less co-adapation. More interestingly, even plain SGD is *implicitly* biased towards networks with less co-adapation. Moreover, for a fixed dropout rate, the regularization effect due to dropout decreases as we increase the width. Thus, it is natural to expect more co-adaptation as the network becomes wider, which is what we observe in the plots.

The generalization gap is plotted in Figure 2 (middle). As expected, increasing dropout rate decreases the generalization gap, uniformly for all widths. In our experiments, the generalization gap increases with the width of the network. The figure on the right shows the quantity $\alpha/\sqrt{n}$ that shows up in the Rademacher complexity bounds in Section 3. We note that, the bound on the Rademacher complexity is predictive of the generalization gap, in the sense that a smaller bound corresponds to a curve with smaller generalization gap.

6 Conclusion

Motivated by the success of dropout in deep learning, we study a dropout algorithm for matrix sensing and show that it enjoys strong generalization guarantees as well as competitive test performance on the MovieLens dataset. We then focus on deep regression under the squared loss and show that the regularizer due to dropout serves as a strong complexity measure for the underlying class of neural networks, using which we give a generalization error bound in terms of the value of the regularizer.

Acknowledgements

This research was supported, in part, by NSF BIGDATA award IIS-1546482 and NSF CAREER award IIS-1943251. The seeds of this work were sown during the summer 2019 workshop on the Foundations of Deep Learning at the Simons Institute for the Theory of Computing. Raman Arora acknowledges the support provided by the Institute for Advanced Study, Princeton, New Jersey as part of the special year on Optimization, Statistics, and Theoretical Machine Learning.

References

Baldi, P. and Sadowski, P. J. Understanding dropout. In *Advances in Neural Information Processing Systems (NIPS)*, pp. 2814–2822, 2013.

Bank, D. and Giryes, R. On the relationship between dropout and equiangular tight frames. *arXiv preprint arXiv:1810.06049*, 2018.

Bartlett, P. L. and Mendelson, S. Rademacher and gaussian complexities: Risk bounds and structural results. *Journal of Machine Learning Research*, 3(Nov):463–482, 2002.

Bartlett, P. L., Foster, D. J., and Telgarsky, M. J. Spectrally-normalized margin bounds for neural networks. In *Advances in Neural Information Processing Systems*, pp. 6240–6249, 2017.

Cavazza, J., Haeffele, B. D., Lane, C., Morerio, P., Murino, V., and Vidal, R. Dropout as a low-rank regularizer for matrix factorization. *Int. Conf. on Artificial Intelligence and Statistics (AISTATS)*, 2018.

Chen, D. and Manning, C. A fast and accurate dependency parser using neural networks. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 740–750, 2014.

Dahl, G. E., Sainath, T. N., and Hinton, G. E. Improving deep neural networks for lvcsr using rectified linear units and dropout. In *2013 IEEE international conference on acoustics, speech and signal processing*, pp. 8609–8613. IEEE, 2013.

Foygel, R., Shamir, O., Srebro, N., and Salakhutdinov, R. R. Learning with the weighted trace-norm under arbitrary sampling distributions. In *Advances in Neural Information Processing Systems*, pp. 2133–2141, 2011.

Gal, Y. and Ghahramani, Z. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *Int. Conf. Machine Learning (ICML)*, 2016.

Gao, W. and Zhou, Z.-H. Dropout rademacher complexity of deep neural networks. *Science China Information Sciences*, 59(7):072104, 2016.

Golowich, N., Rakhlin, A., and Shamir, O. Size-independent sample complexity of neural networks. In *Conference On Learning Theory*, pp. 297–299, 2018.

Gunasekar, S., Woodworth, B. E., Bhojanapalli, S., Neyshabur, B., and Srebro, N. Implicit regularization in matrix factorization. In *Advances in Neural Information Processing Systems*, pp. 6151–6159, 2017.

Harper, F. M. and Konstan, J. A. The movielens datasets: History and context. *Acm transactions on interactive intelligent systems (tiis)*, 5(4):19, 2016.

Havaei, M., Davy, A., Warde-Farley, D., Biard, A., Courville, A., Bengio, Y., Pal, C., Jodoin, P.-M., and Larochelle, H. Brain tumor segmentation with deep neural networks. *Medical image analysis*, 35:18–31, 2017.

Helmbold, D. P. and Long, P. M. On the inductive bias of dropout. *Journal of Machine Learning Research (JMLR)*, 16:3403–3454, 2015.

Helmbold, D. P. and Long, P. M. Surprising properties of dropout in deep networks. *The Journal of Machine Learning Research*, 18(1):7284–7311, 2017.

Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. R. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*, 2012.

Kalchbrenner, N., Grefenstette, E., and Blunsom, P. A convolutional neural network for modelling sentences. *arXiv preprint arXiv:1404.2188*, 2014.

Krizhevsky, A., Hinton, G., et al. Learning multiple layers of features from tiny images. Technical report, Citeseer, 2009.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pp. 1097–1105, 2012.

Li, Y., Ma, T., and Zhang, H. Algorithmic regularization in over-parameterized matrix sensing and neural networks with quadratic activations. In *Conference On Learning Theory*, pp. 2–47, 2018.

Li, Z., Gong, B., and Yang, T. Improved dropout for shallow and deep learning. In *Advances in neural information processing systems*, pp. 2523–2531, 2016.

McAllester, D. A pac-bayesian tutorial with a dropout bound. *arXiv preprint arXiv:1307.2118*, 2013.

Mianjy, P. and Arora, R. On dropout and nuclear norm regularization. In *International Conference on Machine Learning*, 2019.

Mianjy, P., Arora, R., and Vidal, R. On the implicit bias of dropout. In *International Conference on Machine Learning*, pp. 3537–3545, 2018.

Mohri, M., Rostamizadeh, A., and Talwalkar, A. *Foundations of machine learning*. MIT press, 2018.

Mou, W., Zhou, Y., Gao, J., and Wang, L. Dropout training, data-dependent regularization, and generalization bounds. In *International Conference on Machine Learning*, pp. 3642–3650, 2018.

Neyshabur, B., Salakhutdinov, R. R., and Srebro, N. Path-sgd: Path-normalized optimization in deep neural networks. In *Advances in Neural Information Processing Systems*, pp. 2422–2430, 2015a.

Neyshabur, B., Tomioka, R., and Srebro, N. Norm-based capacity control in neural networks. In *Conference on Learning Theory*, pp. 1376–1401, 2015b.

Neyshabur, B., Bhojanapalli, S., and Srebro, N. A pac-bayesian approach to spectrally-normalized margin bounds for neural networks. *arXiv preprint arXiv:1707.09564*, 2017.

Neyshabur, B., Li, Z., Bhojanapalli, S., LeCun, Y., and Srebro, N. Towards understanding the role of over-parametrization in generalization of neural networks. *arXiv preprint arXiv:1805.12076*, 2018.

Pham, V., Bluche, T., Kermorvant, C., and Louradour, J. Dropout improves recurrent neural networks for handwriting recognition. In *2014 14th international conference on frontiers in handwriting recognition*, pp. 285–290. IEEE, 2014.

Srebro, N. and Salakhutdinov, R. R. Collaborative filtering in a non-uniform world: Learning with the weighted trace norm. In *Advances in Neural Information Processing Systems*, pp. 2056–2064, 2010.

Srebro, N., Sridharan, K., and Tewari, A. Optimistic rates for learning with a smooth loss. *arXiv preprint arXiv:1009.3896*, 2010.

Srivastava, N., Hinton, G. E., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research (JMLR)*, 15(1), 2014.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1–9, 2015.

Toshev, A. and Szegedy, C. Deeppose: Human pose estimation via deep neural networks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014.

Vershynin, R. *High-dimensional probability: An introduction with applications in data science*, volume 47. Cambridge University Press, 2018.

Wager, S., Wang, S., and Liang, P. S. Dropout training as adaptive regularization. In *Advances in Neural Information Processing Systems (NIPS)*, 2013.

Wager, S., Fithian, W., Wang, S., and Liang, P. S. Altitude training: Strong bounds for single-layer dropout. In *Adv. Neural Information Processing Systems*, 2014.

Wan, L., Zeiler, M., Zhang, S., Le Cun, Y., and Fergus, R. Regularization of neural networks using dropconnect. In *International conference on machine learning*, pp. 1058–1066, 2013.

Wang, S. and Manning, C. Fast dropout training. In *international conference on machine learning*, pp. 118–126, 2013.

Yang, Z., He, X., Gao, J., Deng, L., and Smola, A. Stacked attention networks for image question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 21–29, 2016.

Zhai, K. and Wang, H. Adaptive dropout with rademacher complexity regularization. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=S1uxsye0Z.

