Randomized Dimension Reduction with Statistical Guarantees

by

Yijun Dong

DISSERTATION

Presented to the Faculty of the Graduate School
of the University of Texas at Austin
in Partial Fulfillment
of the Requirements
for the Degree of

DOCTOR OF PHILOSOPHY

The University of Texas at Austin
August 2023

The Dissertation Committee for Yijun Dong

certifies that this is the approved version of the following dissertation:

Randomized Dimension Reduction with Statistical Guarantees

Committee:

Per-Gunnar Martinsson, Supervisor

Rachel Ward, Co-supervisor

Joseph Kileel

George Biros

Yuji Nakatsukasa

Acknowledgments

Words are insufficient to convey my appreciation to my advisors, Prof. Per-Gunnar Martinsson and Prof. Rachel Ward, for their guidance and support throughout my Ph.D. journey. Their knowledge and insights have been lighthouses that navigate me in the fascinating realms of numerical linear algebra and machine learning. They have provided invaluable advice for both my research and my career, beyond the scope of this thesis. They endow me with the horizon to broaden my view and explore different research areas. Their passion and devotion to research encourage me to follow and try pursuing an academic career.

In addition to my advisors, I have had the fortune to collaborate with and learn from some great applied mathematicians and computer scientists during my doctoral research, including Shuo Yang, Prof. Sujay Sanghavi, Prof. Inderjit Dhillon, Prof. Qi Lei, Yuege Xie, Prof. Yuji Nakatsukasa, Kevin Miller, Kate Pearce, and Chao Chen. Works presented in this thesis and beyond could not have been finished without their efforts. More importantly, their meticulousness and diligence motivate me as a researcher; while their erudition helps diversify my sight from various aspects.

In particular, I would like to give my profound thanks to Prof. Qi Lei, who has been an inspiring mentor, as well as a supportive friend, since the beginning of my graduate study, who introduced me to the splendid field of statistical learning theory, and who constantly influences me with her vision and attitude towards research.

Meanwhile, I am sincerely grateful for the constructive suggestions and generous help of Prof. Joseph Kileel, Prof. George Biros, and Prof. Yuji Nakatsukasa, together with my advisors, who kindly serve on my thesis committee. Especially, I would like to thank Prof. Yuji Nakatsukasa for the insightful discussions and guidance on several projects in numerical linear algebra.

I truly appreciate the experience in Prof. Martinsson's and Prof. Ward's research groups, both of which involve brilliant researchers and collaborative environments where I have learned millions. I am thankful to everyone in both groups, especially my collaborators among them:

Yuege Xie, Kevin Miller, Kate Pearce, and Chao Chen; along with Anna Yesypenko, Ke Chen, Bowei Wu, Heather Wilber, Ruhui Jin, Amelia Henriksen, Xiaoxia Wu, and more for their enlightening advice at different stages of my graduate study. It is also my fortune to be a part of the broader Oden community, learning from its incredible variety of research directions while enjoying its inclusiveness.

Despite the fleeting time, my four years as an undergraduate student at Emory University before graduate school were irreplaceable for my professional and personal development. Looking back, I am genuinely grateful to my undergraduate advisors Prof. Effrosyni Seitaridou and Prof. Eric Weeks for patiently unveiling a corner of the research world to a curious undergraduate, as well as for encouraging me to follow my interests while pursuing doctoral study along a different direction. I was also fortunate enough to make some cherished friendships during my undergraduate, among which I am truly thankful to Xiaoyi Zhang whose wisdom and attitude on life have always inspired and motivated me since our paths first crossed.

Finally, I would like to give my wholehearted gratitude to my parents. I owe them for being mostly away from home in the past nine years and probably more years to come, for the absence during their struggles in the COVID-19 pandemic, and everything. Their unconditional love and support have never been diminished by distance.

Randomized Dimension Reduction with Statistical Guarantees

by

Yijun Dong, Ph.D.
The University of Texas at Austin, 2023

Supervisors: Per-Gunnar Martinsson
Rachel Ward

Large models and enormous data are essential driving forces of the unprecedented successes achieved by modern algorithms, especially in scientific computing and machine learning. Nevertheless, the growing dimensionality and model complexity, as well as the non-negligible workload of data pre-processing, also bring formidable costs to such successes in both computation and data aggregation. As the deceleration of Moore's Law slackens the cost reduction of computation from the hardware level, fast heuristics for expensive classical routines and efficient algorithms for exploiting limited data are increasingly indispensable for pushing the limit of algorithm potency. This thesis explores some of such algorithms for fast execution and efficient data utilization.

1. From the **computational efficiency** perspective, we design and analyze fast randomized low-rank decomposition algorithms for large matrices based on "matrix sketching", which can be regarded as a dimension reduction strategy in the data space. These include the **randomized pivoting-based interpolative and CUR decomposition** discussed in Chapter 2 and the **randomized subspace approximations** discussed in Chapter 3.
2. From the **sample efficiency** perspective, we focus on learning algorithms with various incorporations of data augmentation that improve generalization and distributional robustness provably. Specifically, Chapter 4 presents a sample complexity analysis for **data augmentation consistency regularization** where we view sample efficiency from the lens of dimension reduction in the function space. Then in Chapter 5, we introduce an **adaptively weighted data augmentation consistency regularization** algorithm for distributionally robust optimization with applications in medical image segmentation.

Table of Contents

List of Tables

List of Figures

Chapter 1

Overview

1.1 Computational Efficiency: Randomized Low-rank Decompositions

Low-rank decompositions are dimension reduction techniques that unveil latent low-dimensional structures in large matrices, which are ubiquitous in various applications like principal component analysis and spectral clustering. However, to compute common matrix decompositions (*e.g.*, SVD) of an $m \times n$ matrix, classical deterministic algorithms generally scale as $O(mn^2)$, making them untenable for large-scale problems.

As a remedy, the "matrix sketching" framework [68] embeds high-dimensional matrices into random low-dimensional subspaces via fast linear transforms, commonly known as randomized linear embeddings or (fast) Johnson-Lindenstrauss transforms. Some popular choices include Gaussian random matrices [78], subsampled random trigonometric transforms [166], and sparse embeddings like count sketch [103] and sparse sign matrices [30]. After such dimension reduction through randomized linear embedding, classical matrix decomposition algorithms can be executed efficiently, and low-rank approximations can be reconstructed without much compromise in accuracy.

Chapter 2 and Chapter 3 of this thesis explore the potency of randomization with "matrix sketching" in two classical low-rank decomposition problems, namely the matrix skeletonization and the randomized subspace approximation.

1.1.1 Randomized Pivoting-based Matrix Skeletonization

Given a matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$, the matrix skeletonization problem (*i.e.*, interpolative decomposition (ID) and CUR decomposition) solves for low-rank "natural bases" formed by the original columns (and/or rows) of $\mathbf{A}$. Precisely, the goal is to identify column (or row) skeletons $\mathbf{C} = \mathbf{A}\left(:, J_s\right) \in \mathbb{R}^{m \times k}$ (or $\mathbf{R} = \mathbf{A}\left(I_s, :\right) \in \mathbb{R}^{k \times n}$, in MATLAB notation) indexed by $J_s \subset [n]$ (or $I_s \subset [m]$ where $|J_s| = |I_s| = k$) that serve as good bases,

$$\left\|\mathbf{A} - \mathbf{C}\mathbf{C}^{\dagger}\mathbf{A}\right\|_F \leq (1+\epsilon) \min_{J_s \subset [n]} \left\|\mathbf{A} - \mathbf{A}\left(:, J_s\right) \mathbf{A}\left(:, J_s\right)^{\dagger} \mathbf{A}\right\|_F .$$

Despite the NP-hardness [29] of identifying the nearly optimal skeleton selections like the row/column subset with the maximum spanning volume [61], there exists fast heuristics [40, 96, 137, 153] that enjoys statistical guarantees and/or practical successes. In particular, randomization via "matrix sketching" plays a critical role in many of these heuristics [25, 45, 153].

Randomized pivoting-based matrix skeletonization [45, 137, 153] is a class of such fast heuristics widely used in scientific computing, whose general framework consists of two stages:

(i) dimension reduction via sketching (*e.g.*, constructing row sketch $\mathbf{X} = \mathbf{\Gamma A} \in \mathbb{R}^{l \times n}$ via a Gaussian random matrix $\mathbf{\Gamma}$ with *i.i.d.* entries $\Gamma_{ij} \sim \mathcal{N}\left(0, l^{-1}\right)$ and $l \ll m$) and

(ii) greedy skeleton selection via pivoting on the reduced matrix sketch (*e.g.*, applying LU with partial pivoting on $\mathbf{X}^\top$ and taking the first $|J_s|$ pivots as the column skeletonizations).

In Chapter 2, we first surveyed and compared different options for the two stages, *e.g.*, sketching/randomized SVD [68] for the dimension reduction stage and column pivoted QR (CPQR)/LU with partial pivoting (LUPP) for the pivoting stage. Motivated by the systematic comparison, we then proposed a novel combination of sketching and LU with partial pivoting (LUPP) for efficient randomized matrix skeletonization. Compared to column pivoted QR (CPQR) commonly used in the existing algorithms, LUPP enjoys superior empirical efficiency and parallelizability [62, 136] while compromising the rank-revealing guarantees. Fortunately, for matrix skeletonization, such a trade-off between efficiency and rank-revealing property can be avoided via sketching. In particular, we demonstrated that, instead of relying on rank-revealing properties of the pivoting scheme, *the simple combination of sketching and LUPP exploits the spectrum-preserving capability of sketching and achieves considerable acceleration without compromising accuracy*.

1.1.2 Randomized Subspace Approximations

Theoretical underpinnings of randomized subspace approximation is another key aspect of analyzing randomized low-rank decompositions. A low-rank decomposition can be viewed as a bilinear combination of the associated low-rank bases for the column and row spaces that encapsulate key information for a wide range of tasks (*e.g.*, canonical component analysis and leverage score sampling). Specifically, for a truncated SVD

$$\underset{m\times n}{\mathbf{A}} \approx \mathbf{A}_k = \underset{m\times k}{\mathbf{U}_k}\ \underset{k\times k}{\mathbf{\Sigma}_k}\ \underset{k\times n}{\mathbf{V}_k^\top}, \quad \mathbf{U}_k^\top \mathbf{U}_k = \mathbf{V}_k^\top \mathbf{V}_k = \mathbf{I}_k, \quad \mathbf{\Sigma}_k = \mathrm{diag}\left(\sigma_1, \dots, \sigma_k\right),$$

the left and right leading singular vectors $\mathbf{U}_k$ and $\mathbf{V}_k$ can be approximated efficiently via randomized subspace approximations. In the basic version, randomized subspace approximations leverage the appealing property of randomized linear embeddings (*e.g.*, a Gaussian embedding $\mathbf{\Omega} \in \mathbb{R}^{n\times l}$ with *i.i.d.* entries $\Omega_{ij} \sim \mathcal{N}(0, l^{-1})$) that, with moderate oversampling $l = k + O(1)$, sketching (with q power iterations) $\mathbf{Y} = (\mathbf{A}\mathbf{A}^\top)^q \mathbf{A}\mathbf{\Omega}$ captures the leading singular vectors $\mathbf{U}_k$ with high probability. Under orthonormalization at each iteration (commonly known as randomized subspace iteration [68, Algorithm 4.4]), $\mathrm{Range}\left(\mathbf{Y}\right)$ provides a numerical stable estimate for the leading singular subspace.

Canonical angles [59] (formally defined in Definition 3.1) are commonly used to quantify the difference between two subspaces of the same space, which provides natural error measures for the randomized subspace approximations. For instance, given arbitrary full-rank matrices $\mathbf{U} \in \mathbb{R}^{d\times l}$ and $\mathbf{V} \in \mathbb{R}^{d\times k}$ (assuming $k \le l \le d$ without loss of generality), the k canonical angles $\angle\left(\mathbf{U}, \mathbf{V}\right) \triangleq \angle\left(\mathrm{Range}(\mathbf{U}), \mathrm{Range}(\mathbf{V})\right)$ between their corresponding range subspaces in $\mathbb{R}^d$ are given by the spectra of $\mathbf{Q}_\mathbf{U}^\top \mathbf{Q}_\mathbf{V} \in \mathbb{R}^{l\times k}$ where the columns of $\mathbf{Q}_\mathbf{U} \in \mathbb{R}^{d\times l}$ and $\mathbf{Q}_\mathbf{V} \in \mathbb{R}^{d\times k}$ consist of orthonormal bases of $\mathbf{U}$ and $\mathbf{V}$, respectively. Precisely, for each $i \in [k]$, $\cos\angle_i\left(\mathbf{U}, \mathbf{V}\right) = \sigma_i\left(\mathbf{Q}_\mathbf{U}^\top \mathbf{Q}_\mathbf{V}\right)$, or equivalently, $\sin\angle_i\left(\mathbf{U}, \mathbf{V}\right) = \sigma_{k-i+1}\left(\left(\mathbf{I} - \mathbf{Q}_\mathbf{U}\mathbf{Q}_\mathbf{U}^\top\right)\mathbf{Q}_\mathbf{V}\right)$ (*cf.* [183] Section 3).

In Chapter 3, we extended the existing analysis on the accuracy of singular vectors approximated by the randomized subspace iteration, in terms of the canonical angles $\angle\left(\mathbf{U}_k, \mathbf{Y}\right)$ between the true and approximated leading singular subspaces $\mathrm{Range}\left(\mathbf{U}_k\right)$ and $\mathrm{Range}\left(\mathbf{Y}\right)$. By casting a computational efficiency view on the bounds and estimates of canonical angles, we provided *a set of prior probabilistic bounds that is not only asymptotically tight but also computable in linear time*. Moreover, we derived *unbiased prior estimates*, along with *residual-based posterior bounds*, of canonical angles that can be evaluated efficiently, while further demonstrating the empirical effectiveness of these bounds and estimates with numerical evidence.

1.2 Sample Efficiency: Data Augmentation for Better Generalization

Modern machine learning models, especially deep learning models, require substantially large amounts of samples for training. However, data collection and human annotation often come with non-negligible costs in practice. Therefore, *sample efficiency* and *generalization* are critical properties of learning algorithms. In the most basic setting, a learning algorithm is designed for recovering some unknown ground truths (*e.g.*, descriptions of images) via sampling from some unknown distributions (*e.g.*, images from the Internet with descriptions). The goal is to learn a prediction function (*e.g.*, image captioning) that well approximates the ground truth by providing accurate predictions beyond the training samples (*e.g.*, (in-distribution) generalization to unseen testing samples from the same distribution or out-of-distribution generalization to testing samples from related but different distributions), with as few training samples as possible (*i.e.*, sample efficiency).

Since the seminal work [84], *data augmentation* has been a ubiquitous ingredient in many state-of-the-art machine learning algorithms [34, 71, 85, 132, 133]. It started from simple transformations on samples (*e.g.*, (random) perturbations, distortions, scales, crops, rotations, and horizontal flips on images) that roughly preserve the semantic information. More sophisticated variants were subsequently designed; a non-exhaustive list includes Mixup [177], Cutout [43], and Cutmix [175]. Despite the known capability of improving generalization and sample efficiency empirically, the theoretical understanding of how data augmentation works remain limited due to the wide variety of domain-specific designs [125, 177] and algorithmic choices of utilizing data augmentations [70, 84, 135]. The classical wisdom interprets and leverages data augmentation as a natural expansion of the original training samples [34, 71, 84, 132, 133]. However, simply enlargening the training set is not sufficient to explain the unprecedented successes (in comparison to the classical wisdom) recently achieved by an alternative line of data-augmentation-based learning algorithms — *data augmentation consistency regularization* [5, 13, 88, 124, 135] — that encourages similar predictions among the original sample and its augmentations.

Chapter 4 and Chapter 5 of this thesis discuss the sample efficiency of data augmentation consistency regularization from the dimension reduction aspect, along with an application in medical image segmentation.

1.2.1 Sample Efficiency of Data Augmentation Consistency Regularization

In efforts to interpret the effect of different algorithmic choices on utilizing data augmentations, Chapter 4 conducts *apple-to-apple comparisons* between two popular data-augmentation-based algorithms — the empirical risk minimization on the augmented training set (DA-ERM) and the data augmentation consistency regularization (DAC) — in the *supervised* setting.

Concretely, given a ground truth distribution $P : \mathcal{X} \times \mathcal{Y} \to [0, 1]$ and a well-specified function class $\mathcal{H} \subseteq \{h = f_h \circ \phi_h \mid \phi_h : \mathcal{X} \to \mathcal{W},\ f_h : \mathcal{W} \to \mathcal{Y}\}$ (*e.g.*, a class of neural networks with a sufficiently expressive hidden-layer representation function $\phi_h(\cdot)$) such that for a given loss function $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}_{\geq 0}$, $h^* \triangleq \operatorname{argmin}_{h \in \mathcal{H}} \mathbb{E}_{(\mathbf{x},y)\sim P}\left[\ell\left(h(\mathbf{x}), y\right)\right] \in \mathcal{H}$, let $(\mathbf{x}_i, y_i)_{i\in[N]} \sim P(\mathbf{x}, y)^N$ be a set of N training samples drawn *i.i.d.* from P and

$$\widetilde{\mathcal{A}}(\mathbf{X}) = [\mathbf{x}_1; \cdots; \mathbf{x}_N; \mathbf{x}_{1,1}; \cdots; \mathbf{x}_{N,1}; \cdots; \mathbf{x}_{1,\alpha}; \cdots; \mathbf{x}_{N,\alpha}] \in \mathcal{X}^{(1+\alpha)N}$$

be the features of its (random) augmentation (*i.e.*, $\alpha \in \mathbb{N}$ additional augmentations per sample). Considering the basic version of data augmentation which preserves the labels of original samples, DA-ERM directly includes the augmented samples in the training set and learns via empirical risk minimization (ERM),

$$\widehat{h}^{da-erm} = \operatorname*{argmin}_{h\in\mathcal{H}} \sum_{i=1}^{N} \ell(h(\mathbf{x}_i), y_i) + \sum_{i=1}^{N}\sum_{j=1}^{\alpha} \ell(h(\mathbf{x}_{i,j}), y_i).$$

Instead, DAC regularization rewards $h \in \mathcal{H}$ that provides similar representations ϕ_h among data augmentations of the same sample,

$$\widehat{h}^{dac} = \operatorname*{argmin}_{h\in\mathcal{H}} \sum_{i=1}^{N} l(h(\mathbf{x}_i), y_i) + \underbrace{\lambda \sum_{i=1}^{N}\sum_{j=1}^{\alpha} \varrho\left(\phi_h(\mathbf{x}_i), \phi_h(\mathbf{x}_{i,j})\right)}_{\textit{DAC regularization}},$$

where $\varrho : \mathcal{W} \times \mathcal{W} \to \mathbb{R}_{\geq 0}$ is a metric associated with the metric space $\mathcal{W}$ (*e.g.*, the Euclidean distance).

Previous works [16, 27, 102] generally view augmentations as groups endowed with Haar measures which inevitably assumed access to augmentations over the population. By contrast, we investigate a more realistic circumstance where both the training data and their augmentations are presented as finite random samples, $(\mathbf{X}, \mathbf{y}) \sim P(\mathbf{x}, y)^N$ and $\widetilde{\mathcal{A}}(\mathbf{X}) \in \mathcal{X}^{(1+\alpha)N}$, by considering the DAC regularization as a reduction in the complexity of the function class $\mathcal{H}$ (*e.g.*, a dimension reduction in the linear regression setting). Apart from the well-known semi-supervised learning capability of DAC, in Chapter 4, we demonstrated the *intrinsic efficiency of DAC over DA-ERM in utilizing both samples and their augmentations*, even without unlabeled data, by establishing separations of sample complexities between DAC and DA-ERM in various settings, including different function classes (*e.g.*, linear regression and neural networks), different data augmentations, as well as in-distribution and out-of-distribution generalization.

1.2.2 Adaptively Weighted Data Augmentation Consistency Regularization

Grounding the theoretical insight on data augmentation consistency regularization as guidance for algorithm design, in Chapter 5, we explore the *combination of data augmentation consistency regularization* and *sample reweighting* for a distributionally robust optimization setting commonly encountered in *medical image segmentation*.

Concept shift is a prevailing problem in natural tasks like medical image segmentation where samples usually come from different subpopulations with variant correlations between features and labels. A common type of concept shift in medical image segmentation is the "information imbalance" between the *label-sparse* samples with few (if any) segmentation labels and the *label-dense* ones with plentiful labeled pixels. Existing distributionally robust algorithms have focused on adaptively truncating/down-weighting the "less informative" (*i.e.*, label-sparse) samples. To exploit data features of label-sparse samples more efficiently, in Chapter 5, we propose an adaptively weighted online optimization algorithm — *AdaWAC*— to incorporate data augmentation consistency regularization in sample reweighting. As a simplified overview, when learning from a function class parameterized by $\theta \in \Theta$, *AdaWAC* introduces a set of trainable weights $\boldsymbol{\beta} \in [0,1]^n$ to balance the supervised (cross-entropy) loss $\ell_{CE}\left(\theta; (\mathbf{x}_i, \mathbf{y}_i)\right)$ and unsupervised consistency regularization $\ell_{AC}\left(\theta; \mathbf{x}_i, A_{i,1}, A_{i,2}\right)$[1] of each sample $(\mathbf{x}_i, \mathbf{y}_i)$ separately:

$$\min_{\theta \in \Theta} \max_{\boldsymbol{\beta} \in [0,1]^n} \frac{1}{n} \sum_{i=1}^{n} \beta_i \cdot \ell_{CE}\left(\theta; (\mathbf{x}_i, \mathbf{y}_i)\right) + (1-\beta_i) \cdot \ell_{AC}\left(\theta; \mathbf{x}_i, A_{i,1}, A_{i,2}\right).$$

At the saddle point $\left(\widehat{\theta}, \widehat{\boldsymbol{\beta}}\right)$ of the underlying objective, the weights assign label-dense samples to the supervised loss (*i.e.*, $\widehat{\beta}_i = 1$) and label-sparse samples to the unsupervised consistency regularization (*i.e.*, $\widehat{\beta}_i = 0$). We provide a convergence guarantee by recasting the optimization

[1] Here, $A_{i,1}$ and $A_{i,2}$ denote the (random) augmentations associated with $\mathbf{x}_i$ for each $i \in [n]$ (refer to Section 5.2 for formal definitions).

as online mirror descent on a saddle point problem. Our empirical results further demonstrate that *AdaWAC* not only enhances the segmentation performance and sample efficiency but also improves the robustness to concept shift on various medical image segmentation tasks with different UNet-style backbones.

Overall, this thesis is based on my works [45, 46, 48, 173]. Beyond the score of this thesis, some related topics that I have been working on include knowledge distillation [47] and blockwise adaptive sampling [44]

Chapter 2

Randomized Pivoting Algorithms for CUR and Interpolative Decompositions

Abstract

Matrix skeletonizations like the interpolative and CUR decompositions provide a framework for low-rank approximation in which subsets of a given matrix's columns and/or rows are selected to form approximate spanning sets for its column and/or row space. Such decompositions that rely on "natural" bases have several advantages over traditional low-rank decompositions with orthonormal bases, including preserving properties like sparsity or non-negativity, maintaining semantic information in data, and reducing storage requirements. Matrix skeletonizations can be computed using classical deterministic algorithms such as column-pivoted QR, which work well for small-scale problems in practice, but suffer from slow execution as the dimension increases and can be vulnerable to adversarial inputs. More recently, randomized pivoting schemes have attracted much attention, as they have proven capable of accelerating practical speed, scale well with dimensionality, and sometimes also lead to better theoretical guarantees. This manuscript provides a comparative study of various randomized pivoting-based matrix skeletonization algorithms that leverage classical pivoting schemes as building blocks. We propose a general framework that encapsulates the common structure of these randomized pivoting-based algorithms and provides an a-posteriori-estimable error bound for the framework. Additionally, we propose a novel concretization of the general framework and numerically demonstrate its superior empirical efficiency.[1]

2.1 Introduction

The problem of computing a low-rank approximation to a matrix is a classical one that has drawn increasing attention due to its importance in the analysis of large data sets. At the core of low-rank matrix approximation is the task of constructing bases that approximately

[1] This chapter is based on the following published journal paper:
Yijun Dong and Per-Gunnar Martinsson. Simpler is better: a comparative study of randomized pivoting algorithms for cur and interpolative decompositions. *Advances in Computational Mathematics*, 49(4):66, 2023 [45].

span the column and/or row spaces of a given matrix. This manuscript investigates algorithms for low-rank matrix approximations with “natural bases” of the column and row spaces – bases formed by selecting subsets of the actual columns and rows of the matrix. To be precise, given an $m \times n$ matrix $\mathbf{A}$ and a target rank $k < \min(m, n)$, we seek to determine an $m \times k$ matrix $\mathbf{C}$ holding k of the columns of $\mathbf{A}$, and a $k \times n$ matrix $\mathbf{Z}$ such that

$$\underset{m \times n}{\mathbf{A}} \approx \underset{m \times k}{\mathbf{C}} \; \underset{k \times n}{\mathbf{Z}}. \tag{2.1}$$

We let $J_{\rm s}$ denote the index vector of length k that identifies the k chosen columns, so that, in MATLAB notation,

$$\mathbf{C} = \mathbf{A}(:\,, J_{\rm s}). \tag{2.2}$$

If we additionally identify an index vector $I_{\rm s}$ that marks a subset of the rows that forms an approximate basis for the row space of $\mathbf{A}$, we can then form the “CUR” decomposition

$$\underset{m \times n}{\mathbf{A}} \approx \underset{m \times k}{\mathbf{C}} \; \underset{k \times k}{\mathbf{U}} \; \underset{k \times n}{\mathbf{R}}, \tag{2.3}$$

where $\mathbf{U}$ is a $k \times k$ matrix, and

$$\mathbf{R} = \mathbf{A}(I_{\rm s}, :\,). \tag{2.4}$$

The decomposition (2.3) is also known as a “matrix skeleton” [61] approximation (hence the subscript “s” for “skeleton” in $I_{\rm s}$ and $J_{\rm s}$). Matrix decompositions of the form (2.1) or (2.3) possess several compelling properties: (i) Identifying $I_{\rm s}$ and/or $J_{\rm s}$ is often helpful in data interpretation. (ii) The decompositions (2.1) and (2.3) preserve important properties of the matrix $\mathbf{A}$. For instance, if $\mathbf{A}$ is sparse/non-negative, then $\mathbf{C}$ and $\mathbf{R}$ are also sparse/non-negative. (iii) The decompositions (2.1) and (2.3) are often memory efficient. In particular, when the entries of $\mathbf{A}$ itself are available, or can inexpensively be computed or retrieved, then once $J_{\rm s}$ and $I_{\rm s}$ have been determined, there is no need to store $\mathbf{C}$ and $\mathbf{R}$ explicitly.

Deterministic techniques for identifying close-to-optimal index vectors $I_{\rm s}$ and $J_{\rm s}$ are well established. Greedy algorithms such as the classical column pivoted QR (CPQR) [59, Sec. 5.4.1], and variations of LU with complete pivoting [83, 144] often work well in practice. There also exist specialized pivoting schemes that come with strong theoretical performance guarantees [65].

While effective for smaller dense matrices, classical techniques based on pivoting become computationally inefficient as the matrix sizes grow. The difficulty is that a global update to the matrix is in general required before the next pivot element can be selected. The situation becomes particularly dire for sparse matrices, as each update tends to create substantial fill-in of zero entries.

To better handle large matrices, and huge sparse matrices in particular, a number of algorithms based on *randomized sketching* have been proposed in recent years. The idea is to extract a "sketch" $\mathbf{X}$ of the matrix that is far smaller than the original matrix, yet contains enough information that the index vectors $I_{\rm s}$ and/or $J_{\rm s}$ can be determined by using the information in $\mathbf{X}$ alone. Examples include:

1. *Discrete empirical interpolation method (DEIM):* The sketching step consists of computing approximations to the dominant left and right singular vectors of $\mathbf{A}$, for instance using the randomized SVD (RSVD) [68, 92]. Then a greedy pivoting-based scheme is used to pick the index sets $I_{\rm s}$ and $J_{\rm s}$ [51, 137].
2. *Leverage score sampling:* Again, the procedures start by computing approximations to the dominant left and right singular vectors of $\mathbf{A}$ through a randomized scheme. Then these approximations are used to compute probability distributions on the row and/or column indices, from which a random subset of columns and/or rows is sampled.
3. *Pivoting on a random sketch:* With a random matrix $\mathbf{\Gamma} \in \mathbb{R}^{k \times m}$ drawn from some appropriate distribution, a sketch of $\mathbf{A}$ is formed via $\mathbf{X} = \mathbf{\Gamma}\mathbf{A}$. Then, a classical pivoting strategy such as the CPQR is applied on $\mathbf{X}$ to identify a spanning set of columns.

The existing literature [4, 25, 31, 39, 40, 49, 51, 65, 96, 137, 153] presents compelling evidence in support of each of these frameworks, in the form of mathematical theory and/or empirical numerical experiments.

The objective of the present manuscript is to organize different strategies and to conduct a systematic comparison, with a focus on their empirical accuracy and efficiency. In particular, we compare different strategies for extracting a random sketch, such as techniques based on Gaussian random matrices [68, 78, 100, 165], random fast transforms [19, 68, 100, 118, 146, 166], and random sparse embeddings [30, 100, 103, 111, 147, 165]. We also compare different pivoting strategies such as pivoted QR [59, 153] versus pivoted LU [25, 137]. Finally, we compare how well sampling-based schemes perform in relation to pivoting-based schemes.

In addition to providing a comparison of existing methods, the manuscript proposes a general framework that encapsulates the common structure shared by some popular randomized pivoting-based algorithms and presents an a-posteriori-estimable error bound for the framework. Moreover, the manuscript introduces a novel concretization of the general framework that is faster in execution than the schemes of [137, 153] while picking equally close-to-optimal skeletons in practice. In its most basic version, our simplified method for finding a subset of k columns of $\mathbf{A}$ works as follows:

Sketching step: Draw $\mathbf{\Gamma} \in \mathbb{R}^{k\times m}$ from a Gaussian distribution and form $\mathbf{X} = \mathbf{\Gamma}\mathbf{A}$.

Pivoting step: Perform a *partially pivoted* LU decomposition of $\mathbf{X}^\top \in \mathbb{R}^{n\times k}$. Collect the chosen pivot indices in the index vector J_s. (Since partially pivoted LU picks rows of $\mathbf{X}^*$, J_s indicates columns of $\mathbf{X}$.)

What is particularly interesting about this process is that while the LU factorization with partial pivoting (LUPP) is *not* rank revealing for a general matrix $\mathbf{A}$, the randomized mixing done in the sketching step makes LUPP excel at picking spanning columns. Furthermore, the randomness introduced by sketching empirically serves as a remedy for the vulnerability of classical pivoting schemes like LUPP to adversarial inputs (*e.g.*, the Kahan matrix [79]). The scheme can be accelerated further by incorporating a structured random embedding $\mathbf{\Gamma}$. Alternatively, its accuracy can be enhanced by incorporating one of two steps of power iteration when building the sample matrix $\mathbf{X}$.

The manuscript is organized as follows: Section 2.2 provides a brief overview of the interpolative and CUR decompositions (Section 2.2.2), along with some essential building blocks of the randomized pivoting algorithms, including randomized linear embeddings (Section 2.2.3), randomized low-rank SVD (Section 2.2.4), and matrix decompositions with pivoting (Section 2.2.5). Section 2.3 reviews existing algorithms for matrix skeletonizations (Section 2.3.1, Section 2.3.2), and introduces a general framework that encapsulates the structures of some randomized pivoting-based algorithms. In Section 2.4, we propose a novel concretization of the general framework, and provide an a-posteriori-estimable bound for the associated low-rank approximation error. With the numerical results in Section 2.5, we first compare the efficiency of various choices for the two building blocks in the general framework: randomized linear embeddings (Section 2.5.1) and matrix decompositions with pivoting (Section 2.5.2). Then, we demonstrate the empirical advantages of the proposed algorithm by investigating the accuracy and efficiency of assorted randomized skeleton selection algorithms for the CUR decomposition (Section 2.5.3).

2.2 Background

We first introduce some closely related low-rank matrix decompositions that rely on "natural" bases, including the CUR decomposition, and the column, row, and two-sided interpolative decompositions (ID) in Section 2.2.2. Section 2.2.3 describes techniques for computing randomized sketches of matrices, based on which Section 2.2.4 discusses the randomized construction of low-rank SVD. Section 2.2.5 describes how these can be used to construct matrix decompositions. While introducing the background, we include proofs of some well-established facts that provide key ideas but are hard to extract from the context of relevant references.

2.2.1 Notation

Let $\mathbf{A} \in \mathbb{R}^{m \times n}$ be an arbitrary given matrix of rank $r \leq \min\{m, n\}$, whose SVD is given by

$$\mathbf{A} = \underset{m\times r}{\mathbf{U}_A} \underset{r\times r}{\mathbf{\Sigma}_A} \underset{r\times n}{\mathbf{V}_A^\top} = \left[\mathbf{u}_{A,1}, \dots, \mathbf{u}_{A,r}\right] \mathrm{diag}\left(\sigma_{A,1}, \dots, \sigma_{A,r}\right) \left[\mathbf{v}_{A,1}, \dots, \mathbf{v}_{A,r}\right]^\top$$

such that for any rank parameter $k \leq r$, we denote $\mathbf{U}_{A,k} \triangleq [\mathbf{u}_{A,1}, \dots, \mathbf{u}_{A,k}]$ and $\mathbf{V}_{A,k} \triangleq [\mathbf{v}_{A,1}, \dots, \mathbf{v}_{A,k}]$ as the orthonormal bases of the dimension-k leading left and right singular subspaces of $\mathbf{A}$, while $\mathbf{U}_{A,k}^\perp \triangleq [\mathbf{u}_{A,k+1}, \dots, \mathbf{u}_{A,r}]$ and $\mathbf{V}_{A,k}^\perp \triangleq [\mathbf{v}_{A,k+1}, \dots, \mathbf{v}_{A,r}]$ as the orthonormal bases of the respective orthogonal complements. The diagonal submatrices consisting of the spectrum, $\mathbf{\Sigma}_{A,k} \triangleq \mathrm{diag}\left(\sigma_{A,1}, \dots, \sigma_{A,k}\right)$ and $\mathbf{\Sigma}_{A,k}^\perp \triangleq \mathrm{diag}\left(\sigma_{A,k+1}, \dots, \sigma_{A,r}\right)$, follow analogously. We denote $\mathbf{A}_k \triangleq \mathbf{U}_{A,k}\mathbf{\Sigma}_{A,k}\mathbf{V}_{A,k}^\top$ as the rank-k truncated SVD that minimizes rank-k approximation error of $\mathbf{A}$ [55]. Furthermore, we denote the spectrum of $\mathbf{A}$, $\sigma\left(\mathbf{A}\right)$, as a $r \times r$ diagonal matrix, while for each $i = 1, \dots, r$, let $\sigma_i(\mathbf{A})$ be the i-th singular value of $\mathbf{A}$.

For the QR factorization, given an arbitrary rectangular matrix $\mathbf{M} \in \mathbb{R}^{d \times l}$ with full column rank ($d \geq l$), let $\mathbf{M} = \left[\mathbf{Q}_M, \mathbf{Q}_M^\perp\right] [\mathbf{R}_M; \mathbf{0}]$ be a full QR factorization of $\mathbf{M}$ such that $\mathbf{Q}_M \in \mathbb{R}^{d \times l}$ and $\mathbf{Q}_M^\perp \in \mathbb{R}^{d \times (d-l)}$ consist of orthonormal bases of the subspace spanned by the columns of $\mathbf{M}$ and its orthogonal complement. We denote $\mathrm{ortho} : \left\{\mathbf{M} \in \mathbb{R}^{d \times l} \,\middle|\, \mathrm{rank}\left(\mathbf{M}\right) = l\right\} \to \mathbb{R}^{d \times l}$ ($d \geq l$) as a map that identifies an orthonormal basis (not necessarily unique) for $\mathbf{M}$, $\mathrm{ortho}(\mathbf{M}) = \mathbf{Q}_M$.

We adopt MATLAB notation for matrices throughout this work. Unless specified otherwise (*e.g.*, with subscripts), we use $\|\cdot\|$ to represent either the spectral norm or the Frobenius norm (*i.e.*, holding simultaneously for both norms).

2.2.2 Interpolative and CUR decompositions

We first recall the definitions of the interpolative and CUR decompositions of a given $m \times n$ real matrix $\mathbf{A}$. After providing the basic definitions, we discuss first how well it is theoretically possible to do low-rank approximation under the constraint that natural bases must be used. We then briefly describe the further suboptimality incurred by standard algorithms.

Basic definitions

We consider low-rank approximations for $\mathbf{A}$ with column and/or row subsets as bases. Given an arbitrary linearly independent column subset $\mathbf{C} = \mathbf{A}\left(:, J_s\right)$ ($J_s \subset [n]$), the rank-$|J_s|$ column ID of $\mathbf{A}$ with respect to column skeletons J_s can be formulated as,

$$\widehat{\mathbf{A}}_{*,J_s} \triangleq \mathbf{C}\mathbf{C}^{\dagger}\mathbf{A}, \tag{2.5}$$

where $\mathbf{C}\mathbf{C}^{\dagger}$ is the orthogonal projector onto the spanning subspace of column skeletons. Analogously, given any linearly independent row subset $\mathbf{R} = \mathbf{A}\left(I_s, :\right)$ ($I_s \subset [m]$), the rank-$|I_s|$ column ID of $\mathbf{A}$ with respect to row skeletons I_s takes the form

$$\widehat{\mathbf{A}}_{I_s,*} \triangleq \mathbf{A}\mathbf{R}^{\dagger}\mathbf{R}, \tag{2.6}$$

where $\mathbf{R}^{\dagger}\mathbf{R}$ is the orthogonal projector onto the span of row skeletons. While with both column and row skeletons, we can construct low-rank approximations for $\mathbf{A}$ in two forms – the two-sided ID and CUR decomposition: with $|I_s| = |J_s|$, let $\mathbf{S} \triangleq \mathbf{A}\left(I_s, J_s\right)$ be an invertible two-sided skeleton of $\mathbf{A}$ such that

$$\text{Two-sided ID:} \qquad \widehat{\mathbf{A}}_{I_s,J_s} \triangleq \left(\mathbf{C}\mathbf{S}^{-1}\right)\mathbf{S}\left(\mathbf{C}^{\dagger}\mathbf{A}\right) \tag{2.7}$$

$$\text{CUR decomposition:} \qquad \widetilde{\mathbf{A}}_{I_s,J_s} \triangleq \mathbf{C}\left(\mathbf{C}^{\dagger}\mathbf{A}\mathbf{R}^{\dagger}\right)\mathbf{R} \tag{2.8}$$

where in the exact arithmetic, since $\mathbf{S}^{-1}\mathbf{S} = \mathbf{I}$, the two-sided ID is equivalent to the column ID characterized by $\mathbf{C}$, *i.e.*, $\widehat{\mathbf{A}}_{I_s,J_s} = \widehat{\mathbf{A}}_{*,J_s}$. Nevertheless, the two-sided ID $\widehat{\mathbf{A}}_{I_s,J_s}$ and CUR decomposition $\widetilde{\mathbf{A}}_{I_s,J_s}$ differ in both suboptimality and conditioning.

Remark 2.1 (Suboptimality of ID versus CUR). For any given column and row skeletons $\mathbf{C}$ and $\mathbf{R}$,

$$\left\|\mathbf{A} - \mathbf{C}\mathbf{C}^{\dagger}\mathbf{A}\right\| \leq \left\|\mathbf{A} - \mathbf{C}\mathbf{C}^{\dagger}\mathbf{A}\mathbf{R}^{\dagger}\mathbf{R}\right\| \leq \left(\left\|\mathbf{A} - \mathbf{C}\mathbf{C}^{\dagger}\mathbf{A}\right\|^2 + \left\|\mathbf{A} - \mathbf{A}\mathbf{R}^{\dagger}\mathbf{R}\right\|^2\right)^{\frac{1}{2}}. \tag{2.9}$$

Rationale for Remark 2.1. We observe the simple orthogonal decomposition

$$\mathbf{A} - \mathbf{C}\mathbf{C}^\dagger\mathbf{A}\mathbf{R}^\dagger\mathbf{R} = \left(\mathbf{I}_m - \mathbf{C}\mathbf{C}^\dagger\right)\mathbf{A} + \mathbf{C}\mathbf{C}^\dagger\left(\mathbf{A} - \mathbf{A}\mathbf{R}^\dagger\mathbf{R}\right)$$

where $\left(\mathbf{I}_m - \mathbf{C}\mathbf{C}^\dagger\right)$ and $\mathbf{C}\mathbf{C}^\dagger$ are orthogonal projectors. With the Frobenius norm,

$$\begin{aligned}\left\|\mathbf{A} - \mathbf{C}\mathbf{C}^\dagger\mathbf{A}\mathbf{R}^\dagger\mathbf{R}\right\|_F^2 &= \left\|\mathbf{A} - \mathbf{C}\mathbf{C}^\dagger\mathbf{A}\right\|_F^2 + \left\|\mathbf{C}\mathbf{C}^\dagger\left(\mathbf{A} - \mathbf{A}\mathbf{R}^\dagger\mathbf{R}\right)\right\|_F^2 \\ &\leq \left\|\mathbf{A} - \mathbf{C}\mathbf{C}^\dagger\mathbf{A}\right\|_F^2 + \left\|\mathbf{A} - \mathbf{A}\mathbf{R}^\dagger\mathbf{R}\right\|_F^2,\end{aligned}$$

while with the spectral norm

$$\left\|\mathbf{A} - \mathbf{C}\mathbf{C}^\dagger\mathbf{A}\mathbf{R}^\dagger\mathbf{R}\right\|_2^2 = \max_{\|\mathbf{v}\|_2\leq 1} \left\|\left(\mathbf{A} - \mathbf{C}\mathbf{C}^\dagger\mathbf{A}\right)\mathbf{v}\right\|_2^2 + \left\|\mathbf{C}\mathbf{C}^\dagger\left(\mathbf{A} - \mathbf{A}\mathbf{R}^\dagger\mathbf{R}\right)\mathbf{v}\right\|_2^2$$

where

$$\max_{\|\mathbf{v}\|_2\leq 1} \left\|\left(\mathbf{A} - \mathbf{C}\mathbf{C}^\dagger\mathbf{A}\right)\mathbf{v}\right\|_2^2 = \left\|\mathbf{A} - \mathbf{C}\mathbf{C}^\dagger\mathbf{A}\right\|_2^2$$

and

$$\begin{aligned}&\max_{\|\mathbf{v}\|_2\leq 1} \left\|\left(\mathbf{A} - \mathbf{C}\mathbf{C}^\dagger\mathbf{A}\right)\mathbf{v}\right\|_2^2 + \left\|\mathbf{C}\mathbf{C}^\dagger\left(\mathbf{A} - \mathbf{A}\mathbf{R}^\dagger\mathbf{R}\right)\mathbf{v}\right\|_2^2 \\ \leq &\max_{\|\mathbf{v}\|_2\leq 1} \left\|\left(\mathbf{A} - \mathbf{C}\mathbf{C}^\dagger\mathbf{A}\right)\mathbf{v}\right\|_2^2 + \max_{\|\mathbf{v}\|_2\leq 1} \left\|\mathbf{C}\mathbf{C}^\dagger\left(\mathbf{A} - \mathbf{A}\mathbf{R}^\dagger\mathbf{R}\right)\mathbf{v}\right\|_2^2 \\ \leq &\left\|\mathbf{A} - \mathbf{C}\mathbf{C}^\dagger\mathbf{A}\right\|_2^2 + \left\|\mathbf{A} - \mathbf{A}\mathbf{R}^\dagger\mathbf{R}\right\|_2^2.\end{aligned}$$

■

Remark 2.2 (Conditioning of ID versus CUR). The construction of CUR decomposition tends to be more ill-conditioned than that of two-sided ID. Precisely, for properly selected column and row skeletons J_s and I_s, the corresponding skeletons $\mathbf{S}$, $\mathbf{C}$, and $\mathbf{R}$ share similar spectrum decay as $\mathbf{A}$, which is usually ill-conditioned in the context. In the CUR decomposition, both the bases $\mathbf{C}$, $\mathbf{R}$ and the small matrix $\mathbf{C}^\dagger\mathbf{A}\mathbf{R}^\dagger$ in the middle tend to suffer from large condition numbers as that of $\mathbf{A}$. In contrast, the only potentially ill-conditioned component in the two-sided ID is $\mathbf{S}$ (*i.e.*, despite being expressed in $\mathbf{S}^{-1}$ and $\mathbf{C}^\dagger$, $(\mathbf{C}\mathbf{S}^{-1})$ and $\left(\mathbf{C}^\dagger\mathbf{A}\right)$ in (2.7) are well-conditioned, and can be evaluated without direct inversions).

Remark 2.3 (Stable CUR). Numerically, the stable construction of a CUR decomposition $\widetilde{\mathbf{A}}_{I_s,J_s}$ can be conducted via (unpivoted) QR factorization of $\mathbf{C}$ and $\mathbf{R}$ ([3], Algorithm 2): let $\mathbf{Q}_C \in \mathbb{R}^{m\times|J_s|}$ and $\mathbf{Q}_R \in \mathbb{R}^{n\times|I_s|}$ be matrices from the QR whose columns form orthonormal bases for $\mathbf{C}$ and $\mathbf{R}^\top$, respectively, then

$$\widetilde{\mathbf{A}}_{I_s,J_s} = \mathbf{Q}_C\left(\mathbf{Q}_C^\top\mathbf{A}\mathbf{Q}_R\right)\mathbf{Q}_R^\top. \tag{2.10}$$

Notion of suboptimality

Both interpolative and CUR decompositions share the common goal of identifying proper column and/or row skeletons for $\mathbf{A}$ whose column and/or row spaces are well covered by the respective spans of these skeletons. Without loss of generality, we consider the column skeleton selection problem: for a given rank $k < r$, we aim to find a proper column subset, $\mathbf{C} = \mathbf{A}\left(:, J_s\right)$ ($J_s \subset [n]$, $|J_s| = k$), such that

$$\left\| \mathbf{A} - \widehat{\mathbf{A}}_{*,J_s} \right\| \leq \phi\left(k, m, n\right) \left\| \mathbf{A} - \mathbf{A}_k \right\| \tag{2.11}$$

where common choices of the norm $\left\|\cdot\right\|$ include the spectral norm $\left\|\cdot\right\|_2$ and Frobenius norm $\left\|\cdot\right\|_F$; $\phi\left(k, m, n\right)$ is a function with $\phi\left(k, m, n\right) \geq 1$ for all k, m, n, and depends on the choice of $\left\|\cdot\right\|$; and we recall that $\mathbf{A}_k \triangleq \mathbf{U}_{A,k} \mathbf{\Sigma}_{A,k} \mathbf{V}_{A,k}^\top$ yields the optimal rank-k approximation error. Meanwhile, similar low-rank approximation error bounds are desired for the row ID $\widehat{\mathbf{A}}_{I_s,*}$, two-sided ID $\widehat{\mathbf{A}}_{I_s,J_s}$, and CUR decomposition $\widetilde{\mathbf{A}}_{I_s,J_s}$.

Suboptimality of matrix skeletonization algorithms

The suboptimality of column subset selection, as well as the corresponding ID and CUR decomposition, has been widely studied in a variety of literature.

Specifically, with $\mathbf{S} = \mathbf{A}(I_s, J_s)$ ($|I_s| = |J_s| = k$) being the maximal-volume submatrix in $\mathbf{A}$, the corresponding CUR decomposition (called pseudoskeleton component in the original paper) satisfies (2.11) in $\left\|\cdot\right\|_2$ with $\phi = O\left(\sqrt{k}\left(\sqrt{m} + \sqrt{n}\right)\right)$ [61]. However, it was also pointed out in [61] that skeletons associated with the maximal-volume submatrix are not guaranteed to minimize the low-rank approximation error in (2.11). Moreover, identification of the maximal-volume submatrix is known to be NP-hard [29, 184].

Nevertheless, from the pivoting perspective, the existence of rank-k column IDs with $\phi = \sqrt{1 + k(n-k)}$ can be shown constructively via the strong rank-revealing QR factorization [65], which further provides a polynomial-time relaxation for constructing IDs with $\phi = O\left(\sqrt{k(n-k)}\right)$. From the sampling perspective, the existence of a rank-k column ID with $\phi = \sqrt{(k+1)(n-k)}$ for $\left\|\cdot\right\|_2$ and $\phi = \sqrt{1+k}$ for $\left\|\cdot\right\|_F$ can be shown by upper bounding the expectation of $\left\| \mathbf{A} - \widehat{\mathbf{A}}_{*,J_s} \right\|$ for volume sampling [42]. Later on, polynomial-time algorithms were proposed for selecting such column skeletons [33, 41].

Furthermore, it was unveiled in the recent work [39] that the suboptimality factor $\phi(k, m, n)$ can exhibit a multiple-descent trend with respect to k where depending on spectrum decay,

$\phi(k, m, n)$ can be as tight as $\phi = O\left(k^{1/4}\right)$ for small ks; while for larger ks that fall in certain intervals, $\phi = \Omega(\sqrt{k})$ [39].

2.2.3 Randomized linear embeddings

For a given matrix $\mathbf{A}_k \in \mathbb{R}^{m \times n}$ of rank $k \leq \min(m, n)$ (typically we consider $k \ll \min(m, n)$), and a distortion parameter $\epsilon \in (0, 1)$, a linear map $\mathbf{\Gamma} : \mathbb{R}^m \to \mathbb{R}^l$ (*i.e.*, $\mathbf{\Gamma} \in \mathbb{R}^{l \times m}$, typically we consider $l \ll m$ for embeddings) is called an ℓ_2 *linear embedding* of $\mathbf{A}_k$ with distortion ϵ if

$$(1 - \epsilon)\|\mathbf{A}_k\mathbf{x}\|_2 \leq \|\mathbf{\Gamma}\mathbf{A}_k\mathbf{x}\|_2 \leq (1 + \epsilon)\|\mathbf{A}_k\mathbf{x}\|_2 \quad \forall\, \mathbf{x} \in \mathbb{R}^n. \tag{2.12}$$

A distribution $\mathcal{S}$ over linear maps $\mathbb{R}^m \to \mathbb{R}^l$ (or equivalently, over $\mathbb{R}^{l \times m}$) generates *randomized oblivious ℓ_2 linear embeddings* (abbreviated as *randomized linear embeddings*) if over $\mathbf{\Gamma} \sim \mathcal{S}$, (2.12) holds for all $\mathbf{A}_k$ with at least constant probability. Given $\mathbf{A}_k$ and a randomized linear embedding $\mathbf{\Gamma} \sim \mathcal{S}$, $\mathbf{\Gamma}\mathbf{A}_k$ provides a (row) sketch of $\mathbf{A}_k$, and the process of forming $\mathbf{\Gamma}\mathbf{A}_k$ is known as sketching [100, 165].

Randomized linear embeddings are closely related to various concepts like the Johnson-Lindenstrauss lemma and the restricted isometry property, and are studied in a broad scope of literature. Some popular randomized linear embeddings (*cf.* [100] Section 8, 9) include:

1. Gaussian embeddings: $\mathbf{\Gamma} \in \mathbb{R}^{l \times m}$ with *i.i.d.* Gaussian entries drawn from $\mathcal{N}(0, 1/l)$ [68, 78, 100, 165];
2. subsampled randomized trigonometric transforms (SRTT):
$$\mathbf{\Gamma} = \sqrt{\frac{m}{l}}\mathbf{\Pi}_{m \to l}\mathbf{T}\mathbf{\Phi}\mathbf{\Pi}_{m \to m}$$
where $\mathbf{\Pi}_{m \to l} \in \mathbb{R}^{l \times m}$ is a uniformly random selection of l out of m rows; $\mathbf{T}$ is an $m \times m$ unitary trigonometric transform (*e.g.*, discrete Hartley transform for $\mathbb{R}$, and discrete Fourier transform for $\mathbb{C}$); $\mathbf{\Phi} \triangleq \operatorname{diag}(\varphi_1, \dots, \varphi_m)$ with *i.i.d.* Rademacher random variables $\{\varphi_i\}_{i \in [m]}$ flips signs randomly; and $\mathbf{\Pi}_{m \to m}$ is a random permutation [19, 68, 100, 118, 146, 166]; and
3. sparse sign matrices: $\mathbf{\Gamma} = \sqrt{\frac{m}{\zeta}}\left[\mathbf{s}_1, \dots, \mathbf{s}_m\right]$ for some $2 \leq \zeta \leq l$, with *i.i.d.* ζ-sparse columns $\left\{\mathbf{s}_j \in \mathbb{R}^l\right\}_{j \in [m]}$ constructed such that each $\mathbf{s}_j$ is filled with ζ independent Rademacher random variables at uniformly random coordinates [30, 100, 103, 111, 147, 165].

Table 2.1 summarizes lower bounds on ls that provide theoretical guarantee for (2.12), along with asymptotic complexities of sketching, denoted as $T_s(l, \mathbf{A}_k)$, for these randomized linear

Table 2.1: Lower bounds of ls that provide theoretical guarantee for (2.12), and asymptotic complexities of sketching, $T_s(l, \mathbf{A}_k)$, for some common randomized linear embeddings.

Randomized linear embedding	Theoretical best dimension reduction	$T_s(l, \mathbf{A}_k)$
Gaussian embedding	$l = \Omega\left(k/\epsilon^2\right)$	$O(\mathrm{nnz}(\mathbf{A}_k)l)$
SRTT	$l = \Omega\left(k \log k/\epsilon^2\right)$	$O(mn \log l)$
Sparse sign matrix	$l = \Omega\left(k \log k/\epsilon^2\right)$, $\zeta = \Omega\left(\log k/\epsilon\right)$	$O(\mathrm{nnz}(\mathbf{A}_k)\zeta)$

embeddings. In spite of the weaker guarantees for structured randomized embeddings (*i.e.*, SRTTs and sparse sign matrices) in the theory by a logarithmic factor, from the empirical perspective, $l = \Omega\left(k/\epsilon^2\right)$ is usually sufficient for all the embeddings in Table 2.1 when considering tasks such as constructing randomized rangefinders (which we subsequently leverage for fast skeleton selection). For instance, it was suggested in [68, 100] to take $l = k + \Omega(1)$ (*e.g.*, $l = k + 10$) for Gaussian embeddings, $l = \Omega\left(k\right)$ for SRTTs, and $l = \Omega\left(k\right)$, $\zeta = \min\left(l, 8\right)$ for sparse sign matrices in practice [148].

2.2.4 Randomized rangefinder and low-rank SVD

Given $\mathbf{A} \in \mathbb{R}^{m\times n}$, the randomized rangefinder problem aims to construct a matrix $\mathbf{X} \in \mathbb{R}^{l\times n}$ such that the row space of $\mathbf{X}$ aligns well with the leading right singular subspace of $\mathbf{A}$ [100]: $\left\|\mathbf{A} - \mathbf{A}\mathbf{X}^\dagger\mathbf{X}\right\|$ is sufficiently small for some unitary invariant norm $\left\|\cdot\right\|$ (*e.g.*, $\left\|\cdot\right\|_2$ or $\left\|\cdot\right\|_F$). When $\mathbf{X}$ admits full row rank, we call $\mathbf{X}$ a rank-l row basis approximator of $\mathbf{A}$. The well-known optimality result from [55] demonstrated that, for a fixed rank k, the optimal rank-k row basis approximator of $\mathbf{A}$ is given by its leading k right singular vectors: $\left\|\mathbf{A} - \mathbf{A}\mathbf{V}_{A,k}\mathbf{V}_{A,k}^\top\right\|_F^2 = \left\|\mathbf{A} - \mathbf{A}_k\right\|_F^2 = \sum_{i=k+1}^{\min\{m,n\}} \sigma_i\left(\mathbf{A}\right)^2$.

A row sketch $\mathbf{X} = \mathbf{\Gamma}\mathbf{A}$ generated by some proper randomized linear embedding $\mathbf{\Gamma}$ is known to serve as a good solution for the randomized rangefinder problem with high probability. For instance, with the Gaussian embedding, a small constant oversampling $l - k \geq 4$ is sufficient for a good approximation [68]:

$$\mathbb{E}\left[\left\|\mathbf{A} - \mathbf{A}\mathbf{X}^\dagger\mathbf{X}\right\|_F^2\right] \leq \frac{l-1}{l-k-1}\left\|\mathbf{A} - \mathbf{A}_k\right\|_F^2, \tag{2.13}$$

and moreover, $\left\|\mathbf{A} - \mathbf{A}\mathbf{X}^\dagger\mathbf{X}\right\|_F^2 \lesssim l(l-k)\log\left(l-k\right)\left\|\mathbf{A} - \mathbf{A}_k\right\|_F^2$ with high probability. Similar guarantees hold for spectral norm ([68], Section 10). The randomized rangefinder error depends on the spectral decay of $\mathbf{A}$, and can be aggravated by a flat spectrum. In this scenario, power iterations (with proper orthogonalization, [68]), as well as Krylov and block Krylov subspace iterations ([108]), may be incorporated after the initial sketching as a remedy. For example, with a randomized linear embedding $\mathbf{\Omega}$ of size $l \times n$, a row basis approximator with q power iterations ($q \geq 1$) is given by

$$\mathbf{X} = \mathbf{\Omega} \left(\mathbf{A}^\top \mathbf{A}\right)^q, \tag{2.14}$$

and takes $\mathbf{X}$ takes $O\left(T_s(l, \mathbf{A}) + (2q-1)\,\mathrm{nnz}(\mathbf{A}) l\right)$ operations to construct. However, such plain power iteration in (2.14) is numerically unstable and can lead to large errors when $\mathbf{A}$ is ill-conditioned and $q > 1$. For a stable construction, orthogonalization can be applied at each iteration:

$$\begin{aligned}
&\mathbf{Y}^{(1)} = \mathbf{A}\mathbf{\Omega}^\top \\
&\mathbf{Y}^{(i)} = \mathrm{ortho}\left(\mathbf{A}\,\mathrm{ortho}\left(\mathbf{A}^\top \mathbf{Y}^{(i-1)}\right)\right) \ \forall\, i = 2, \dots, q \ (\text{if } q > 1) \\
&\mathbf{X} = \mathrm{ortho}\left(\mathbf{Y}^{(q)}\right)^\top \mathbf{A}
\end{aligned} \tag{2.15}$$

with an additional cost of $O\left(q(m+n)l^2\right)$ overall.

In addition, with a proper l that does not exceed the exact rank of $\mathbf{A}$, the row sketch $\mathbf{X} \in \mathbb{R}^{l \times n}$ has a full row rank almost surely. Precisely, recall $r = \mathrm{rank}(\mathbf{A}) \leq \min\{m, n\}$ from Section 2.2.1:

Remark 2.4. For a Gaussian embedding $\mathbf{\Gamma} \in \mathbb{R}^{l \times m}$ with *i.i.d.* entries from $\mathcal{N}(0, 1/l)$ and $l \leq r$, the row sketch $\mathbf{X} = \mathbf{\Gamma}\mathbf{A}$ has full row rank almost surely.

Rationale for Remark 2.4. Recall the reduced SVD of $\mathbf{A}$, $\mathbf{A} = \mathbf{U_A}\mathbf{\Sigma_A}\mathbf{V_A}^\top$. Given $l \leq r$, it is sufficient to show that $\mathrm{rank}\left(\mathbf{\Gamma}\mathbf{U_A}\right) = l$. Since $\mathbf{U_A}$ consists of orthonormal columns, by the rotation invariance of Gaussian distribution, $\mathbf{\Gamma}\mathbf{U_A} \in \mathbb{R}^{l \times r}$ can also be viewed as a Gaussian random matrix with *i.i.d.* entries from $\mathcal{N}(0, 1/l)$. Since all square submatrices of a Gaussian random matrix are invertible almost surely [37, 120], we have $\mathrm{rank}\left(\mathbf{\Gamma}\mathbf{U_A}\right) = l$ almost surely. ■

A low-rank row basis approximator $\mathbf{X}$ can be subsequently leveraged to construct a randomized rank-l SVD. Assuming l is properly chosen such that $\mathbf{X}$ has full row rank, let $\mathbf{Q}_X \in \mathbb{R}^{n \times l}$ be an orthonormal basis for the row space of $\mathbf{X}$. The exact SVD of the smaller matrix $\mathbf{A}\mathbf{Q}_X \in \mathbb{R}^{m \times l}$,

$$\Big[\underset{m\times l}{\widehat{\mathbf{U}}_A}, \underset{l\times l}{\widehat{\mathbf{\Sigma}}_A}, \underset{l\times l}{\widetilde{\mathbf{V}}_A}\Big] = \mathrm{svd}\left(\underbrace{\mathbf{A}\mathbf{Q}_X}_{m\times l}, \text{‘econ’}\right), \quad \underset{n\times l}{\widehat{\mathbf{V}}_A} = \mathbf{Q}_X \widetilde{\mathbf{V}}_A, \tag{2.16}$$

can be evaluated efficiently in $O\left(ml^2\right)$ operations (and $O\left(nl^2\right)$ additional operations for constructing $\widehat{\mathbf{V}}_A$) such that $\mathbf{A} \approx \mathbf{A}\mathbf{X}^\dagger\mathbf{X} = \widehat{\mathbf{U}}_A\widehat{\mathbf{\Sigma}}_A\widehat{\mathbf{V}}_A^\top$ [68].

2.2.5 Matrix decompositions with pivoting

We next briefly survey how pivoted QR and LU decompositions can be leveraged to resolve the matrix skeleton selection problem. In this section, $\mathbf{X} \in \mathbb{R}^{l\times n}$ denotes a matrix of full row rank (that will typically arise as a "row space approximator"). Let $\mathbf{X}^{(t)} \in \mathbb{R}^{l\times n}$ be the resulted matrix after the t-th step of pivoting and matrix updating, so that $\mathbf{X}^{(0)} = \mathbf{X}$.

Column pivoted QR (CPQR)

Applying the CPQR to $\mathbf{X}$ gives:

$$\mathbf{X} \underset{n\times n}{\mathbf{\Pi}_n} = \mathbf{X} \left[\underset{n\times l}{\mathbf{\Pi}_{n,1}},\ \underset{n\times(n-l)}{\mathbf{\Pi}_{n,2}}\right] = \underset{l\times l}{\mathbf{Q}_l}\ \underset{l\times n}{\mathbf{R}^{QR}} = \mathbf{Q}_l \left[\underset{l\times l}{\mathbf{R}_1^{QR}},\ \underset{l\times(n-l)}{\mathbf{R}_2^{QR}}\right], \tag{2.17}$$

where $\mathbf{Q}_l$ is an orthogonal matrix; $\mathbf{R}_1^{QR}$ is upper triangular; and $\mathbf{\Pi}_n \in \mathbb{R}^{n\times n}$ is a column permutation. QR decompositions rank-1 update the active submatrix at each step for orthogonalization (*e.g.*, [76], [144], Algorithm 10.1). For each $t = 0, \ldots, l-2$, at the $(t+1)$-th step, the CPQR searches the entire active submatrix $\mathbf{X}^{(t)}\left(t+1:l, t+1:n\right)$ for the $(t+1)$-th column pivot with the maximal ℓ_2-norm:

$$j_{t+1} = \underset{t+1\leq j\leq n}{\operatorname{argmax}} \left\|\mathbf{X}^{(t)}\left(t+1:l, j\right)\right\|_2.$$

As illustrated in [65], CPQR satisfies $\max_{i,j}\left|\left(\left(\mathbf{R}_1^{QR}\right)^{-1}\mathbf{R}_2^{QR}\right)_{ij}\right| \leq 2^{l-i}$; while the upper bound is tight with the classical Kahan matrix [79]. Nevertheless, these adversarial inputs are scarce and sensitive to perturbations. The empirical success of CPQR also suggests that exponential growth with respect to l almost never occurs in practice [145]. Meanwhile, there exist more sophisticated variations of CPQR, like the rank-revealing [24, 74] and strong rank-revealing QR [65], guaranteeing that $\max_{i,j}\left|\left(\left(\mathbf{R}_1^{QR}\right)^{-1}\mathbf{R}_2^{QR}\right)_{ij}\right|$ is upper bounded by some low-degree polynomial in l, but coming with higher complexities as trade-off.

LU with partial pivoting (LUPP)

Applying the LUPP to $\mathbf{X}^\top$ yields:

$$\mathbf{X} \underset{n\times n}{\mathbf{\Pi}_n} = \mathbf{X} \left[\underset{n\times l}{\mathbf{\Pi}_{n,1}},\ \underset{n\times(n-l)}{\mathbf{\Pi}_{n,2}}\right] = \underset{l\times l}{\mathbf{L}_l}\ \underset{l\times n}{\mathbf{R}^{LU}} = \mathbf{L}_l \left[\underset{l\times l}{\mathbf{R}_1^{LU}},\ \underset{l\times(n-l)}{\mathbf{R}_2^{LU}}\right], \tag{2.18}$$

where $\mathbf{L}_l$ is lower triangular; $\mathbf{R}_1^{LU}$ is upper triangular; $\mathbf{R}_1^{LU}(i,i) = 1$ and $\left|\mathbf{R}^{LU}(i,j)\right| \leq 1$ for all $i \in [l]$, $i \leq j \leq n$; and $\mathbf{\Pi}_n \in \mathbb{R}^{n\times n}$ is a column permutation. LU decompositions update active submatrices via Shur complements (*e.g.*, [144], Algorithm 21.1): for $t = 0, \dots, l-2$,

$$\begin{aligned}\mathbf{X}^{(t+1)}(t+2:l, t+2:n) =& \mathbf{X}^{(t)}(t+2:l, t+2:n) \\ & - \mathbf{X}^{(t)}(t+2:l, t)\,\mathbf{X}^{(t)}(t, t+2:n) / \mathbf{X}^{(t)}(t,t).\end{aligned}$$

At the $(t+1)$-th step, the LUPP on $\mathbf{X}^\top$ searches only the $(t+1)$-th row in the active submatrix and pivots

$$j_{t+1} = \underset{t+1\leq j\leq n}{\operatorname{argmax}} \left|\mathbf{X}^{(t)}(t+1, j)\right|,$$

such that $\mathbf{R}^{LU}(i,j) = \mathbf{X}^{(i-1)}(i,j) / \mathbf{X}^{(i)}(i,i)$ for all $i \in [l]$, $i+1 \leq j \leq n$ (except for $\mathbf{R}^{LU}(i,j_i) = \mathbf{X}^{(i-1)}(i,i) / \mathbf{X}^{(i)}(i,i)$), and therefore $\left|\mathbf{R}^{LU}(i,j)\right| \leq 1$.

Analogous to CPQR, the pivoting strategy of LUPP leads to a loose, exponential upper bound:

Remark 2.5. The LUPP in (2.18) satisfies that

$$\max_{i,j} \left|\left(\left(\mathbf{R}_1^{LU}\right)^{-1} \mathbf{R}_2^{LU}\right)_{ij}\right| \leq 2^{l-i},$$

where the upper bound is tight, for instance, when $\mathbf{R}_1^{LU}(i,j) = -1$ for all $i \in [l-1]$, $i+1 \leq j \leq l$ and $\mathbf{R}_2^{LU}(i,j) = 1$ for all $i \in [l]$, $j \in [n-l]$ (*i.e.*, a Kahan-type matrix [79, 116]).

Rationale for Remark 2.5. In reminiscence of the exponential worse-case growth factor of Gaussian elimination with partial pivoting (*e.g.*, [59] Section 3.4.5), we start by observing the following recursive relations: for all $j = 1, \dots, n-l$ and $i = l-1, \dots, 1$,

$$\begin{aligned}\left(\left(\mathbf{R}_1^{LU}\right)^{-1} \mathbf{R}_2^{LU}\right)_{lj} &= \mathbf{R}_2^{LU}(l,j) \\ \left(\left(\mathbf{R}_1^{LU}\right)^{-1} \mathbf{R}_2^{LU}\right)_{ij} &= \mathbf{R}_2^{LU}(i,j) - \sum_{\iota=i+1}^{l} \mathbf{R}_1^{LU}(i,\iota) \left(\left(\mathbf{R}_1^{LU}\right)^{-1} \mathbf{R}_2^{LU}\right)_{\iota,j}\end{aligned}$$

given $\mathbf{R}_1^{LU}(i,i) = 1$. Then both the upper bound and the adversarial examples of Kahan-type matrices follow from the fact that $\left|\mathbf{R}^{LU}(i,j)\right| \leq 1$ for all $i \in [l]$, $i \leq j \leq n$. ■

In addition to the exponential worst-case scenario in Remark 2.5, LUPP is also vulnerable to rank deficiency since it only views one row for each pivoting step (in contrast to CPQR which searches the entire active submatrix). The advantage of the LUPP type pivoting scheme is its superior empirical efficiency and parallelizability [58, 62, 87, 136]. Fortunately, as with CPQR, adversarial inputs for LUPP are sensitive to perturbations (*e.g.*, flip the signs of random off-diagonal entries in $\mathbf{R}_1^{LU}$), and are rarely encountered in practice.

LUPP can be further stabilized with randomization [114, 115, 145]. In terms of the worse-case exponential entry-wise bound in Remark 2.5, the average-case growth factors of LUPP on random matrices drawn from a variety of distributions (*e.g.*, the Gaussian distribution, uniform distributions, Rademacher distribution, symmetry / Toeplitz matrices with Gaussian entries, and orthogonal matrices following Haar measure) were investigated in [145] where it was conjectured that the growth factor increases sublinearly with respect to the problem size in average cases.

Remark 2.6 (Conjectured in [145]). With randomized preprocessing like sketching, LUPP is robust to adversarial inputs in practice, with $\max_{i,j} \left| \left(\left(\mathbf{R}_1^{LU} \right)^{-1} \mathbf{R}_2^{LU} \right)_{ij} \right| = O(l)$ in average cases.

Some common alternatives to partial pivoting for LU decompositions include (adaptive) cross approximations [9, 83, 150], and complete pivoting. Specifically, complete pivoting is a more robust (*e.g.*, to rank deficiency) alternative to partial pivoting that searches the entire active submatrix, and permutes rows and columns simultaneously. Despite lacking theoretical guarantees for the plain complete pivoting, like for QR decompositions, there exists modified complete pivoting strategies for LU that come with better rank-revealing guarantees [4, 25, 107, 113], but higher computational cost as a trade-off.

2.3 Summary of existing algorithms

A vast assortment of algorithms for interpolative and CUR decompositions have been proposed and analyzed in the past decades [1, 3, 4, 15, 18, 21, 25, 31, 33, 39, 41, 42, 50, 96, 126, 137, 153, 157]. From the skeleton selection perspective, these algorithms broadly fall into two categories:

1. sampling-based methods that draw matrix skeletons (directly, adaptively, or iteratively) from some proper distributions, and
2. pivoting-based methods that pick matrix skeletons greedily by constructing low-rank matrix decompositions with pivoting.

In this section, we discuss existing algorithms for matrix skeletonizations, with a focus on algorithms based on randomized linear embeddings and matrix decompositions with pivoting.