Supplementary Materials for "Dropout: Explicit Forms and Capacity Control"

A Auxiliary Results

Lemma 1 (Khintchine-Kahane inequality). *Let $\{\epsilon_i\}_{i=1}^n$ be i.i.d. Rademacher random variables, and $\{\mathrm{x}\}_{i=1}^n \subset \mathbb{R}^d$. Then there exist a universal constants $c > 0$ such that*

$$\mathbb{E}\|\sum_{i=1}^n \epsilon_i \mathrm{x}_i\| \geq c\sqrt{\sum_{i=1}^n \|\mathrm{x}_i\|^2}.$$

Theorem 4 (Hoeffding's inequality: Theorem 2.6.2 Vershynin (2018)). *Let $X_1, \ldots, X_N$ be independent, mean zero, sub-Gaussian random variables. Then, for every $t \geq 0$, we have*

$$\mathbb{P}\left(\left|\widehat{\mathbb{E}}_i X_i\right| \geq t\right) \leq 2e^{-\frac{ct^2N^2}{\sum_{i=1}^N \|X_i\|_{\psi_2}^2}}$$

Theorem 5 (Theorem 3.1 of Mohri et al. (2018)). *Let $\mathcal{G}$ be a family of functions mapping from $\mathcal{Z}$ to $[0,1]$. Then, for any $\delta > 0$, with probability at least $1-\delta$ over a sample $\mathcal{S} = \{z_1, \ldots, z_n\}$, the following holds for all $g \in \mathcal{G}$*

$$E[g(z)] \leq \frac{1}{n}\sum_{i=1}^n g(z_i) + 2\mathfrak{R}_n(\mathcal{G}) + \sqrt{\frac{\log(1/\delta)}{2n}}$$
$$E[g(z)] \leq \frac{1}{n}\sum_{i=1}^n g(z_i) + 2\mathfrak{R}_{\mathcal{S}}(\mathcal{G}) + 3\sqrt{\frac{\log(1/\delta)}{2n}}$$

Theorem 6 (Theorem 10.3 of Mohri et al. (2018)). *Assume that $\|h - f\|_\infty \leq M$ for all $h \in \mathcal{H}$. Then, for any $\delta > 0$, with probability at least $1-\delta$ over a sample $\mathcal{S} = \{(\mathrm{x}_i, \mathrm{y}_i),\ i \in [n]\}$ of size n, the following inequalities holds uniformly for all $h \in \mathcal{H}$.*

$$\mathbb{E}[|h(\mathrm{x}) - f(\mathrm{x})|^2] \leq \widehat{\mathbb{E}}_i|h(\mathrm{x}_i) - f(\mathrm{x}_i)|^2 + 4M\mathfrak{R}_n(\mathcal{H}) + M^2\sqrt{\frac{\log(2/\delta)}{2n}}$$
$$\mathbb{E}[|h(\mathrm{x}) - f(\mathrm{x})|^2] \leq \widehat{\mathbb{E}}_i|h(\mathrm{x}_i) - f(\mathrm{x}_i)|^2 + 4M\mathfrak{R}_{\mathcal{S}}(\mathcal{H}) + 3M^2\sqrt{\frac{\log(2/\delta)}{2n}}$$

Theorem 7 (Based on Theorem 1 in Srebro et al. (2010)). *Let $\mathcal{X}$ and $\mathcal{Y} = [-1,1]$ denote the input space and the label space, respectively. Let $\mathcal{H} \subseteq \{f : \mathcal{X} \to \mathcal{Y}\}$ be the target function class. For any $f \in \mathcal{H}$, and any $(\mathrm{x}, y) \in \mathcal{X} \times \mathcal{Y}$, let $\ell(f, \mathrm{x}, y) := (f(\mathrm{x}) - y)^2$ be the squared loss. Let $L(f) = \mathbb{E}_{\mathcal{D}}[\ell(f, \mathrm{x}, y)]$ be the population risk with respect to the joint distribution $\mathcal{D}$ on $\mathcal{X} \times \mathcal{Y}$. For any $\delta > 0$, with probability at least $1-\delta$ over a sample of size n, we have for any $f \in \mathcal{H}$:*

$$L(f) \leq L_* + K\left(\sqrt{L_*}\left(\sqrt{2}\log(n)^{1.5}\mathfrak{R}_n(\mathcal{H}) + \sqrt{\frac{4\log\frac{1}{\delta}}{n}}\right) + 2\log(n)^3\mathfrak{R}_n^2(\mathcal{H}) + \frac{4\log\frac{1}{\delta}}{n}\right)$$

where $L_* := \min_{f\in\mathcal{H}} L(f)$, and K is a numeric constant derived from Srebro et al. (2010).

Theorem 8 (Theorem 3.3 in Mianjy et al. (2018)). *For any pair of matrices $U \in \mathbb{R}^{d_2 \times d_1}$, $V \in \mathbb{R}^{d_0 \times d_1}$, there exist a rotation matrix $Q \in SO(d_1)$ such that rotated matrices $\tilde{U} := UQ$, $\tilde{V} := VQ$ satisfy $\|\tilde{u}_i\|\|\tilde{v}_i\| = \frac{1}{d_1}\|UV^\top\|_*$, for all $i \in [d_1]$.*

Theorem 9 (Theorem 1 in Foygel et al. (2011)). *Assume that $p(i)q(j) \geq \frac{\log(d_2)}{n\sqrt{d_2 d_0}}$ for all $i \in [d_2], j \in [d_0]$. For any $\alpha > 0$, let $\mathcal{M}_\alpha := \{M \in \mathbb{R}^{d_2 \times d_1} : \|\operatorname{diag}(\sqrt{p})M\operatorname{diag}(\sqrt{q})\|_*^2 \leq \alpha\}$ be the class of linear transformations with weighted trace-norm bounded with $\sqrt{\alpha}$. Then the expected Rademacher complexity of $\mathcal{M}_\alpha$ is bounded as follows:*

$$\mathfrak{R}_n(\mathcal{M}_\alpha) \leq O\left(\sqrt{\frac{\alpha d_2 \log(d_2)}{n}}\right)$$

B Matrix Sensing

Proposition 2 (Dropout regularizer in matrix sensing). *The following holds for any $p \in [0, 1)$:*

$$\widehat{L}_{drop}(U, V) = \widehat{L}(U, V) + \lambda \widehat{R}(U, V), \tag{6}$$

where $\widehat{R}(U, V) = \sum_{i=1}^{d_1} \widehat{\mathbb{E}}_j (u_i^\top A^{(j)} v_i)^2$ and $\lambda = \frac{p}{1-p}$ is the regularization parameter.

Proof of Proposition 2. Similar statements and proofs can be found in several previous works Srivastava et al. (2014); Wang & Manning (2013); Cavazza et al. (2018); Mianjy et al. (2018). For completeness, we include a proof here. The following equality follows from the definition of variance:

$$\mathbb{E}_{\mathrm{b}}[(y_i - \langle \mathrm{UBV}^\top, \mathrm{A}^{(i)} \rangle)^2] = \Big(\mathbb{E}_{\mathrm{b}}[y_i - \langle \mathrm{UBV}^\top, \mathrm{A}^{(i)} \rangle]\Big)^2 + \mathrm{Var}(y_i - \langle \mathrm{UBV}^\top, \mathrm{A}^{(i)} \rangle) \tag{7}$$

Recall that for a Bernoulli random variable B_{ii}, we have $\mathbb{E}[\mathrm{B}_{ii}] = 1$ and $\mathrm{Var}(\mathrm{B}_{ii}) = \frac{p}{1-p}$. Thus, the first term on right hand side is equal to $(y_i - \langle \mathrm{UV}^\top, \mathrm{A}^{(i)} \rangle)^2$. For the second term we have

$$\mathrm{Var}(y_i - \langle \mathrm{UBV}^\top, \mathrm{A}^{(i)} \rangle) = \mathrm{Var}(\sum_{j=1}^{d_1} \mathrm{B}_{jj} \mathrm{u}_j^\top \mathrm{A}^{(i)} \mathrm{v}_j) = \sum_{j=1}^{d_1} (\mathrm{u}_j^\top \mathrm{A}^{(i)} \mathrm{v}_j)^2 \, \mathrm{Var}(\mathrm{B}_{jj}) = \frac{p}{1-p} \sum_{j=1}^{d_1} (\mathrm{u}_j^\top \mathrm{A}^{(i)} \mathrm{v}_j)^2$$

Plugging the above into Equation (7) and averaging over samples we get

$$\begin{aligned}
\widehat{L}_{\mathrm{drop}}(\mathrm{U}, \mathrm{V}) &= \widehat{\mathbb{E}}_i \mathbb{E}_{\mathrm{b}}[(y_i - \langle \mathrm{UBV}^\top, \mathrm{A}^{(i)} \rangle)^2] \\
&= \widehat{\mathbb{E}}_i (y_i - \langle \mathrm{UV}^\top, \mathrm{A}^{(i)} \rangle)^2 + \widehat{\mathbb{E}}_i \frac{p}{1-p} \sum_{j=1}^{d_1} (\mathrm{u}_j^\top \mathrm{A}^{(i)} \mathrm{v}_j)^2 \\
&= \widehat{L}(\mathrm{U}, \mathrm{V}) + \frac{p}{1-p} \widehat{R}(\mathrm{U}, \mathrm{V}).
\end{aligned}$$

which completes the proof. □

Lemma 2 (Concentration in matrix completion). *For $\ell \in [n]$, let $A^{(\ell)}$ be an indicator matrix whose (i, j)-th element is selected according to some distribution. Assume U, V is such that $\|U^\top\|_{2,\infty}\|V\|_{\infty,\infty} \leq \gamma$. Then, with probability at least $1 - \delta$ over a sample of size n, we have that*

$$|R(U, V) - \widehat{R}(U, V)| \leq \frac{C\gamma^2\sqrt{\log(2/\delta)}}{\sqrt{n}}.$$

Proof of Lemma 2. Define $X_\ell := \sum_{w=1}^{d_1} (\mathrm{u}_w^\top \mathrm{A}^{(\ell)} \mathrm{v}_w)^2$ and observe that

$$\begin{aligned} X_\ell &= \sum_{w=1}^{d_1} \left(\sum_{i,j} \mathrm{U}_{iw} \mathrm{V}_{jw} \mathrm{A}_{ij}^{(\ell)} \right)^2 = \sum_{w=1}^{d_1} \sum_{i,i',j,j'} \mathrm{U}_{iw} \mathrm{U}_{i'w} \mathrm{V}_{jw} \mathrm{V}_{j'w} \mathrm{A}_{ij}^{(\ell)} \mathrm{A}_{i'j'}^{(\ell)} \\ &= \sum_{w=1}^{d_1} \sum_{i,j} \mathrm{U}_{iw}^2 \mathrm{V}_{jw}^2 \mathrm{A}_{ij}^{(\ell)} \leq \max_{i,j} \sum_{w=1}^{d_1} \mathrm{U}_{iw}^2 \mathrm{V}_{jw}^2 \\ &\leq \max_{i,j} \|\mathrm{U}(i,:)\|^2 \|\mathrm{V}(j,:)\|_\infty^2 = \|\mathrm{U}^\top\|_{2,\infty}^2 \|\mathrm{V}\|_{\infty,\infty}^2 \leq \gamma^2 \end{aligned}$$