2.3.1 Sampling based skeleton selection

The idea of skeleton selection via sampling is closely related to various topics including graph sparsification [8] and volume sampling [42]. Concerning volume sampling, adaptive sampling strategies [3, 33, 39, 41] that lead to matrix skeletons with close-to-optimal error guarantees are reviewed in Section 2.2.2. Meanwhile, leverage score sampling for constructing CUR decompositions, as well as efficient estimations for the leverage scores, are extensively studied in [15, 49, 50, 96]. Furthermore, some sophisticated variations of sampling-based skeleton selection algorithms were proposed in [18, 21, 31, 157] where iterative sampling and/or combinations of different sampling schemes were incorporated.

2.3.2 Skeleton selection via deterministic pivoting

Greedy algorithms based on column and row pivoting can also be used for matrix skeletonizations. For instance, with proper rank-revealing pivoting like the strong rank-revealing QR proposed in [65], a rank-k ($k < r$) column ID can be constructed with the first k column pivots

$$\begin{aligned}\mathbf{A}[\underset{n\times l}{\mathbf{\Pi}_{n,1}}, \underset{n\times(n-l)}{\mathbf{\Pi}_{n,2}}] &= [\underset{m\times k}{\mathbf{Q}_{A,1}}, \underset{m\times(r-k)}{\mathbf{Q}_{A,2}}] \begin{bmatrix} \mathbf{R}_{A,11} & \mathbf{R}_{A,12} \\ \mathbf{0} & \mathbf{R}_{A,22} \end{bmatrix} \\ &\approx (\mathbf{A}\mathbf{\Pi}_{n,1}) \left[\mathbf{I}_k, \ \mathbf{R}_{A,11}^{-1}\mathbf{R}_{A,12}\right]\end{aligned}$$

where $\mathbf{\Pi}_n = [\mathbf{\Pi}_{n,1}, \mathbf{\Pi}_{n,2}]$ is a permutation of columns; and $\mathbf{R}_{A,11}$ and $\mathbf{R}_{A,22}$ are non-singular and upper triangular. $\mathbf{C} = (\mathbf{A}\mathbf{\Pi}_{n,1})$ are the selected column skeletons that satisfies $\left\|\mathbf{A} - \mathbf{C}\mathbf{C}^\dagger\mathbf{A}\right\| = \left\|\mathbf{R}_{A,22}\right\| \lesssim \sqrt{k(n-k)}\left\|\mathbf{A} - \mathbf{A}_k\right\|$.

As a more affordable alternative to the rank-revealing pivoting, the CPQR discussed in Section 2.2.5 also works well for skeleton selection in practice [153], despite the weaker theoretical guarantee due to the known existence of adversarial inputs (*i.e.*, $\left\|\mathbf{A} - \mathbf{C}\mathbf{C}^\dagger\mathbf{A}\right\| \lesssim 2^k\left\|\mathbf{A} - \mathbf{A}_k\right\|$).

In addition to the QR-based pivoting schemes, (randomized) LU-based pivoting algorithms with rank-revealing guarantees [1, 4, 25, 107, 113, 126] can also be leveraged for greedy matrix skeleton selection (as discussed in Section 2.2.5). Alternatively, the DEIM skeleton selection algorithm [51, 137] relaxes the rank-revealing requirements on pivoting schemes by applying LUPP on the leading singular vectors of $\mathbf{A}$.

2.3.3 Randomized pivoting-based skeleton selection

In comparison to the sampling-based skeleton selection, the deterministic pivoting-based skeleton selection methods suffer from two major drawbacks. First, pivoting is usually unaffordable for large-scale problems in common modern applications. Second, classical pivoting schemes like the CPQR and LUPP are vulnerable to antagonistic inputs. Fortunately, randomized pre-processing with sketching provides remedies to both problems:

1. Faster execution speed is attained by executing classical pivoting schemes on a sketch $\mathbf{X} = \mathbf{\Gamma}\mathbf{A} \in \mathbb{R}^{l \times n}$, for some randomized embedding $\mathbf{\Gamma}$, instead on $\mathbf{A}$ directly.
2. With randomization, classical pivoting schemes like the CPQR and LUPP are robust to adversarial inputs in practice (Remark 2.6, [145]).

Algorithm 1 describes a general framework for randomized pivoting-based skeleton selection. Grounding this framework down with different combinations of row basis approximators and pivoting schemes, it was proposed in [153] to take $\mathbf{X} = \mathbf{\Gamma}\mathbf{A}$ as a row sketch and apply CPQR to $\mathbf{X}$ for column skeleton selection. Alternatively, the DEIM skeleton selection algorithm proposed in [137] can be accelerated by taking $\mathbf{X}$ as an approximation of the leading-l right singular vectors of $\mathbf{A}$ ((2.16)), where LUPP is applied for skeleton selection.

Algorithm 1 Randomized pivoting-based skeleton selection: a general framework

```
Require: $\mathbf{A} \in \mathbb{R}^{m \times n}$ of rank $r$, rank $l \leq r$ (typically $l \ll \min(m, n)$).
Ensure: Column and/or row skeleton indices, $J_s \subset [n]$ and/or $I_s \subset [m]$, $|J_s| = |I_s| = l$.
1: Draw an oblivious $\ell_2$-embedding $\mathbf{\Gamma} \in \mathbb{R}^{l \times m}$.
2: Construct a row basis approximator $\mathbf{X} \in R^{l \times n}$ via sketching with $\mathbf{\Gamma}$.
   e.g., $\mathbf{X}$ can be 1) a row sketch or 2) approximations of right singular vectors.
3: Perform column-wise pivoting on $\mathbf{X}$. Let $J_s$ index the $l$ column pivots.
4: Perform row-wise pivoting on $\mathbf{C} = \mathbf{A}(:, J_s)$. Let $I_s$ index the $l$ row pivots.
```

First, we recall from Remark 2.4 that when taking $\mathbf{X}$ as a row sketch, with Gaussian embeddings, $\mathbf{\Gamma}$ and $\mathbf{X} = \mathbf{\Gamma}\mathbf{A}$ are both full row rank with probability 1. Moreover, when taking $\mathbf{X}$ as an approximation of right singular vectors constructed with a row basis approximator consisting of l linearly independent rows, $\mathbf{X}$ also admits full row rank.

Second, when both column and row skeletons are inquired, Algorithm 1 selects the column skeletons first with randomized pivoting and subsequently identifies the row skeletons by pivoting on the selected columns. With $\mathbf{X}$ being full row rank (almost surely when $\mathbf{\Gamma}$ is a Gaussian embedding), the column skeletons $\mathbf{C}$ are linearly independent. Therefore, the row-wise skeletonization of $\mathbf{C}$ is exact, without introducing additional errors. That is, the two-sided ID constructed by Algorithm 1 is equal to the associated column ID in exact arithmetic, $\widehat{\mathbf{A}}_{I_s,J_s} = \widehat{\mathbf{A}}_{*,J_s}$.

2.4 A simple but effective modification: LUPP on sketches

Inspired by the idea of pivoting on sketches [153] and the remarkably competitive performance of LUPP when applied to leading singular vectors [137], we propose a simple but effective modification – applying LUPP directly to a sketch of $\mathbf{A}$. In terms of the general framework in Algorithm 1, this corresponds to taking $\mathbf{X}$ as a row sketch of $\mathbf{A}$, and then selecting skeletons via LUPP on $\mathbf{X}$ and $\mathbf{C}$, as summarized in Algorithm 2.

Algorithm 2 Randomized LUPP skeleton selection

```
Require: $\mathbf{A} \in \mathbb{R}^{m \times n}$ of rank $r$, rank $l \leq r$ (typically $l \ll \min(m,n)$).
Ensure: Column and/or row skeleton indices, $J_s \subset [n]$ and/or $I_s \subset [m]$, $|J_s| = |I_s| = l$.
  Draw an oblivious $\ell_2$-embedding $\mathbf{\Gamma} \in \mathbb{R}^{l \times m}$.
  Construct a row sketch $\mathbf{X} = \mathbf{\Gamma}\mathbf{A}$.
  Perform LUPP on $\mathbf{X}^\top$. Let $J_s$ index the $l$ column pivots of $\mathbf{X}$ (i.e., row pivots of $\mathbf{X}^\top$).
  Perform LUPP on $\mathbf{C} = \mathbf{A}(:, J_s)$. Let $I_s$ index the $l$ row pivots.
```

Comparing to pivoting with CPQR [153], Algorithm 2 with LUPP is empirically faster, as discussed in Section 2.2.5, and illustrated in Figure 2.2. Meanwhile, assuming that the true SVD of $\mathbf{A}$ is unavailable, in comparison to pivoting on the approximated leading singular vectors [137] from (2.16), Algorithm 2 saves the effort of constructing randomized SVD which takes $O\left(\mathrm{nnz}(\mathbf{A})l + (m+n)l^2\right)$ additional operations. Additionally, with randomization, the stability of LUPP conjectured in [145] (Remark 2.6) applies, and Algorithm 2 effectively circumvents the potential vulnerability of LUPP to adversarial inputs in practice. A formal error analysis of Algorithm 1 in general reflects these points:

Theorem 2.1 (Column skeleton selection by pivoting on a row basis approximator)**.** *Given a row basis approximator $\mathbf{X} \in \mathbb{R}^{l \times n}$ ($l \leq r$) of $\mathbf{A}$ that admits full row rank, let $\mathbf{\Pi}_n \in \mathbb{R}^{n \times n}$ be the resulted permutation after applying some proper column pivoting scheme on $\mathbf{X}$ that identifies l linearly independent column pivots: for the $(l, n-l)$ column-wise partition $\mathbf{X}\mathbf{\Pi}_n = \mathbf{X}\left[\mathbf{\Pi}_{n,1}, \mathbf{\Pi}_{n,2}\right] = \left[\mathbf{X}_1, \mathbf{X}_2\right]$, the first l column pivots $\mathbf{X}_1 = \mathbf{X}\mathbf{\Pi}_{n,1} \in \mathbb{R}^{l \times l}$ admits full column rank. Moreover, the rank-l column ID $\widehat{\mathbf{A}}_{*,J_s} = \mathbf{C}\mathbf{C}^\dagger\mathbf{A}$, with linearly independent column skeletons $\mathbf{C} = \mathbf{A}\mathbf{\Pi}_{n,1}$, satisfies that*

$$\left\| \mathbf{A} - \mathbf{C}\mathbf{C}^\dagger\mathbf{A} \right\| \leq \eta \left\| \mathbf{A} - \mathbf{A}\mathbf{X}^\dagger\mathbf{X} \right\|, \tag{2.19}$$

where $\eta \leq \sqrt{1 + \left\|\mathbf{X}_1^\dagger \mathbf{X}_2\right\|_2^2}$*, and* $\|\cdot\|$ *represents the spectral or Frobenius norm.*

Theorem 2.1 states that when selecting column skeletons by pivoting on a row basis approximator, the low-rank approximation error of the resulting column ID is upper bounded by that of the associated row basis approximator up to a factor $\eta > 1$ that can be computed a posteriori efficiently in $O\left(l^2(n-l)\right)$ operations. (2.19) essentially decouples the error from the row basis approximation with $\mathbf{X}$ ($\left\|\mathbf{A} - \mathbf{A}\mathbf{X}^\dagger\mathbf{X}\right\|$ corresponding to Line 1 and 2 of Algorithm 1, as reviewed in Section 2.2.4) and that from the skeleton selection by pivoting on $\mathbf{X}$ (η corresponding to Line 3 and 4 of Algorithm 1).

Now we ground Theorem 2.1 with different choices of row basis approximation and pivoting strategies:

1. With Algorithm 2, $\left\|\mathbf{A} - \mathbf{A}\mathbf{X}^\dagger\mathbf{X}\right\|$ is the randomized rangefinder error ((2.13), [68] Section 10), and $\eta \leq \sqrt{1 + \left\|\left(\mathbf{R}_1^{LU}\right)^{-1}\mathbf{R}_2^{LU}\right\|_2^2}$ (recall (2.18)). Although in the worst-case scenario (where the entry-wise upper bound in Remark 2.5 is tight), $\eta = \Theta\left(2^l\sqrt{n-l}\right)$, with a randomized row sketch $\mathbf{X}$, assuming the stability of LUPP conjectured in [145] holds (Remark 2.6), $\eta = O\left(l^{3/2}\sqrt{n-l}\right)$.
2. Skeleton selection with CPQR on row sketches (*i.e.*, randomized CPQR proposed in [153]) shares the same error bound as Algorithm 2 (*i.e.*, analogous arguments hold for $\left\|\left(\mathbf{R}_1^{QR}\right)^{-1}\mathbf{R}_2^{QR}\right\|_2^2$).
3. When applying LUPP on the true leading singular vectors (*i.e.*, DEIM proposed in [137], assuming that the true SVD is available), $\left\|\mathbf{A} - \mathbf{A}\mathbf{X}^\dagger\mathbf{X}\right\| = \|\mathbf{A} - \mathbf{A}_l\|$, but without randomization, LUPP is vulnerable to adversarial inputs which can lead to $\eta = \Theta\left(2^l\sqrt{n-l}\right)$ in the worse case.
4. When applying LUPP on approximations of leading singular vectors (constructed via (2.16), *i.e.*, randomized DEIM suggested in [137]), $\left\|\mathbf{A} - \mathbf{A}\mathbf{X}^\dagger\mathbf{X}\right\|$ corresponds to the randomized rangefinder error with power itertions ([68] Corollary 10.10), while η follows the analogous analysis as for Algorithm 2.

Proof of Theorem 2.1. We start by defining two oblique projectors

$$\underset{n\times n}{\mathbf{P}_X} \triangleq \mathbf{\Pi}_{n,1}\left(\mathbf{X}\mathbf{\Pi}_{n,1}\right)^\dagger \mathbf{X}, \quad \underset{n\times n}{\mathbf{P}_C} \triangleq \mathbf{\Pi}_{n,1}\left(\mathbf{C}^\top\mathbf{C}\right)^\dagger \mathbf{C}^\top\mathbf{A},$$

and observe that, since $\mathbf{C}$ consists of linearly independent columns, $\left(\mathbf{C}^\top\mathbf{C}\right)^\dagger \mathbf{C}^\top\mathbf{A}\mathbf{\Pi}_{n,1} = \mathbf{I}_l$, and

$$\mathbf{P}_C \mathbf{P}_X = \mathbf{\Pi}_{n,1} \left(\mathbf{C}^\top \mathbf{C}\right)^\dagger \mathbf{C}^\top \mathbf{A} \mathbf{\Pi}_{n,1} \left(\mathbf{X}\mathbf{\Pi}_{n,1}\right)^\dagger \mathbf{X} = \mathbf{P}_X.$$

With $\mathbf{P}_C$, we can express the column ID as

$$\widehat{\mathbf{A}}_{*,J_s} = \mathbf{C}\mathbf{C}^\dagger \mathbf{A} = \mathbf{A}\left(\mathbf{\Pi}_{n,1} \left(\mathbf{C}^\top \mathbf{C}\right)^\dagger \mathbf{C}^\top \mathbf{A}\right) = \mathbf{A}\mathbf{P}_C,$$

Therefore, the low-rank approximation error of $\widehat{\mathbf{A}}_{*,J_s}$ satisfies

$$\begin{aligned}
\left\|\mathbf{A} - \mathbf{C}\mathbf{C}^\dagger \mathbf{A}\right\| &= \left\|\mathbf{A}\left(\mathbf{I} - \mathbf{P}_C\right)\right\| \\
&= \left\|\mathbf{A}\left(\mathbf{I}_n - \mathbf{P}_C\right)\left(\mathbf{I}_n - \mathbf{P}_X\right)\right\| \\
&= \left\|\left(\mathbf{I}_m - \mathbf{C}\mathbf{C}^\dagger\right)\mathbf{A}\left(\mathbf{I}_n - \mathbf{P}_X\right)\right\| \\
&\leq \left\|\mathbf{I}_m - \mathbf{C}\mathbf{C}^\dagger\right\|_2 \left\|\mathbf{A}\left(\mathbf{I}_n - \mathbf{P}_X\right)\right\|,
\end{aligned}$$

where $\left\|\mathbf{I}_m - \mathbf{C}\mathbf{C}^\dagger\right\|_2 = 1$, and since $\mathbf{X}\mathbf{P}_X = \mathbf{X}_1\mathbf{X}_1^\dagger \mathbf{X} = \mathbf{X}$ with $\mathbf{X}_1$ being full-rank,

$$\left\|\mathbf{A}\left(\mathbf{I}_n - \mathbf{P}_X\right)\right\| = \left\|\mathbf{A}\left(\mathbf{I}_n - \mathbf{X}^\dagger\mathbf{X}\right)\left(\mathbf{I}_n - \mathbf{P}_X\right)\right\| = \left\|\mathbf{I}_n - \mathbf{P}_X\right\|_2 \left\|\mathbf{A}\left(\mathbf{I}_n - \mathbf{X}^\dagger\mathbf{X}\right)\right\|.$$

As a result, we have

$$\left\|\mathbf{A} - \mathbf{C}\mathbf{C}^\dagger \mathbf{A}\right\| \leq \left\|\mathbf{I}_n - \mathbf{P}_X\right\|_2 \left\|\mathbf{A}\left(\mathbf{I}_n - \mathbf{X}^\dagger\mathbf{X}\right)\right\|,$$

and it is sufficient to show that $\eta \triangleq \left\|\mathbf{I}_n - \mathbf{P}_X\right\|_2 \leq \sqrt{1 + \left\|\mathbf{X}_1^\dagger \mathbf{X}_2\right\|_2^2}$. Indeed,

$$\begin{aligned}
\mathbf{I}_n - \mathbf{P}_X =& \mathbf{\Pi}_n^\top \mathbf{\Pi}_n - \mathbf{\Pi}_n^\top \mathbf{\Pi}_{n,1} \left(\mathbf{X}\mathbf{\Pi}_{n,1}\right)^\dagger \mathbf{X}\mathbf{\Pi}_n \\
=& \mathbf{I}_n - \begin{bmatrix} \mathbf{I}_l \\ \mathbf{0} \end{bmatrix} \mathbf{X}_1^\dagger \left[\mathbf{X}_1 \mathbf{X}_2\right] = \begin{bmatrix} \mathbf{0} & -\mathbf{X}_1^\dagger \mathbf{X}_2 \\ \mathbf{0} & \mathbf{I}_{n-l} \end{bmatrix}
\end{aligned}$$

such that $\eta = \left\|\left[-\mathbf{X}_1^\dagger \mathbf{X}_2; \mathbf{I}_{n-l}\right]\right\|_2 \leq \sqrt{1 + \left\|\mathbf{X}_1^\dagger \mathbf{X}_2\right\|_2^2}$. ■

Here, the proof of Theorem 2.1 is reminiscent of [137], while it generalizes the result for fixed right leading singular vectors to any proper row basis approximators of $\mathbf{A}$ (*e.g.*, $\widehat{\mathbf{V}}_A$ in (2.16), or simply a row sketch). The generalization of Theorem 2.1 leads to a factor η that is efficiently computable a posteriori, which can serve as an empirical replacement of the exponential upper bound induced by the scarce adversarial inputs.

In addition to the empirical efficiency and robustness discussed above, Algorithm 2 has another potential advantage: the skeleton selection algorithm can be easily adapted to the

Algorithm 3 Streaming LUPP/CPQR skeleton selection

```
Require: $\mathbf{A} \in \mathbb{R}^{m\times n}$ of rank $r$, rank $l \leq r$ (typically $l \ll \min(m,n)$).
Ensure: Column and/or row skeleton indices, $J_s \subset [n]$ and/or $I_s \subset [m]$, $|J_s| = |I_s| = l$.
Draw independent oblivious $\ell_2$-embeddings $\mathbf{\Gamma} \in \mathbb{R}^{l\times m}$ and $\mathbf{\Omega} \in \mathbb{R}^{l\times n}$.
Construct row and column sketches, $\mathbf{X} = \mathbf{\Gamma A}$ and $\mathbf{Y} = \mathbf{A\Omega}^\top$, in a single pass through $\mathbf{A}$.
Perform column-wise pivoting (LUPP on $\mathbf{X}^\top$ or CPQR on $\mathbf{X}$). Let $J_s$ index the $l$ column pivots.
Perform row-wise pivoting (LUPP on $\mathbf{Y}$ or CPQR on $\mathbf{Y}^\top$). Let $I_s$ index the $l$ row pivots.
```

streaming setting. The streaming setting considers $\mathbf{A}$ as a data stream that can only be accessed as a sequence of snapshots. Each snapshot of $\mathbf{A}$ can be viewed only once, and the storage of the entire matrix $\mathbf{A}$ is infeasible [100, 147, 148].

Remark 2.7. When only the column and/or row skeleton *indices* are required (and not the explicit construction of the corresponding interpolative or CUR decomposition), Algorithm 2 can be adapted to the streaming setting (as shown in Algorithm 3) by sketching both sides of $\mathbf{A}$ independently in a single pass, and pivoting on the resulting column and row sketches. Moreover, with the column and row skeletons J_s and I_s from Algorithm 3, Theorem 2.1 and its row-wise analog together, along with (2.9), imply that

$$\left\|\mathbf{A} - \widetilde{\mathbf{A}}_{I_s,J_s}\right\| \leq \eta_X \left\|\mathbf{A} - \mathbf{A}\mathbf{X}^\dagger\mathbf{X}\right\| + \eta_Y \left\|\mathbf{A} - \mathbf{Y}\mathbf{Y}^\dagger\mathbf{A}\right\|,$$

where η_X and η_Y are small in practice with pivoting on randomized sketches and have efficiently a posteriori computable upper bounds given $\mathbf{X}$ and $\mathbf{Y}$, as discussed previously. $\left\|\mathbf{A} - \mathbf{A}\mathbf{X}^\dagger\mathbf{X}\right\|_F^2$ and $\left\|\mathbf{A} - \mathbf{Y}\mathbf{Y}^\dagger\mathbf{A}\right\|$ are the randomized rangefinder errors with well-established upper bounds ([68] Section 10).

We point out that, although the column and row skeleton selection can be conducted in a streaming fashion, the explicit stable construction of ID or CUR requires two additional passes through $\mathbf{A}$: one pass for retrieving the skeletons $\mathbf{C}$ and/or $\mathbf{R}$, and the other pass to construct $\mathbf{C}^\dagger\mathbf{A}\mathbf{R}^\dagger$ for CUR, or $\mathbf{C}^\dagger\mathbf{A}$, $\mathbf{A}\mathbf{R}^\dagger$ for IDs. In practice, for efficient estimations of the ID or CUR when revisiting $\mathbf{A}$ is expensive, it is possible to circumvent the second pass through $\mathbf{A}$ with compromise on accuracy and stability, albeit the inevitability of the first pass for skeleton retrieval.

Precisely for the ID, $\mathbf{C}^\dagger\mathbf{A}$ (in (2.5)) or $\mathbf{A}\mathbf{R}^\dagger$ (in (2.6)) can be estimated without revisiting

$\mathbf{A}$ leveraging the associated row and column sketches:

$$\mathbf{C}^{\dagger}\mathbf{A} \approx \mathbf{X}_1^{\dagger}\mathbf{X}, \quad \mathbf{A}\mathbf{R}^{\dagger} \approx \mathbf{Y}\mathbf{Y}_1^{\dagger},$$

where $\mathbf{X}_1 = \mathbf{X}\left(:, J_s\right)$ and $\mathbf{Y}_1 = \mathbf{Y}\left(I_s, :\right)$ are the l column and row pivots in $\mathbf{X}$ and $\mathbf{Y}$, respectively. Meanwhile for the CUR, by retrieving the skeletons $\mathbf{S} = \mathbf{A}\left(I_s, J_s\right)$, $\mathbf{C} = \mathbf{A}\left(:, J_s\right)$ and $\mathbf{R} = \mathbf{A}\left(I_s, :\right)$, we can construct a CUR decomposition $\mathbf{C}\mathbf{S}^{-1}\mathbf{R}$, despite the compromise on both accuracy and stability.

2.5 Numerical experiments

In this section, we study the empirical performance of various randomized skeleton selection algorithms. Starting with the randomized pivoting-based algorithms, we investigate the efficiency of two major components of Algorithm 1: (1) the sketching step for row basis approximator construction, and (2) the pivoting step for greedy skeleton selection. Then we explore the suboptimality (in terms of low-rank approximation errors of the resulting CUR decompositions $\left\|\mathbf{A} - \widetilde{\mathbf{A}}_{I_s,J_s}\right\|$), as well as the efficiency (in terms of empirical run time), of different randomized skeleton selection algorithms.

We conduct all the experiments, except for those in Figure 2.1 on the efficiency of sketching, in MATLAB R2020a. In the implementation, the computationally dominant processes, including the sketching, LUPP, CPQR, and SVD, are performed by the MATLAB built-in functions. The experiments in Figure 2.1 are conducted in Julia Version 1.5.3 with the JuliaMatrices/LowRankApprox.jl package [73].

2.5.1 Computational speeds of different embeddings

Here, we compare the empirical efficiency of constructing sketches with some common randomized embeddings listed in Table 2.1. We consider applying an embedding $\mathbf{\Gamma}$ of size $l \times m$ to a matrix $\mathbf{A}$ of size $m \times n$, which can be interpreted as embedding n vectors in an ambient space $\mathbb{R}^m$ to a lower dimensional space $\mathbb{R}^l$. We scale the experiments with respect to the ambient dimension m, at several different embedding dimensions l, with a fixed number of repetitions $n = 1000$. Figure 2.1 suggests that, with proper implementation, the sparse sign matrices are more efficient than the Gaussian embeddings and the SRTTs, especially for large-scale problems. The SRTTs outperform Gaussian embeddings in terms of efficiency, and such an advantage can be amplified as l increases. These observations align with the asymptotic complexity in Table 2.1. While we also observe that, with MATLAB default implementation, the Gaussian embeddings usually enjoy matching efficiency as sparse sign matrices for moderate-size problems, and are more efficient than SRTTs.

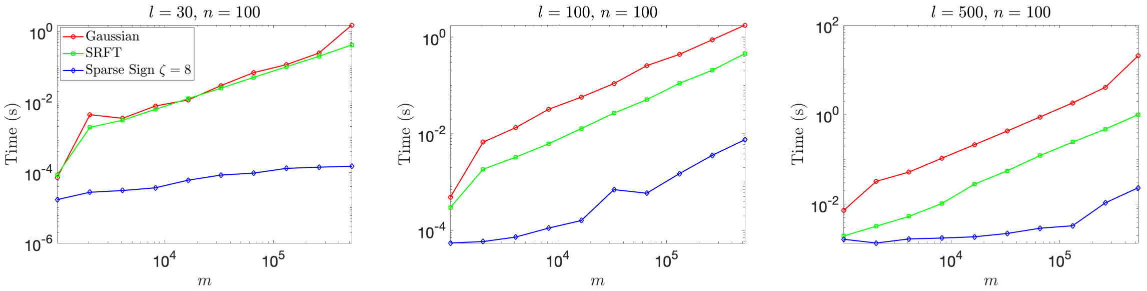


Figure 2.1: Run time of applying different randomized embeddings $\mathbf{\Gamma} \in \mathbb{R}^{l \times m}$ to some dense matrices of size $m \times n$, scaled with respect to the ambience dimension m, with different embedding dimension l, and a fixed number of embeddings $n = 100$.

2.5.2 Computational speeds of different pivoting schemes

Given a sketch of $\mathbf{A}$, we isolate different pivoting schemes in Algorithm 1 and compare their run time as the problem size n increases. Specifically, the LUPP and CPQR pivot directly on the given row sketch $\mathbf{X} = \mathbf{\Gamma A} \in \mathbb{R}^{l \times n}$, while the DEIM involves one additional power iteration with orthogonalization ((2.15)) before applying the LUPP (*i.e.*, with a given column sketch $\mathbf{Y} = \mathbf{A\Omega} \in \mathbb{R}^{m \times l}$, for DEIM, we first construct an orthonormal basis $\mathbf{Q}_Y \in \mathbb{R}^{m \times l}$ for columns of the sketch, and then we compute the reduced SVD for $\mathbf{Q}_Y^\top \mathbf{A} \in l \times n$, and finally we column-wisely pivot on the resulting right singular vectors of size $l \times n$). In Figure 2.2, we observe a considerable run time advantage of the LUPP over the CPQR and DEIM, especially when l is large. (Additionally, we see that DEIM slightly outperforms CPQR, which is perhaps surprising, given the substantially larger number of flops required by DEIM.)

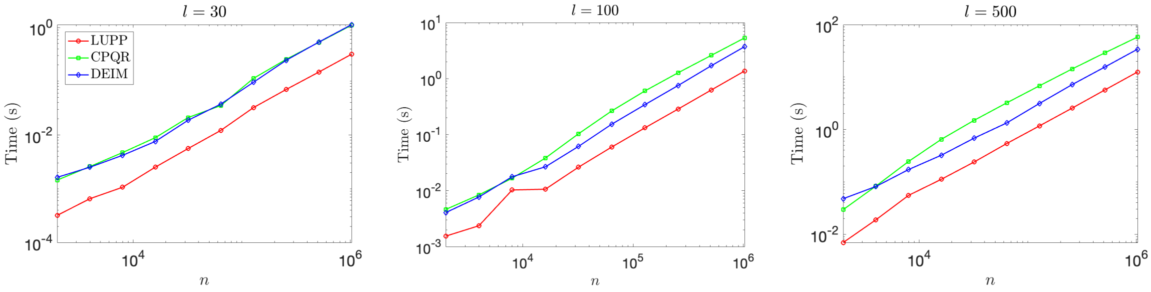


Figure 2.2: Run time of different pivoting schemes, scaled with respect to the problem size n, with different embedding dimension l.

2.5.3 Randomized skeleton selection algorithms: accuracy and efficiency

As we move from measuring speed to measuring the precision of revealing the numerical rank of a matrix, the choice of test matrix becomes important. We consider four different classes of test matrices, including some synthetic random matrices with different spectral patterns, as well as some empirical datasets, as summarized below:

1. `large`: a full-rank $4,282 \times 8,617$ sparse matrix with $20,635$ nonzero entries from the SuiteSparse matrix collection, generated by a linear programming problem sequence [104].
2. `YaleFace64x64`: a full-rank 165×4096 dense matrix, consisting of 165 face images each of size 64×64. The flattened image vectors are centered and normalized such that the average image vector is zero, and the entries are bounded within $[-1, 1]$.
3. `MNIST` training set consists of $60,000$ images of hand-written digits from 0 to 9. Each image is of size 28×28. The images are flattened and normalized to form a full-rank matrix of size $N \times d$ where N is the number of images and $d = 784$ is the size of the flattened images, with entries bounded in $[0, 1]$. The nonzero entries take approximately 20% of the matrix for both the training and the testing sets.
4. Random *sparse non-negative (SNN)* matrices are synthetic random sparse matrices used in [137, 153] for testing skeleton selection algorithms. Given $s_1 \geq \cdots \geq s_r > 0$, a random SNN matrix $\mathbf{A}$ of size $m \times n$ takes the form,

$$\mathbf{A} = \mathrm{SNN}\left(\{s_i\}_{i=1}^{r};\ m, n\right) := \sum_{i=1}^{r} s_i \mathbf{x}_i \mathbf{y}_i^T \tag{2.20}$$

   where $\mathbf{x}_i \in \mathbb{R}^m$, $\mathbf{y}_i \in \mathbb{R}^n$, $i \in [r]$ are random sparse vectors with non-negative entries. In the experiments, we use two random SNN matrices of distinct sizes:

   (i) `SNN1e3` is a 1000×1000 SNN matrix with $r = 1000$, $s_i = \frac{2}{i}$ for $i = 1, \dots, 100$, and $s_i = \frac{1}{i}$ for $i = 101, \dots, 1000$;

   (ii) `SNN1e6` is a $10^6 \times 10^6$ SNN matrix with $r = 400$, $s_i = \frac{2}{i}$ for $i = 1, \dots, 100$, and $s_i = \frac{1}{i}$ for $i = 101, \dots, 400$.

Scaled with respect to the approximation ranks k, we compare the accuracy and efficiency of the following randomized CUR algorithms:

1. Rand-LUPP (and Rand-LUPP-1piter): Algorithm 1 with $\mathbf{X} = \mathbf{\Gamma A}$ being a row sketch (or with one plain power iteration as in (2.14)), and pivoting with LUPP;

2. Rand-CPQR (and Rand-CPQR-1piter): Algorithm 1 with $\mathbf{X} = \mathbf{\Gamma A}$ being a row sketch (or with one power iteration as in (2.14)), and pivoting with CPQR [153];
3. RSVD-DEIM: Algorithm 1 with $\mathbf{X}$ being an approximation of leading-k right singular vectors ((2.16)), and pivoting with LUPP [137];
4. RSVD-LS: Skeleton sampling based on approximated leverage scores [96] from a rank-k SVD approximation ((2.16));
5. SRCUR: Spectrum-revealing CUR decomposition proposed in [25].

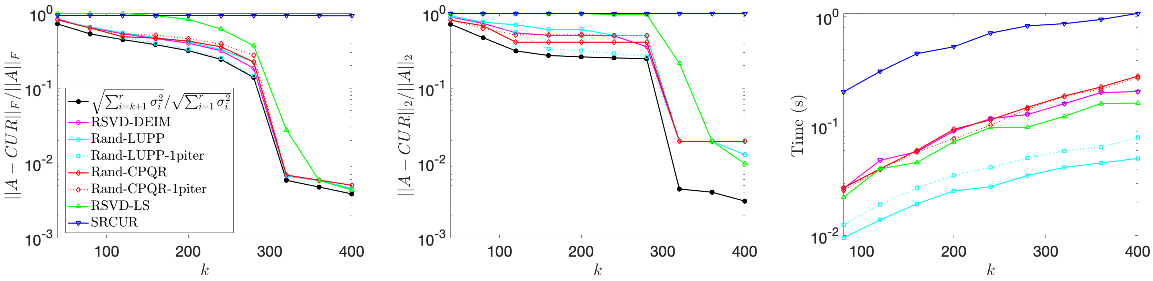
The asymptotic complexities of the first three randomized pivoting-based skeleton selection algorithms based on Algorithm 1 are summarized in Table 2.2.

Table 2.2: Asymptotic complexities of various randomized pivoting-based skeleton selection algorithms based on Algorithm 1.

Algorithm	Row basis approximator construction (Line 1,2)	Pivoting (Line 3)
Rand-LUPP	$O(T_s(l, \mathbf{A}))$	$O(nl^2)$
Rand-LUPP-1piter	$O(T_s(l, \mathbf{A}) + \mathrm{nnz}(\mathbf{A})l)$	$O(nl^2)$
Rand-CPQR	$O(T_s(l, \mathbf{A}))$	$O(nl^2)$
Rand-CPQR-1piter	$O(T_s(l, \mathbf{A}) + \mathrm{nnz}(\mathbf{A})l)$	$O(nl^2)$
RSVD-DEIM	$O\left(T_s(l, \mathbf{A}) + (m+n)l^2 + \mathrm{nnz}(\mathbf{A})l\right)$	$O(nl^2)$

For consistency, we use Gaussian embeddings for sketching throughout the experiments. With the selected column and row skeletons, we leverage the stable construction in (2.10) to form the corresponding CUR decompositions $\widetilde{\mathbf{A}}_{I_s,J_s}$. Although oversampling (*i.e.*, $l > k$) is necessary for multiplicative error bounds with respect to the optimal rank-k approximation error ((2.13), Theorem 2.1), since oversampling can be interpreted as a shift of curves along the axis of the approximation rank, for the comparison purpose, we simply treat $l = k$, and compare the rank-k approximation errors of the CUR decompositions against the optimal rank-k approximation error $\|\mathbf{A} - \mathbf{A}_k\|$.