where the third equality follows because for an indicator matrix $\mathrm{A}^{(\ell)}$, it holds that $\mathrm{A}_{ij}^{(\ell)} \mathrm{A}_{i'j'}^{(\ell)} = 0$ if $(i,j) \neq (i',j')$. Thus, $X_{w,\ell}$ is a sub-Gaussian (more strongly, bounded) random variable with mean $\mathbb{E}[X_\ell] = R(\mathrm{U},\mathrm{V})$ and sub-Gaussian norm $\|X_\ell\|_{\psi_2} \leq \gamma^2/\ln(2)$. Furthermore, $\|X_\ell - R(\mathrm{U},\mathrm{V})\|_{\psi_2} \leq C' \|X_\ell\|_{\psi_2} \leq C\gamma^2$, for some absolute constants C', C (Lemma 2.6.8 of Vershynin (2018)). Using Theorem 4, for $t = C d_1 \sqrt{\frac{\log 2/\delta}{n}}$ we get that:

$$\mathbb{P}\left(\left| \widehat{R}(\mathrm{U},\mathrm{V}) - R(\mathrm{U},\mathrm{V}) \right| \geq t \right) = \mathbb{P}\left(\left| \frac{1}{n} \sum_{\ell=1}^{n} X_\ell - R(\mathrm{U},\mathrm{V}) \right| \geq C\gamma^2 \sqrt{\frac{\log 2/\delta}{n}} \right) \leq \delta$$

which completes the proof. □

Proposition 3. *[Induced regularizer] For $j \in [n]$, let $A^{(j)}$ be an indicator matrix whose (i,k)-th element is selected randomly with probability $p(i)q(k)$, where $p(i)$ and $q(k)$ denote the probability of choosing the i-th row and the k-th column. Then $\Theta(M) = \frac{1}{d_1} \|\operatorname{diag}(\sqrt{p}) U V^\top \operatorname{diag}(\sqrt{q})\|_*^2$.*

Proof of Proposition 3. For any pair of factors (U,V) it holds that

$$\begin{aligned} R(\mathrm{U},\mathrm{V}) &= \sum_{i=1}^{d_1} \mathbb{E}(\mathrm{u}_i^\top \mathrm{A} \mathrm{v}_i)^2 = \sum_{i=1}^{d_1} \sum_{j=1}^{d_2} \sum_{k=1}^{d_0} p(j)q(k) (\mathrm{u}_i^\top \mathrm{e}_j \mathrm{e}_k^\top \mathrm{v}_i)^2 \\ &= \sum_{i=1}^{d_1} \sum_{j=1}^{d_2} \sum_{k=1}^{d_0} p(j)q(k) \mathrm{U}(j,i)^2 \mathrm{V}(k,i)^2 = \sum_{i=1}^{d_1} \|\operatorname{diag}(\sqrt{p})\mathrm{u}_i\|^2 \|\operatorname{diag}(\sqrt{q})\mathrm{v}_i\|^2 \end{aligned}$$

We can now lower bound the right hand side above as follows:

$$\begin{aligned} R(\mathrm{U},\mathrm{V}) &\geq \frac{1}{d_1} \left(\sum_{i=1}^{d_1} \|\operatorname{diag}(\sqrt{p})\mathrm{u}_i\| \|\operatorname{diag}(\sqrt{q})\mathrm{v}_i\| \right)^2 \\ &= \frac{1}{d_1} \left(\sum_{i=1}^{d_1} \|\operatorname{diag}(\sqrt{p})\mathrm{u}_i \mathrm{v}_i^\top \operatorname{diag}(\sqrt{q})\|_* \right)^2 \\ &\geq \frac{1}{d_1} \left(\|\operatorname{diag}(\sqrt{p}) \sum_{i=1}^{d_1} \mathrm{u}_i \mathrm{v}_i^\top \operatorname{diag}(\sqrt{q})\|_* \right)^2 = \frac{1}{d_1} \|\operatorname{diag}(\sqrt{p})\mathrm{U}\mathrm{V}^\top \operatorname{diag}(\sqrt{q})\|_*^2 \end{aligned}$$

where the first inequality is due to Cauchy-Schwartz and the second inequality follows from the triangle inequality. The equality right after the first inequality follows from the fact that for any two vectors a, b, $\|\mathrm{a}\mathrm{b}^\top\|_* = \|\mathrm{a}\mathrm{b}^\top\| = \|\mathrm{a}\|\|\mathrm{b}\|$. Since the inequalities hold for any U, V, it implies that

$$\Theta(\mathrm{U}\mathrm{V}^\top) \geq \frac{1}{d_1} \|\operatorname{diag}(\sqrt{p})\mathrm{U}\mathrm{V}^\top \operatorname{diag}(\sqrt{q})\|_*^2.$$

Applying Theorem 8 on $(\operatorname{diag}(\sqrt{p})\mathrm{U}, \operatorname{diag}(\sqrt{p})\mathrm{V})$, there exist a rotation matrix Q such that

$$\|\operatorname{diag}(\sqrt{p})\mathrm{U}\mathrm{q}_i\|\|\operatorname{diag}(\sqrt{q})\mathrm{V}\mathrm{q}_i\| = \frac{1}{d_1}\|\operatorname{diag}(\sqrt{p})\mathrm{U}\mathrm{V}^\top \operatorname{diag}(\sqrt{q})\|_*$$

We evaluate the expected dropout regularizer at UQ, VQ:

$$\begin{aligned} R(\mathrm{UQ}, \mathrm{VQ}) &= \sum_{i=1}^{d_1} \|\operatorname{diag}(\sqrt{p})\mathrm{U}\mathrm{q}_i\|^2 \|\operatorname{diag}(\sqrt{q})\mathrm{V}\mathrm{q}_i\|^2 \\ &= \sum_{i=1}^{d_1} \frac{1}{d_1^2}\|\operatorname{diag}(\sqrt{p})\mathrm{U}\mathrm{V}^\top \operatorname{diag}(\sqrt{q})\|_*^2 = \frac{1}{d_1}\|\operatorname{diag}(\sqrt{p})\mathrm{U}\mathrm{V}^\top \operatorname{diag}(\sqrt{q})\|_*^2 \le \Theta(\mathrm{U}\mathrm{V}^\top) \end{aligned}$$

which completes the proof of the first part. □

Proof of Theorem 1. We use Theorem 6 to bound the population risk in terms of the Rademacher complexity of the target class. Define the class of predictors with weighted trace-norm bounded by $\sqrt{\alpha}$, i.e.

$$\mathcal{M}_\alpha = \{\mathrm{M} : \ \|\operatorname{diag}(\sqrt{\mathrm{p}})\mathrm{M}\operatorname{diag}(\sqrt{\mathrm{q}})\|_*^2 \le \alpha\}.$$

In particular dropout empirical risk minimizers U, V belong to this class:

$$\|\operatorname{diag}(\sqrt{p})\mathrm{U}\mathrm{V}^\top \operatorname{diag}(\sqrt{q})\|_*^2 = d_1\Theta(\mathrm{U}\mathrm{V}^\top) \le d_1 R(\mathrm{U}, \mathrm{V}) \le \alpha$$

where the first inequality holds by definition of the induced regularizer, and the second inequality follows from the assumption of the theorem. Since g is a contraction, by Talagrand's lemma and Theorem 9, we have that $\mathfrak{R}_n(g \circ \mathcal{M}_\alpha) \le \mathfrak{R}_n(\mathcal{M}_\alpha) \le \sqrt{\frac{\alpha d_2 \log(d_2)}{n}}$. To obtain the maximum deviation parameter M in Theorem 6, we note that the assumption $\|\mathrm{M}_*\| \le 1$ implies that $|\mathrm{M}_*(i,j)| \le 1$ for all i, j, so that $g(\mathrm{M}_*) = \mathrm{M}_*$. We have that:

$$\max_{\mathrm{A}} |\langle \mathrm{M}_* - g(\mathrm{U}\mathrm{V}^\top), \mathrm{A}\rangle| = \max_{i,j} |\langle \mathrm{M}_* - g(\mathrm{U}\mathrm{V}^\top), \mathrm{e}_i\mathrm{e}_j^\top\rangle| \le \max_{i,j} |\mathrm{M}_*(i,j)| + \max_{i,j} |\langle \mathrm{U}\mathrm{V}^\top, \mathrm{e}_i\mathrm{e}_j^\top\rangle| \le \|\mathrm{M}_*\| + 1 \le 2$$

Let $L(g(\mathrm{U}\mathrm{V}^\top)) := \mathbb{E}(y - \langle g(\mathrm{U}\mathrm{V}^\top), \mathrm{A}\rangle)^2$ and $\widehat{L}(g(\mathrm{U}\mathrm{V}^\top)) := \widehat{\mathbb{E}}_i(y_i - \langle g(\mathrm{U}\mathrm{V}^\top), \mathrm{A}^{(i)}\rangle)^2$ denote the *true risk* and the *empirical risk* of $g(\mathrm{U}\mathrm{V}^\top)$, respectively. Plugging the above results in Theorem 6, we get

$$\begin{aligned} L(g(\mathrm{U}, \mathrm{V})) &\le \widehat{L}(g(\mathrm{U}, \mathrm{V})) + 8\mathfrak{R}_n(g \circ \mathcal{M}_\alpha) + 4\sqrt{\frac{\log(2/\delta)}{2n}} \\ &\le \widehat{L}(\mathrm{U}, \mathrm{V}) + 8\sqrt{\frac{\alpha d_2 \log(d_2)}{n}} + 4\sqrt{\frac{\log(2/\delta)}{2n}} \\ &\le \widehat{L}(\mathrm{U}, \mathrm{V}) + 8\sqrt{\frac{2\alpha d_2 \log(d_2) + \frac{1}{4}\log(2/\delta)}{n}} \end{aligned}$$

where the second inequality holds since $\widehat{L}(g(\mathrm{U}, \mathrm{V})) \le \widehat{L}(\mathrm{U}, \mathrm{V})$. □

B.1 Optimistic Rates

As we discussed in the main text, under additional assumptions on the value of α, it is possible to give optimistic generalization bounds that decay as $\tilde{O}(\alpha d_2/n)$. This result is given as the following theorem.

Theorem 10. Assume that $d_2 \ge d_0$ and $\|\mathrm{M}_*\| \le 1$. Furthermore, assume that $\min_{i,k} p(i)q(k) \ge \frac{\log(d_2)}{n\sqrt{d_2 d_0}}$. Let (U, V) be a minimizer of the dropout ERM objective in equation (3). Let α be such that $\max\{R(\mathrm{U}, \mathrm{V}), \Theta(\mathrm{M}_*)\} \le$

α/d_1. Then, for any $\delta \in (0,1)$, the following generalization bounds holds with probability at least $1-\delta$ over a sample of size n:

$$L(g(\mathrm{UV}^\top)) \leq \frac{2K\log(n)^3 \alpha d_2 \log(d_2) + 4K\log(1/\delta)}{n}$$

where K is an absolute constant Srebro et al. (2010), $g(\mathrm{M})$ thresholds M between $[-1,1]$, and $L(g(\mathrm{UV}^\top)) := \mathbb{E}(y - \langle g(\mathrm{UV}^\top), \mathrm{A}\rangle)^2$ is the *true risk* of $g(\mathrm{UV}^\top)$.

Proof of Theorem 10. We use Theorem 7 to bound the population risk in terms of the Rademacher complexity of the target class. Define the class of predictors with weighted trace-norm bounded by $\sqrt{\alpha}$, i.e.

$$\mathcal{M}_\alpha = \{\mathrm{M} : \ \|\operatorname{diag}(\sqrt{\mathrm{p}})\mathrm{M}\operatorname{diag}(\sqrt{\mathrm{q}})\|_*^2 \leq \alpha\}.$$

In particular dropout empirical risk minimizers U, V belong to this class:

$$\|\operatorname{diag}(\sqrt{p})\mathrm{UV}^\top \operatorname{diag}(\sqrt{q})\|_*^2 = d_1\Theta(\mathrm{UV}^\top) \leq d_1 R(\mathrm{U},\mathrm{V}) \leq \alpha$$

where the first inequality holds by definition of the induced regularizer, and the second inequality follows from the assumption of the theorem. Moreover, by assumption $\Theta(\mathrm{M}_*) \leq \alpha$, we have that $\mathrm{M}_* \in \mathcal{M}_\alpha$. With this, we get that

$$L_* := \min_{M \in g\circ\mathcal{M}_\alpha} L(\mathrm{M}) \leq L(g(\mathrm{M}_*)) = L(g(\mathrm{M}_*)) = 0.$$

Since g is a contraction, by Talagrand's lemma and Theorem 9, we have that $\mathfrak{R}_n(g\circ\mathcal{M}_\alpha) \leq \mathfrak{R}_n(\mathcal{M}_\alpha) \leq \sqrt{\frac{\alpha d_2 \log(d_2)}{n}}$. Plugging the above in Theorem 6, we get

$$\begin{aligned} L(g(\mathrm{U},\mathrm{V})) &\leq 2K\log(n)^3\mathfrak{R}_n^2(g\circ\mathcal{M}_\alpha) + \frac{4K\log\frac{1}{\delta}}{n} \\ &\leq \frac{2K\log(n)^3\alpha d_2\log(d_2) + 4K\log\frac{1}{\delta}}{n} \end{aligned}$$

□

C Non-linear Neural Networks

Proposition 4 (Dropout regularizer in deep regression)**.**

$$\widehat{L}_{drop}(\mathrm{w}) = \widehat{L}(\mathrm{w}) + \widehat{R}(\mathrm{w}), \quad \textit{where} \quad \widehat{R}(\mathrm{w}) = \lambda\sum_{j=1}^{d_1}\|u_j\|^2\widehat{a}_j^2.$$

where $\widehat{a}_j = \sqrt{\widehat{\mathbb{E}}_i a_j(\mathrm{x}_i)^2}$ *and* $\lambda = \frac{p}{1-p}$ *is the regularization parameter.*

Proof of Proposition 4. Similar statements and proofs can be found in several previous works Srivastava et al. (2014); Wang & Manning (2013); Cavazza et al. (2018); Mianjy et al. (2018). Here we include a proof for completeness. Recall that $\mathbb{E}[\mathrm{B}_{ii}] = 1$ and $\mathrm{Var}(\mathrm{B}_{ii}) = \frac{p}{1-p}$. Conditioned on x, y in the current mini-batch, we have that:

$$\mathbb{E}_\mathrm{B}\|\mathrm{y} - \mathrm{U}^\top\mathrm{Ba}(\mathrm{x})\|^2 = \sum_{i=1}^{d_2}\left(\mathbb{E}_\mathrm{B}[y_i - \mathrm{u}_i^\top\mathrm{Ba}(\mathrm{x})]\right)^2 + \sum_{i=1}^{d_2}\mathrm{Var}(y_i - \mathrm{u}_i^\top\mathrm{Ba}(\mathrm{x}))$$

Since $\mathbb{E}[\mathrm{B}] = \mathrm{I}$, the first term on right hand side is equal to $\|\mathrm{y} - \mathrm{U}^\top\mathrm{a}(\mathrm{x})\|^2$. For the second term we have

$$\sum_{i=1}^{d_2}\mathrm{Var}(y_i - \mathrm{u}_i^\top\mathrm{Ba}(\mathrm{x})) = \sum_{i=1}^{d_2}\mathrm{Var}(\sum_{j=1}^{d_1}\mathrm{U}_{j,i}\mathrm{B}_{jj}a_j(\mathrm{x})) = \sum_{i=1}^{d_2}\sum_{j=1}^{d_1}(\mathrm{U}_{j,i}a_j(\mathrm{x}))^2\,\mathrm{Var}(\mathrm{B}_{jj}) = \frac{p}{1-p}\sum_{j=1}^{d_1}\|\mathrm{u}_j\|^2 a_j(\mathrm{x})^2$$

Thus, conditioned on the sample (x, y), we have that

$$\mathbb{E}_{\mathrm{B}}[\|\mathrm{y} - \mathrm{U}^\top \mathrm{B}\mathrm{a}(\mathrm{x})\|^2] = \|\mathrm{y} - \mathrm{U}^\top \mathrm{a}(\mathrm{x})\|^2 + \frac{p}{1-p}\sum_{j=1}^{d_1} \|\mathrm{u}_j\|^2 a_j(\mathrm{x})^2$$

Now taking the empirical average with respect to x, y, we get

$$\widehat{L}_{\mathrm{drop}}(\mathrm{w}) = \widehat{L}(\mathrm{w}) + \frac{p}{1-p}\sum_{j=1}^{d_1} \|\mathrm{u}_j\|^2 \widehat{a}_j^2 = \widehat{L}(\mathrm{w}) + \widehat{R}(\mathrm{w})$$

which completes the proof. □

Proposition 5. *Consider a two layer neural network $f_{\mathrm{w}}(\cdot)$ with ReLU activation functions in the hidden layer. Furthermore, assume that the marginal input distribution $\mathbb{P}_{\mathcal{X}}(\mathrm{x})$ is symmetric and isotropic, i.e., $\mathbb{P}_{\mathcal{X}}(\mathrm{x}) = \mathbb{P}_{\mathcal{X}}(-\mathrm{x})$ and $\mathbb{E}[\mathrm{x}\mathrm{x}^\top] = I$. Then the following holds for the expected 5 due to dropout:*

$$R(\mathrm{w}) := \mathbb{E}[\widehat{R}(\mathrm{w})] = \frac{\lambda}{2}\sum_{i_0, i_1, i_2 = 1}^{d_0, d_1, d_2} U(i_1, i_2)^2\, V(i_1, i_0)^2, \tag{8}$$

Proof of Proposition 5. Using Proposition 4, we have that:

$$R(\mathrm{w}) = \mathbb{E}[\widehat{R}(\mathrm{w})] = \lambda \sum_{j=1}^{d_1} \|\mathrm{u}_j\|^2 \mathbb{E}[\sigma(\mathrm{V}(j,:)^\top \mathrm{x})^2]$$

It remains to calculate the quantity $\mathbb{E}_{\mathrm{x}}[\sigma(\mathrm{V}(j,:)^\top \mathrm{x})^2]$. By symmetry assumption, we have that $\mathbb{P}_{\mathcal{X}}(\mathrm{x}) = \mathbb{P}_{\mathcal{X}}(-\mathrm{x})$. As a result, for any $\mathrm{v} \in \mathbb{R}^{d_0}$, we have that $\mathbb{P}(\mathrm{v}^\top \mathrm{x}) = \mathbb{P}(-\mathrm{v}^\top \mathrm{x})$ as well. That is, the random variable $z_j := \mathrm{W}_1(j,:)^\top \mathrm{x}$ is also symmetric about the origin. It is easy to see that $\mathbb{E}_z[\sigma(z)^2] = \frac{1}{2}\mathbb{E}_z[z^2]$.

$$\mathbb{E}_z[\sigma(z)^2] = \int_{-\infty}^{\infty} \sigma(z)^2 d\mu(z) = \int_0^{\infty} \sigma(z)^2 d\mu(z) = \int_0^{\infty} z^2 d\mu(z) = \frac{1}{2}\int_{\infty}^{\infty} z^2 d\mu(z) = \frac{1}{2}\mathbb{E}_z[z^2].$$

Plugging back the above identity in the expression of $R(\mathrm{w})$, we get that

$$R(\mathrm{w}) = \lambda \sum_{j=1}^{d_1} \|\mathrm{u}_j\|^2 \mathbb{E}[(\mathrm{V}(j,:)^\top \mathrm{x})^2] = \frac{\lambda}{2}\sum_{j=1}^{d_1} \|\mathrm{u}_j\|^2 \|\mathrm{V}(j,:)\|^2$$

where the second equality follows from the assumption that the distribution is isotropic. □

Proof of Proposition 1. For $\delta \in (0, \frac{1}{2})$, consider the following random variable:

$$\mathrm{x} = \begin{cases} [1; 0] & \text{with probability } \delta \\ [\frac{-\delta}{1-\delta}; \frac{\sqrt{1-2\delta}}{1-\delta}] & \text{with probability } \frac{1-\delta}{2} \\ [\frac{-\delta}{1-\delta}; -\frac{\sqrt{1-2\delta}}{1-\delta}] & \text{with probability } \frac{1-\delta}{2} \end{cases}$$