From Figure 2.3-2.6, we observe that the randomized pivoting-based skeleton selection algorithms that fall in Algorithm 1 (*i.e.*, Rand-LUPP, Rand-CPQR, and RSVD-DEIM) share the similar approximation accuracy, which is considerably higher than that of the RSVD-LS and SRCUR. From the efficiency perspective, Rand-LUPP provides the most competitive run time among all the algorithms, especially when $\mathbf{A}$ is sparse. Meanwhile, we observe that, for both Rand-CPQR and Rand-LUPP, constructing the sketches with one plain power iteration (*i.e.*, with (2.14)) can observably improve the accuracy, without sacrificing the efficiency sig-

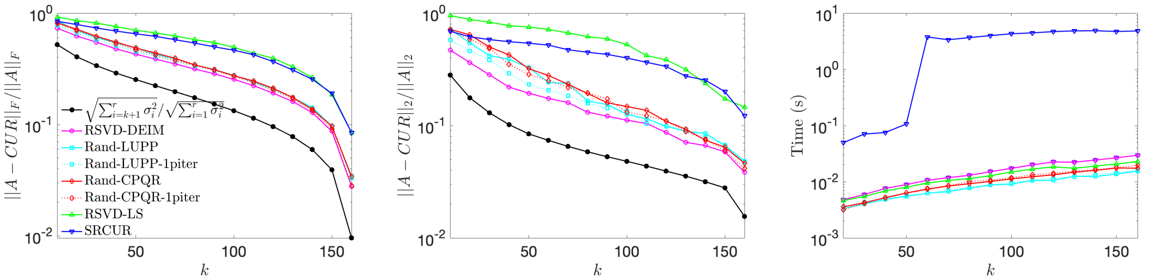


(a) Frobenius norm error.

(b) Spectral norm error.

(c) Runtime.

Figure 2.3: Relative error and run time of randomized skeleton selection on the `large` data set.

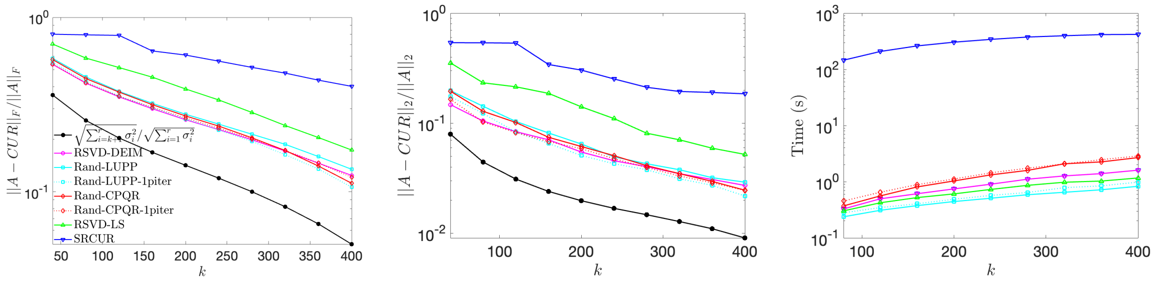


(a) Frobenius norm error.

(b) Spectral norm error.

(c) Runtime.

Figure 2.4: Relative error and run time of randomized skeleton selection on the `YaleFace64x64` data set.



(a) Frobenius norm error.

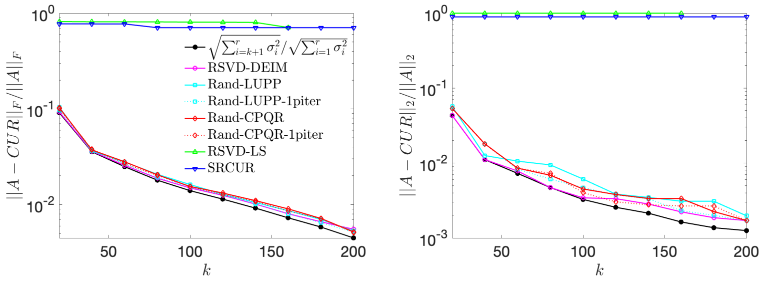
(b) Spectral norm error.

(c) Runtime.

Figure 2.5: Relative error and run time of randomized skeleton selection on the training set of MNIST.

nificantly (*e.g.*, in comparison to the randomized DEIM which involves one power iteration with orthogonalization as in (2.15)).

In Figure 2.7, a similar performance is also observed on a synthetic large-scale problem, `SNN1e6`, where the matrix is only accessible as a fast matrix-vector multiplication (matvec) oracle such that each matvec takes $o(mn)$ (i.e., $O((m+n)r)$ in our construction) opera-

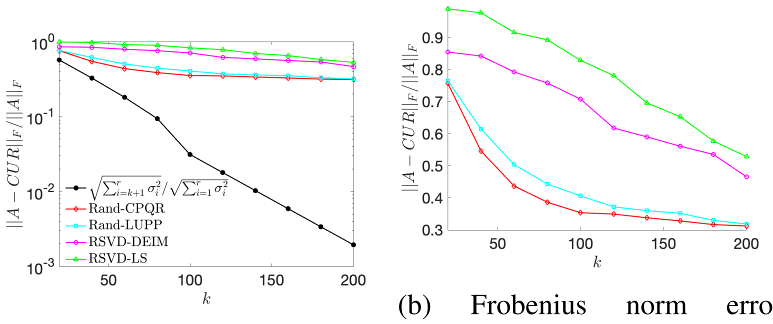


(a) Frobenius norm error.

(b) Spectral norm error.

(c) Runtime.

Figure 2.6: Relative error and run time of randomized skeleton selection on a 1000×1000 sparse non-negative random matrix, `SNN1e3`.



(a) Frobenius norm error.

(b) Frobenius norm error zoomed.

(c) Runtime.

Figure 2.7: Relative error and run time of randomized skeleton selection on a $10^6 \times 10^6$ sparse non-negative random matrix, `SNN1e6`.

tions. It is worth noticing that the gap between the optimal rank-k approximation error and the CUR approximation error consists of two components: (i) the suboptimality of skeleton selection from the algorithms, and (ii) the gap between the optimal rank-k approximation error and the CUR approximation error with the optimal skeleton selection from the matrix skeletonization problem. Despite the unknown optimal skeleton selection due to its intractability [29], intuitively, the suboptimality from the matrix skeletonization problem itself accounts for the larger CUR approximation error in Figure 2.7 (*cf.* Figure 2.3-2.6) due to the significantly more challenging skeleton selection problem brought by the large matrix size $m = n = 10^6 \gg r = \mathrm{rank}(\mathbf{A}) = 400$.

Chapter 3

Randomized Subspace Approximations: Efficient Bounds and Estimates for Canonical Angles

Abstract

Randomized subspace approximation with “matrix sketching” is an effective approach for constructing approximate partial singular value decompositions (SVDs) of large matrices. The performance of such techniques has been extensively analyzed, and very precise estimates on the distribution of the residual errors have been derived. However, our understanding of the accuracy of the computed singular vectors (measured in terms of the canonical angles between the spaces spanned by the exact and the computed singular vectors, respectively) remains relatively limited. In this work, we present bounds and estimates for canonical angles of randomized subspace approximation that *can be computed efficiently either a priori or a posteriori*. Under moderate oversampling in the randomized SVD, our prior probabilistic bounds are asymptotically tight and can be computed efficiently, while bringing a clear insight into the balance between oversampling and power iterations given a fixed budget on the number of matrix-vector multiplications. The numerical experiments demonstrate the empirical effectiveness of these canonical angle bounds and estimates on different matrices under various algorithmic choices for the randomized SVD.[1]

3.1 Introduction

In light of the ubiquity of high-dimensional data in modern computation, dimension reduction tools like the low-rank matrix decompositions are becoming indispensable tools for managing large data sets. In general, the goal of low-rank matrix decomposition is to identify bases of proper low-dimensional subspaces that well encapsulate the dominant components in the original column and row spaces. As one of the most well-established forms of matrix decompositions, the truncated singular value decomposition (SVD) is known to achieve the

[1] This chapter is based on the following arXiv paper:
Yijun Dong, Per-Gunnar Martinsson, and Yuji Nakatsukasa. Efficient bounds and estimates for canonical angles in randomized subspace approximations. *arXiv preprint arXiv:2211.04676*, 2022 [46].

optimal low-rank approximation errors for any given ranks [55]. Moreover, the corresponding left and right leading singular subspaces can be broadly leveraged for problems like principal component analysis, canonical correlation analysis, spectral clustering [20], and leverage score sampling for matrix skeleton selection [49, 50, 96].

However, for large matrices, the computational cost of classical algorithms for computing the SVD (*cf.* [144, Lec. 31] or [59, Sec. 8.6.3]) quickly becomes prohibitive. Fortunately, a randomization framework known as "matrix sketching" [68, 165] provides a simple yet effective remedy for this challenge by embedding large matrices to random low-dimensional subspaces where the classical SVD algorithms can be executed efficiently.

Concretely, for an input matrix $\mathbf{A} \in \mathbb{C}^{m \times n}$ and a target rank $k \ll \min(m,n)$, the basic version of the randomized SVD [68, Alg. 4.1] starts by drawing a Gaussian random matrix $\mathbf{\Omega} \in \mathbb{C}^{n \times l}$ for a sample size l that is slightly larger than k so that $k < l \ll \min(m,n)$. Then through a matrix-matrix multiplication $\mathbf{X} = \mathbf{A}\mathbf{\Omega}$ with $O(mnl)$ complexity, the n-dimensional row space of $\mathbf{A}$ is embedded to a random l-dimensional subspace. With the low-dimensional range approximation $\mathbf{X}$, a rank-l randomized SVD $\widehat{\mathbf{A}}_l = \widehat{\mathbf{U}}_l \widehat{\mathbf{\Sigma}}_l \widehat{\mathbf{V}}_l^*$ can be constructed efficiently by computing the QR and SVD of small matrices in $O\left((m+n)l^2\right)$ time. When the spectral decay in $\mathbf{A}$ is slow, a few power iterations $\mathbf{X} = \left(\mathbf{A}\mathbf{A}^*\right)^q \mathbf{A}\mathbf{\Omega}$ (usually $q = 1, 2$) can be incorporated to enhance the accuracy, *cf.* [68] Algorithms 4.3 and 4.4.

Let $\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^*$[2] denote the (unknown) full SVD of $\mathbf{A}$. In this work, we explore the alignment between the true leading rank-k singular subspaces $\mathbf{U}_k, \mathbf{V}_k$ and their respective rank-l approximations $\widehat{\mathbf{U}}_l, \widehat{\mathbf{V}}_l$ in terms of the canonical angles $\angle\left(\mathbf{U}_k, \widehat{\mathbf{U}}_l\right)$ and $\angle\left(\mathbf{V}_k, \widehat{\mathbf{V}}_l\right)$. We introduce prior statistical guarantees and unbiased estimates for these angles with respect to $\mathbf{\Sigma}$, as well as posterior deterministic bounds with the additional dependence on $\widehat{\mathbf{A}}_l$, as synopsized below.

3.1.1 Our Contributions

Prior probabilistic bounds and estimates with insight on oversampling-power iteration balance. Evaluating the randomized SVD with a fixed budget on the number of matrix-vector multiplications, the computational resource can be leveraged in two ways – oversampling (characterized by $l-k$) and power iterations (characterized by q). A natural question is *how to*

[2] Here $\mathbf{U} \in \mathbb{C}^{m \times r}$, $\mathbf{V} \in \mathbb{C}^{n \times r}$, and $\mathbf{\Sigma} \in \mathbb{C}^{r \times r}$. $\mathbf{\Sigma}$ is a diagonal matrix with positive non-increasing diagonal entries, and $r \leq \min(m,n)$.

distribute the computation between oversampling and power iterations for better subspace approximations?

Answers to this question are problem-dependent: when aiming to minimize the canonical angles between the true and approximated leading singular subspaces, the prior probabilistic bounds and estimates on the canonical angles provide primary insights. To be precise, with isotropic random subspace embeddings and sufficient oversampling, the accuracy of subspace approximations depends jointly on the spectra of the target matrices, oversampling, and the number of power iterations. In this work, we present a set of prior probabilistic bounds that precisely quantify the relative benefits of oversampling versus power iterations. Specifically, the canonical angle bounds in Theorem 3.1

(i) provide statistical guarantees that are asymptotically tight under sufficient oversampling (*i.e.*, $l = \Omega\left(k\right)$),

(ii) unveil a clear balance between oversampling and power iterations for random subspace approximations with given spectra,

(iii) can be evaluated in $O(\mathrm{rank}\left(\mathbf{A}\right))$ time given access to the (true/estimated) spectra and provide valuable estimations for canonical angles in practice with moderate oversampling (*e.g.*, $l \geq 1.6k$).

Further, inspired by the derivation of the prior probabilistic bounds, we propose unbiased estimates for the canonical angles with respect to given spectra that admit efficient evaluation and concentrate well empirically.

Posterior residual-based guarantees. Alongside the prior probabilistic bounds, we present two sets of posterior canonical angle bounds that hold in the deterministic sense and can be approximated efficiently based on the residuals and the spectrum of $\mathbf{A}$.

Numerical comparisons. With numerical comparisons among different canonical angle bounds on a variety of data matrices, we aim to explore the question on *how the spectral decay and different algorithmic choices of randomized SVD affect the empirical effectiveness of different canonical angle bounds*. In particular, our numerical experiments suggest that, for matrices with subexponential spectral decay, the prior probabilistic bounds usually provide tighter (statistical) guarantees than the (deterministic) guarantees from the posterior residual-based bounds, especially with power iterations. By contrast, for matrices with exponential spectral decay, the posterior residual-based bounds can be as tight as the prior probabilistic bounds, especially with large oversampling. The code for numerical comparisons is available at https://github.com/dyjdongyijun/Randomized_Subspace_Approximation.

3.1.2 Related Work

The randomized SVD algorithm (with power iterations) [68, 99] has been extensively analyzed as a low-rank approximation problem where the accuracy is usually measured in terms of residual norms, as well as the discrepancy between the approximated and true spectra [64, 68, 98, 101]. For instance, [68, Thm. 10.7, Thm. 10.8] show that for a given target rank k (usually $k \ll \min(m, n)$ for the randomized acceleration to be useful), a small constant oversampling $l \geq 1.1k$ is sufficient to guarantee that the residual norm of the resulting rank-l approximation is close to the optimal rank-k approximation (*i.e.*, the rank-k truncated SVD) error with high probability. Alternatively, [64] investigates the accuracy of the individual approximated singular values $\widehat{\sigma}_i$ and provides upper and lower bounds for each $\widehat{\sigma}_i$ with respect to the true singular value σ_i.

In addition to providing accurate low-rank approximations, the randomized SVD algorithm also produces estimates of the leading left and right singular subspaces corresponding to the top singular values. When coupled with power iterations ([68] Algorithms 4.3 & 4.4), such randomized subspace approximations are commonly known as randomized power (subspace) iterations. Their accuracy is explored in terms of canonical angles that measure differences between the unknown true subspaces and their random approximations [20, 109, 123]. Generally, upper bounds on the canonical angles can be categorized into two types: (i) probabilistic bounds that establish prior statistical guarantees by exploring the concentration of the alignment between random subspace embeddings and the unknown true subspace, and (ii) residual-based bounds that can be computed a posteriori from the residual of the resulting approximation.

The existing prior probabilistic bounds on canonical angles [20, 123] mainly focus on the setting where the randomized SVD is evaluated without oversampling or with a small constant oversampling. Concretely, [20] derives guarantees for the canonical angles evaluated without oversampling (*i.e.*, $l = k$) in the context of spectral clustering. Further, by taking a small constant oversampling (*e.g.*, $l \geq k + 2$) into account, Saibaba [123] provides a comprehensive analysis for an assortment of canonical angles between the true and approximated leading singular spaces. Compared with our results (Theorem 3.1), in both no-oversampling and constant-oversampling regimes, the basic forms of the existing prior probabilistic bounds (*e.g.*, [123] Theorem 1) generally depend on the unknown singular subspace $\mathbf{V}$. Although such dependence is later lifted using the isotropicity and the concentration of the randomized subspace embedding $\mathbf{\Omega}$ (*e.g.*, [123] Theorem 6), the separated consideration on the spectra and the singular subspaces introduces unnecessary compromise to the upper bounds (as we will discuss in Remark 3.1). In contrast, by allowing a more generous choice of multiplicative oversampling $l = \Omega\left(k\right)$, we present a set of space-agnostic bounds (i.e., bounds that hold regardless of the singular vectors of $\mathbf{A}$) based on an integrated analysis of the spectra and the singular subspaces that appears to be tighter both from derivation and in practice.

The classical Davis-Kahan $\sin\theta$ and $\tan\theta$ theorems [38] for eigenvector perturbation can be used to compute deterministic and computable bounds for the canonical angles. These bounds have the advantage that they give strict bounds (up to the estimation of the so-called gap) rather than estimates or bounds that hold with high probability (although, as we argue below, the failure probability can be taken to be negligibly low). The Davis-Kahan theorems have been extended to perturbation of singular vectors by Wedin [161], and recent work [109] derives perturbation bounds for singular vectors computed using a subspace projection method. In this work, we establish canonical angle bounds for the singular vectors in the context of (randomized) subspace iterations. Our results indicate clearly that the accuracy of the right and left singular vectors are usually not identical (i.e., $\mathbf{V}$ is more accurate with Algorithm 4).

As a roadmap, we formalize the problem setup in Section 3.2, including a brief review of the randomized SVD and canonical angles. In Section 3.3, we present the prior probabilistic space-agnostic bounds. Subsequently, in Section 3.4, we describe a set of unbiased canonical angle estimates that is closely related to the space-agnostic bounds. Then in Section 3.5, we introduce two sets of posterior residual-based bounds. Finally, in Section 3.6, we instantiate the insight cast by the space-agnostic bounds on the balance between oversampling and power iterations and demonstrate the empirical effectiveness of different canonical angle bounds and estimates with numerical comparisons.

3.2 Problem Setup

In this section, we first recapitulate the randomized SVD algorithm (with power iterations) [68] for which we analyze the accuracy of the resulting singular subspace approximations. Then, we review the notion of canonical angles [59] that quantify the difference between two subspaces of the same Euclidean space.

3.2.1 Notation

We start by introducing notations for the SVD of a given matrix $\mathbf{A} \in \mathbb{C}^{m \times n}$ of rank r:

$$\mathbf{A} = \underset{m\times r}{\mathbf{U}} \underset{r\times r}{\mathbf{\Sigma}} \underset{r\times n}{\mathbf{V}^*} = \begin{bmatrix} \mathbf{u}_1 & \dots & \mathbf{u}_r \end{bmatrix} \begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_r \end{bmatrix} \begin{bmatrix} \mathbf{v}_1^* \\ \vdots \\ \mathbf{v}_r^* \end{bmatrix}.$$

For any $1 \leq k \leq r$, we let $\mathbf{U}_k \triangleq [\mathbf{u}_1, \dots, \mathbf{u}_k]$ and $\mathbf{V}_k \triangleq [\mathbf{v}_1, \dots, \mathbf{v}_k]$ denote the orthonormal bases of the dimension-k left and right singular subspaces of $\mathbf{A}$ corresponding to the top-k singular values, while $\mathbf{U}_{r\setminus k} \triangleq [\mathbf{u}_{k+1}, \dots, \mathbf{u}_r]$ and $\mathbf{V}_{r\setminus k} \triangleq [\mathbf{v}_{k+1}, \dots, \mathbf{v}_r]$ are orthonormal bases of the respective orthogonal complements. The diagonal submatrices consisting of the spectrum, $\mathbf{\Sigma}_k \triangleq \operatorname{diag}(\sigma_1, \dots, \sigma_k)$ and $\mathbf{\Sigma}_{r\setminus k} \triangleq \operatorname{diag}(\sigma_{k+1}, \dots, \sigma_r)$, follow analogously.

Meanwhile, for the QR decomposition of an arbitrary matrix $\mathbf{M} \in \mathbb{C}^{d \times l}$ $(d \geq l)$, we denote $\mathbf{M} = [\mathbf{Q}_{\mathbf{M}}, \mathbf{Q}_{\mathbf{M},\perp}] \begin{bmatrix} \mathbf{R}_{\mathbf{M}} \\ \mathbf{0} \end{bmatrix}$ such that $\mathbf{Q}_{\mathbf{M}} \in \mathbb{C}^{d \times l}$ and $\mathbf{Q}_{\mathbf{M},\perp} \in \mathbb{C}^{d \times (d-l)}$ consist of orthonormal bases of the subspace spanned by the columns of $\mathbf{M}$ and its orthogonal complement.

Furthermore, we denote the spectrum of $\mathbf{M}$ by $\sigma(\mathbf{M})$, a $\operatorname{rank}(\mathbf{M}) \times \operatorname{rank}(\mathbf{M})$ diagonal matrix with singular values $\sigma_1(\mathbf{M}) \geq \dots \geq \sigma_{\operatorname{rank}(\mathbf{M})}(\mathbf{M}) > 0$ on the diagonal.

Generally, we adapt the MATLAB notation for matrix slicing throughout this work. For any $k \in \mathbb{N}$, we denote $[k] = \{1, \dots, k\}$.

3.2.2 Randomized SVD and Power Iterations

Algorithm 4 Randomized SVD (with power iterations) [68]

```
Require: $\mathbf{A} \in \mathbb{C}^{m \times n}$, power $q \in \{0, 1, 2, \dots\}$, oversampled rank $l \in \mathbb{N}$ $(l < r = \operatorname{rank}(\mathbf{A}))$
Ensure: $\widehat{\mathbf{U}}_l \in \mathbb{C}^{m \times l}$, $\widehat{\mathbf{V}}_l \in \mathbb{C}^{n \times l}$, $\widehat{\mathbf{\Sigma}}_l \in \mathbb{C}^{l \times l}$ such that $\widehat{\mathbf{A}}_l = \widehat{\mathbf{U}}_l \widehat{\mathbf{\Sigma}}_l \widehat{\mathbf{V}}_l^*$
1: Draw $\mathbf{\Omega} \sim P\left(\mathbb{C}^{n \times l}\right)$ with $\Omega_{ij} \sim \mathcal{N}(0, l^{-1})$ i.i.d. such that $\mathbb{E}[\mathbf{\Omega}\mathbf{\Omega}^*] = \mathbf{I}_n$
2: $\mathbf{X}^{(q)} = (\mathbf{A}\mathbf{A}^*)^q \mathbf{A}\mathbf{\Omega}$
3: $\mathbf{Q}_{\mathbf{X}} = \operatorname{ortho}\left(\mathbf{X}^{(q)}\right)$
4: $\left[\widetilde{\mathbf{U}}_l, \widehat{\mathbf{\Sigma}}_l, \widehat{\mathbf{V}}_l\right] = \operatorname{svd}(\mathbf{A}^*\mathbf{Q}_{\mathbf{X}})$ (where $\widetilde{\mathbf{U}}_l \in \mathbb{C}^{l \times l}$)
5: $\widehat{\mathbf{U}}_l = \mathbf{Q}_{\mathbf{X}} \widetilde{\mathbf{U}}_l$
```

As described in Algorithm 4, the randomized SVD provides a rank-l $(l \ll \min(m, n))$

approximation of $\mathbf{A} \in \mathbb{C}^{m \times n}$ while grants provable acceleration to the truncated SVD evaluation – $O\left(mnl(2q+1)\right)$ with the Gaussian random matrix[3]. Such efficiency improvement is achieved by first embedding the high-dimensional row (column) space of $\mathbf{A}$ to a low-dimensional subspace via a Johnson-Lindenstrauss transform[4] (JLT) $\mathbf{\Omega}$ (known as "sketching"). Then, SVD of the resulting column (row) sketch $\mathbf{X} = \mathbf{A\Omega}$ can be evaluated efficiently in $O\left(ml^2\right)$ time, and the rank-l approximation can be constructed accordingly.

The spectral decay in $\mathbf{A}$ has a significant impact on the accuracy of the resulting low-rank approximation from Algorithm 4 (as suggested in [68] Theorem 10.7 and Theorem 10.8). To remediate the performance of Algorithm 4 on matrices with flat spectra, power iterations (Algorithm 4, Line 2) are usually incorporated to enhance the spectral decay. However, without proper orthogonalization, plain power iterations can be numerically unstable, especially for ill-conditioned matrices. For stable power iterations, starting with $\mathbf{X} = \mathbf{A\Omega} \in \mathbb{C}^{m \times l}$ of full column rank (which holds almost surely for Gaussian random matrices), we incorporate orthogonalization in each power iteration via the reduced unpivoted QR factorization (each with complexity $O(ml^2)$). Let $\mathrm{ortho}\left(\mathbf{X}\right) = \mathbf{Q}_{\mathbf{X}} \in \mathbb{C}^{m \times l}$ be an orthonormal basis of $\mathbf{X}$ produced by the QR factorization. Then, the stable evaluation of q power iterations (Algorithm 4, Line 2) can be expressed as:

$$\mathbf{X}^{(0)} \leftarrow \mathrm{ortho}\left(\mathbf{A\Omega}\right), \quad \mathbf{X}^{(i)} \leftarrow \mathrm{ortho}\left(\mathbf{A}\,\mathrm{ortho}\left(\mathbf{A}^*\mathbf{X}^{(i-1)}\right)\right) \ \forall\, i \in [q]. \tag{3.1}$$

Notice that in Algorithm 4, with $\mathbf{X} = \mathbf{X}^{(q)}$, the approximated rank-$l$ SVD of $\mathbf{A}$ can be expressed as $\widehat{\mathbf{A}}_l = \widehat{\mathbf{U}}_l \widehat{\mathbf{\Sigma}}_l \widehat{\mathbf{V}}_l^* = \mathbf{X}\mathbf{Y}^*$ where $\mathbf{Y} = \mathbf{X}^\dagger \mathbf{A}$. With $\widehat{\mathbf{U}}_l$ and $\widehat{\mathbf{V}}_l$ characterizing the approximated l-dimensional left and right leading singular subspaces, $\widehat{\mathbf{U}}_{m\setminus l} \in \mathbb{C}^{m \times (m-l)}$ and $\widehat{\mathbf{V}}_{n\setminus l} \in \mathbb{C}^{n \times (n-l)}$ denote an arbitrary pair of their respective orthogonal complements. For any $1 \leq k < l$, we further denote the partitions $\widehat{\mathbf{U}}_l = \left[\widehat{\mathbf{U}}_k, \widehat{\mathbf{U}}_{l\setminus k}\right]$ and $\widehat{\mathbf{V}}_l = \left[\widehat{\mathbf{V}}_k, \widehat{\mathbf{V}}_{l\setminus k}\right]$ where $\widehat{\mathbf{U}}_k \in \mathbb{C}^{m \times k}$ and $\widehat{\mathbf{V}}_k \in \mathbb{C}^{n \times k}$, respectively.

[3] Asymptotically, there exist deterministic iterative algorithms for the truncated SVD (*e.g.*, based on Lanczos iterations ([144] Algorithm 36.1)) that run in $O\left(mnl\right)$ time. However, compared with these inherently sequential iterative algorithms, the randomized SVD can be executed much more efficiently in practice, even with power iterations (*i.e.*, $q > 0$), since the $O\left(mnl(2q+1)\right)$ computation bottleneck in Algorithm 4 involves only matrix-matrix multiplications which are easily parallelizable and highly optimized.

[4] Throughout this work, we focus on Gaussian random matrices (Algorithm 4, Line 1) in the sake of theoretical guarantees, *i.e.*, $\mathbf{\Omega}$ being isotropic and rotationally invariant.

3.2.3 Canonical Angles

Now, we review the notion of canonical angles [59] that measure distances between two subspaces $\mathcal{U}$, $\mathcal{V}$ of an arbitrary Euclidean space $\mathbb{C}^d$.

Definition 3.1 (Canonical angles, [59])**.** Given two subspaces $\mathcal{U}, \mathcal{V} \subseteq \mathbb{C}^d$ with dimensions $\dim(\mathcal{U}) = l$ and $\dim(\mathcal{V}) = k$ (assuming $l \geq k$ without loss of generality), the canonical angles, denoted by $\angle(\mathcal{U}, \mathcal{V}) = \operatorname{diag}(\theta_1, \dots, \theta_k)$, consist of k angles that measure the alignment between $\mathcal{U}$ and $\mathcal{V}$, defined recursively such that

$$\begin{aligned} \mathbf{u}_i, \mathbf{v}_i \ &\triangleq \ \operatorname{argmax} \ \mathbf{u}_i^* \mathbf{v}_i \\ &\quad \text{s.t. } \mathbf{u}_i \in \left(\mathcal{U} \setminus \operatorname{span}\left\{\mathbf{u}_\iota\right\}_{\iota=1}^{i-1}\right) \cap \mathbb{S}^{d-1}, \\ &\qquad\quad \mathbf{v}_i \in \left(\mathcal{V} \setminus \operatorname{span}\left\{\mathbf{v}_\iota\right\}_{\iota=1}^{i-1}\right) \cap \mathbb{S}^{d-1} \\ \cos\theta_i \ &= \mathbf{u}_i^* \mathbf{v}_i \quad \forall\, i = 1, \dots, k, \quad 0 \leq \theta_1 \leq \dots \leq \theta_k \leq \pi/2. \end{aligned}$$

For arbitrary full-rank matrices $\mathbf{U} \in \mathbb{C}^{d \times l}$ and $\mathbf{V} \in \mathbb{C}^{d \times k}$ (assuming $k \leq l \leq d$ without loss of generality), let $\angle(\mathbf{M}, \mathbf{N}) \triangleq \angle(\operatorname{span}(\mathbf{M}), \operatorname{span}(\mathbf{N}))$ denote the canonical angles between the corresponding spanning subspaces in $\mathbb{C}^d$. For each $i \in [k]$, let $\angle_i(\mathbf{M}, \mathbf{N})$ be the i-th (smallest) canonical angle such that $\cos\angle_i(\mathbf{M}, \mathbf{N}) = \sigma_i(\mathbf{Q}_\mathbf{M}^* \mathbf{Q}_\mathbf{N})$ and $\sin\angle_i(\mathbf{M}, \mathbf{N}) = \sigma_{k-i+1}((\mathbf{I} - \mathbf{Q}_\mathbf{M}\mathbf{Q}_\mathbf{M}^*)\mathbf{Q}_\mathbf{N})$ (*cf.* [183] Section 3).

With the unknown true rank-k truncated SVD $\mathbf{A}_k = \mathbf{U}_k \boldsymbol{\Sigma}_k \mathbf{V}_k^*$ and an approximated rank-l SVD $\widehat{\mathbf{A}}_l = \widehat{\mathbf{U}}_l \widehat{\boldsymbol{\Sigma}}_l \widehat{\mathbf{V}}_l^*$ from Algorithm 4, in this work, we mainly focus on the prior and posterior guarantees for the canonical angles $\angle\left(\mathbf{U}_k, \widehat{\mathbf{U}}_l\right)$ and $\angle\left(\mathbf{V}_k, \widehat{\mathbf{V}}_l\right)$. Meanwhile, in Theorem 3.4, we present a set of posterior residual-based upper bounds for the canonical angles $\angle\left(\mathbf{U}_k, \widehat{\mathbf{U}}_k\right)$ and $\angle\left(\mathbf{V}_k, \widehat{\mathbf{V}}_k\right)$ as corollaries.

3.3 Space-agnostic Bounds under Sufficient Oversampling

We start by pointing out the intuition that, under sufficient oversampling, with Gaussian random matrices whose distribution is orthogonally invariant, the alignment between the approximated and true subspaces are independent of the unknown true subspaces, *i.e.*, the canonical angles are space-agnostic, as reflected in the following theorem.

Theorem 3.1. *For a rank-l randomized SVD (Algorithm 4) with a Gaussian embedding $\boldsymbol{\Omega}$ and $q \geq 0$ power iterations, when the oversampled rank l satisfies $l = \Omega(k)$ (where k is the target* *rank, $k < l < r = \mathrm{rank}(\mathbf{A})$) and q is reasonably small such that $\eta \triangleq \frac{\left(\sum_{j=k+1}^{r} \sigma_j^{4q+2}\right)^2}{\sum_{j=k+1}^{r} \sigma_j^{2(4q+2)}}$* [5] *satisfies $\eta = \Omega(l)$, with high probability (at least $1 - e^{-\Theta(k)} - e^{-\Theta(l)}$), there exist distortion factors $0 < \epsilon_1, \epsilon_2 < 1$ such that*

$$\sin \angle_i \left(\mathbf{U}_k, \widehat{\mathbf{U}}_l\right) \leq \left(1 + \frac{1-\epsilon_1}{1+\epsilon_2} \cdot \frac{l}{\sum_{j=k+1}^{r} \sigma_j^{4q+2}} \cdot \sigma_i^{4q+2}\right)^{-\frac{1}{2}} \tag{3.2}$$

$$\sin \angle_i \left(\mathbf{V}_k, \widehat{\mathbf{V}}_l\right) \leq \left(1 + \frac{1-\epsilon_1}{1+\epsilon_2} \cdot \frac{l}{\sum_{j=k+1}^{r} \sigma_j^{4q+4}} \cdot \sigma_i^{4q+4}\right)^{-\frac{1}{2}} \tag{3.3}$$

for all $i \in [k]$, where $\epsilon_1 = \Theta\left(\sqrt{\frac{k}{l}}\right)$ and $\epsilon_2 = \Theta\left(\sqrt{\frac{l}{\eta}}\right)$. Furthermore, both bounds are asymptotically tight:

$$\sin \angle_i \left(\mathbf{U}_k, \widehat{\mathbf{U}}_l\right) \geq \left(1 + O\left(\frac{l \cdot \sigma_i^{4q+2}}{\sum_{j=k+1}^{r} \sigma_j^{4q+2}}\right)\right)^{-\frac{1}{2}} \tag{3.4}$$

$$\sin \angle_i \left(\mathbf{V}_k, \widehat{\mathbf{V}}_l\right) \geq \left(1 + O\left(\frac{l \cdot \sigma_i^{4q+4}}{\sum_{j=k+1}^{r} \sigma_j^{4q+4}}\right)\right)^{-\frac{1}{2}} \tag{3.5}$$

where $O(\cdot)$ suppresses the distortion factors $\frac{1+\epsilon_1}{1-\epsilon_2}$ [6].

The main insights provided by Theorem 3.1 include: (i) improved statistical guarantees for canonical angles under sufficient oversampling (*i.e.*, $l = \Omega(k)$), as discussed later in Remark 3.1, (ii) a clear view of the balance between oversampling and power iterations for random subspace approximations with given spectra, as instantiated in Section 3.6.3, and (iii) affordable upper bounds that can be evaluated in $O(\mathrm{rank}(\mathbf{A}))$ time with access to the (true/estimated) spectra and hold in practice with only moderate oversampling (*e.g.*, $l \geq 1.6k$), as shown in Section 3.6.2.

Proof of Theorem 3.1. We show the derivation of (3.2) for left canonical angles $\sin \angle_i \left(\mathbf{U}_k, \widehat{\mathbf{U}}_l\right)$. The derivation for right canonical angles $\sin \angle_i \left(\mathbf{V}_k, \widehat{\mathbf{V}}_l\right)$ in (3.3) follows directly by replacing

[5] Notice that $1 < \eta \leq r - k$. To the extremes, $\eta = r - k$ when the tail is flat $\sigma_{k+1} = \cdots = \sigma_r$; while $\eta \to 1$ when $\sigma_{k+1} \gg \sigma_j$ for all $j = k+2, \ldots, r$. In particular, with a relatively flat tail $\mathbf{\Sigma}_{r \setminus k}$ and a reasonably small q (recall that $q = 1, 2$ is usually sufficient in practice), we have $\eta = \Theta(r - k)$, and the assumption can be simplified as $r - k = \Omega(l)$. Although exponential tail decay can lead to small η and may render the assumption infeasible in theory, in practice, simply taking $r - k = \Omega(l)$, $l = \Omega(k)$, $\epsilon_1 = \sqrt{\frac{k}{l}}$ and $\epsilon_2 = \sqrt{\frac{l}{r-k}}$ is sufficient to ensure the validity of upper bounds when $q \leq 10$ even for matrices with rapid tail decay, as shown in Section 3.6.2.