It is easy to check that the x has zero mean and is supported on the unit sphere. Consider the vector $\mathrm{w} = [\frac{1}{\sqrt{\delta}}; 0]$. It is easy to check that x satisfies $R(\mathrm{w}) = \sqrt{\mathbb{E}\sigma(\mathrm{w}^\top \mathrm{x})^2} = 1$; however, for any given C, it holds that $\|\mathrm{w}\| \geq C$ as long as we let $\delta = C^2$. □

Proof of Theorem 2. For any $j \in [h]$, let $a_j^2 := \mathbb{E}[\sigma(\mathrm{v}_j^\top \mathrm{x})^2]$ denote the average squared activation of the j-th node with respect to the input distribution. Given n i.i.d. samples $\mathcal{S} = \{\mathrm{x}_1, \cdots, \mathrm{x}_n\}$, the empirical Rademahcer complexity is bounded as follows:

$$\begin{aligned}
\mathfrak{R}_{\mathcal{S}}(\mathcal{F}_\alpha) &= \mathbb{E}_\zeta \sup_{f_{\{\mathrm{u},\mathrm{V}\}} \in \mathcal{F}_\alpha} \frac{1}{n} \sum_{j=1}^h u_j a_j \sum_{i=1}^n \zeta_i \frac{\sigma(\mathrm{v}_j^\top \mathrm{x}_i)}{a_j} \\
&\le \mathbb{E}_\zeta \sup_{f_{\{\mathrm{u},\mathrm{V}\}} \in \mathcal{F}_\alpha} \frac{1}{n} \sum_{j=1}^h |u_j a_j| \, |\sum_{i=1}^n \zeta_i \frac{\sigma(\mathrm{v}_j^\top \mathrm{x}_i)}{a_j}| \\
&\le \mathbb{E}_\zeta \left[\left(\sup_{f_{\{\mathrm{u},\mathrm{V}\}} \in \mathcal{F}_\alpha} \sum_{j=1}^h |u_j a_j| \right) \left(\sup_{\mathrm{V}} \max_{j \in [h]} |\frac{1}{n} \sum_{i=1}^n \zeta_i \frac{\sigma(\mathrm{v}_j^\top \mathrm{x}_i)}{a_j}| \right) \right]
\end{aligned}$$

where we used the fact that the supremum of product of positive functions is upperbounded by the product of the supremums. By definition of $\mathcal{F}_\alpha$, the first term on the right hand side is bounded by α. To bound the second term in the right hand side, we note that the maximum over rows of $\mathrm{V}^\top$ can be absorbed into the supremum.

$$\begin{aligned}
\frac{1}{n} \mathbb{E}_\zeta \sup_{\mathrm{v}} |\sum_{i=1}^n \zeta_i \frac{\sigma(\mathrm{v}^\top \mathrm{x}_i)}{\sqrt{\mathbb{E}[\sigma(\mathrm{v}^\top \mathrm{x})^2]}}| &= \frac{1}{n} \mathbb{E}_\zeta \sup_{\mathbb{E}[\sigma(\mathrm{v}^\top \mathrm{x})^2] \le 1} |\sum_{i=1}^n \zeta_i \sigma(\mathrm{v}^\top \mathrm{x}_i)| \\
&\le \frac{2}{n} \mathbb{E}_\zeta \sup_{\mathbb{E}[\sigma(\mathrm{v}^\top \mathrm{x})^2] \le 1} \sum_{i=1}^n \zeta_i \sigma(\mathrm{v}^\top \mathrm{x}_i) \\
&\le \frac{2}{n} \mathbb{E}_\zeta \sup_{\beta \mathbb{E}(\mathrm{v}^\top \mathrm{x})^2 \le 1} \sum_{i=1}^n \zeta_i \sigma(\mathrm{v}^\top \mathrm{x}_i) \qquad (\beta\text{-retentiveness})
\end{aligned}$$

Let $\mathrm{C}^\dagger$ be the pseudo-inverse of C. We perform the following change the variable: $\mathrm{w} \leftarrow \mathrm{C}^{-\dagger/2} \mathrm{v}$.

$$\begin{aligned}
\text{R.H.S.} &\le \frac{2}{n} \mathbb{E}_\zeta \sup_{\mathbb{E}[(\mathrm{w}^\top \mathrm{C}^{\dagger/2} \mathrm{x})^2] \le 1/\beta} \sum_{i=1}^n \zeta_i \mathrm{w}^\top \mathrm{C}^{\dagger/2} \mathrm{x}_i \\
&= \frac{2}{n} \mathbb{E}_\zeta \sup_{\|\mathrm{w}\|^2 \le 1/\beta} \langle \mathrm{w}, \sum_{i=1}^n \zeta_i \mathrm{C}^{\dagger/2} \mathrm{x}_i \rangle \\
&= \frac{2}{n\sqrt{\beta}} \mathbb{E}_\zeta \| \sum_{i=1}^n \zeta_i \mathrm{C}^{\dagger/2} \mathrm{x}_i \| \\
&\le \frac{2}{n\sqrt{\beta}} \sqrt{\mathbb{E}_\zeta \| \sum_{i=1}^n \zeta_i \mathrm{C}^{\dagger/2} \mathrm{x}_i \|^2} = \frac{2}{n\sqrt{\beta}} \sqrt{\sum_{i=1}^n \mathrm{x}_i^\top \mathrm{C}^\dagger \mathrm{x}_i}
\end{aligned}$$

where the last inequality holds due to Jensen's inequality. To bound the expected Rademacher complexity, we take the expected value of both sides with respected to sample $\mathcal{S}$, which gives the following:

$$\mathfrak{R}_n(\mathcal{F}_\alpha) = \mathbb{E}_{\mathrm{x}}[\mathfrak{R}_{\mathcal{S}}(\mathcal{F}_\alpha)] \le \frac{2}{n\sqrt{\beta}} \mathbb{E}_{\mathcal{S}} \sqrt{\sum_{i=1}^n \mathrm{x}_i^\top \mathrm{C}^\dagger \mathrm{x}_i} \le \frac{2}{n\sqrt{\beta}} \sqrt{\sum_{i=1}^n \mathbb{E}_{\mathrm{x}_i}[\mathrm{x}_i^\top \mathrm{C}^\dagger \mathrm{x}_i]},$$

where the last inequality holds again due to Jensen's inequality. Finally, we have that $\mathbb{E}_{\mathrm{x}_i} \mathrm{x}_i^\top \mathrm{C}^\dagger \mathrm{x}_i = \mathbb{E}_{\mathrm{x}_i} \langle \mathrm{x}_i \mathrm{x}_i^\top, \mathrm{C}^\dagger \rangle = \langle \mathrm{C}, \mathrm{C}^\dagger \rangle = \mathrm{Rank}(\mathrm{C})$, which completes the proof of the Theorem. □

Proof of Theorem 3. For simplicity, assume that the width of the hidden layer is even. Consider the linear function class:

$$\mathcal{G}_r := \{g_{\mathrm{w}} : \mathrm{x} \mapsto \mathrm{w}^\top \mathrm{x}, \ \mathbb{E}(\mathrm{w}^\top \mathrm{x})^2 \le d_1 r/2\}.$$

Recall that $\mathcal{H}_r := \{h_{\mathrm{w}} : \mathrm{x} \mapsto \mathrm{u}^\top \sigma(\mathrm{V}^\top \mathrm{x}),\ R(\mathrm{u}, \mathrm{V}) \le r\}$. First, we argue that $\mathcal{G}_r \subset \mathcal{H}_r$. Let $g_{\mathrm{w}} \in \mathcal{G}_r$; we show that there exist u, V such that $g_{\mathrm{w}} = f_{\mathrm{u},\mathrm{V}}$ and $f_{\mathrm{u},\mathrm{V}} \in \mathcal{H}_r$. Define $\mathrm{u} := \frac{2}{d_1}[1; -1; \cdots 1; -1] \in \mathbb{R}^{d_1}$, and let $\mathrm{V} = \mathrm{w}(\mathrm{e}_1 - \mathrm{e}_2 + \mathrm{e}_3 - \mathrm{e}_4 + \cdots + \mathrm{e}_{d_1-1} - \mathrm{e}_{d_1})^\top$, where $\mathrm{e}_i \in \mathbb{R}^{d_1}$ is the i-th standard basis vector. It's easy to see that

$$\begin{aligned}
f_{\mathrm{u},\mathrm{V}}(\mathrm{x}) = \mathrm{u}^\top \sigma(\mathrm{V}^\top \mathrm{x}) &= \sum_{i=1}^{d_1} u_i \sigma(\mathrm{v}_i^\top \mathrm{x}) \\
&= \sum_{i=1}^{d_1} \frac{2}{d_1} (-1)^{i-1} \sigma(\mathrm{v}_i^\top \mathrm{x}) \\
&= \sum_{i=1}^{d_1/2} \frac{2}{d_1} (\sigma(\mathrm{v}_{2i-1}^\top \mathrm{x}) - \sigma(\mathrm{v}_{2i}^\top \mathrm{x})) \\
&= \sum_{i=1}^{d_1/2} \frac{2}{d_1} (\sigma(\mathrm{w}^\top \mathrm{x}) - \sigma(-\mathrm{w}^\top \mathrm{x})) = \mathrm{w}^\top \mathrm{x} = g_{\mathrm{w}}.
\end{aligned}$$

Furthermore, it holds for the explicit regularizer that

$$\begin{aligned}
R(\mathrm{u}, \mathrm{V}) = \sum_{i=1}^{d_1} u_i^2 \mathbb{E}\sigma(\mathrm{v}_i^\top \mathrm{x})^2 &= \sum_{i=1}^{d_1/2} \frac{4}{d_1^2} \left(\mathbb{E}\sigma(\mathrm{v}_{2i-1}^\top \mathrm{x})^2 + \mathbb{E}\sigma(\mathrm{v}_{2i}^\top \mathrm{x})^2 \right) \\
&= \sum_{i=1}^{d_1/2} \frac{4}{d_1^2} \mathbb{E}[\sigma(\mathrm{w}^\top \mathrm{x})^2 + \sigma(-\mathrm{w}^\top \mathrm{x})^2] \\
&= \frac{2}{d_1} \mathbb{E}(\mathrm{w}^\top \mathrm{x})^2 \le r
\end{aligned}$$

Thus, we have that $\mathcal{G}_r \subset \mathcal{H}_r$, and the following inequalities follow.

$$\begin{aligned}
\mathfrak{R}_{\mathcal{S}}(\mathcal{H}_r) \ge \mathfrak{R}_{\mathcal{S}}(\mathcal{G}_r) &= \mathbb{E}_{\epsilon_i} \sup_{g_{\mathrm{w}} \in \mathcal{G}_r} \frac{1}{n} \sum_{i=1}^n \epsilon_i g_{\mathrm{w}}(\mathrm{x}_i) \\
&= \mathbb{E}_{\epsilon_i} \sup_{\mathbb{E}(\mathrm{w}^\top \mathrm{x})^2 \le d_1 r/2} \frac{1}{n} \sum_{i=1}^n \epsilon_i \mathrm{w}^\top \mathrm{x}_i \\
&= \mathbb{E}_{\epsilon_i} \sup_{\mathrm{w}^\top \mathrm{C} \mathrm{w} \le d_1 r/2} \frac{1}{n} \langle \mathrm{w}, \sum_{i=1}^n \epsilon_i \mathrm{x}_i \rangle \\
&= \mathbb{E}_{\epsilon_i} \sup_{\|\mathrm{C}^{1/2}\mathrm{w}\|^2 \le d_1 r/2} \frac{1}{n} \langle \mathrm{C}^{1/2}\mathrm{w}, \sum_{i=1}^n \epsilon_i \mathrm{C}^{-\dagger/2} \mathrm{x}_i \rangle \\
&= \frac{\sqrt{d_1 r}}{\sqrt{2} n} \mathbb{E}_{\epsilon_i} \| \sum_{i=1}^n \epsilon_i \mathrm{C}^{\dagger/2} \mathrm{x}_i \| \\
&\ge \frac{c\sqrt{d_1 r}}{\sqrt{2} n} \sqrt{\sum_{i=1}^n \|\mathrm{C}^{\dagger/2} \mathrm{x}_i\|^2} = \frac{c\sqrt{d_1 r} \|\mathrm{X}\|_{\mathrm{C}^\dagger}}{\sqrt{2} n}
\end{aligned}$$

where the last inequality follows from Khintchine-Kahane inequality in Lemma 1. □

Next, we define some function classes that will be used frequently in the proofs.

Definition 1. *For any closed subset $[a,b] \subset \mathbb{R}$, let $\Pi_{[a,b]}(y) := \max\{a, \min\{b, y\}\}$. For $z := (\mathrm{x}, y)$ and $f : \mathcal{X} \to \mathcal{Y}$, define the squared loss $\ell_2(f, z) := (1 - yf(\mathrm{x}))^2$. For a given value $\alpha > 0$, consider the following classes*

$$\begin{aligned}
\mathcal{W}_\alpha &:= \{\mathrm{w} = (u, V) \in \mathbb{R}^{d_1} \times \mathbb{R}^{d_0 \times d_1},\ \sum_{i=1}^{d_1} |u_i| \sqrt{\mathbb{E}\sigma(v_i^\top \mathrm{x})^2} \le \alpha\} \\
\mathcal{F}_\alpha &:= \{f_\mathrm{w} : \mathrm{x} \mapsto u^\top \sigma(V^\top \mathrm{x}),\ \mathrm{w} \in \mathcal{W}_\alpha\}, \\
\mathcal{G}_\alpha &:= \Pi_{[-1,1]} \circ \mathcal{F}_\alpha = \{g_\mathrm{w} = \Pi_{[-1,-1]} \circ f_\mathrm{w},\ f_\mathrm{w} \in \mathcal{F}_\alpha\} \\
\mathcal{L}_\alpha &:= \{\ell_2 : (g_\mathrm{w}, z) \mapsto (y - g_\mathrm{w}(\mathrm{x}))^2,\ g_\mathrm{w} \in \mathcal{G}_\alpha\}
\end{aligned}$$

Lemma 3. *Let $\mathcal{W}_\alpha, \mathcal{F}_\alpha, \mathcal{G}_\alpha, \mathcal{L}_\alpha$ be as defined in Definition 1. Then the following holds true:*

1. $\mathfrak{R}_\mathcal{S}(\mathcal{G}_\alpha) \le \mathfrak{R}_\mathcal{S}(\mathcal{F}_\alpha)$.
2. *If $\mathcal{Y} = \{-1, +1\}$ (binary classification), then it holds that $\mathfrak{R}_\mathcal{S}(\mathcal{L}_\alpha) \le 2\mathfrak{R}_\mathcal{S}(\mathcal{G}_\alpha)$.*

Proof. Since $\Pi_{[-1,-1]}(\cdot)$ is 1-Lipschitz, by Talagrand's contraction lemma, we have that $\mathfrak{R}_\mathcal{S}(\mathcal{G}_\alpha) \le \mathfrak{R}_\mathcal{S}(\mathcal{F}_\alpha)$. The second claim follows from

$$\begin{aligned}
\mathfrak{R}_\mathcal{S}(\mathcal{L}_\alpha) &= \mathbb{E}_\zeta \sup_{\mathrm{w} \in \mathcal{W}} \frac{1}{n} \sum_{i=1}^n \zeta_i (y_i - g_\mathrm{w}(\mathrm{x}_i))^2 \\
&= \mathbb{E}_\zeta \sup_{\mathrm{w} \in \mathcal{W}} \frac{1}{n} \sum_{i=1}^n \zeta_i (1 - y_i g_\mathrm{w}(\mathrm{x}_i))^2 && (y_i \in \{-1, +1\}) \\
&\le 2\mathbb{E}_\zeta \sup_{\mathrm{w} \in \mathcal{W}} \frac{1}{n} \sum_{i=1}^n \zeta_i y_i g_\mathrm{w}(\mathrm{x}_i) \\
&= 2\mathbb{E}_\zeta \sup_{\mathrm{w} \in \mathcal{W}} \frac{1}{n} \sum_{i=1}^n \zeta_i g_\mathrm{w}(\mathrm{x}_i) = 2\mathfrak{R}_\mathcal{S}(\mathcal{G}_\alpha)
\end{aligned}$$

where the first inequality follows from Talagrand's contraction lemma due to the fact that $h(z) = (1 - z)^2$ is 2-Lipschitz for $z \in [-1, 1]$, and the penultimate holds true since for any fixed $(y_i)_{i=1}^n \in \{-1, +1\}^n$, the distribution of $(\zeta_1 y_1, \ldots, \zeta_n y_n)$ is the same as that of $(\zeta_1, \ldots, \zeta_n)$. □

Proof of Corollary 1. We use the standard generalization bound in Theorem 6 for class $\mathcal{G}_\alpha$:

$$\begin{aligned}
L_\mathcal{D}(g_\mathrm{w}) &\le \widehat{L}_\mathcal{S}(g_\mathrm{w}) + 4M\mathfrak{R}_\mathcal{S}(\mathcal{G}_\alpha) + 3M^2 \sqrt{\frac{\log(2/\delta)}{2n}} \\
&\le \widehat{L}_\mathcal{S}(g_\mathrm{w}) + 8\mathfrak{R}_\mathcal{S}(\mathcal{F}_\alpha) + 12\sqrt{\frac{\log(2/\delta)}{2n}} && \text{(Lemma 3)} \\
&\le \widehat{L}_\mathcal{S}(g_\mathrm{w}) + \frac{16\alpha \|\mathrm{X}\|_{\mathrm{C}^\dagger}}{\sqrt{\beta n}} + 12\sqrt{\frac{\log(2/\delta)}{2n}} && \text{(Theorem 2)}
\end{aligned}$$

where second inequality follows because the maximum deviation parameter M in Theorem 6 is bounded as

$$M = \sup_{\mathrm{w} \in \mathcal{W}} \sup_{(\mathrm{x}, y) \in \mathcal{X} \times \mathcal{Y}} |y - g_\mathrm{w}(\mathrm{x})| \le \sup_{\mathrm{w} \in \mathcal{W}} \sup_{(\mathrm{x}, y) \in \mathcal{X} \times \mathcal{Y}} |y| + |g_\mathrm{w}(\mathrm{x})| \le 2.$$

□

Proof of Corollary 2. Recall that the input is jointly distributed as $(\mathrm{x}, y) \sim \mathcal{D}$. For $\mathcal{X} \subseteq \mathbb{R}_+^{d_0}$, let $\mathcal{X}' = \mathcal{X} \cup -\mathcal{X}$ be the *symmetrized input domain*. Let ζ be a Rademacher random variable. Denote the *symmetrized input* by $\mathrm{x}' = \zeta \mathrm{x}$, and the joint distribution of (x', y) by $\mathcal{D}'$. By construction, $\mathcal{D}'$ is centrally symmetric w.r.t. x',

i.e., it holds for all $(\mathrm{x}, y) \in \mathcal{X} \times \mathcal{Y}$ that $\mathcal{D}'(\mathrm{x}, y) = \mathcal{D}'(-\mathrm{x}, y) = \frac{1}{2}\mathcal{D}(\mathrm{x}, y)$. As a result, population risk with respect to the original distribution $\mathcal{D}$ can be bounded in terms of the population risk with respect to the *symmetrized distribution* $\mathcal{D}'$ as follows:

$$\begin{aligned} L_{\mathcal{D}}(f) &:= \mathbb{E}_{\mathcal{D}}[\ell(f(\mathrm{x}), y)] \\ &\le \mathbb{E}_{\mathcal{D}}[\ell(f(\mathrm{x}), y) + \ell(f(-\mathrm{x}), y)] \\ &= 2\mathbb{E}_{\mathcal{D}}[\frac{1}{2}\ell(f(\mathrm{x}), y) + \frac{1}{2}\ell(f(-\mathrm{x}), y)] \\ &= 2\mathbb{E}_{\mathcal{D}}\mathbb{E}_{\zeta}[\ell(f(\zeta\mathrm{x}), y) \mid \mathrm{x}, y] \\ &= 2\mathbb{E}_{\mathcal{D}'}[\ell(f(\mathrm{x}'), y)] = 2L_{\mathcal{D}'}(f) \end{aligned} \tag{9}$$

Moreover, since $\mathcal{D}'$ is centrally symmetric, Assumption 1 holds with $\beta = \frac{1}{2}$. The proof of Corollary 2 follows by doubling the right hand side of inequalities in Corollary 1, and substituting $\beta = \frac{1}{2}$. □

C.1 Classification

Although in the main text we only focus on the task of regression with squared loss, it is not hard to extend the results to binary classification. In particular, the following two Corollaries bound the miss-classification error in terms of the training error and the Rademacher complexity of the target class, with and without symmetrization.