[6] Despite the asymptotic tightness of Theorem 3.1 theoretically, in practice, we observe that the empirical validity of lower bounds is more restrictive on oversampling than that of upper bounds. In specific, the numerical observations in Section 3.6.2 suggest that $l \geq 1.6k$ is usually sufficient for the upper bounds to hold; whereas the empirical validity of lower bounds generally requires more aggressive oversampling of at least $l \geq 4k$, also with slightly larger constants associated with ϵ_1 and ϵ_2, as demonstrated in Section A.2.1.

the exponent $4q+2$ in (3.2) with $4q+4$ in (3.3). This slightly larger exponent comes from the additional half power iteration associated with $\widehat{\mathbf{V}}_l$ in Algorithm 4 (as discussed in Remark 3.3), an observation made also in [123].

For the rank-l randomized SVD with a Gaussian embedding $\mathbf{\Omega} \in \mathbb{C}^{n\times l}$ and q power iterations, we denote the projected embeddings onto the singular subspaces $\mathbf{\Omega}_1 \triangleq \mathbf{V}_k^*\mathbf{\Omega}$ and $\mathbf{\Omega}_2 \triangleq \mathbf{V}_{r\setminus k}^*\mathbf{\Omega}$, as well as their weighted correspondences $\widetilde{\mathbf{\Omega}}_1 \triangleq \mathbf{\Sigma}_k^{2q+1}\mathbf{\Omega}_1$ and $\widetilde{\mathbf{\Omega}}_2 \triangleq \mathbf{\Sigma}_{r\setminus k}^{2q+1}\mathbf{\Omega}_2$, such that

$$\widetilde{\mathbf{X}} \triangleq \mathbf{U}^*\mathbf{X} = \mathbf{U}^*\mathbf{A}\mathbf{\Omega} = \begin{bmatrix} \mathbf{\Sigma}_k^{2q+1}\mathbf{\Omega}_1 \\ \mathbf{\Sigma}_{r\setminus k}^{2q+1}\mathbf{\Omega}_2 \end{bmatrix} = \begin{bmatrix} \widetilde{\mathbf{\Omega}}_1 \\ \widetilde{\mathbf{\Omega}}_2 \end{bmatrix}.$$

Then for all $i \in [k]$,

$$\begin{aligned}
\sin\angle_{k-i+1}\left(\mathbf{U}_k, \widehat{\mathbf{U}}_l\right) =& \sigma_i\left(\left(\mathbf{I}_m - \widehat{\mathbf{U}}_l\widehat{\mathbf{U}}_l^*\right)\mathbf{U}_k\right) \\
(\text{span}(\mathbf{U}_l) = \text{span}(\mathbf{X}) \text{ and } \mathbf{U}\mathbf{U}^*\mathbf{U}_k = \mathbf{U}_k) =& \sigma_i\left(\left(\mathbf{I}_m - \mathbf{X}\mathbf{X}^\dagger\right)\mathbf{U}\mathbf{U}^*\mathbf{U}_k\right) \\
(\mathbf{U}\mathbf{U}^*\mathbf{X} = \mathbf{X}) =& \sigma_i\left(\mathbf{U}\mathbf{U}^*\left(\mathbf{I}_m - \mathbf{X}\mathbf{X}^\dagger\right)\mathbf{U}\mathbf{U}^*\mathbf{U}_k\right) \\
(\mathbf{U} \text{ consists of orthonormal columns}) =& \sigma_i\left(\mathbf{U}^*\left(\mathbf{I}_m - \mathbf{X}\mathbf{X}^\dagger\right)\mathbf{U}\mathbf{U}^*\mathbf{U}_k\right) \\
\left(\mathbf{U}^*\mathbf{U} = \mathbf{I}_r,\ \mathbf{X}^\dagger\mathbf{U} = \left(\mathbf{U}^*\mathbf{X}\right)^\dagger = \widetilde{\mathbf{X}}^\dagger\right) =& \sigma_i\left(\left(\mathbf{I}_r - \widetilde{\mathbf{X}}\widetilde{\mathbf{X}}^\dagger\right)\begin{bmatrix} \mathbf{I}_k \\ \mathbf{0} \end{bmatrix}\right).
\end{aligned}$$

Since $\mathbf{X}$ is assumed to have full column rank, we have $\widetilde{\mathbf{X}}\widetilde{\mathbf{X}}^\dagger = \widetilde{\mathbf{X}}\left(\widetilde{\mathbf{X}}^*\widetilde{\mathbf{X}}\right)^{-1}\widetilde{\mathbf{X}}^*$ (which is an orthogonal projection), and

$$\begin{aligned}
&\begin{bmatrix} \mathbf{I}_k & \mathbf{0} \end{bmatrix}\left(\mathbf{I}_r - \widetilde{\mathbf{X}}\widetilde{\mathbf{X}}^\dagger\right)\begin{bmatrix} \mathbf{I}_k \\ \mathbf{0} \end{bmatrix} \\
=&\begin{bmatrix} \mathbf{I}_k & \mathbf{0} \end{bmatrix}\left(\mathbf{I}_r - \begin{bmatrix} \widetilde{\mathbf{\Omega}}_1 \\ \widetilde{\mathbf{\Omega}}_2 \end{bmatrix}\left(\widetilde{\mathbf{\Omega}}_1^*\widetilde{\mathbf{\Omega}}_1 + \widetilde{\mathbf{\Omega}}_2^*\widetilde{\mathbf{\Omega}}_2\right)^{-1}\begin{bmatrix} \widetilde{\mathbf{\Omega}}_1^* & \widetilde{\mathbf{\Omega}}_2^* \end{bmatrix}\right)\begin{bmatrix} \mathbf{I}_k \\ \mathbf{0} \end{bmatrix} \\
=&\mathbf{I}_k - \widetilde{\mathbf{\Omega}}_1\left(\widetilde{\mathbf{\Omega}}_1^*\widetilde{\mathbf{\Omega}}_1 + \widetilde{\mathbf{\Omega}}_2^*\widetilde{\mathbf{\Omega}}_2\right)^{-1}\widetilde{\mathbf{\Omega}}_1^* \\
(\text{Woodbury identity}) =&\left(\mathbf{I}_k + \widetilde{\mathbf{\Omega}}_1\left(\widetilde{\mathbf{\Omega}}_2^*\widetilde{\mathbf{\Omega}}_2\right)\widetilde{\mathbf{\Omega}}_1^*\right)^{-1}.
\end{aligned}$$

Therefore for all $i \in [k]$,

$$\begin{aligned}
\sin^2\angle_{k-i+1}\left(\mathbf{U}_k, \widehat{\mathbf{U}}_l\right) =& \sigma_i\left(\begin{bmatrix} \mathbf{I}_k & \mathbf{0} \end{bmatrix}\left(\mathbf{I}_r - \widetilde{\mathbf{X}}\widetilde{\mathbf{X}}^\dagger\right)\begin{bmatrix} \mathbf{I}_k \\ \mathbf{0} \end{bmatrix}\right) \\
=& \sigma_i\left(\left(\mathbf{I}_k + \widetilde{\mathbf{\Omega}}_1\left(\widetilde{\mathbf{\Omega}}_2^*\widetilde{\mathbf{\Omega}}_2\right)^{-1}\widetilde{\mathbf{\Omega}}_1^*\right)^{-1}\right).
\end{aligned} \tag{3.6}$$

By the orthogonal invariance of the distribution of Gaussian embeddings $\mathbf{\Omega}$ together with the orthonormality of $\mathbf{V}_k \perp \mathbf{V}_{r\setminus k}$, we see that $\mathbf{\Omega}_1 \sim P\left(\mathbb{C}^{k\times l}\right)$ and $\mathbf{\Omega}_2 \sim P\left(\mathbb{C}^{(r-k)\times l}\right)$ are independent Gaussian random matrices with the same entry-wise *i.i.d.* distribution $\mathcal{N}\left(0, l^{-1}\right)$ as $\mathbf{\Omega}$. Therefore by Lemma A.1, when $l = \Omega(k)$, with high probability (at least $1 - e^{-\Theta(k)}$),

$$\left(1 - \epsilon_1\right) \mathbf{\Sigma}_k^{4q+2} \preccurlyeq \widetilde{\mathbf{\Omega}}_1 \widetilde{\mathbf{\Omega}}_1^* \preccurlyeq \left(1 + \epsilon_1\right) \mathbf{\Sigma}_k^{4q+2}$$

for some $\epsilon_1 = \Theta\left(\sqrt{\frac{k}{l}}\right)$.

Analogously, when $r - k \geq \eta = \frac{\operatorname{tr}\left(\mathbf{\Sigma}_{r\setminus k}^{4q+2}\right)^2}{\operatorname{tr}\left(\mathbf{\Sigma}_{r\setminus k}^{2(4q+2)}\right)} = \Omega(l)$, with high probability (at least $1 - e^{-\Theta(l)}$),

$$\frac{1 - \epsilon_2}{l} \operatorname{tr}\left(\mathbf{\Sigma}_{r\setminus k}^{4q+2}\right) \mathbf{I}_l \preccurlyeq \widetilde{\mathbf{\Omega}}_2^* \widetilde{\mathbf{\Omega}}_2 \preccurlyeq \frac{1 + \epsilon_2}{l} \operatorname{tr}\left(\mathbf{\Sigma}_{r\setminus k}^{4q+2}\right) \mathbf{I}_l$$

for some $\epsilon_2 = \Theta\left(\sqrt{\frac{l}{\eta}}\right)$.

Therefore by the union bound, we have with high probability (at least $1 - e^{-\Theta(k)} - e^{-\Theta(l)}$) that,

$$\left(\mathbf{I}_k + \widetilde{\mathbf{\Omega}}_1 \left(\widetilde{\mathbf{\Omega}}_2^* \widetilde{\mathbf{\Omega}}_2\right)^{-1} \widetilde{\mathbf{\Omega}}_1^*\right)^{-1} \preccurlyeq \left(\mathbf{I}_k + \frac{1 - \epsilon_1}{1 + \epsilon_2} \cdot \frac{l}{\operatorname{tr}\left(\mathbf{\Sigma}_{r\setminus k}^{4q+2}\right)} \cdot \mathbf{\Sigma}_k^{4q+2}\right)^{-1},$$

which leads to (3.2), while the tightness is implied by

$$\left(\mathbf{I}_k + \widetilde{\mathbf{\Omega}}_1 \left(\widetilde{\mathbf{\Omega}}_2^* \widetilde{\mathbf{\Omega}}_2\right)^{-1} \widetilde{\mathbf{\Omega}}_1^*\right)^{-1} \succcurlyeq \left(\mathbf{I}_k + \frac{1 + \epsilon_1}{1 - \epsilon_2} \cdot \frac{l}{\operatorname{tr}\left(\mathbf{\Sigma}_{r\setminus k}^{4q+2}\right)} \cdot \mathbf{\Sigma}_k^{4q+2}\right)^{-1}.$$

The proof of (3.3) follows analogously by replacing the exponents $2q + 1$ and $4q + 2$ with $2q + 2$ and $4q + 4$, respectively. ■

Remark 3.1 (Comparison with existing probabilistic bounds). With access to the unknown right singular subspace $\mathbf{V}$, let $\mathbf{\Omega}_1 \triangleq \mathbf{V}_k^* \mathbf{\Omega}$ and $\mathbf{\Omega}_2 \triangleq \mathbf{V}_{r\setminus k}^* \mathbf{\Omega}$. Then, Saibaba [123, Thm. 1]indicates that, for all $i \in [k]$,

$$\sin \angle_i \left(\mathbf{U}_k, \widehat{\mathbf{U}}_l\right) \leq \left(1 + \frac{\sigma_i^{4q+2}}{\sigma_{k+1}^{4q+2} \left\|\mathbf{\Omega}_2 \mathbf{\Omega}_1^\dagger\right\|_2^2}\right)^{-\frac{1}{2}}, \tag{3.7}$$

$$\sin \angle_i \left(\mathbf{V}_k, \widehat{\mathbf{V}}_l\right) \leq \left(1 + \frac{\sigma_i^{4q+4}}{\sigma_{k+1}^{4q+4} \left\|\mathbf{\Omega}_2 \mathbf{\Omega}_1^\dagger\right\|_2^2}\right)^{-\frac{1}{2}}. \tag{3.8}$$

Further, leveraging existing results on concentration properties of the independent and isotropic Gaussian random matrices $\mathbf{\Omega}_1$ and $\mathbf{\Omega}_2$ (*e.g.*, from the proof of [68] Theorem 10.8), [123] shows that, when $l \geq k+2$, for any $\delta \in (0,1)$, with probability at least $1-\delta$,

$$\left\|\mathbf{\Omega}_2\mathbf{\Omega}_1^\dagger\right\|_2 \leq \frac{e\sqrt{l}}{l-k+1}\left(\frac{2}{\delta}\right)^{\frac{1}{l-k+1}}\left(\sqrt{n-k}+\sqrt{l}+\sqrt{2\log\frac{2}{\delta}}\right).$$

Without loss of generality, we consider the bounds on $\sin\angle\left(\mathbf{U}_k, \widehat{\mathbf{U}}_l\right)$. Comparing to the existing bound in (3.7), under multiplicative oversampling ($l = \Omega(k)$, $r = \Omega(l)$), (3.2) in Theorem 3.1 captures the spectral decay on the tail by replacing the denominator term

$$\sigma_{k+1}^{4q+2}\left\|\mathbf{\Omega}_2\mathbf{\Omega}_1^\dagger\right\|_2^2 \quad \text{with} \quad \frac{1+\epsilon_2}{1-\epsilon_1}\cdot\frac{1}{l}\sum_{j=k+1}^{r}\sigma_j^{4q+2} = \Theta\left(\frac{1}{l}\sum_{j=k+1}^{r}\sigma_j^{4q+2}\right).$$

We observe that $\frac{1}{l}\sum_{j=k+1}^{r}\sigma_j^{4q+2} \leq \frac{r-k}{l}\sigma_{k+1}^{4q+2}$; while Lemma A.1 implies that, for independent Gaussian random matrices $\mathbf{\Omega}_1 \sim P\left(\mathbb{C}^{k\times l}\right)$ and $\mathbf{\Omega}_2 \sim P\left(\mathbb{C}^{(r-k)\times l}\right)$ with *i.i.d.* entries from $\mathcal{N}\left(0, l^{-1}\right)$,

$$\mathbb{E}\left[\left\|\mathbf{\Omega}_2\mathbf{\Omega}_1^\dagger\right\|_2^2\right] = \mathbb{E}_{\mathbf{\Omega}_1}\left[\left\|\left(\mathbf{\Omega}_1^\dagger\right)^*\mathbb{E}_{\mathbf{\Omega}_2}\left[\mathbf{\Omega}_2^*\mathbf{\Omega}_2\right]\mathbf{\Omega}_1^\dagger\right\|_2\right] = \frac{r-k}{l}\mathbb{E}_{\mathbf{\Omega}_1}\left[\left\|\left(\mathbf{\Omega}_1\mathbf{\Omega}_1^*\right)^\dagger\right\|_2\right].$$

With non-negligible spectral decay on the tail such that $\frac{1}{l}\sum_{j=k+1}^{r}\sigma_j^{4q+2} \ll \frac{r-k}{l}\sigma_{k+1}^{4q+2}$, when $\frac{1+\epsilon_2}{1-\epsilon_1}\cdot\frac{1}{l}\sum_{j=k+1}^{r}\sigma_j^{4q+2} \ll \sigma_{k+1}^{4q+2}\left\|\mathbf{\Omega}_2\mathbf{\Omega}_1^\dagger\right\|_2^2$, (3.2) provides a tighter statistical guarantee than (3.7), which is also confirmed by numerical observations in Section 3.6.2.

From the derivation perspective, such improvement is achieved by taking an integrated view on the concentration of $\mathbf{\Sigma}_{r\backslash k}^{2q+1}\mathbf{\Omega}_2$, instead of considering the spectrum and the unknown singular subspace separately.

3.4 Unbiased Space-agnostic Estimates

A natural corollary from the proof of Theorem 3.4 is unbiased estimates for the canonical angles that hold for arbitrary oversampling (*i.e.*, for all $l \geq k$). Further, we will subsequently see in Section 3.6 that such unbiased estimates also enjoy good empirical concentration.

Proposition 3.2. *For a rank-l randomized SVD (Algorithm 4) with the Gaussian embedding $\mathbf{\Omega} \sim P\left(\mathbb{C}^{n\times l}\right)$ such that $\Omega_{ij} \sim \mathcal{N}\left(0, l^{-1}\right)$* i.i.d. *and $q \geq 0$ power iterations, for all $i \in [k]$,*

$$\mathbb{E}_{\mathbf{\Omega}}\left[\sin\angle_i\left(\mathbf{U}_k, \widehat{\mathbf{U}}_l\right)\right] = \mathbb{E}_{\mathbf{\Omega}_1', \mathbf{\Omega}_2'}\left[\sigma_i^{-\frac{1}{2}}\left(\mathbf{I}_k + \mathbf{\Sigma}_k^{2q+1}\mathbf{\Omega}_1'\left(\mathbf{\Omega}_2'^*\mathbf{\Sigma}_{r\backslash k}^{4q+2}\mathbf{\Omega}_2'\right)^{-1}\mathbf{\Omega}_1'^*\mathbf{\Sigma}_k^{2q+1}\right)\right], \tag{3.9}$$

and analogously,

$$\mathbb{E}_{\mathbf{\Omega}}\left[\sin\angle_i\left(\mathbf{V}_k, \widehat{\mathbf{V}}_l\right)\right] = \mathbb{E}_{\mathbf{\Omega}_1', \mathbf{\Omega}_2'}\left[\sigma_i^{-\frac{1}{2}}\left(\mathbf{I}_k + \mathbf{\Sigma}_k^{2q+2}\mathbf{\Omega}_1'\left(\mathbf{\Omega}_2'^*\mathbf{\Sigma}_{r\backslash k}^{4q+4}\mathbf{\Omega}_2'\right)^{-1}\mathbf{\Omega}_1'^*\mathbf{\Sigma}_k^{2q+2}\right)\right], \tag{3.10}$$

where $\boldsymbol{\Omega}_1' \sim P\left(\mathbb{C}^{k\times l}\right)$ *and* $\boldsymbol{\Omega}_2' \sim P\left(\mathbb{C}^{(r-k)\times l}\right)$ *are independent Gaussian random matrices with* i.i.d. *entries drawn from* $\mathcal{N}\left(0, l^{-1}\right)$.

To calculate the unbiased estimate, for a modest integer N we draw a set of independent Gaussian random matrices $\left\{\boldsymbol{\Omega}_1^{(j)} \sim P\left(\mathbb{C}^{k\times l}\right), \boldsymbol{\Omega}_2^{(j)} \sim P\left(\mathbb{C}^{(r-k)\times l}\right) \;\middle|\; j \in [N]\right\}$ and evaluate

$$\sin\angle_i\left(\mathbf{U}_k, \widehat{\mathbf{U}}_l\right) \approx \alpha_i = \frac{1}{N}\sum_{j=1}^{N}\left(1+\sigma_i^2\left(\boldsymbol{\Sigma}_k^{2q+1}\boldsymbol{\Omega}_1^{(j)}\left(\boldsymbol{\Sigma}_{r\setminus k}^{2q+1}\boldsymbol{\Omega}_2^{(j)}\right)^{\dagger}\right)\right)^{-\frac{1}{2}},$$

$$\sin\angle_i\left(\mathbf{V}_k, \widehat{\mathbf{V}}_l\right) \approx \beta_i = \frac{1}{N}\sum_{j=1}^{N}\left(1+\sigma_i^2\left(\boldsymbol{\Sigma}_k^{2q+2}\boldsymbol{\Omega}_1^{(j)}\left(\boldsymbol{\Sigma}_{r\setminus k}^{2q+2}\boldsymbol{\Omega}_2^{(j)}\right)^{\dagger}\right)\right)^{-\frac{1}{2}},$$

for all $i \in [k]$, which can be conducted efficiently in $O\left(Nrl^2\right)$ time. Algorithm 5 demonstrates the construction of unbiased estimates for $\mathbb{E}\left[\sin\angle_i\left(\mathbf{U}_k, \widehat{\mathbf{U}}_l\right)\right] = \alpha_i$; while the unbiased estimates for $\mathbb{E}\left[\sin\angle_i\left(\mathbf{V}_k, \widehat{\mathbf{V}}_l\right)\right] = \beta_i$ can be evaluated analogously by replacing Line 4 with $\widetilde{\boldsymbol{\Omega}}_1^{(j)} = \boldsymbol{\Sigma}_k^{2q+2}\boldsymbol{\Omega}^{(j)}(1:k,:)$, $\widetilde{\boldsymbol{\Omega}}_2^{(j)} = \boldsymbol{\Sigma}_{r\setminus k}^{2q+2}\boldsymbol{\Omega}^{(j)}(k+1:r,:)$.

Algorithm 5 Unbiased canonical angle estimates

```
Require: (Exact or estimated) singular values $\boldsymbol{\Sigma}$, rank $k$, sample size $l \geq k$, number of power
    iterations $q$, number of trials $N$
Ensure: Unbiased estimates $\mathbb{E}\left[\sin\angle_i\left(\mathbf{U}_k, \widehat{\mathbf{U}}_l\right)\right] = \alpha_i$ for all $i \in [k]$
 1: Partition $\boldsymbol{\Sigma}$ into $\boldsymbol{\Sigma}_k = \boldsymbol{\Sigma}(1:k, 1:k)$ and $\boldsymbol{\Sigma}_{r\setminus k} = \boldsymbol{\Sigma}(k+1:r, k+1:r)$
 2: for $j = 1, \dots, N$ do
 3:     Draw $\boldsymbol{\Omega}^{(j)} \sim P\left(\mathbb{C}^{r\times l}\right)$ such that $\Omega_{ij}^{(j)} \sim \mathcal{N}\left(0, l^{-1}\right)$ i.i.d.
 4:     $\widetilde{\boldsymbol{\Omega}}_1^{(j)} = \boldsymbol{\Sigma}_k^{2q+1}\boldsymbol{\Omega}^{(j)}(1:k,:)$, $\widetilde{\boldsymbol{\Omega}}_2^{(j)} = \boldsymbol{\Sigma}_{r\setminus k}^{2q+1}\boldsymbol{\Omega}^{(j)}(k+1:r,:)$
 5:     $\left[\mathbf{U}_{\widetilde{\boldsymbol{\Omega}}_2^{(j)}}, \boldsymbol{\Sigma}_{\widetilde{\boldsymbol{\Omega}}_2^{(j)}}, \mathbf{V}_{\widetilde{\boldsymbol{\Omega}}_2^{(j)}}\right] = \mathrm{svd}\left(\widetilde{\boldsymbol{\Omega}}_2^{(j)}, \text{“econ”}\right)$
 6:     $\boldsymbol{\nu}^{(j)} = \mathrm{svd}\left(\widetilde{\boldsymbol{\Omega}}_1^{(j)}\mathbf{V}_{\widetilde{\boldsymbol{\Omega}}_2^{(j)}}\boldsymbol{\Sigma}_{\widetilde{\boldsymbol{\Omega}}_2^{(j)}}^{-1}\mathbf{U}_{\widetilde{\boldsymbol{\Omega}}_2^{(j)}}^{*}\right)$
 7:     $\theta_i^{(j)} = 1/\sqrt{1+\left(\nu_i^{(j)}\right)^2}$ for all $i \in [k]$
 8: $\alpha_i = \frac{1}{N}\sum_{j=1}^{N}\theta_i^{(j)}$ for all $i \in [k]$
```

As demonstrated in Section 3.6, the unbiased estimates concentrate well in practice such that a sample size as small as $N = 3$ is seen to be sufficient to provide good estimates. Further, with independent Gaussian random matrices, the unbiased estimates in Proposition 3.2 are also space agnostic, *i.e.*, (3.9) and (3.10) only depend on the spectrum $\boldsymbol{\Sigma}$ but not on the unknown true singular subspaces $\mathbf{U}$ and $\mathbf{V}$.

Proof of Proposition 3.2. To show (3.9), we recall from the proof of Theorem 3.1 that, for the rank-l randomized SVD with a Gaussian embedding $\boldsymbol{\Omega} \sim P\left(\mathbb{C}^{n\times l}\right)$ and q power iterations,

$\boldsymbol{\Omega}_1 \triangleq \mathbf{V}_k^* \boldsymbol{\Omega}$ and $\boldsymbol{\Omega}_2 \triangleq \mathbf{V}_{r\setminus k}^* \boldsymbol{\Omega}$ are independent Gaussian random matrices with the same entry-wise distribution as $\boldsymbol{\Omega}$. Therefore, with $\mathrm{rank}\left(\mathbf{A}\right) = r$, for all $i \in [k]$,

$$
\begin{aligned}
&\mathbb{E}_{\boldsymbol{\Omega}}\left[\sin \angle_i \left(\mathbf{U}_k, \widehat{\mathbf{U}}_l\right)\right] \quad \text{(Recall (3.6))} \\
=&\mathbb{E}_{\boldsymbol{\Omega}}\left[\sigma^{\frac{1}{2}}_{k-i+1}\left(\left(\mathbf{I}_k + \boldsymbol{\Sigma}_k^{2q+1}\boldsymbol{\Omega}_1\left(\boldsymbol{\Omega}_2^*\boldsymbol{\Sigma}_{r\setminus k}^{4q+2}\boldsymbol{\Omega}_2\right)^{-1}\boldsymbol{\Omega}_1^*\boldsymbol{\Sigma}_k^{2q+1}\right)^{-1}\right)\right] \\
=&\mathbb{E}_{\boldsymbol{\Omega}}\left[\sigma_i^{-\frac{1}{2}}\left(\mathbf{I}_k + \boldsymbol{\Sigma}_k^{2q+1}\boldsymbol{\Omega}_1\left(\boldsymbol{\Omega}_2^*\boldsymbol{\Sigma}_{r\setminus k}^{4q+2}\boldsymbol{\Omega}_2\right)^{-1}\boldsymbol{\Omega}_1^*\boldsymbol{\Sigma}_k^{2q+1}\right)\right] \\
=&\mathop{\mathbb{E}}_{\boldsymbol{\Omega}_1', \boldsymbol{\Omega}_2'}\left[\sigma_i^{-\frac{1}{2}}\left(\mathbf{I}_k + \boldsymbol{\Sigma}_k^{2q+1}\boldsymbol{\Omega}_1'\left(\boldsymbol{\Omega}_2'^*\boldsymbol{\Sigma}_{r\setminus k}^{4q+2}\boldsymbol{\Omega}_2'\right)^{-1}\boldsymbol{\Omega}_1'^*\boldsymbol{\Sigma}_k^{2q+1}\right)\right].
\end{aligned}
$$

The unbiased estimate in (3.10) follows analogously. ■

As a side note, we point out that, compared with the probabilistic upper bounds (3.2) and (3.3), the estimates (3.9) and (3.10) circumvent overestimation from the operator-convexity of inversion $\sigma \to \sigma^{-1}$,

$$
\mathbb{E}_{\boldsymbol{\Omega}_2'}\left[\left(\boldsymbol{\Omega}_2'^*\boldsymbol{\Sigma}_{r\setminus k}^{4q+2}\boldsymbol{\Omega}_2'\right)^{-1}\right] \succcurlyeq \left(\mathbb{E}_{\boldsymbol{\Omega}_2'}\left[\boldsymbol{\Omega}_2'^*\boldsymbol{\Sigma}_{r\setminus k}^{4q+2}\boldsymbol{\Omega}_2'\right]\right)^{-1},
$$

which implies that

$$
\begin{aligned}
&\mathbb{E}_{\boldsymbol{\Omega}_1', \boldsymbol{\Omega}_2'}\left[\left(\mathbf{I}_k + \boldsymbol{\Sigma}_k^{2q+1}\boldsymbol{\Omega}_1'\left(\boldsymbol{\Omega}_2'^*\boldsymbol{\Sigma}_{r\setminus k}^{4q+2}\boldsymbol{\Omega}_2'\right)^{-1}\boldsymbol{\Omega}_1'^*\boldsymbol{\Sigma}_k^{2q+1}\right)^{-1}\right] \\
\succcurlyeq\ &\mathbb{E}_{\boldsymbol{\Omega}_1'}\left[\left(\mathbf{I}_k + \boldsymbol{\Sigma}_k^{2q+1}\boldsymbol{\Omega}_1'\left(\mathbb{E}_{\boldsymbol{\Omega}_2'}\left[\boldsymbol{\Omega}_2'^*\boldsymbol{\Sigma}_{r\setminus k}^{4q+2}\boldsymbol{\Omega}_2'\right]\right)^{-1}\boldsymbol{\Omega}_1'^*\boldsymbol{\Sigma}_k^{2q+1}\right)^{-1}\right].
\end{aligned}
$$

3.5 Posterior Residual-based Bounds

In addition to the prior probabilistic bounds and unbiased estimates, in this section, we introduce two sets of posterior guarantees for the canonical angles that hold deterministically and can be evaluated/approximated efficiently based on the residual of the resulting low-rank approximation $\widehat{\mathbf{A}}_l = \widehat{\mathbf{U}}_l \widehat{\boldsymbol{\Sigma}}_l \widehat{\mathbf{V}}_l^*$ from Algorithm 4.

Remark 3.2 (Generality of residual-based bounds). It is worth highlighting that both the statements and the proofs of the posterior guarantees Theorems 3.3 and 3.4 to be presented are algorithm-independent. In contrast to Theorem 3.1 and Proposition 3.2 whose derivation depends explicitly on the algorithm (*e.g.*, assuming $\boldsymbol{\Omega}$ being Gaussian in Algorithm 4), the residual-based bounds in Theorems 3.3 and 3.4 hold for general low-rank approximations $\mathbf{A} \approx \widehat{\mathbf{U}}_l \widehat{\boldsymbol{\Sigma}}_l \widehat{\mathbf{V}}_l^T$.

We start with the following proposition that establishes relations between the canonical angles and the residuals and the true spectrum $\sigma\left(\mathbf{A}\right)$.

Theorem 3.3. *Given any* $\widehat{\mathbf{U}}_l \in \mathbb{C}^{m\times l}$ *and* $\widehat{\mathbf{V}}_l \in \mathbb{C}^{n\times l}$ *with orthonormal columns such that* $\mathrm{Range}\left(\widehat{\mathbf{U}}_l\right) \subseteq \mathrm{col}\left(\mathbf{A}\right)$ *and* $\mathrm{Range}\left(\widehat{\mathbf{V}}_l\right) \subseteq \mathrm{row}\left(\mathbf{A}\right)$*, we have for each* $i = 1, \dots, k$ *(*$k \leq l$*),*

$$\sin\angle_i\left(\mathbf{U}_k, \widehat{\mathbf{U}}_l\right) \leq \min\left\{\frac{\sigma_{k-i+1}\left(\left(\mathbf{I}_m - \widehat{\mathbf{U}}_l\widehat{\mathbf{U}}_l^*\right)\mathbf{A}\right)}{\sigma_k}, \frac{\sigma_1\left(\left(\mathbf{I}_m - \widehat{\mathbf{U}}_l\widehat{\mathbf{U}}_l^*\right)\mathbf{A}\right)}{\sigma_i}\right\}, \tag{3.11}$$

while

$$\sin\angle_i\left(\mathbf{V}_k, \widehat{\mathbf{V}}_l\right) \leq \min\left\{\frac{\sigma_{k-i+1}\left(\mathbf{A}\left(\mathbf{I}_n - \widehat{\mathbf{V}}_l\widehat{\mathbf{V}}_l^*\right)\right)}{\sigma_k}, \frac{\sigma_1\left(\mathbf{A}\left(\mathbf{I}_n - \widehat{\mathbf{V}}_l\widehat{\mathbf{V}}_l^*\right)\right)}{\sigma_i}\right\}. \tag{3.12}$$

Remark 3.3 (Left versus right singular subspaces). When $\widehat{\mathbf{U}}_l$ and $\widehat{\mathbf{V}}_l$ consist of approximated left and right singular vectors from Algorithm 4, upper bounds on $\sin\angle_i\left(\mathbf{V}_k, \widehat{\mathbf{V}}_l\right)$ tend to be smaller than those on $\sin\angle_i\left(\mathbf{U}_k, \widehat{\mathbf{U}}_l\right)$. This is induced by the algorithmic fact that, in Algorithm 4, $\widehat{\mathbf{V}}_l$ is an orthonormal basis of $\mathbf{A}^*\mathbf{Q}_{\mathbf{X}}$, while $\widehat{\mathbf{U}}_l$ and $\mathbf{Q}_{\mathbf{X}}$ are orthonormal bases of $\mathbf{X} = \mathbf{A}\mathbf{\Omega}$. That is, the evaluation of $\widehat{\mathbf{V}}_l$ is enhanced by an additional half power iteration compared with that of $\widehat{\mathbf{U}}_l$, which is also reflected by the differences in exponents on $\mathbf{\Sigma}$ (*i.e.*, $2q+1$ versus $2q+2$) in Theorem 3.1 and Proposition 3.2. This difference can be important especially when q is small (*e.g.*, $q = 0$). When higher accuracy in the left singular subspace is desirable, one can work with $\mathbf{A}^*$.

Proof of Theorem 3.3. Starting with the leading left singular subspace, by definition, for each $i = 1, \dots, k$, we have

$$\begin{aligned}
\sin\angle_i\left(\mathbf{U}_k, \widehat{\mathbf{U}}_l\right) =& \sigma_{k-i+1}\left(\left(\mathbf{I}_m - \widehat{\mathbf{U}}_l\widehat{\mathbf{U}}_l^*\right)\mathbf{U}_k\right) \\
=& \sigma_{k-i+1}\left(\left(\mathbf{I}_m - \widehat{\mathbf{U}}_l\widehat{\mathbf{U}}_l^*\right)\mathbf{A}\left(\mathbf{V}\mathbf{\Sigma}^{-1}\mathbf{U}^*\mathbf{U}_k\right)\right) \\
=& \sigma_{k-i+1}\left(\left(\left(\mathbf{I}_m - \widehat{\mathbf{U}}_l\widehat{\mathbf{U}}_l^*\right)\mathbf{A}\mathbf{V}_k\right)\mathbf{\Sigma}_k^{-1}\right).
\end{aligned}$$

Then, we observe that the following holds simultaneously,

$$\begin{aligned}
\sigma_{k-i+1}\left(\left(\left(\mathbf{I}_m - \widehat{\mathbf{U}}_l\widehat{\mathbf{U}}_l^*\right)\mathbf{A}\mathbf{V}_k\right)\mathbf{\Sigma}_k^{-1}\right) \leq \sigma_1\left(\left(\mathbf{I}_m - \widehat{\mathbf{U}}_l\widehat{\mathbf{U}}_l^*\right)\mathbf{A}\mathbf{V}_k\right)\cdot\sigma_{k-i+1}\left(\mathbf{\Sigma}_k^{-1}\right), \\
\sigma_{k-i+1}\left(\left(\left(\mathbf{I}_m - \widehat{\mathbf{U}}_l\widehat{\mathbf{U}}_l^*\right)\mathbf{A}\mathbf{V}_k\right)\mathbf{\Sigma}_k^{-1}\right) \leq \sigma_{k-i+1}\left(\left(\mathbf{I}_m - \widehat{\mathbf{U}}_l\widehat{\mathbf{U}}_l^*\right)\mathbf{A}\mathbf{V}_k\right)\cdot\sigma_1\left(\mathbf{\Sigma}_k^{-1}\right),
\end{aligned}$$

where $\sigma_{k-i+1}\left(\mathbf{\Sigma}_k^{-1}\right) = 1/\sigma_i$ and $\sigma_1\left(\mathbf{\Sigma}_k^{-1}\right) = 1/\sigma_k$. Finally by Lemma A.2, we have

$$\sigma_{k-i+1}\left(\left(\mathbf{I}_m - \widehat{\mathbf{U}}_l\widehat{\mathbf{U}}_l^*\right)\mathbf{A}\mathbf{V}_k\right) \leq \sigma_{k-i+1}\left(\left(\mathbf{I}_m - \widehat{\mathbf{U}}_l\widehat{\mathbf{U}}_l^*\right)\mathbf{A}\right).$$

Meanwhile, the upper bound for the leading right singular subspace can be derived analogously by observing that

$$\sin\angle_i\left(\mathbf{V}_k,\widehat{\mathbf{V}}_l\right)=\sigma_{k-i+1}\left(\mathbf{V}_k^*\left(\mathbf{I}_n-\widehat{\mathbf{V}}_l\widehat{\mathbf{V}}_l^*\right)\right)=\sigma_{k-i+1}\left(\mathbf{\Sigma}_k^{-1}\mathbf{U}_k^*\mathbf{A}\left(\mathbf{I}_n-\widehat{\mathbf{V}}_l\widehat{\mathbf{V}}_l^*\right)\right).$$

■

As a potential drawback, although the residuals $\left(\mathbf{I}_m-\widehat{\mathbf{U}}_l\widehat{\mathbf{U}}_l^*\right)\mathbf{A}$ and $\mathbf{A}\left(\mathbf{I}_n-\widehat{\mathbf{V}}_l\widehat{\mathbf{V}}_l^*\right)$ in Theorem 3.3 can be evaluated efficiently in $O(mn)$ and $O(mnl)$ time[7], respectively, the exact evaluation of their full spectra can be unaffordable. A straightforward remedy for this problem is to use only the second terms in the right-hand-sides of (3.11) and (3.12) while estimating $\left\|\left(\mathbf{I}_m-\widehat{\mathbf{U}}_l\widehat{\mathbf{U}}_l^*\right)\mathbf{A}\right\|_2$ and $\left\|\mathbf{A}\left(\mathbf{I}_n-\widehat{\mathbf{V}}_l\widehat{\mathbf{V}}_l^*\right)\right\|_2$ with the randomized power method (*cf.* [86], [101] Algorithm 4). Alternatively, we leverage the analysis from [109] Theorem 6.1 and present the following posterior bounds based only on norms of the residuals which can be estimated efficiently via sampling.

Theorem 3.4. *For any rank-l approximation in the SVD form $\mathbf{A}\approx\widehat{\mathbf{U}}_l\widehat{\mathbf{\Sigma}}_l\widehat{\mathbf{V}}_l^*$ (not necessarily obtained by Algorithm 4), recall the notation that $\widehat{\mathbf{U}}_l=\left[\widehat{\mathbf{U}}_k,\widehat{\mathbf{U}}_{l\backslash k}\right]$, $\widehat{\mathbf{V}}_l=\left[\widehat{\mathbf{V}}_k,\widehat{\mathbf{V}}_{l\backslash k}\right]$, while $\widehat{\mathbf{U}}_{m\backslash l},\widehat{\mathbf{U}}_{n\backslash l}$ are the orthogonal complements of $\widehat{\mathbf{U}}_l,\widehat{\mathbf{V}}_l$, respectively. Then, with $\mathbf{E}_{31}\triangleq\widehat{\mathbf{U}}_{m\backslash l}^*\mathbf{A}\widehat{\mathbf{V}}_k$, $\mathbf{E}_{32}\triangleq\widehat{\mathbf{U}}_{m\backslash l}^*\mathbf{A}\widehat{\mathbf{V}}_{l\backslash k}$, and $\mathbf{E}_{33}\triangleq\widehat{\mathbf{U}}_{m\backslash l}^*\mathbf{A}\widehat{\mathbf{V}}_{n\backslash l}$, assuming $\sigma_k>\widehat{\sigma}_{k+1}$ and $\sigma_k>\left\|\mathbf{E}_{33}\right\|_2$, we define the spectral gaps*

$$\gamma_1\triangleq\frac{\sigma_k^2-\widehat{\sigma}_{k+1}^2}{\sigma_k},\qquad\gamma_2\triangleq\frac{\sigma_k^2-\widehat{\sigma}_{k+1}^2}{\widehat{\sigma}_{k+1}},\qquad\Gamma_1=\frac{\sigma_k^2-\left\|\mathbf{E}_{33}\right\|_2^2}{\sigma_k},\qquad\Gamma_2=\frac{\sigma_k^2-\left\|\mathbf{E}_{33}\right\|_2^2}{\left\|\mathbf{E}_{33}\right\|_2}.$$

Then for an arbitrary unitarily invariant norm $|||\cdot|||$,

$$\left|\left|\left|\sin\angle\left(\mathbf{U}_k,\widehat{\mathbf{U}}_l\right)\right|\right|\right|\leq\frac{|||[\mathbf{E}_{31},\mathbf{E}_{32}]|||}{\Gamma_1},\tag{3.13}$$

$$\left|\left|\left|\sin\angle\left(\mathbf{V}_k,\widehat{\mathbf{V}}_l\right)\right|\right|\right|\leq\frac{|||[\mathbf{E}_{31},\mathbf{E}_{32}]|||}{\Gamma_2},\tag{3.14}$$

and specifically for the spectral or Frobenius norm $\left\|\cdot\right\|_\xi$ ($\xi=2,F$),

$$\left\|\sin\angle\left(\mathbf{U}_k,\widehat{\mathbf{U}}_k\right)\right\|_\xi\leq\frac{\left\|[\mathbf{E}_{31},\mathbf{E}_{32}]\right\|_\xi}{\Gamma_1}\sqrt{1+\frac{\left\|\mathbf{E}_{32}\right\|_2^2}{\gamma_2^2}},\tag{3.15}$$

$$\left\|\sin\angle\left(\mathbf{V}_k,\widehat{\mathbf{V}}_k\right)\right\|_\xi\leq\frac{\left\|[\mathbf{E}_{31},\mathbf{E}_{32}]\right\|_\xi}{\Gamma_1}\sqrt{\frac{\left\|\mathbf{E}_{32}\right\|_2^2}{\gamma_1^2}+\frac{\left\|\mathbf{E}_{33}\right\|_2^2}{\sigma_k^2}}.\tag{3.16}$$

[7] For the $O(mn)$ complexity of computing $\left(\mathbf{I}_m-\widehat{\mathbf{U}}_l\widehat{\mathbf{U}}_l^*\right)\mathbf{A}$, we assume that $\widehat{\mathbf{A}}_l=\widehat{\mathbf{U}}_l\widehat{\mathbf{U}}_l^*\mathbf{A}=\widehat{\mathbf{U}}_l\widehat{\mathbf{\Sigma}}_l\widehat{\mathbf{V}}_l^*$ is readily available from Algorithm 4. Otherwise (*e.g.*, when Algorithm 4 returns $\left(\widehat{\mathbf{U}}_l,\widehat{\mathbf{\Sigma}}_l,\widehat{\mathbf{V}}_l\right)$ but not $\widehat{\mathbf{A}}_l$), the evaluation of $\left(\mathbf{I}_m-\widehat{\mathbf{U}}_l\widehat{\mathbf{U}}_l^*\right)\mathbf{A}$ will inevitably take $O(mnl)$ as that of $\mathbf{A}\left(\mathbf{I}_n-\widehat{\mathbf{V}}_l\widehat{\mathbf{V}}_l^*\right)$.

Furthermore, for all $i \in [k]$,

$$\sin \angle_i \left(\mathbf{U}_k, \widehat{\mathbf{U}}_l\right) \leq \frac{\sigma_k}{\sigma_{k-i+1}} \cdot \frac{\|[\mathbf{E}_{31}, \mathbf{E}_{32}]\|_2}{\Gamma_1}, \tag{3.17}$$

$$\sin \angle_i \left(\mathbf{V}_k, \widehat{\mathbf{V}}_l\right) \leq \frac{\sigma_k}{\sigma_{k-i+1}} \cdot \frac{\|[\mathbf{E}_{31}, \mathbf{E}_{32}]\|_2}{\Gamma_2}, \tag{3.18}$$

$$\sin \angle_i \left(\mathbf{U}_k, \widehat{\mathbf{U}}_k\right) \leq \frac{\|[\mathbf{E}_{31}, \mathbf{E}_{32}]\|_2}{\Gamma_1} \sqrt{1 + \left(\frac{\sigma_k}{\sigma_{k-i+1}} \cdot \frac{\|\mathbf{E}_{32}\|_2}{\gamma_2}\right)^2}, \tag{3.19}$$

$$\sin \angle_i \left(\mathbf{V}_k, \widehat{\mathbf{V}}_k\right) \leq \frac{\|[\mathbf{E}_{31}, \mathbf{E}_{32}]\|_2}{\Gamma_1} \sqrt{\left(\frac{\sigma_k}{\sigma_{k-i+1}} \cdot \frac{\|\mathbf{E}_{32}\|_2}{\gamma_1}\right)^2 + \left(\frac{\|\mathbf{E}_{33}\|_2}{\sigma_k}\right)^2}. \tag{3.20}$$

In practice, norms of the residuals can be computed as

$$\begin{aligned}
&|||[\mathbf{E}_{31}, \mathbf{E}_{32}]||| = \left|\left|\left|\widehat{\mathbf{U}}_{m\backslash l}^* \mathbf{A} \widehat{\mathbf{V}}_l\right|\right|\right| = \left|\left|\left|\left(\mathbf{I}_m - \widehat{\mathbf{U}}_l \widehat{\mathbf{U}}_l^*\right) \mathbf{A} \widehat{\mathbf{V}}_l\right|\right|\right| = \left|\left|\left|\left(\mathbf{A} - \widehat{\mathbf{A}}_l\right) \widehat{\mathbf{V}}_l\right|\right|\right|, \\
&\|\mathbf{E}_{32}\|_2 = \left\|\widehat{\mathbf{U}}_{m\backslash l}^* \mathbf{A} \widehat{\mathbf{V}}_{l\backslash k}\right\|_2 = \left\|\left(\mathbf{A} - \widehat{\mathbf{A}}_l\right) \widehat{\mathbf{V}}_{l\backslash k}\right\|_2, \\
&|||\mathbf{E}_{33}|||_2 = \left\|\widehat{\mathbf{U}}_{m\backslash l}^* \mathbf{A} \widehat{\mathbf{V}}_{n\backslash l}\right\| = \left\|\left(\mathbf{A} - \widehat{\mathbf{A}}_l\right) \left(\mathbf{I}_n - \widehat{\mathbf{V}}_l \widehat{\mathbf{V}}_l^*\right)\right\|_2 = \left\|\mathbf{A} - \mathbf{A} \widehat{\mathbf{V}}_l \widehat{\mathbf{V}}_l^*\right\|_2,
\end{aligned}$$

where the construction of $\left(\mathbf{A} - \widehat{\mathbf{A}}_l\right) \widehat{\mathbf{V}}_l$, $\left(\mathbf{A} - \widehat{\mathbf{A}}_l\right) \widehat{\mathbf{V}}_{l\backslash k}$, and $\mathbf{A} - \mathbf{A}\widehat{\mathbf{V}}_l \widehat{\mathbf{V}}_l^*$ takes $O\left(mnl\right)$ time, while the respective norms can be estimated efficiently via sampling (*cf.* [101] Algorithm 1-4, [105] Algorithm 1-3, etc.).

The proof of Theorem 3.4 is similar to that of [109, Thm. 6.1].

Proof of Theorem 3.4. Let $\widetilde{\mathbf{U}}_{11} \triangleq \widehat{\mathbf{U}}_k^* \mathbf{U}_k$, $\widetilde{\mathbf{U}}_{21} \triangleq \widehat{\mathbf{U}}_{l\backslash k}^* \mathbf{U}_k$, $\widetilde{\mathbf{U}}_{31} \triangleq \widehat{\mathbf{U}}_{m\backslash l}^* \mathbf{U}_k$, and $\widetilde{\mathbf{V}}_{11} \triangleq \widehat{\mathbf{V}}_k^* \mathbf{V}_k$, $\widetilde{\mathbf{V}}_{21} \triangleq \widehat{\mathbf{V}}_{l\backslash k}^* \mathbf{V}_k$, and $\widetilde{\mathbf{V}}_{31} \triangleq \widehat{\mathbf{V}}_{n\backslash l}^* \mathbf{V}_k$. We start by expressing the canonical angles in terms of $\widetilde{\mathbf{U}}_{31}$ and $\widetilde{\mathbf{U}}_{21}$:

$$\sin \angle \left(\mathbf{U}_k, \widehat{\mathbf{U}}_l\right) = \sigma\left(\widehat{\mathbf{U}}_{n\backslash l}^* \mathbf{U}_k\right) = \sigma\left(\widetilde{\mathbf{U}}_{31}\right),$$

$$\sin \angle \left(\mathbf{V}_k, \widehat{\mathbf{V}}_l\right) = \sigma\left(\widehat{\mathbf{V}}_{n\backslash l}^* \mathbf{V}_k\right) = \sigma\left(\widetilde{\mathbf{V}}_{31}\right),$$

$$\sin \angle \left(\mathbf{U}_k, \widehat{\mathbf{U}}_k\right) = \sigma\left(\begin{bmatrix} \widehat{\mathbf{U}}_{l\backslash k}^* \\ \widehat{\mathbf{U}}_{m\backslash l}^* \end{bmatrix} \mathbf{U}_k\right) = \sigma\left(\begin{bmatrix} \widetilde{\mathbf{U}}_{21} \\ \widetilde{\mathbf{U}}_{31} \end{bmatrix}\right),$$

$$\sin \angle \left(\mathbf{V}_k, \widehat{\mathbf{V}}_k\right) = \sigma\left(\begin{bmatrix} \widehat{\mathbf{V}}_{l\backslash k}^* \\ \widehat{\mathbf{V}}_{n\backslash l}^* \end{bmatrix} \mathbf{V}_k\right) = \sigma\left(\begin{bmatrix} \widetilde{\mathbf{V}}_{21} \\ \widetilde{\mathbf{V}}_{31} \end{bmatrix}\right).$$

By observing that for any rank-l approximation in the SVD form $\mathbf{A} \approx \widehat{\mathbf{U}}_l \widehat{\mathbf{\Sigma}}_l \widehat{\mathbf{V}}_l^*$,

$$\mathbf{A} = \widehat{\mathbf{U}}_l \widehat{\mathbf{\Sigma}}_l \widehat{\mathbf{V}}_l^* + \widehat{\mathbf{U}}_{m\setminus l} \widehat{\mathbf{U}}_{m\setminus l}^* \mathbf{A} = \begin{bmatrix} \widehat{\mathbf{U}}_k & \widehat{\mathbf{U}}_{l\setminus k} & \widehat{\mathbf{U}}_{m\setminus l} \end{bmatrix} \begin{bmatrix} \widehat{\mathbf{\Sigma}}_k & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \widehat{\mathbf{\Sigma}}_{l\setminus k} & \mathbf{0} \\ \mathbf{E}_{31} & \mathbf{E}_{32} & \mathbf{E}_{33} \end{bmatrix} \begin{bmatrix} \widehat{\mathbf{V}}_k^* \\ \widehat{\mathbf{V}}_{l\setminus k}^* \\ \widehat{\mathbf{V}}_{n\setminus l}^* \end{bmatrix},$$

we left multiply $\widehat{\mathbf{U}}^* = \left[\widehat{\mathbf{U}}_k^*; \widehat{\mathbf{U}}_{l\setminus k}^*; \widehat{\mathbf{U}}_{m\setminus l}^*\right] \in \mathbb{C}^{m\times m}$ and right multiply $\mathbf{V}_k$ on both sides and get

$$\begin{bmatrix} \widetilde{\mathbf{U}}_{11} \\ \widetilde{\mathbf{U}}_{21} \\ \widetilde{\mathbf{U}}_{31} \end{bmatrix} \mathbf{\Sigma}_k = \begin{bmatrix} \widehat{\mathbf{\Sigma}}_k & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \widehat{\mathbf{\Sigma}}_{l\setminus k} & \mathbf{0} \\ \mathbf{E}_{31} & \mathbf{E}_{32} & \mathbf{E}_{33} \end{bmatrix} \begin{bmatrix} \widetilde{\mathbf{V}}_{11} \\ \widetilde{\mathbf{V}}_{21} \\ \widetilde{\mathbf{V}}_{31} \end{bmatrix}, \tag{3.21}$$

while left multiplying $\mathbf{U}_k^*$ and right multiplying $\widehat{\mathbf{V}} = \left[\widehat{\mathbf{V}}_k, \widehat{\mathbf{V}}_{l\setminus k}, \widehat{\mathbf{V}}_{n\setminus l}\right]$ yield

$$\mathbf{\Sigma}_k \begin{bmatrix} \widetilde{\mathbf{V}}_{11}^* & \widetilde{\mathbf{V}}_{21}^* & \widetilde{\mathbf{V}}_{31}^* \end{bmatrix} = \begin{bmatrix} \widetilde{\mathbf{U}}_{11}^* & \widetilde{\mathbf{U}}_{21}^* & \widetilde{\mathbf{U}}_{31}^* \end{bmatrix} \begin{bmatrix} \widehat{\mathbf{\Sigma}}_k & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \widehat{\mathbf{\Sigma}}_{l\setminus k} & \mathbf{0} \\ \mathbf{E}_{31} & \mathbf{E}_{32} & \mathbf{E}_{33} \end{bmatrix}. \tag{3.22}$$

Bounding $\sigma\left(\widetilde{\mathbf{U}}_{31}\right)$ and $\sigma\left(\widetilde{\mathbf{V}}_{31}\right)$. To bound $\sigma\left(\widetilde{\mathbf{U}}_{31}\right)$, we observe the following from the third row of (3.21) and the third column of (3.22),

$$\widetilde{\mathbf{U}}_{31}\mathbf{\Sigma}_k = \mathbf{E}_{31}\widetilde{\mathbf{V}}_{11} + \mathbf{E}_{32}\widetilde{\mathbf{V}}_{21} + \mathbf{E}_{33}\widetilde{\mathbf{V}}_{31}, \quad \widetilde{\mathbf{U}}_{31}^*\mathbf{E}_{33} = \mathbf{\Sigma}_k\widetilde{\mathbf{V}}_{31}^*.$$

Noticing that $\left[\widetilde{\mathbf{V}}_{11}; \widetilde{\mathbf{V}}_{21}\right] = \widehat{\mathbf{V}}_l^*\mathbf{V}_k$ and $\left\|\widehat{\mathbf{V}}_l^*\mathbf{V}_k\right\|_2 \leq 1$, we have

$$\begin{aligned} \left|\!\left|\!\left|\widetilde{\mathbf{U}}_{31}\mathbf{\Sigma}_k\right|\!\right|\!\right| \leq& \left|\!\left|\!\left|[\mathbf{E}_{31}, \mathbf{E}_{32}]\right|\!\right|\!\right| \left\|\widehat{\mathbf{V}}_l^*\mathbf{V}_k\right\|_2 + \left|\!\left|\!\left|\mathbf{E}_{33}\mathbf{E}_{33}^*\widetilde{\mathbf{U}}_{31}\mathbf{\Sigma}_k^{-1}\right|\!\right|\!\right| \\ \leq& \left|\!\left|\!\left|[\mathbf{E}_{31}, \mathbf{E}_{32}]\right|\!\right|\!\right| + \frac{\|\mathbf{E}_{33}\|_2^2}{\sigma_k^2}\left|\!\left|\!\left|\widetilde{\mathbf{U}}_{31}\mathbf{\Sigma}_k\right|\!\right|\!\right| \end{aligned}$$

for all $i \in [\min(k, m-l)]$, which implies that

$$\left|\!\left|\!\left|\widetilde{\mathbf{U}}_{31}\mathbf{\Sigma}_k\right|\!\right|\!\right| \leq \left(1 - \frac{\|\mathbf{E}_{33}\|_2^2}{\sigma_k^2}\right)^{-1} \left|\!\left|\!\left|[\mathbf{E}_{31}, \mathbf{E}_{32}]\right|\!\right|\!\right| = \sigma_k \cdot \frac{\left|\!\left|\!\left|[\mathbf{E}_{31}, \mathbf{E}_{32}]\right|\!\right|\!\right|}{\Gamma_1},$$

and leads to

$$\begin{aligned} \left|\!\left|\!\left|\widetilde{\mathbf{U}}_{31}\right|\!\right|\!\right| \leq& \frac{1}{\sigma_k}\left|\!\left|\!\left|\widetilde{\mathbf{U}}_{31}\mathbf{\Sigma}_k\right|\!\right|\!\right| \leq \frac{\left|\!\left|\!\left|[\mathbf{E}_{31}, \mathbf{E}_{32}]\right|\!\right|\!\right|}{\Gamma_1}, \\ \sigma_i\left(\widetilde{\mathbf{U}}_{31}\right) \leq& \frac{1}{\sigma_{k-i+1}}\left\|\widetilde{\mathbf{U}}_{31}\mathbf{\Sigma}_k\right\|_2 \leq \frac{\sigma_k}{\sigma_{k-i+1}} \cdot \frac{\|[\mathbf{E}_{31}, \mathbf{E}_{32}]\|_2}{\Gamma_1} \quad \forall\, i \in [k], \end{aligned}$$

where the second line follows from Lemma A.3. These lead to Equations (3.13) and (3.17)

To bound $\sigma\left(\widetilde{\mathbf{V}}_{31}\right)$, we use the relation $\widetilde{\mathbf{U}}_{31}^*\mathbf{E}_{33} = \mathbf{\Sigma}_k\widetilde{\mathbf{V}}_{31}^*$,

$$\begin{aligned} \left|\!\left|\!\left|\widetilde{\mathbf{V}}_{31}\right|\!\right|\!\right| \leq& \frac{\|\mathbf{E}_{33}\|_2}{\sigma_k}\left|\!\left|\!\left|\widetilde{\mathbf{U}}_{31}\right|\!\right|\!\right| \leq \frac{\left|\!\left|\!\left|[\mathbf{E}_{31}, \mathbf{E}_{32}]\right|\!\right|\!\right|}{\Gamma_2}, \\ \sigma_i\left(\widetilde{\mathbf{V}}_{31}\right) \leq& \frac{1}{\sigma_k}\sigma_i\left(\mathbf{E}_{33}^*\widetilde{\mathbf{U}}_{31}\right) \leq \frac{\|\mathbf{E}_{33}\|_2}{\sigma_k}\sigma_i\left(\widetilde{\mathbf{U}}_{31}\right) \leq \frac{\sigma_k}{\sigma_{k-i+1}} \cdot \frac{\|[\mathbf{E}_{31}, \mathbf{E}_{32}]\|_2}{\Gamma_2} \quad \forall\, i \in [k]. \end{aligned}$$

We therefore have upper bounds Equations (3.14) and (3.18).

Bounding $\sigma\left(\widetilde{\mathbf{U}}_{21}\right)$ and $\sigma\left(\widetilde{\mathbf{V}}_{21}\right)$. To bound $\sigma\left(\widetilde{\mathbf{U}}_{21}\right)$, we leverage the second row of (3.21) and the second column of (3.22),

$$\widetilde{\mathbf{U}}_{21}\boldsymbol{\Sigma}_k = \widehat{\boldsymbol{\Sigma}}_{l\setminus k}\widetilde{\mathbf{V}}_{21}, \quad \boldsymbol{\Sigma}_k\widetilde{\mathbf{V}}_{21}^* = \widetilde{\mathbf{U}}_{21}^*\widehat{\boldsymbol{\Sigma}}_{l\setminus k} + \widetilde{\mathbf{U}}_{31}^*\mathbf{E}_{32}.$$

Up to rearrangement, we observe that

$$\begin{aligned}\left\vert\!\left\vert\!\left\vert\widetilde{\mathbf{U}}_{21}\boldsymbol{\Sigma}_k\right\vert\!\right\vert\!\right\vert =& \left\vert\!\left\vert\!\left\vert\widehat{\boldsymbol{\Sigma}}_{l\setminus k}\left(\widehat{\boldsymbol{\Sigma}}_{l\setminus k}\widetilde{\mathbf{U}}_{21} + \mathbf{E}_{32}^*\widetilde{\mathbf{U}}_{31}\right)\boldsymbol{\Sigma}_k^{-1}\right\vert\!\right\vert\!\right\vert \\ \leq& \frac{\left\|\widehat{\boldsymbol{\Sigma}}_{l\setminus k}\right\|_2^2}{\sigma_k^2}\left\vert\!\left\vert\!\left\vert\widetilde{\mathbf{U}}_{21}\boldsymbol{\Sigma}_k\right\vert\!\right\vert\!\right\vert + \frac{\left\|\widehat{\boldsymbol{\Sigma}}_{l\setminus k}\right\|_2}{\sigma_k}\left\vert\!\left\vert\!\left\vert\mathbf{E}_{32}^*\widetilde{\mathbf{U}}_{31}\right\vert\!\right\vert\!\right\vert,\end{aligned}$$

which implies that

$$\left\vert\!\left\vert\!\left\vert\widetilde{\mathbf{U}}_{21}\boldsymbol{\Sigma}_k\right\vert\!\right\vert\!\right\vert \leq \left(1 - \frac{\widehat{\sigma}_{k+1}^2}{\sigma_k^2}\right)^{-1}\frac{\widehat{\sigma}_{k+1}}{\sigma_k}\left\vert\!\left\vert\!\left\vert\mathbf{E}_{32}^*\widetilde{\mathbf{U}}_{31}\right\vert\!\right\vert\!\right\vert \leq \sigma_k \cdot \frac{\|\mathbf{E}_{32}\|_2}{\gamma_2}\left\vert\!\left\vert\!\left\vert\widetilde{\mathbf{U}}_{31}\right\vert\!\right\vert\!\right\vert,$$

and therefore, with Lemma A.3, for all $i \in [k]$,

$$\begin{aligned}\left\vert\!\left\vert\!\left\vert\widetilde{\mathbf{U}}_{21}\right\vert\!\right\vert\!\right\vert &\leq \frac{1}{\sigma_k}\left\vert\!\left\vert\!\left\vert\widetilde{\mathbf{U}}_{21}\boldsymbol{\Sigma}_k\right\vert\!\right\vert\!\right\vert \leq \frac{\|\mathbf{E}_{32}\|_2}{\gamma_2}\left\vert\!\left\vert\!\left\vert\widetilde{\mathbf{U}}_{31}\right\vert\!\right\vert\!\right\vert, \\ \sigma_i\left(\widetilde{\mathbf{U}}_{21}\right) &\leq \frac{1}{\sigma_{k-i+1}}\left\|\widetilde{\mathbf{U}}_{21}\boldsymbol{\Sigma}_k\right\|_2 \leq \frac{\sigma_k}{\sigma_{k-i+1}}\cdot\frac{\|\mathbf{E}_{32}\|_2}{\gamma_2}\left\|\widetilde{\mathbf{U}}_{31}\right\|_2.\end{aligned}$$

Then, with the stronger inequality for the spectral or Frobenius norm $\|\cdot\|_\xi$ ($\xi = 2, F$),

$$\begin{aligned}\left\|\begin{bmatrix}\widetilde{\mathbf{U}}_{21} \\ \widetilde{\mathbf{U}}_{31}\end{bmatrix}\right\|_\xi \leq& \sqrt{\left\|\widetilde{\mathbf{U}}_{31}\right\|_\xi^2 + \left\|\widetilde{\mathbf{U}}_{21}\right\|_\xi^2} \leq \left\|\widetilde{\mathbf{U}}_{31}\right\|_\xi\sqrt{1 + \frac{\|\mathbf{E}_{32}\|_2^2}{\gamma_2^2}} \\ \leq& \frac{\|[\mathbf{E}_{31}, \mathbf{E}_{32}]\|_\xi}{\Gamma_1}\sqrt{1 + \frac{\|\mathbf{E}_{32}\|_2^2}{\gamma_2^2}}\end{aligned}$$

leads to (3.15). Meanwhile for (3.19), the individual canonical angles are upper bounded by

$$\begin{aligned}\sigma_i\left(\begin{bmatrix}\widetilde{\mathbf{U}}_{21} \\ \widetilde{\mathbf{U}}_{31}\end{bmatrix}\right) =& \sqrt{\sigma_i\left(\widetilde{\mathbf{U}}_{21}^*\widetilde{\mathbf{U}}_{21} + \widetilde{\mathbf{U}}_{31}^*\widetilde{\mathbf{U}}_{31}\right)} \\ \text{(Lemma A.4)} \qquad \leq& \sqrt{\left\|\widetilde{\mathbf{U}}_{31}\right\|_2^2 + \sigma_i\left(\widetilde{\mathbf{U}}_{21}\right)^2} \\ \leq& \left\|\widetilde{\mathbf{U}}_{31}\right\|_2\sqrt{1 + \left(\frac{\sigma_k}{\sigma_{k-i+1}}\cdot\frac{\|\mathbf{E}_{32}\|_2}{\gamma_2}\right)^2} \\ \leq& \frac{\|[\mathbf{E}_{31}, \mathbf{E}_{32}]\|_2}{\Gamma_1}\sqrt{1 + \left(\frac{\sigma_k}{\sigma_{k-i+1}}\cdot\frac{\|\mathbf{E}_{32}\|_2}{\gamma_2}\right)^2}.\end{aligned}$$

Analogously, by observing that

$$\left\vert\left\vert\left\vert \widetilde{\mathbf{V}}_{21}\mathbf{\Sigma}_k \right\vert\right\vert\right\vert = \left\vert\left\vert\left\vert \widehat{\mathbf{\Sigma}}_{l\setminus k}^2 \widetilde{\mathbf{V}}_{21}\mathbf{\Sigma}_k^{-1} + \mathbf{E}_{32}^* \widetilde{\mathbf{U}}_{31} \right\vert\right\vert\right\vert \leq \frac{\left\| \widehat{\mathbf{\Sigma}}_{l\setminus k} \right\|_2^2}{\sigma_k^2} \left\vert\left\vert\left\vert \widetilde{\mathbf{V}}_{21}\mathbf{\Sigma}_k \right\vert\right\vert\right\vert + \left\vert\left\vert\left\vert \mathbf{E}_{32}^* \widetilde{\mathbf{U}}_{31} \right\vert\right\vert\right\vert,$$

we have that, by Lemma A.3, for all $i \in [k]$,

$$\begin{aligned}
&\left\vert\left\vert\left\vert \widetilde{\mathbf{V}}_{21}\mathbf{\Sigma}_k \right\vert\right\vert\right\vert \leq \left(1 - \frac{\widehat{\sigma}_{k+1}^2}{\sigma_k^2}\right)^{-1} \left\vert\left\vert\left\vert \mathbf{E}_{32}^* \widetilde{\mathbf{U}}_{31} \right\vert\right\vert\right\vert \leq \sigma_k \cdot \frac{\|\mathbf{E}_{32}\|_2}{\gamma_1} \left\vert\left\vert\left\vert \widetilde{\mathbf{U}}_{31} \right\vert\right\vert\right\vert, \\
&\left\vert\left\vert\left\vert \widetilde{\mathbf{V}}_{21} \right\vert\right\vert\right\vert \leq \frac{1}{\sigma_k} \left\vert\left\vert\left\vert \widetilde{\mathbf{V}}_{21}\mathbf{\Sigma}_k \right\vert\right\vert\right\vert \leq \frac{\|\mathbf{E}_{32}\|_2}{\gamma_1} \left\vert\left\vert\left\vert \widetilde{\mathbf{U}}_{31} \right\vert\right\vert\right\vert, \\
&\sigma_i\left(\widetilde{\mathbf{V}}_{21}\right) \leq \frac{1}{\sigma_{k-i+1}} \left\| \widetilde{\mathbf{V}}_{21}\mathbf{\Sigma}_k \right\|_2 \leq \frac{\sigma_k}{\sigma_{k-i+1}} \cdot \frac{\|\mathbf{E}_{32}\|_2}{\gamma_1} \left\| \widetilde{\mathbf{U}}_{31} \right\|_2,
\end{aligned}$$

and therefore for the spectral or Frobenius norm $\|\cdot\|_\xi$ ($\xi = 2, F$),

$$\begin{aligned}
\left\| \begin{bmatrix} \widetilde{\mathbf{V}}_{21} \\ \widetilde{\mathbf{V}}_{31} \end{bmatrix} \right\|_\xi &\leq \sqrt{\left\| \widetilde{\mathbf{V}}_{21} \right\|_\xi^2 + \left\| \widetilde{\mathbf{V}}_{31} \right\|_\xi^2} \leq \left\| \widetilde{\mathbf{U}}_{31} \right\|_\xi \sqrt{\frac{\|\mathbf{E}_{32}\|_2^2}{\gamma_1^2} + \frac{\|\mathbf{E}_{33}\|_2^2}{\sigma_k^2}} \\
&\leq \frac{\|[\mathbf{E}_{31}, \mathbf{E}_{32}]\|_\xi}{\Gamma_1} \sqrt{\frac{\|\mathbf{E}_{32}\|_2^2}{\gamma_1^2} + \frac{\|\mathbf{E}_{33}\|_2^2}{\sigma_k^2}},
\end{aligned}$$

which leads to (3.16). Additionally for individual canonical angles $i \in [k]$,

$$\begin{aligned}
\sigma_i\left(\begin{bmatrix} \widetilde{\mathbf{V}}_{21} \\ \widetilde{\mathbf{V}}_{31} \end{bmatrix} \right) &\leq \sqrt{\sigma_i\left(\widetilde{\mathbf{V}}_{21}\right)^2 + \left\| \widetilde{\mathbf{V}}_{31} \right\|_2^2} \qquad \text{(Lemma A.4)} \\
&\leq \left\| \widetilde{\mathbf{U}}_{31} \right\|_2 \sqrt{\left(\frac{\sigma_k}{\sigma_{k-i+1}} \cdot \frac{\|\mathbf{E}_{32}\|_2}{\gamma_1} \right)^2 + \left(\frac{\|\mathbf{E}_{33}\|_2}{\sigma_k} \right)^2} \\
&\leq \frac{\|[\mathbf{E}_{31}, \mathbf{E}_{32}]\|_2}{\Gamma_1} \sqrt{\left(\frac{\sigma_k}{\sigma_{k-i+1}} \cdot \frac{\|\mathbf{E}_{32}\|_2}{\gamma_1} \right)^2 + \left(\frac{\|\mathbf{E}_{33}\|_2}{\sigma_k} \right)^2}.
\end{aligned}$$

This yields (3.20) and completes the proof. ■

3.6 Numerical Experiments

First, we present numerical comparisons among different canonical angle upper bounds and the unbiased estimates on the left and right leading singular subspaces of various synthetic and real data matrices. We start by describing the target matrices in Section 3.6.1. In Section 3.6.2, we discuss the performance of the unbiased estimates, as well as the relative tightness of the canonical angle bounds, for different algorithmic choices based on the numerical observations. Second, in Section 3.6.3, we present an illustrative example that provides insight into the balance between oversampling and power iterations brought by the space-agnostic bounds.

3.6.1 Target Matrices

We consider several different classes of target matrices, including some synthetic random matrices with different spectral patterns, as well as an empirical dataset, as summarized below:

1. A random sparse non-negative (SNN) matrix [138] $\mathbf{A}$ of size $m \times n$ takes the form,

$$\mathbf{A} = \text{SNN}\left(a, r_1\right) := \sum_{i=1}^{r_1} \frac{a}{i} \mathbf{x}_i \mathbf{y}_i^T + \sum_{i=r_1+1}^{\min(m,n)} \frac{1}{i} \mathbf{x}_i \mathbf{y}_i^T \tag{3.23}$$

   where $a > 1$ and $r_1 < \min\left(m, n\right)$ control the spectral decay, and $\mathbf{x}_i \in \mathbb{C}^m$, $\mathbf{y}_i \in \mathbb{C}^n$ are random sparse vectors with non-negative entries. In the experiments, we test on two random SNN matrices of size 500×500 with $r_1 = 20$ and $a = 1, 100$, respectively.