Corollary 3. *Consider a binary classification setting where $\mathcal{Y} = \{-1, +1\}$. For any $\mathrm{w} \in \mathcal{F}_\alpha$, for any $\delta \in (0, 1)$, the following generalization bound holds with probability at least $1 - \delta$ over $\mathcal{S} = \{(\mathrm{x}_i, y_i)\}_{i=1}^n \sim \mathcal{D}^n$:*

$$\mathbb{P}\{y f_{\mathrm{w}}(\mathrm{x}) < 0\} \le \widehat{L}_{\mathcal{S}}(g_{\mathrm{w}}) + \frac{8\alpha\|\mathrm{X}\|_{\mathrm{C}^\dagger}}{\sqrt{\beta}n} + 4\sqrt{\frac{\log(1/\delta)}{2n}}$$

where $g_{\mathrm{w}}(\cdot) = \max\{-1, \min\{1, f_{\mathrm{w}}(\cdot)\}\}$ projects the network output onto the range $[-1, 1]$.

Proof of Corollary 3. We use the standard generalization bound in Theorem 5. Recall that $g_w = \Pi_{[-1,1]}(f_{\mathrm{w}})$, where $\Pi_{[-1,1]}(y) = \max\{-1, \max\{1, y\}\}$ projects onto the range $[-1, 1]$. It is easy to bound the classification error of f_{w} in terms of the ℓ_2-loss of g_{w}:

$$\mathbb{P}\{\mathrm{sgn}(f_{\mathrm{w}}(\mathrm{x})) \neq y\} = \mathbb{P}\{y f_{\mathrm{w}}(\mathrm{x}) < 0\} = \mathbb{E}[1_{y f_{\mathrm{w}}(\mathrm{x}) < 0}] = \mathbb{E}[1_{y g_{\mathrm{w}}(\mathrm{x}) < 0}] \le \mathbb{E}(1 - y g_{\mathrm{w}}(\mathrm{x}))^2 = L_{\mathcal{D}}(g_{\mathrm{w}}). \tag{10}$$

We use Theorem 5 for class $\frac{1}{4}\mathcal{L}_\alpha$ to get the generalization bound as follows:

$$\begin{aligned} &\frac{1}{4}L_{\mathcal{D}}(g_{\mathrm{w}}) \le \frac{1}{4}\widehat{L}_{\mathcal{S}}(g_{\mathrm{w}}) + 2\mathfrak{R}_{\mathcal{S}}(\frac{1}{4}\mathcal{L}_\alpha) + \sqrt{\frac{\log(1/\delta)}{2n}} \\ \implies &L_{\mathcal{D}}(g_{\mathrm{w}}) \le \widehat{L}_{\mathcal{S}}(g_{\mathrm{w}}) + 4\mathfrak{R}_{\mathcal{S}}(\mathcal{F}_\alpha) + 4\sqrt{\frac{\log(1/\delta)}{2n}} && \text{(by Lemma 3)} \\ \implies &L_{\mathcal{D}}(g_{\mathrm{w}}) \le \widehat{L}_{\mathcal{S}}(g_{\mathrm{w}}) + \frac{8\alpha\|\mathrm{X}\|_{\mathrm{C}^\dagger}}{\sqrt{\beta}n} + 4\sqrt{\frac{\log(1/\delta)}{2n}} && \text{(by Theorem 2)} \end{aligned}$$

□

Corollary 4. *Consider a binary classification setting where $\mathcal{Y} = \{-1, +1\}$. For any $\mathrm{w} \in \mathcal{F}'_\alpha$, for any $\delta \in (0, 1)$, the following generalization bound holds with probability at least $1 - \delta$ over a sample of size n and the randomization in symmetrization*

$$\mathbb{P}\{y g_{\mathrm{w}}(\mathrm{x}) < 0\} \le 2\widehat{L}_{\mathcal{S}'}(g_{\mathrm{w}}) + \frac{23\alpha'\|\mathrm{X}\|_{\mathrm{C}^\dagger}}{n} + 8\sqrt{\frac{\log(1/\delta)}{2n}}$$

Proof of Corollary 4. Akin to proof of Corollary 2, we have that $L_{\mathcal{D}}(f) \le 2L_{\mathcal{D}'}(f)$, and the marginal distribution is $\frac{1}{2}$-retentive. Proof of Corollary 4 follows by doubling the right hand side of inequalities in Corollary 3, and substituting $\beta = \frac{1}{2}$. □

D Additional Experiments

In this section, we include additional plots which was not reported in the main paper due to the space limitations.

D.1 Matrix Completion

Figure 1 in the main paper shows comparisons between plain SGD and the dropout algorithm on the MovieLens dataset for a factorization size of $d_1 = 70$. The observation that we make with regard to those plots is not at all limited to the specific choice of the factorization size. In Figure 1 here, we report similar experiments with factorization sizes $d_1 \in \{30, 110, 150, 190\}$. It can be seen that the overall behaviour of plain SGD and dropout are very similar in all experiments. In particular, plain SGD always achieves the best training error but it has the largest generalization gap. Furthermore, increasing the dropout rate increases the training error but results in a tighter generalization gap.

It can be seen that an appropriate choice of the dropout rate *always* perform better than the plain SGD in terms of the test error. For instance, a dropout rate of $p = 0.2$ seems to always outperform plain SGD. Moreover, as the factorization size increases, the function class becomes more complex, and a larger value of the dropout rate is more helpful. For example, when $d_1 = 30$, the dropout with rates $p = 0.3, 0.4$ fail to achieve a good test performance, where as for larger factorization sizes ($d_1 \in \{110, 150, 190\}$), they consistently outperform plain SGD as well as other dropout rates.

D.2 Shallow Neural Networks

In Figure 2, we plot the co-adaptation measure, the generalization gap, as well as the complexity measure $\alpha/\sqrt{n}$ as a function of width of the network, for FashionMNIST with symmetrization, and for MNIST without symmetrization.

The co-adaptation plot is very similar to Figure 2 in the main text. In particular, 1) increasing the dropout rate results in less co-adaptation; 2) even plain SGD is biased towards networks with less co-adapation; and 3) as the networks becomes wider, the co-adaptation curves corresponding to plain SGD converge to those of dropout. We also make similar observations for the generalization gap as well as the complexity term $\alpha/\sqrt{n}$. In particular, 1) a higher dropout rate corresponds to a lower generalization gap, uniformly for all widths; 2) the generalization gap is higher for wider networks; and 3) curves with smaller complexity terms in the right plot correspond to curves with smaller generalization gaps in the middle plot.

D.3 Deep Neural Networks

In Section 3, we derived generalization bounds that scale with the explicit regularizer as $O(\sqrt{\frac{\texttt{width}\cdot R(\mathrm{w})}{n}})$. Although our theoretical analysis is limited to two-layer networks; empirically, we show in Figure 3 that the generalization gap correlates well with this measure even for deep neural networks. In particular, we train deep convolutional neural networks with a dropout layer on top of the feature extractor, i.e. the top hidden layer. Let $\texttt{feature}_i$ denote the i-th hidden node in the top hidden layer. Akin to the derivation presented in Proposition 4, it is easy to see that the (expected) explicit regularizer is given by $R(\mathrm{w}) = \frac{p}{1-p}\sum_{i=1}^{\texttt{width}} \|\mathrm{u}_i\|^2 a_i^2$, where $\texttt{width}$ is the width of the top hidden layer, U denotes the top layer weight matrix, and $a_i^2 = \mathbb{E}_{\mathrm{x}}[\texttt{feature}_i(\mathrm{x})^2]$ is the second moment of the i-th node in the top hidden layer.

We train convolutional neural networks with and without dropout, on MNIST, Fashion MNIST, and CIFAR-10. The CIFAR-10 dataset consists of 60K 32×32 color images in 10 classes, with 6k images per class, divided into a training set and a test set of sizes 50K and 10K respectively Krizhevsky et al. (2009). We do not perform symmetrization in these experiments. In contrast with the experiments in the previous section, here we run the experiments on full datasets, representing each of the ten classes as a one-hot target vector.