2. Gaussian dense matrices with controlled spectral decay are randomly generated via a similar construction to the SNN matrix, with $\mathbf{x}_j \in \mathbb{S}^{m-1}$ and $\mathbf{y}_j \in \mathbb{S}^{n-1}$ in (3.23) replaced by uniformly random dense orthonormal vectors. The generating procedures for $\mathbf{A} \in \mathbb{C}^{m \times n}$ with rank $r \leq \min\left(m, n\right)$ can be summarized as following:
   (i) Draw Gaussian random matrices, $\mathbf{G}_m \in \mathbb{C}^{m \times r}$ and $\mathbf{G}_n \in \mathbb{C}^{m \times r}$.
   (ii) Compute $\mathbf{U} = \text{ortho}(\mathbf{G}_m) \in \mathbb{C}^{m \times r}$, $\mathbf{V} = \text{ortho}(\mathbf{G}_n) \in \mathbb{C}^{n \times r}$ as orthonormal bases.
   (iii) Given the spectrum $\mathbf{\Sigma} = \text{diag}\left(\sigma_1, \ldots, \sigma_r\right)$, we construct $\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^*$.

   In the experiments, we consider two types of spectral decay:
   (i) slower decay with $r_1 = 20$, $\sigma_i = 1$ for all $i \in [r_1]$, $\sigma_i = 1/\sqrt{i - r_1 + 1}$ for all $i = r_1 + 1, \ldots, r$, and
   (ii) faster decay with $r_1 = 20$, $\sigma_i = 1$ for all $i \in [r_1]$, $\sigma_i = \max(0.99^{i-r_1}, 10^{-3})$ for all $i = r_1 + 1, \ldots, r$.

3. MNIST training set consists of $60,000$ images of hand-written digits from 0 to 9. Each image is of size 28×28. We form the target matrices by uniformly sampling $N = 800$ images from the MNIST training set. The images are flattened and normalized to form a full-rank matrix of size $N \times d$ where $d = 784$ is the size of the flattened images, with entries bounded in $[0, 1]$. The nonzero entries take approximately 20% of the matrix for both the training and the testing sets.

3.6.2 Canonical Angle Bounds and Estimates

Now we present numerical comparisons of the performance of the canonical angle bounds and the unbiased estimates under different algorithmic choices. Considering the scenario where the true matrix spectra may not be available in practice, we calculate two sets of upper bounds, one from the true spectra $\Sigma \in \mathbb{C}^{r\times r}$ and the other from the l approximated singular values from Algorithm 4. For the later, we pad the approximated spectrum $\widehat{\Sigma}_l = \mathrm{diag}\left(\widehat{\sigma}_1, \ldots, \widehat{\sigma}_l\right)$ with $r-l$ copies of $\widehat{\sigma}_l$ and evaluate the canonical angle bounds and estimates with $\widetilde{\Sigma} = \mathrm{diag}\left(\widehat{\sigma}_1, \ldots, \widehat{\sigma}_l, \ldots, \widehat{\sigma}_l\right) \in \mathbb{C}^{r\times r}$.

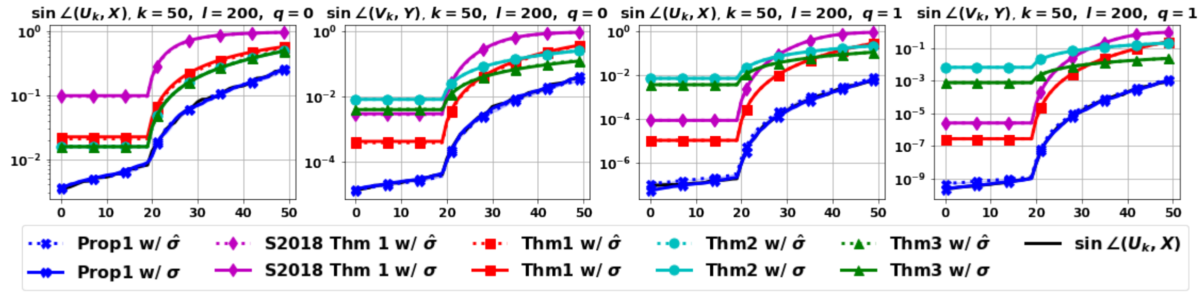


Figure 3.1: Synthetic Gaussian with the slower spectral decay. $k = 50$, $l = 200$, $q = 0, 1$.

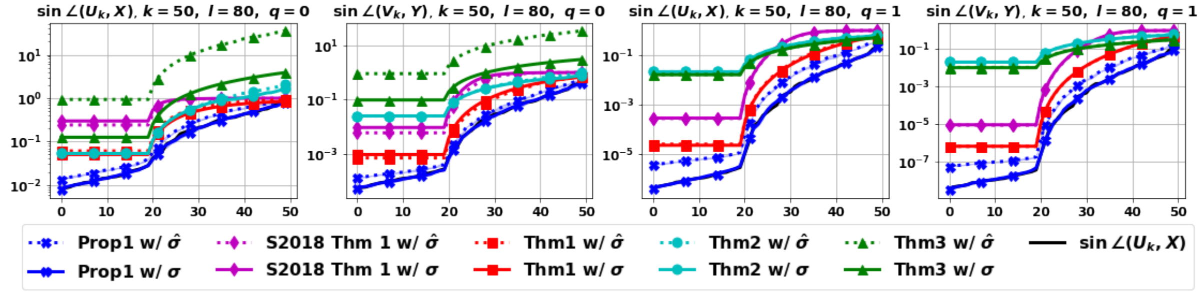


Figure 3.2: Synthetic Gaussian with the slower spectral decay. $k = 50$, $l = 80$, $q = 0, 1$.

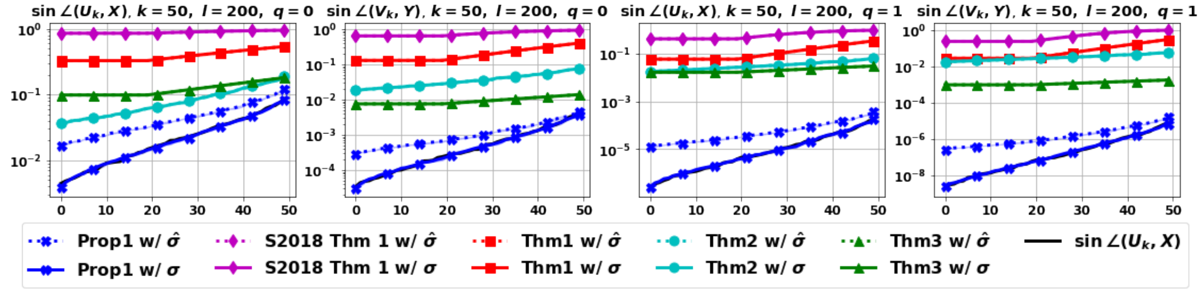


Figure 3.3: Synthetic Gaussian with the faster spectral decay. $k = 50$, $l = 200$, $q = 0, 1$.

From Figure 3.1 to Figure 3.11,

1. Red lines and dashes (**Thm1 w/ $\boldsymbol{\sigma}$ and $\widehat{\boldsymbol{\sigma}}$**) represent the space-agnostic probabilistic bounds in Theorem 3.1 evaluated with the true (lines) and approximated (dashes) singular

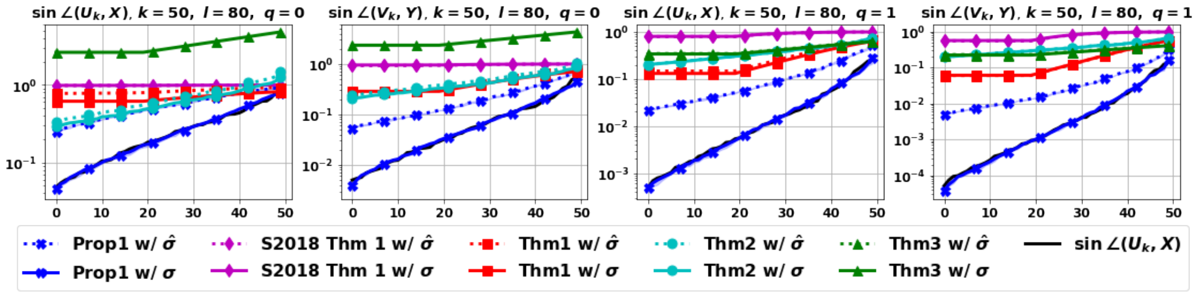


Figure 3.4: Synthetic Gaussian with the faster spectral decay. $k = 50$, $l = 80$, $q = 0, 1$.

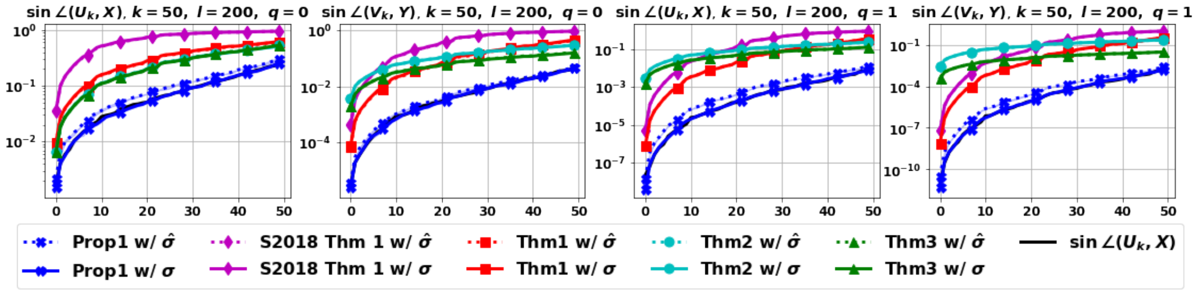


Figure 3.5: SNN with $r_1 = 20$, $a = 1$. $k = 50$, $l = 200$, $q = 0, 1$.

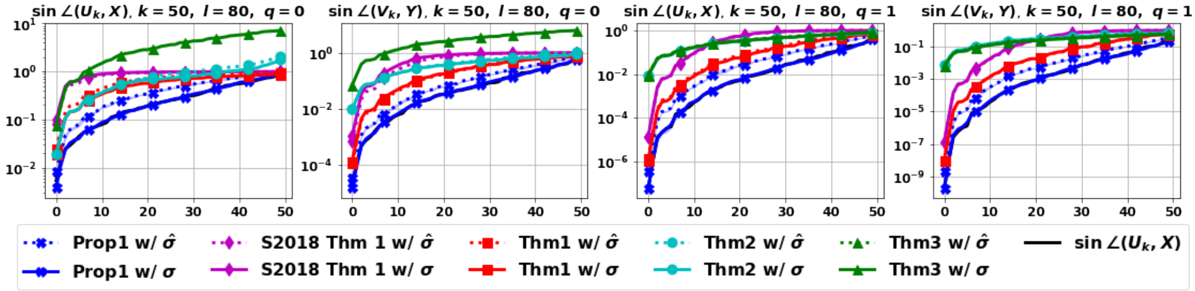


Figure 3.6: SNN with $r_1 = 20$, $a = 1$. $k = 50$, $l = 80$, $q = 0, 1$.

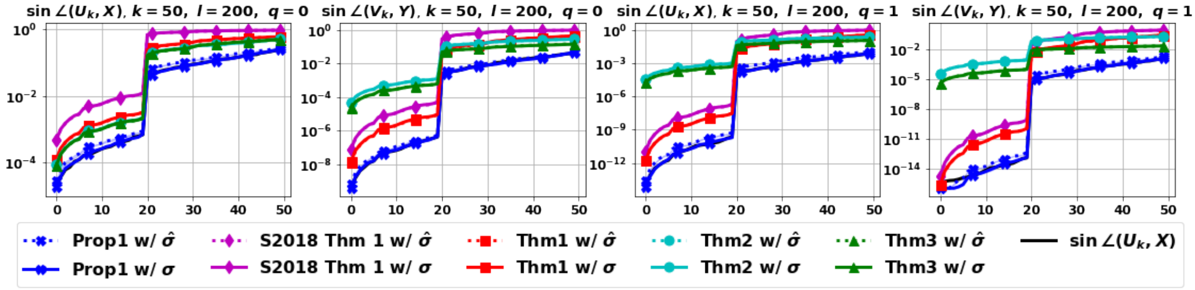


Figure 3.7: SNN with $r_1 = 20$, $a = 100$. $k = 50$, $l = 200$, $q = 0, 1$.

values, Σ, and $\widetilde{\Sigma}$, respectively, where we simply ignore tail decay and suppress constants for the distortion factors and set $\epsilon_1 = \sqrt{\frac{k}{l}}$ and $\epsilon_2 = \sqrt{\frac{l}{r-k}}$ in Equations (3.2) and (3.3);

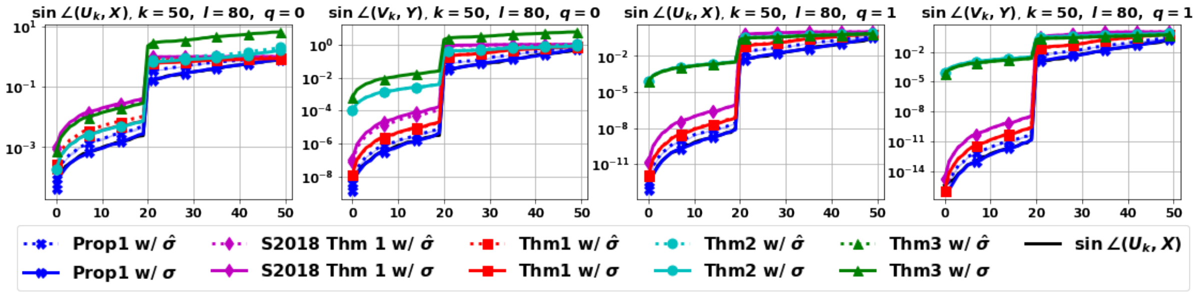


Figure 3.8: SNN with $r_1 = 20$, $a = 100$. $k = 50$, $l = 80$, $q = 0, 1$.

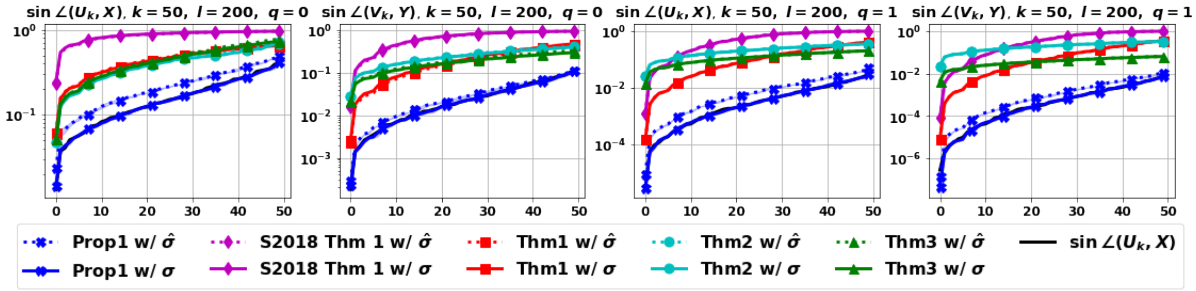


Figure 3.9: 800 randomly sampled images from the MNIST training set. $k = 50$, $l = 200$, $q = 0, 1$.

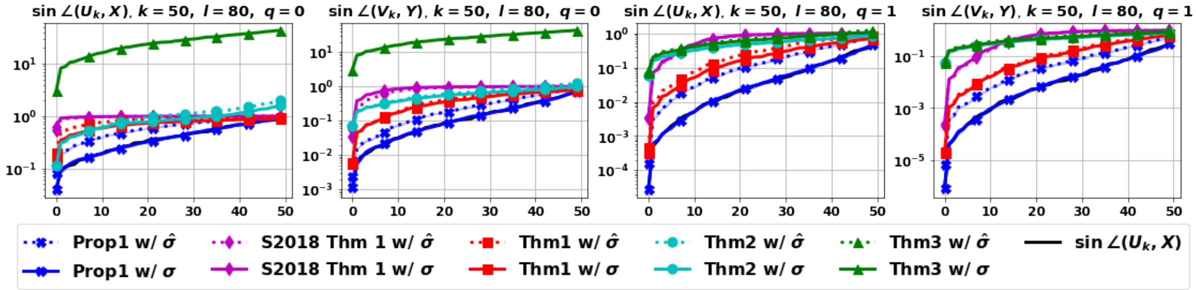


Figure 3.10: 800 randomly sampled images from the MNIST training set. $k = 50$, $l = 80$, $q = 0, 1$.

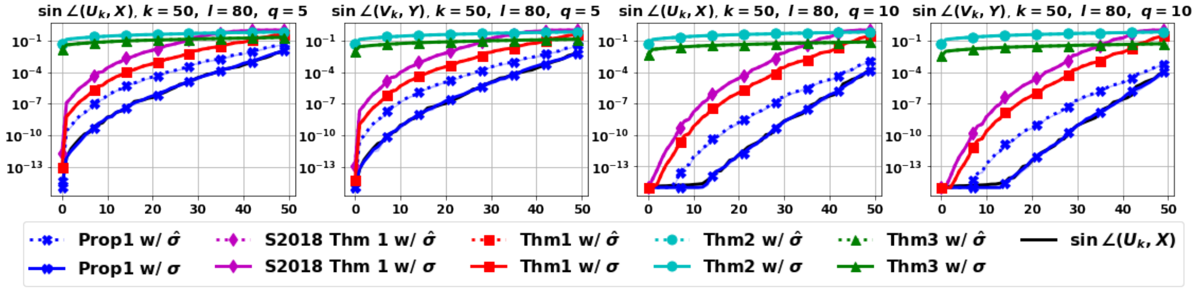


Figure 3.11: 800 randomly sampled images from the MNIST training set. $k = 50$, $l = 80$, $q = 5, 10$.

2. Blue lines and dashes (**Prop1 w/ σ and $\widehat{\sigma}$**) represent the unbiased space-agnostic estimates in Proposition 3.2 (averages of $N = 3$ independent trials with blue shades marking the corresponding minima and maxima in the trials) evaluated with the true (lines) and approximated (dashes) singular values, $\mathbf{\Sigma}$ and $\widetilde{\mathbf{\Sigma}}$, respectively;
3. Cyan lines and dashes (**Thm2 w/ σ and $\widehat{\sigma}$**) represent the posterior residual-based bounds in Theorem 3.3 evaluated with the true (lines) and approximated (dashes) singular values, $\mathbf{\Sigma}$, and $\widetilde{\mathbf{\Sigma}}$, respectively;
4. Green lines and dashes (**Thm3 w/ σ and $\widehat{\sigma}$**) represent the posterior residual-based bounds (3.13) and (3.14) in Theorem 3.4 evaluated with the true (lines) and approximated (dashes) singular values, $\mathbf{\Sigma}$, and $\widetilde{\mathbf{\Sigma}}$, respectively;
5. Magenta lines and dashes (**S2018 Thm1 w/ σ and $\widehat{\sigma}$**) represent the upper bounds in [123] Theorem 1 (*i.e.*, (3.7) and (3.8)) evaluated with the true (lines) and approximated (dashes) singular values, $\mathbf{\Sigma}$ and $\widetilde{\mathbf{\Sigma}}$, respectively, and the unknown true singular subspace such that $\mathbf{\Omega}_1 = \mathbf{V}_k^* \mathbf{\Omega}$ and $\mathbf{\Omega}_2 = \mathbf{V}_{r\setminus k}^* \mathbf{\Omega}$;
6. Black lines mark the true canonical angles $\sin\angle\left(\mathbf{U}_k, \widehat{\mathbf{U}}_l\right)$.

We recall from Remark 3.3 that, by the algorithmic construction of Algorithm 4, for given q, canonical angles of the right singular spaces $\sin\angle\left(\mathbf{V}_k, \widehat{\mathbf{V}}_l\right)$ are evaluated with half more power iterations than those of the left singular spaces $\sin\angle\left(\mathbf{U}_k, \widehat{\mathbf{U}}_l\right)$. That is, $\sin\angle\left(\mathbf{U}_k, \widehat{\mathbf{U}}_l\right)$, $\sin\angle\left(\mathbf{V}_k, \widehat{\mathbf{V}}_l\right)$ with $q = 0, 1$ in Figure 3.1-Figure 3.10 can be viewed as canonical angles of randomized subspace approximation with $q = 0, 0.5, 1, 1.5$ power iterations, respectively; while Figure 3.11 corresponds to randomized subspace approximations constructed with $q = 5, 5.5, 10, 10.5$ power iterations analogously.

For each set of upper bounds/unbiased estimates, we observe the following.

1. The space-agnostic probabilistic bounds (**Thm1 w/ σ and $\widehat{\sigma}$**) in Theorem 3.1 provide tighter statistical guarantees for the canonical angles of all the tested target matrices in comparison to those from [123] Theorem 1 (**S2018 Thm1 w/ σ and $\widehat{\sigma}$**) , as explained in Remark 3.1.
2. The unbiased estimators (**Prop1 w/ σ and $\widehat{\sigma}$**) in Proposition 3.2 yield accurate approximations for the true canonical angles on all the tested target matrices with as few as $N = 3$ trials, while enjoying good empirical concentrate. As a potential drawback, the accuracy of the unbiased estimates may be compromised when approaching the machine epsilon (as observed in Figure 3.7, $\sin\angle\left(\mathbf{V}_k, \mathbf{Y}\right)$, $q = 1$).

3. The posterior residual-based bounds (**Thm2 w/ $\boldsymbol{\sigma}$ and $\widehat{\boldsymbol{\sigma}}$**) in Theorem 3.3 are relatively tighter among the compared bounds in the setting with larger oversampling ($l = 4k$), and no power iterations ($\sin\angle\left(\mathbf{U}_k, \widehat{\mathbf{U}}_l\right)$ with $q = 0$) or exponential spectral decay (Figure 3.3)

4. The posterior residual-based bounds Equations (3.13) and (3.14) (**Thm3 w/ $\boldsymbol{\sigma}$ and $\widehat{\boldsymbol{\sigma}}$**) in Theorem 3.4 share the similar relative tightness as the posterior residual-based bounds in Theorem 3.3, but are slightly more sensitive to power iterations. As shown in Figure 3.3, on a target matrix with exponential spectral decay and large oversampling ($l = 4k$), Theorem 3.4 gives tighter posterior guarantees when $q > 0$. However, with the addition assumptions $\sigma_k > \widehat{\sigma}_{k+1}$ and $\sigma_k > \|\mathbf{E}_{33}\|_2$, Theorem 3.4 usually requires large oversampling ($l = 4k$) in order to provide non-trivial (*i.e.*, within the range $[0, 1]$) bounds.

For target matrices with various patterns of spectral decay, with different combinations of oversampling ($l = 1.6k, 4k$) and power iterations ($q = 0, 1$), we make the following observations on the relative tightness of upper bounds in Theorem 3.1, Theorem 3.3, and Theorem 3.4.

1. For target matrices with subexponential spectral decay, the space-agnostic bounds in Theorem 3.1 are relatively tighter in most tested settings, except for the setting in Figure 3.3 with larger oversampling ($l = 200$) and no power iterations ($q = 0$).

2. For target matrices with exponential spectral decay (Figure 3.3 and Figure 3.4), the posterior residual-based bounds in Theorem 3.3 and Theorem 3.4 tend to be relatively tighter, especially with large oversampling (Figure 3.3 with $l = 4k$). Meanwhile, with power iterations $q > 0$, Theorem 3.4 tend to be tighter than Theorem 3.3.

Furthermore, considering the scenario with an unknown true spectrum $\boldsymbol{\Sigma}$, we plot estimations for the upper bounds in Theorem 3.1, Theorem 3.3, Theorem 3.4, and the unbiased estimates in Proposition 3.2, evaluated with a padded approximation of the spectrum $\widetilde{\boldsymbol{\Sigma}} = \mathrm{diag}\left(\widehat{\sigma}_1, \dots, \widehat{\sigma}_l, \dots, \widehat{\sigma}_l\right)$, which leads to mild overestimations, as marked in dashes from Figure 3.1 to Figure 3.11.

3.6.3 Balance between Oversampling and Power Iterations

To illustrate the insight cast by Theorem 3.1 on the balance between oversampling and power iterations, we consider the following synthetic example.

Example 1. Given a target rank $k \in \mathbb{N}$, we consider a simple synthetic matrix $\mathbf{A} \in \mathbb{C}^{r \times r}$ of size $r = (1+\beta)k$, consisting of random singular subspaces (generated by orthonormalizing Gaussian matrices) and a step spectrum:

$$\sigma\left(\mathbf{A}\right) = \mathrm{diag}(\underbrace{\sigma_1, \dots, \sigma_1}_{\sigma_i=\sigma_1\ \forall\ i\leq k}, \underbrace{\sigma_{k+1}, \dots, \sigma_{k+1}}_{\sigma_i=\sigma_{k+1}\ \forall\ i\geq k+1}).$$

We fix a budget of $N = \alpha k$ matrix-vector multiplications with $\mathbf{A}$ in total. The goal is to distribute the computational budget between the sample size l and the number of power iterations q for the smaller canonical angles $\angle\left(\mathbf{U}_k, \widehat{\mathbf{U}}_l\right)$.

Leveraging Theorem 3.1, we start by fixing $\gamma > 1$ associated with the constants $\epsilon_1 = \gamma\sqrt{k/l}$ and $\epsilon_2 = \gamma\sqrt{l/(r-k)}$ in (3.2) such that $l \geq \gamma^2 k$ and $2q+1 < \alpha/\gamma^2$. Characterized by γ, the right-hand-side of (3.2) under fixed budget N (*i.e.*, $N \geq l(2q+1)$) is defined as:

$$\begin{aligned}\phi_\gamma\left(q\right) &\triangleq \left(1 + \frac{1-\epsilon_1}{1+\epsilon_2}\cdot\frac{l}{r-k}\left(\frac{\sigma_1}{\sigma_{k+1}}\right)^{4q+2}\right)^{-\frac{1}{2}} \qquad (3.24)\\ &= \left(1 + \frac{\alpha - \gamma\sqrt{\alpha(2q+1)}}{\beta(2q+1)+\gamma\sqrt{\alpha\beta(2q+1)}}\left(\frac{\sigma_1}{\sigma_{k+1}}\right)^{4q+2}\right)^{-\frac{1}{2}}.\end{aligned}$$

With the synthetic step spectrum, the dependence of (3.2) on $\sigma\left(\mathbf{A}\right)$ is reduced to the spectral gap σ_1/σ_{k+1} in (3.24).

As a synopsis, Table 3.1 summarizes the relevant parameters that characterize the problem setup.

Table 3.1: Given $\mathbf{A} \in \mathbb{C}^{r\times r}$ with a spectral gap σ_1/σ_{k+1}, a target rank k, and a budget of N matrix-vector multiplications, we consider applying Algorithm 4 with a sample size l and q power iterations.

α	budget parameter	$N = \alpha k$
β	size parameter	$r = (1+\beta)k$
γ	oversampling parameter	$l \geq \gamma^2 k$ and $2q+1 \leq \frac{\alpha}{\gamma^2}$

With $k = 10$, $\alpha = 32$, $\beta = 64$, and $\gamma \in \{1.05, 2.00\}$, Figure 3.12 and Figure 3.13 illustrate (i) how the balance between oversampling and power iterations is affected by the spectral gap σ_1/σ_{k+1}, and more importantly, (ii) how (3.24) unveils the trend in true canonical angles $\sin\angle_i\left(\mathbf{U}_k, \widehat{\mathbf{U}}_l\right)$ among different configurations $\{(l,q) \mid l \geq \gamma^2 k, 2q+1 \leq \alpha/\gamma^2\}$.

Concretely, both Figure 3.12 and Figure 3.13 imply that more oversampling (*e.g.*, $q = 0$) is preferred when the spectral gap is small (*e.g.*, $\sigma_1/\sigma_{k+1} = 1.01$); while more power iterations

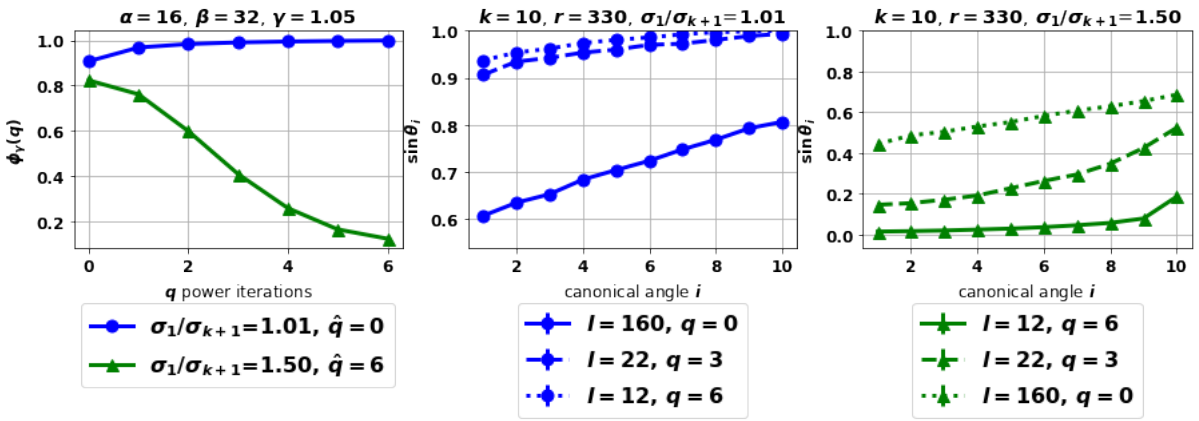


Figure 3.12: For $k = 10$, $\alpha = 16$, $\beta = 32$, $\gamma = 1.05$, the left figure marks $\phi_\gamma(q)$ (*i.e.*, the right-hand side of (3.2) under the fixed budget N) with two different spectral gaps ($\widehat{q} = \operatorname{argmin}_{1 \le 2q+1 \le \alpha/\gamma^2} \phi_\gamma(q)$), while the middle and the right figures demonstrate how the relative magnitudes of canonical angles $\sin\angle_i\left(\mathbf{U}_k, \widehat{\mathbf{U}}_l\right)$ ($i \in [k]$) under different configurations (*i.e.*, choices of (l, q), showing the averages and ranges of 5 trials) align with the trends in $\phi_\gamma(q)$.

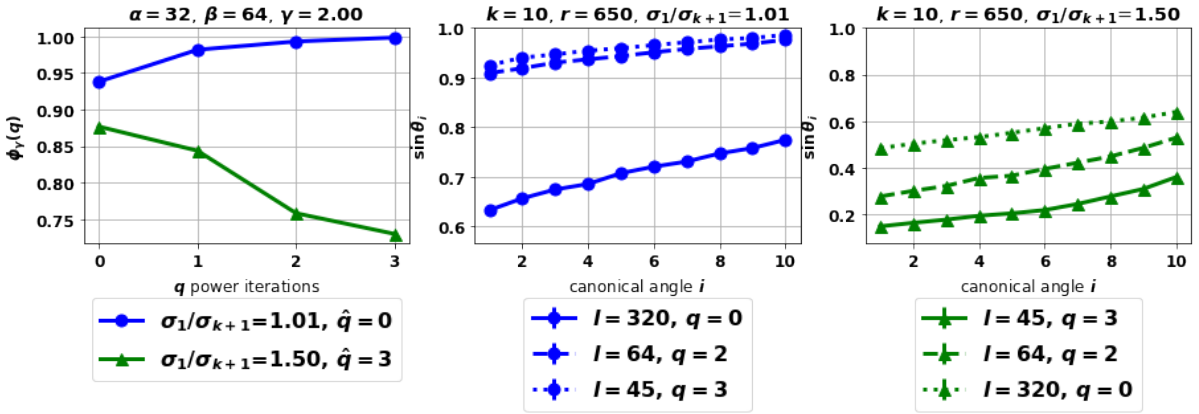


Figure 3.13: Under the same setup as Figure 3.12, for $k = 10$, $\alpha = 32$, $\beta = 64$, $\gamma = 2.00$, the trend in $\phi_\gamma(q)$ also aligns well with that in true canonical angles $\sin\angle_i\left(\mathbf{U}_k, \widehat{\mathbf{U}}_l\right)$ ($i \in [k]$).

(*e.g.*, $q = \lfloor \frac{\alpha/\gamma^2 - 1}{2} \rfloor$) are preferred when the spectral gap is large (*e.g.*, $\sigma_1/\sigma_{k+1} = 1.5$). Such trends are both observed in the true canonical angles $\sin\angle_i\left(\mathbf{U}_k, \widehat{\mathbf{U}}_l\right)$ ($i \in [k]$) and well reflected by $\phi_\gamma(q)$.

3.7 Discussion

We presented prior and posterior bounds and estimates that can be computed efficiently for canonical angles of the randomized subspace approximation. Under moderate multiplicative oversampling, our prior probabilistic bounds are space-agnostic (*i.e.*, independent of the unknown true subspaces), asymptotically tight, and can be computed in linear ($O(\mathrm{rank}\left(\mathbf{A}\right))$) time, while casting a clear guidance on the balance between oversampling and power iterations for a fixed budget of matrix-vector multiplications. As corollaries of the prior probabilistic bounds, we introduce a set of unbiased canonical angle estimates that are efficiently computable and applicable to arbitrary choices of oversampling with good empirical concentrations. In addition to the prior bounds and estimates, we further discuss two sets of posterior bounds that provide deterministic guarantees for canonical angles given the computed low-rank approximations. With numerical experiments, we compare the empirical tightness of different canonical angle bounds and estimates on various data matrices under a diverse set of algorithmic choices for the randomized subspace approximation.

Chapter 4

Sample Efficiency of Data Augmentation Consistency Regularization

Abstract

Data augmentation is popular in the training of large neural networks; however, currently, theoretical understanding of the discrepancy between different algorithmic choices of leveraging augmented data remains limited. In this paper, we take a step in this direction – we first present a simple and novel analysis for linear regression with label invariant augmentations, demonstrating that data augmentation consistency (DAC) is intrinsically more efficient than empirical risk minimization on augmented data (DA-ERM). The analysis is then generalized to misspecified augmentations (i.e., augmentations that change the labels), which again demonstrates the merit of DAC over DA-ERM. Further, we extend our analysis to non-linear models (e.g., neural networks) and present generalization bounds. Finally, we perform experiments that make a clean and apples-to-apples comparison (i.e., with no extra modeling or data tweaks) between DAC and DA-ERM using CIFAR-100 and WideResNet; these together demonstrate the superior efficacy of DAC.[1]

4.1 Introduction

Modern machine learning models, especially deep learning models, require abundant training samples. Since data collection and human annotation are expensive, data augmentation has become a ubiquitous practice in creating artificial labeled samples and improving generalization performance. This practice is corroborated by the fact that the semantics of images remain the same through simple translations like obscuring, flipping, rotation, color jitter, rescaling [131]. Conventional algorithms use data augmentation to expand the training data set [34, 71, 84, 132, 133].

[1] This chapter is based on the following published conference paper:
Shuo Yang, Yijun Dong, Rachel Ward, Inderjit S Dhillon, Sujay Sanghavi, and Qi Lei. Sample efficiency of data augmentation consistency regularization. In *International Conference on Artificial Intelligence and Statistics*, pages 3825–3853. PMLR, 2023 [173].

Data Augmentation Consistency (DAC) regularization, as an alternative, enforces the model to output similar predictions on the original and augmented samples and has contributed to many recent state-of-the-art supervised or semi-supervised algorithms. This idea was first proposed in [5] and popularized by [88, 124], and gained more attention recently with the success of FixMatch [135] for semi-supervised few-shot learning as well as AdaMatch [13] for domain adaptation. DAC can utilize unlabeled samples, as one can augment the training samples and enforce consistent predictions without knowing the true labels. This bypasses the limitation of the conventional algorithms that can only augment labeled samples and add them to the training set (referred to as DA-ERM). However, it is not well-understood whether DAC has additional algorithmic benefits compared to DA-ERM. We are, therefore, seeking a theoretical answer.