For MNIST and Fashion MNIST datasets, we use a convolutional neural network with one convolutional layer and two fully connected layers. The convolutional layer has 16 convolutional filters, padding and stride

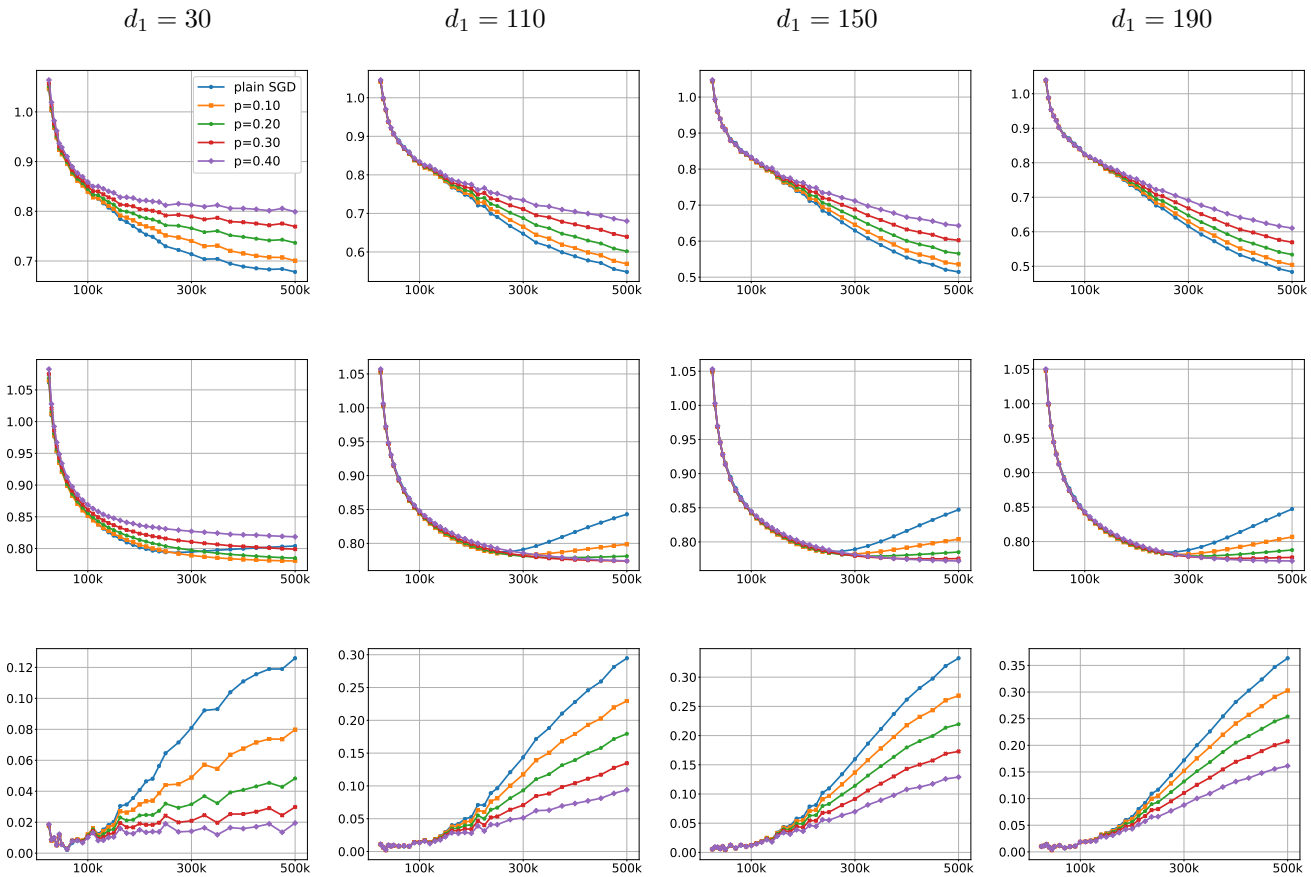

Figure 1: MovieLens dataset: the training error (**top**), the test error (**middle**), and the generalization gap for plain SGD as well as dropout with $p \in \{0.25, 0.50, 0.75\}$ as a function of the number of iterates, for different factorization sizes $d_1 = 30$ (first column), $d_1 = 110$ (second column), $d_1 = 150$ (third column), and $d_1 = 190$ (forth column).

of 2, and kernel size of 5. We report experiments on networks with the width of the top hidden layer chosen from $\texttt{width} \in \{2^6, 2^7, 2^8, 2^9, 2^{10}, 2^{11}\}$. In all the experiments, a fixed learning rate $\texttt{lr} = 0.5$ and a mini-batch of size 256 is used to perform the updates. We train the models for 30 epochs over the whole training set.

For CIFAR-10, we use an AlexNet Krizhevsky et al. (2012), where the layers are modified accordingly to match the dataset. The only difference here is that we apply dropout to the top hidden layer, whereas in Krizhevsky et al. (2012), dropout is used on top of the second and the third hidden layers from the top. We report experiments on networks with the width of the top hidden layer chosen from $\texttt{width} \in \{2^5, 2^6, 2^7, 2^8, 2^9, 2^{10}, 2^{11}, 2^{12}\}$. In all the experiments, an initial learning rate $\texttt{lr} = 5$ and a mini-batch of size 256 is used to perform the updates. We train the models for 100 epochs over the whole training set. We decay the learning rate by a factor of 10 every 30 epochs.

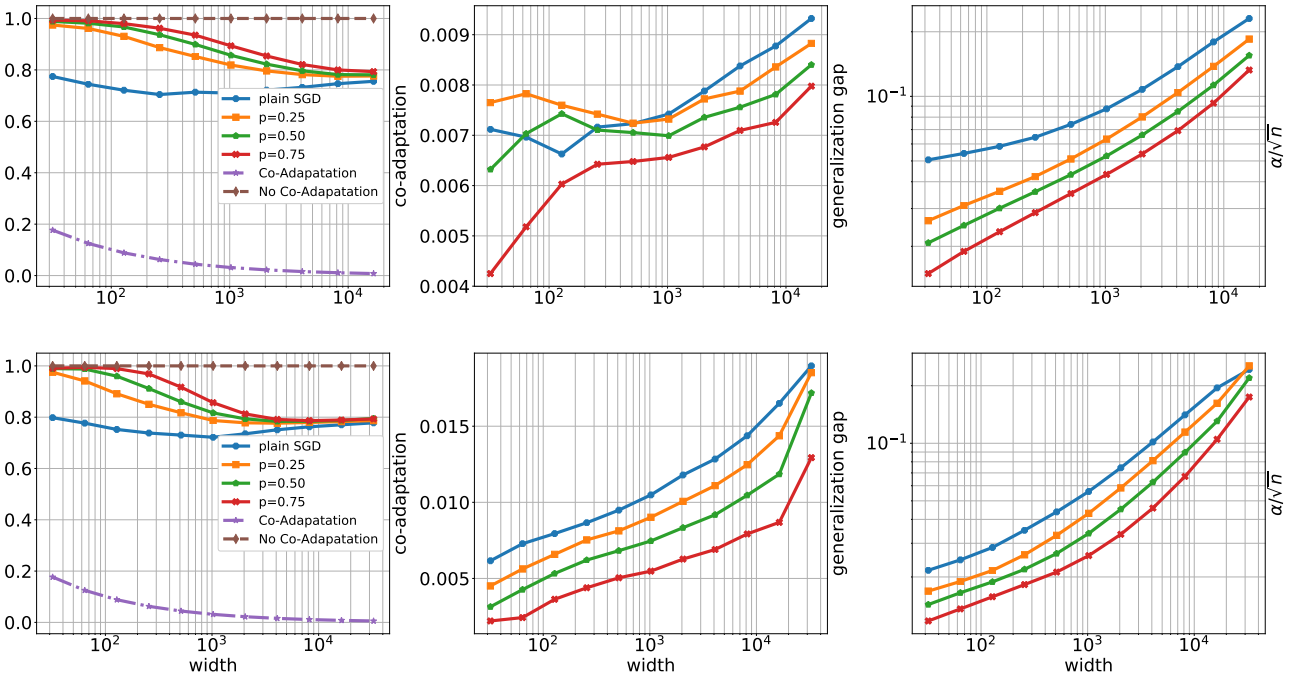

Figure 2: (**left**) *"co-adaptation"*; (**middle**) generalization gap; and (**right**) $\alpha/\sqrt{n}$ (**top**) with symmetrization on FashionMNIST; and (**bottom**) without symmetrization on MNIST. In left column, the dashed brown and dotted purple lines represent minimal and maximal co-adaptations, respectively.

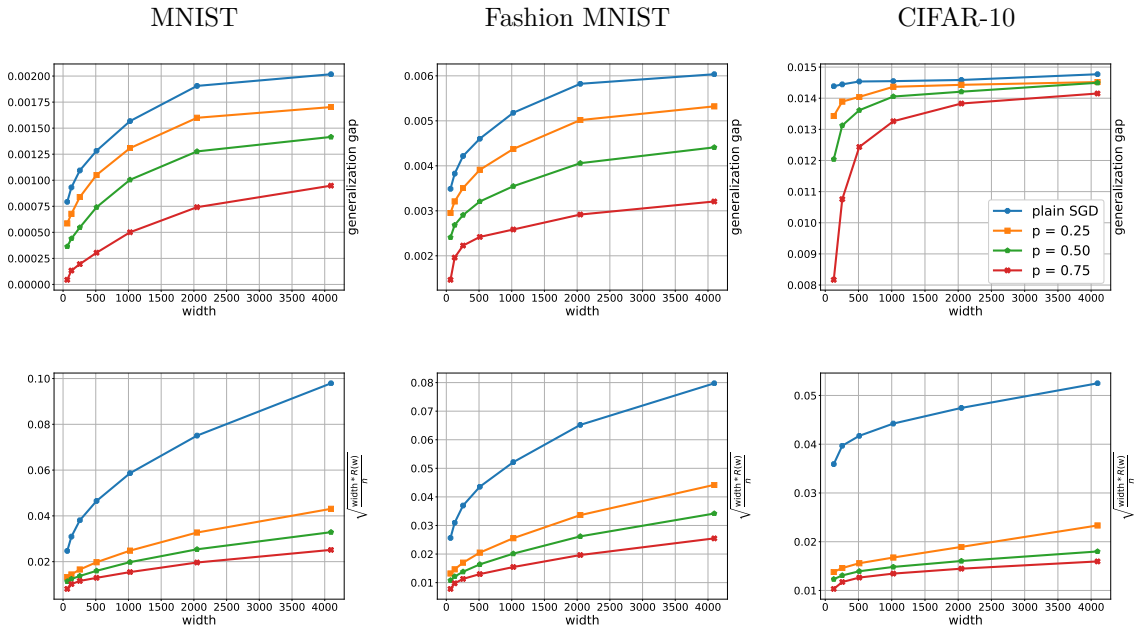

Figure 3: (**top**) generalization gap and (**bottom**) the complexity measure ($\sqrt{\frac{\texttt{width}\cdot R(\mathrm{w})}{n}}$) as a function of the width of the top hidden layer on (**left**) MNIST, (**middle**) Fashion MNIST, and (**right**) CIFAR-10.