Despite the empirical success, the theoretical understanding of data augmentation (DA) remains limited. Existing work [27, 95, 102] focused on establishing that augmenting data saves on the number of labeled samples needed for the same level of accuracy. However, none of these explicitly compare the efficacy (in terms of the number of augmented samples) between different algorithmic choices on *how to use the augmented samples* in an apples-to-apples way.

In this paper, we focus on the following research question:

Is DAC intrinsically more efficient than DA-ERM (even without unlabeled samples)?

We answer the question affirmatively. We show that DAC is intrinsically more efficient than DA-ERM with a simple and novel analysis for linear regression under label invariant augmentations. We then extend the analysis to misspecified augmentations (i.e., those that change the labels). We further provide generalization bounds under consistency regularization for non-linear models like two-layer neural networks and DNN-based classifiers with expansion-based augmentations. Intuitively, we show DAC is better than DA-ERM in the following sense: 1) DAC enforces stronger invariance in the learned models, yielding smaller estimation error; and 2) DAC better tolerates mis-specified augmentations and incurs smaller approximation error. Our theoretical findings can also explain and guide some technical choices, e.g. why we can use stronger augmentation in consistency regularization but only weaker augmentation when creating pseudo-labels [135].

Specifically, our **main contributions** are:

- **Theoretical comparisons between DAC and DA-ERM.** We first present a simple and novel result for linear regression, which shows that DAC yields a strictly smaller generalization error than DA-ERM using the same augmented data. Further, we demonstrate that with with the flexibility of hyper-parameter tuning, DAC can better handle data augmentation with small misspecification in the labels.

- **Extended analysis for non-linear models.** We derive generalization bounds for DAC under two-layer neural networks, and classification with expansion-based augmentations.
- **Empirical comparisons between DAC and DA-ERM.** We perform experiments that make a clean and apples-to-apples comparison (i.e., with no extra modeling or data tweaks) between DAC and DA-ERM using CIFAR-100 and WideResNet. Our empirical results demonstrate the superior efficacy of DAC.

4.2 Related Work

Empirical findings. Data augmentation (DA) is an essential ingredient for almost every state-of-the-art supervised learning algorithm since the seminal work of [84] (see reference therein [34, 71, 85, 132, 133]). It started from adding augmented data to the training samples via (random) perturbations, distortions, scales, crops, rotations, and horizontal flips. More sophisticated variants were subsequently designed; a non-exhaustive list includes Mixup [177], Cutout [43], and Cutmix [175]. The choice of data augmentation and their combinations require domain knowledge and experts' heuristics, which triggered some automated search algorithms to find the best augmentation strategies [34, 93]. The effects of different DAs are systematically explored in [143].

Recent practices not only add augmented data to the training set but also enforce similar predictions by adding consistency regularization [5, 88, 135]. One benefit of DAC is the feasibility of exploiting unlabeled data. Therefore input consistency on augmented data also formed a major component to state-of-the-art algorithms for semi-supervised learning [88, 124, 135, 172], self-supervised learning [28], and unsupervised domain adaptation [13, 56].

Theoretical studies. Many interpret the effect of DA as some form of regularization [72]. Some work focuses on linear transformations and linear models [169] or kernel classifiers [36]. Convolutional neural networks by design enforce translation equivariance symmetry [11, 91]; further studies have hard-coded CNN's invariance or equivariance to rotation [32, 97, 167, 182], scaling [139, 168] and other types of transformations.

Another line of works view data augmentation as invariant learning by averaging over group actions [16, 27, 95, 102, 128, 156]. They consider an ideal setting that is equivalent to ERM with all possible augmented data, bringing a clean mathematical interpretation. In contrast, we are interested in a more realistic setting with limited augmented data. In this setting, it is crucial to utilize the limited data with proper training methods, the difference of which cannot be revealed under previously studied settings.

Some more recent work investigates the feature representation learning procedure with DA for self-supervised learning tasks [57, 69, 152, 163]. [22, 162] studied the effect of data augmentation with label propagation. Data augmentation is also deployed to improve robustness [117], to facilitate domain adaptation and domain generalization [22, 122].

4.3 Problem Setup and Data Augmentation Consistency

Consider the standard supervised learning problem setup: $\mathbf{x} \in \mathcal{X}$ is input feature, and $y \in \mathcal{Y}$ is its label (or response). Let P be the true distribution of $(\mathbf{x}, y)$ (i.e., the label distribution follows $y \sim P(y|\mathbf{x})$). We have the following definition for label invariant augmentation.

Definition 4.1 (Label Invariant Augmentation)**.** For any sample $\mathbf{x} \in \mathcal{X}$, we say that a random transformation $A : \mathcal{X} \to \mathcal{X}$ is a label invariant augmentation if and only if $A(\mathbf{x})$ satisfies $P(y|\mathbf{x}) = P(y|A(\mathbf{x}))$.

Our work largely relies on label invariant augmentation but also extends to augmentations that incur small misspecification in their labels. Therefore our results apply to the augmentations achieved via certain transformations (e.g., random cropping, rotation), and we do not intend to cover augmentations that can largely alter the semantic meanings (e.g., MixUp [177]). See examples of data augmentation in Sections 4.4 and 4.6.

Now we introduce the learning problem on an augmented dataset. Let $(\mathbf{X}, \mathbf{y}) \in \mathcal{X}^N \times \mathcal{Y}^N$ be a training set consisting of N *i.i.d.* samples. Besides the original $(\mathbf{X}, \mathbf{y})$, each training sample is provided with α augmented samples. The features of the augmented dataset $\widetilde{\mathcal{A}}(\mathbf{x}) \in \mathcal{X}^{(1+\alpha)N}$ is:

$$\widetilde{\mathcal{A}}(\mathbf{X}) = [\mathbf{x}_1; \cdots; \mathbf{x}_N; \mathbf{x}_{1,1}; \cdots; \mathbf{x}_{N,1}; \cdots; \mathbf{x}_{1,\alpha}; \cdots; \mathbf{x}_{N,\alpha}] \in \mathcal{X}^{(1+\alpha)N},$$

where $\mathbf{x}_i$ is in the original training set and $\mathbf{x}_{i,j}, \forall j \in [\alpha]$ are the augmentations of $\mathbf{x}_i$. The labels of the augmented samples are kept the same, which can be denoted as $\widetilde{\mathbf{M}}\mathbf{y} \in \mathcal{Y}^{(1+\alpha)N}$, where $\widetilde{\mathbf{M}} \in \mathbb{R}^{(1+\alpha)N \times N}$ is a vertical stack of $(1 + \alpha)$ identity mappings.

Data Augmentation Consistency Regularization. Let $\mathcal{H} = \{h : \mathcal{X} \to \mathcal{Y}\}$ be a well-specified function class (e.g., for linear regression problems, $\exists h^* \in \mathcal{H}$, s.t. $h^*(\mathbf{x}) = \mathbb{E}[y|\mathbf{x}]$) that we hope to learn from. Without loss of generality, we assume that each function $h \in \mathcal{H}$ can be expressed as $h = f_h \circ \phi_h$, where $\phi_h \in \Phi = \{\phi : \mathcal{X} \to \mathcal{W}\}$ is a proper representation mapping and $f_h \in \mathcal{F} = \{f : \mathcal{W} \to \mathcal{Y}\}$ is a predictor on top of the learned representation. We tend to decompose h such that ϕ_h is a powerful feature extraction function whereas f_h can be as simple as a linear combiner. For instance, in a deep neural network, all the layers before the final layer can be viewed as feature extraction ϕ_h, and the predictor f_h is the final linear combination layer.

For a loss function $l : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ and a metric ϱ properly defined on the representation space $\mathcal{W}$, learning with data augmentation consistency (DAC) regularization is:

$$\underset{h\in\mathcal{H}}{\operatorname{argmin}} \sum_{i=1}^{N} l(h(\mathbf{x}_i), y_i) + \underbrace{\lambda \sum_{i=1}^{N}\sum_{j=1}^{\alpha} \varrho\left(\phi_h(\mathbf{x}_i), \phi_h(\mathbf{x}_{i,j})\right)}_{\textit{DAC regularization}}. \tag{4.1}$$

Note that the DAC regularization in (4.1) can be easily implemented empirically as a regularizer. Intuitively, DAC regularization penalizes the representation difference between the original sample $\phi_h(\mathbf{x}_i)$ and the augmented sample $\phi_h(\mathbf{x}_{i,j})$, with the belief that similar samples (i.e., original and augmented samples) should have similar representations. When the data augmentations do not alter the labels, it is reasonable to enforce a strong regularization (i.e., $\lambda \to \infty$) – since the conditional distribution of y does not change. The learned function $\widehat{h}^{dac}$ can then be written as the solution of a constrained optimization problem:

$$\widehat{h}^{dac} \triangleq \underset{h\in\mathcal{H}}{\operatorname{argmin}} \sum_{i=1}^{N} l(h(\mathbf{x}_i), y_i) \quad \text{s.t.} \quad \phi_h(\mathbf{x}_i) = \phi_h(\mathbf{x}_{i,j}),\ \forall i \in [N], j \in [\alpha]. \tag{4.2}$$

In the rest of the paper, we mainly focus on the data augmentations satisfying Definition 4.1 and our analysis relies on the formulation of (4.2). When the data augmentations alter the label distributions (i.e., not satisfying Definition 4.1), it becomes necessary to adopt a finite λ for (4.1), and such extension is discussed in Section 4.5.

4.4 Linear Model and Label Invariant Augmentations

In this section, we show the efficacy of DAC regularization with linear regression under label invariant augmentations (Definition 4.1).

To see the efficacy of DAC regularization (i.e., (4.2)), we revisit a more commonly adopted training method here – empirical risk minimization on augmented data (DA-ERM):

$$\widehat{h}^{da-erm} \triangleq \underset{h \in \mathcal{H}}{\operatorname{argmin}} \sum_{i=1}^{N} l(h(\mathbf{x}_i), y_i) + \sum_{i=1}^{N} \sum_{j=1}^{\alpha} l(h(\mathbf{x}_{i,j}), y_i). \tag{4.3}$$

Now we show that the DAC regularization ((4.2)) learns more efficiently than DA-ERM. Consider the following setting: given N observations $\mathbf{X} \in \mathbb{R}^{N \times d}$, the responses $\mathbf{y} \in \mathbb{R}^N$ are generated from a linear model $\mathbf{y} = \mathbf{X}\boldsymbol{\theta}^* + \boldsymbol{\epsilon}$, where $\boldsymbol{\epsilon} \in \mathbb{R}^N$ is zero-mean noise with $\mathbb{E}\left[\boldsymbol{\epsilon}\boldsymbol{\epsilon}^\top\right] = \sigma^2 \mathbf{I}_N$. Recall that $\widetilde{\mathcal{A}}(\mathbf{X})$ is the entire augmented dataset, and $\widetilde{\mathbf{M}}\mathbf{y}$ corresponds to the labels. We focus on the fixed design excess risk of $\boldsymbol{\theta}$ on $\widetilde{\mathcal{A}}(\mathbf{X})$, which is defined as $L(\boldsymbol{\theta}) \triangleq \frac{1}{(1+\alpha)N} \|\widetilde{\mathcal{A}}(\mathbf{X})\boldsymbol{\theta} - \widetilde{\mathcal{A}}(\mathbf{X})\boldsymbol{\theta}^*\|_2^2$.

Let $\boldsymbol{\Delta} \triangleq \widetilde{\mathcal{A}}\left(\mathbf{X}\right) - \widetilde{\mathbf{M}}\mathbf{X}$ and $d_{aug} \triangleq \operatorname{rank}\left(\boldsymbol{\Delta}\right)$ measure the number of dimensions in the row space of $\mathbf{X}$ perturbed by augmentations (which can be intuitively view as the "strength" of data augmentations where the larger d_{aug} implies the stronger perturbation brought by $\widetilde{\mathcal{A}}(\mathbf{X})$ to $\mathbf{X}$). Assuming that $\widetilde{\mathcal{A}}(\mathbf{X})$ has full column rank (such that the linear regression problem has a unique solution), we have the following result for learning by DAC versus DA-ERM.

Theorem 4.1 (Informal result on linear regression (formally in Theorem B.1))**.** *Learning with DAC regularization,*

$$\mathbb{E}_{\boldsymbol{\epsilon}}\left[L(\widehat{\boldsymbol{\theta}}^{dac}) - L(\boldsymbol{\theta}^*)\right] = \frac{(d - d_{aug})\sigma^2}{N},$$

while learning with ERM directly on the augmented dataset, there exists $d' \in [0, d_{aug}]$ such that

$$\mathbb{E}_{\boldsymbol{\epsilon}}\left[L(\widehat{\boldsymbol{\theta}}^{da-erm}) - L(\boldsymbol{\theta}^*)\right] = \frac{(d - d_{aug} + d')\sigma^2}{N}.$$

Formally, we have $d' \triangleq \frac{\operatorname{tr}\left(\left(\mathbf{P}_{\widetilde{\mathcal{A}}(\mathbf{X})} - \mathbf{P}_{\mathcal{S}}\right)\widetilde{\mathbf{M}}\widetilde{\mathbf{M}}^\top\right)}{1+\alpha}$, where $\mathbf{P}_{\widetilde{\mathcal{A}}(\mathbf{X})} \triangleq \widetilde{\mathcal{A}}(\mathbf{X})\widetilde{\mathcal{A}}(\mathbf{X})^\dagger$, and $\mathbf{P}_{\mathcal{S}}$ is the projector onto $\mathcal{S} \triangleq \left\{\widetilde{\mathbf{M}}\mathbf{X}\boldsymbol{\theta} \mid \forall \boldsymbol{\theta} \in \mathbb{R}^d, s.t. \left(\widetilde{\mathcal{A}}(\mathbf{X}) - \widetilde{\mathbf{M}}\mathbf{X}\right)\boldsymbol{\theta} = 0\right\}$. Under standard conditions (e.g., $\mathbf{x}$ is sub-Gaussian and N is not too small), it is not hard to extend Theorem 4.1 to random design (i.e., the more commonly acknowledged generalization bound) with the same order.

Remark 4.1 (Why DAC is more effective). Intuitively, DAC reduces the dimensions from d to $d - d_{aug}$ by enforcing consistency regularization. DA-ERM, on the other hand, still learns in the original d-dimensional space. d' characterizes such difference.

Now we take a closer look at $d' \triangleq \frac{\operatorname{tr}\left(\left(\mathbf{P}_{\widetilde{\mathcal{A}}(\mathbf{X})} - \mathbf{P}_{\mathcal{S}}\right)\widetilde{\mathbf{M}}\widetilde{\mathbf{M}}^\top\right)}{1+\alpha}$ characterizing the discrepancy between DAC and DA-ERM. We first observe that $\sigma^2 \cdot \widetilde{\mathbf{M}}\widetilde{\mathbf{M}}^\top$ is the noise covariance matrix of the augmented dataset. $\operatorname{tr}\left(\mathbf{P}_{\mathcal{S}}\widetilde{\mathbf{M}}\widetilde{\mathbf{M}}^\top\right)$ represents the variance of $\widehat{\boldsymbol{\theta}}^{dac}$, while $\operatorname{tr}\left(\mathbf{P}_{\widetilde{\mathcal{A}}(\mathbf{X})}\widetilde{\mathbf{M}}\widetilde{\mathbf{M}}^\top\right)$ denotes the variance of $\widehat{\boldsymbol{\theta}}^{da-erm}$. Therefore, $d' \propto \operatorname{tr}\left(\left(\mathbf{P}_{\widetilde{\mathcal{A}}(\mathbf{X})} - \mathbf{P}_{\mathcal{S}}\right)\widetilde{\mathbf{M}}\widetilde{\mathbf{M}}^\top\right)$ measures the excess variance of $\widehat{\boldsymbol{\theta}}^{da-erm}$ in comparison to $\widehat{\boldsymbol{\theta}}^{dac}$. When $\mathbf{P}_{\widetilde{\mathcal{A}}(\mathbf{X})} \neq \mathbf{P}_{\mathcal{S}}$ (a common scenario as instantiated in Example 2), DAC is strictly better than DA-ERM.

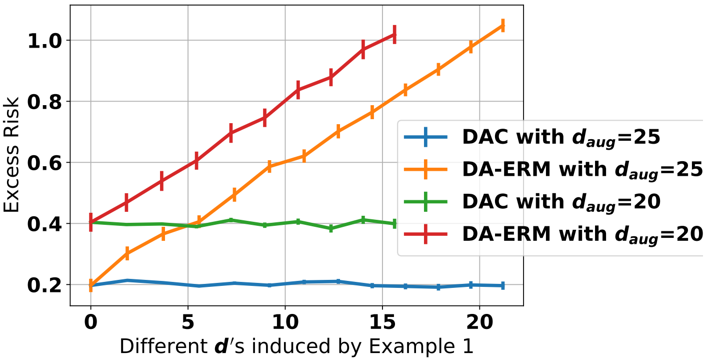


Figure 4.1: Comparison of DAC regularization and DA-ERM (Example 2). The results precisely match Theorem 4.1. DA-ERM depends on the d' induced by different augmentations, while the DAC regularization works equally well for all d' and better than the DA-ERM. Further, both DAC and DA-ERM are affected by d_{aug}, the number of dimensions perturbed by $\widetilde{\mathcal{A}}(\mathbf{X})$.

Example 2. Consider a 30-dimensional linear regression. The original training set contains 50 samples. The inputs $\mathbf{x}_i$s are generated independently from $\mathcal{N}(0, \mathbf{I}_{30})$ and we set $\boldsymbol{\theta}^* = [\boldsymbol{\theta}_c^*; \mathbf{0}]$ with $\boldsymbol{\theta}_c^* \sim \mathcal{N}(0, \mathbf{I}_5)$ and $\mathbf{0} \in \mathbb{R}^{25}$. The noise variance σ is set to 1. We partition $\mathbf{x}$ into 3 parts $[x_{c1}, x_{e_1}, x_{e2}]$ and take the following augmentations: $A([x_{c1}; x_{e1}; x_{e2}]) = [x_{c1}; 2x_{e1}; -x_{e2}], x_{c1} \in \mathbb{R}^{d_{c1}}, x_{e1} \in \mathbb{R}^{d_{e1}}, x_{e2} \in \mathbb{R}^{d_{e2}}$, where $d_{c1} + d_{e1} + d_{e2} = 30$.

Notice that the augmentation perturbs x_{e1} and x_{e2} and leaving x_{c1} unchanged, we therefore have $d_{aug} = 30 - d_{c1}$. By changing d_{c1} and d_{e1}, we can have different augmentations with different d_{aug}, d'. The results for $d_{aug} \in \{20, 25\}$ and various d's are presented in Figure 4.1. The excess risks precisely match Theorem 4.1. It confirms that the DAC regularization is strictly better than DA-ERM for a wide variety of augmentations.

4.5 Beyond Label Invariant Augmentation

In this section, we extend our analysis to misspecified augmentations by relaxing the label invariance assumption (such that $P(y|\mathbf{x}) \neq P(y|A(\mathbf{x}))$). With an illustrative linear regression problem, we show that DAC also brings advantages over DA-ERM for misspecified augmentations.

We first recall the linear regression setup: given a set of N *i.i.d.* samples $(\mathbf{X}, \mathbf{y})$ that follows $\mathbf{y} = \mathbf{X}\boldsymbol{\theta}^* + \boldsymbol{\epsilon}$ where $\boldsymbol{\epsilon}$ are zero-mean independent noise with $\mathbb{E}\left[\boldsymbol{\epsilon}\boldsymbol{\epsilon}^\top\right] = \sigma^2 \mathbf{I}_N$, we aim to learn the unknown ground truth $\boldsymbol{\theta}^*$. For randomly generated misspecified augmentations $\widetilde{\mathcal{A}}(\mathbf{X})$ that alter the labels (*i.e.*, $\widetilde{\mathcal{A}}\left(\mathbf{X}\right)\boldsymbol{\theta}^* \neq \widetilde{\mathbf{M}}\mathbf{X}\boldsymbol{\theta}^*$), a proper consistency constraint is $\|\phi_h(\mathbf{x}_i) - \phi_h(\mathbf{x}_{i,j})\|_2 \leq C_{mis}$ (where $\mathbf{x}_{i,j}$ is an augmentation of $\mathbf{x}_i$, noticing that $C_{mis} = 0$ corresponds to label invariant augmentations in Definition 4.1). For $C_{mis} > 0$, the constrained optimization is equivalent to:

$$\widehat{\boldsymbol{\theta}}^{dac} = \underset{\boldsymbol{\theta} \in \mathbb{R}^d}{\operatorname{argmin}} \frac{1}{N} \|\mathbf{X}\boldsymbol{\theta} - \mathbf{y}\|_2^2 + \frac{\lambda}{(1+\alpha)\,N} \| \left(\widetilde{\mathcal{A}}\left(\mathbf{X}\right) - \widetilde{\mathbf{M}}\mathbf{X}\right)\boldsymbol{\theta}\|_2^2 \tag{4.4}$$

for some finite $0 < \lambda < \infty$. We compare $\widehat{\boldsymbol{\theta}}^{dac}$ to the solution learned with ERM on augmented data (as in (4.3)):

$$\widehat{\boldsymbol{\theta}}^{da-erm} = \underset{\boldsymbol{\theta} \in \mathbb{R}^d}{\operatorname{argmin}} \frac{1}{(1+\alpha)\,N} \left\|\widetilde{\mathcal{A}}\left(\mathbf{X}\right)\boldsymbol{\theta} - \widetilde{\mathbf{M}}\mathbf{y}\right\|_2^2.$$

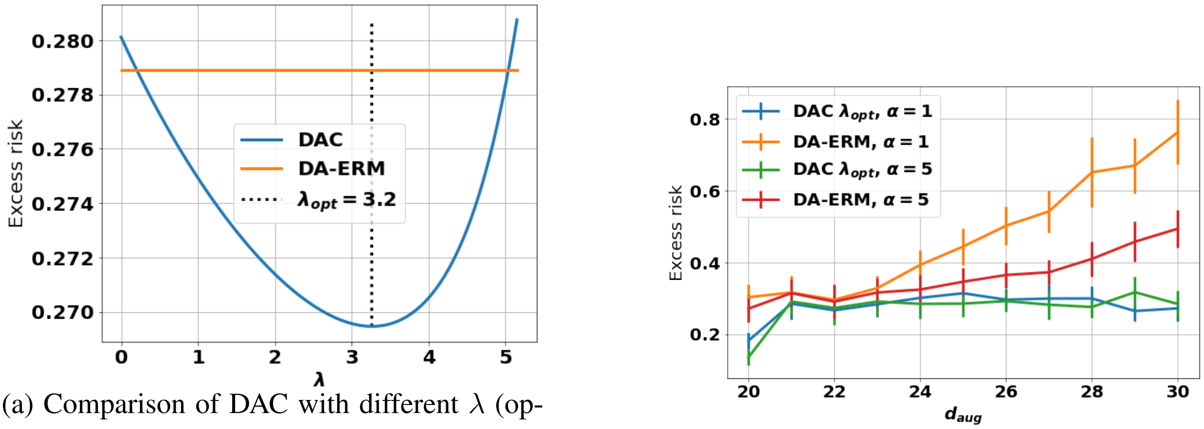
Let $\boldsymbol{\Sigma}_{\mathbf{X}} \triangleq \frac{1}{N}\mathbf{X}^\top\mathbf{X}$ and $\boldsymbol{\Sigma}_{\widetilde{\mathcal{A}}(\mathbf{X})} \triangleq \frac{1}{(1+\alpha)N}\widetilde{\mathcal{A}}\left(\mathbf{X}\right)^\top \widetilde{\mathcal{A}}\left(\mathbf{X}\right)$. With $\mathbf{S} = \frac{1}{1+\alpha}\widetilde{\mathbf{M}}^\top\widetilde{\mathcal{A}}\left(\mathbf{X}\right)$, $\boldsymbol{\Delta} \triangleq \widetilde{\mathcal{A}}\left(\mathbf{X}\right) - \widetilde{\mathbf{M}}\mathbf{X}$, and its reweighted analog $\widetilde{\boldsymbol{\Delta}} \triangleq \left(\widetilde{\mathbf{M}}\mathbf{X}\right)\widetilde{\mathcal{A}}\left(\mathbf{X}\right)^\dagger \boldsymbol{\Delta}$, we further introduce positive semidefinite matrices: $\boldsymbol{\Sigma}_{\mathbf{S}} \triangleq \frac{1}{N}\mathbf{S}^\top\mathbf{S}$, $\boldsymbol{\Sigma}_{\boldsymbol{\Delta}} \triangleq \frac{1}{(1+\alpha)N}\boldsymbol{\Delta}^\top\boldsymbol{\Delta}$, and $\boldsymbol{\Sigma}_{\widetilde{\boldsymbol{\Delta}}} \triangleq \frac{1}{(1+\alpha)N}\widetilde{\boldsymbol{\Delta}}^\top\widetilde{\boldsymbol{\Delta}}$. For demonstration purpose, we consider fixed $\mathbf{X}$ and $\widetilde{\mathcal{A}}\left(\mathbf{X}\right)$, with respect to which we introduce distortion factors $c_X, c_S > 0$ as the minimum constants that satisfy $\boldsymbol{\Sigma}_{\widetilde{\mathcal{A}}(\mathbf{X})} \preccurlyeq c_X \boldsymbol{\Sigma}_{\mathbf{X}}$ and $\boldsymbol{\Sigma}_{\widetilde{\mathcal{A}}(\mathbf{X})} \preccurlyeq c_S \boldsymbol{\Sigma}_{\mathbf{S}}$ (notice that such c_X, c_S exist almost surely when $\mathbf{X}$ and $\widetilde{\mathcal{A}}\left(\mathbf{X}\right)$ are drawn from absolutely continuous marginal distributions).

Recall $d_{aug} \triangleq \operatorname{rank}\left(\boldsymbol{\Delta}\right)$ from Section 4.4. Let $\mathbf{P}_{\boldsymbol{\Delta}} \triangleq \boldsymbol{\Delta}^\dagger\boldsymbol{\Delta}$ denote the rank-d_{aug} orthogonal projector onto $\operatorname{Range}\left(\boldsymbol{\Delta}^\top\right)$. Then, for $L(\boldsymbol{\theta}) = \frac{1}{N}\|\mathbf{X}\boldsymbol{\theta} - \mathbf{y}\|_2^2$, we have the following result:

Theorem 4.2. *Learning with DAC regularization ((4.4)), we have that, at the optimal λ*[2]*,*

$$\mathbb{E}_{\boldsymbol{\epsilon}}\left[L(\widehat{\boldsymbol{\theta}}^{dac}) - L\left(\boldsymbol{\theta}^*\right)\right] \leq \frac{\sigma^2\left(d - d_{aug}\right)}{N} \|\mathbf{P}_{\boldsymbol{\Delta}}\boldsymbol{\theta}^*\|_{\boldsymbol{\Sigma}_{\boldsymbol{\Delta}}} \sqrt{\frac{\sigma^2}{N}\operatorname{tr}\left(\boldsymbol{\Sigma}_{\mathbf{X}}\boldsymbol{\Sigma}_{\boldsymbol{\Delta}}^\dagger\right)},$$

[2] A positive (semi)definite matrix $\boldsymbol{\Sigma}$ induces a (semi)norm: $\|\mathbf{u}\|_{\boldsymbol{\Sigma}} = \left(\mathbf{u}^\top\boldsymbol{\Sigma}\mathbf{u}\right)^{1/2}$ for all conformable $\mathbf{u}$.



(a) Comparison of DAC with different λ (optimal choice at $\lambda_{opt} = 3.2$) and DA-ERM in Example 3, where $d_{aug} = 24$ and $\alpha = 1$. The results demonstrate that, with a proper λ, DAC can outperform DA-ERM under misspecified augmentations.

(b) Comparison of DAC with the optimal λ and DA-ERM in Example 3 for different augmentation strength d_{aug}. $d_{aug} = 20$ corresponds to the label-invariance augmentations, whereas increasing d_{aug} leads to more misspecification.

Figure 4.2: Comparisons of DAC and DA-ERM under misspecification.

whereas learning with DA-ERM ((4.3)),

$$\mathbb{E}_{\epsilon}\left[L(\widehat{\boldsymbol{\theta}}^{da-erm}) - L\left(\boldsymbol{\theta}^*\right)\right] \geq \frac{\sigma^2 d}{N c_X c_S} + \|\mathbf{P}_{\boldsymbol{\Delta}}\boldsymbol{\theta}^*\|_{\boldsymbol{\Sigma}_{\widetilde{\boldsymbol{\Delta}}}}^2 .$$

Here, $\mathbf{P}_{\boldsymbol{\Delta}}\boldsymbol{\theta}^*$ *measures the misspecification in* $\boldsymbol{\theta}^*$ *by the augmentations* $\widetilde{\mathcal{A}}\left(\mathbf{X}\right)$.

One advantage of DAC regularization derives from its flexibility in choosing regularization parameter λ. With a proper λ (*e.g.*, see Figure 4.2a) that matches the misspecification $C_{mis}^2 = \frac{1}{(1+\alpha)N}\|\left(\widetilde{\mathcal{A}}\left(\mathbf{X}\right) - \widetilde{\mathbf{M}}\mathbf{X}\right)\boldsymbol{\theta}^*\|_2^2 = \|\mathbf{P}_{\boldsymbol{\Delta}}\boldsymbol{\theta}^*\|_{\boldsymbol{\Sigma}_{\boldsymbol{\Delta}}}^2$, DAC effectively reduces the function class from $\mathbb{R}^d$ to $\left\{\boldsymbol{\theta} \;\middle|\; \|\mathbf{P}_{\boldsymbol{\Delta}}\boldsymbol{\theta}\|_{\boldsymbol{\Sigma}_{\boldsymbol{\Delta}}} \leq C_{mis}\right\}$ and therefore improves the sample efficiency.

Another advantage of DAC is that, in contrast to DA-ERM, the consistency regularization term in (4.4) refrains from learning the original labels with misspecified augmentations $\mathbb{E}_{\epsilon}\left[\widetilde{\mathbf{M}}\mathbf{y}\right] \neq \widetilde{\mathcal{A}}\left(\mathbf{X}\right)\boldsymbol{\theta}^*$ when a suitable C_{mis} is identified implicitly via λ. This allows DAC to learn from fewer but stronger (potentially more severely misspecified) augmentations (*e.g.*, Figure 4.2b). Specifically, as $N \to \infty$, the excess risk of DAC with the optimal λ converges to zero by learning from unbiased labels $\mathbb{E}_{\epsilon}\left[\mathbf{y}\right] = \mathbf{X}\boldsymbol{\theta}^*$, whereas DA-ERM suffers from a bias term $\|\mathbf{P}_{\boldsymbol{\Delta}}\boldsymbol{\theta}^*\|_{\boldsymbol{\Sigma}_{\widetilde{\boldsymbol{\Delta}}}}^2 > 0$ due to the bias from misspecified augmentations.

Example 3. As in Example 2, we consider a linear regression problem of dimension $d = 30$ with $\alpha \geq 1$ misspecified augmentations on $N = 50$ *i.i.d.* training samples drawn from $\mathcal{N}(\mathbf{0}, \mathbf{I}_d)$. We aim to learn $\boldsymbol{\theta}^* = [\boldsymbol{\theta}_c^*; \mathbf{0}] \in \mathbb{R}^d$ (where $\boldsymbol{\theta}_c^* \in \{-1, +1\}^{d_c}$, $d_c = 10$) under label noise $\sigma = 0.1$. The misspecified augmentations mimic the effect of color jitter by adding *i.i.d.* Gaussian noise entry-wisely to the last d_{aug} feature coordinates: $\widetilde{\mathcal{A}}\left(\mathbf{X}\right) = [\mathbf{X}; \mathbf{X}']$ where $\mathbf{X}'_{ij} = \mathbf{X}_{ij} + \mathcal{N}\left(0, 0.1\right)$ for all $i \in [N]$, $d - d_{aug} + 1 \leq j \leq d$ – such that $d_{aug} = \mathrm{rank}\left(\boldsymbol{\Delta}\right)$ with probability 1. The $(d - d_{aug} + 1), \dots, d_c$-th coordinates of $\boldsymbol{\theta}^*$ are misspecified by the augmentations.

As previously discussed on Theorem 4.2, DAC is more robust than DA-ERM to misspecified augmentations, and therefore can learn with fewer (smaller α) and stronger (larger d_{aug}) augmentations. In addition, DAC generally achieves better generalization than DA-ERM with limited samples.

4.6 Beyond Linear Model

In this section, we extend our analysis of DAC regularization to non-linear models, including the two-layer neural networks, and DNN-based classifiers with expansion-based augmentations.

Further, in addition to the popular in-distribution setting where we consider a unique distribution P for both training and testing, DAC regularization is also known to improve out-of-distribution generalization for settings like domain adaptation. We defer detailed discussion on such advantage of DAC regularization for linear regression in the domain adaptation setting to Section B.4.

4.6.1 Two-layer Neural Network

We first generalize our analysis to an illustrative nonlinear model – two-layer ReLU network. With $\mathcal{X} = \mathbb{R}^d$ and $\mathcal{Y} = \mathbb{R}$, we consider a ground truth distribution $P\left(y|\mathbf{x}\right)$ induced by $y = \left(\mathbf{x}^\top \mathbf{B}^*\right)_+ \mathbf{w}^* + \epsilon$. For the unknown ground truth function $h^*\left(\mathbf{x}\right) \triangleq \left(\mathbf{x}^\top \mathbf{B}^*\right)_+ \mathbf{w}^*$, $(\cdot)_+ \triangleq \max(0, \cdot)$ denotes the element-wisely ReLU function; $\mathbf{B}^* = \left[\mathbf{b}_1^* \dots \mathbf{b}_k^* \dots \mathbf{b}_q^*\right] \in \mathbb{R}^{d \times q}$ consists of $\mathbf{b}_k^* \in \mathbb{S}^{d-1}$ for all $k \in [q]$; and $\epsilon \sim \mathcal{N}\left(0, \sigma^2\right)$ is *i.i.d.* Gaussian noise. In terms of the function class $\mathcal{H}$, for some constant $C_w \geq \|\mathbf{w}^*\|_1$, let

$$\mathcal{H} = \left\{ h(\mathbf{x}) = (\mathbf{x}^\top \mathbf{B})_+ \mathbf{w} \;\middle|\; \mathbf{B} = [\mathbf{b}_1 \dots \mathbf{b}_q] \in \mathbb{R}^{d \times q}, \|\mathbf{b}_k\|_2 = 1 \; \forall\, j \in [q], \|\mathbf{w}\|_1 \leq C_w \right\},$$

such that $h^* \in \mathcal{H}$. For regression, we again consider square loss $l(h(\mathbf{x}), y) = \frac{1}{2}(h(\mathbf{x}) - y)^2$ and learn with DAC on the first layer: $\left(\mathbf{x}_i^\top \mathbf{B}\right)_+ = \left(\mathbf{x}_{i,j}^\top \mathbf{B}\right)_+$.

Let $\boldsymbol{\Delta} \triangleq \widetilde{\mathcal{A}}(\mathbf{X}) - \widetilde{\mathbf{M}}\mathbf{X}$, and $\mathbf{P}_{\boldsymbol{\Delta}}^{\perp}$ be the projector onto the null space of $\boldsymbol{\Delta}$. Under mild regularity conditions (*i.e.*, αN being sufficiently large, $\mathbf{x}$ being subgaussian, and distribution of $\boldsymbol{\Delta}$ being absolutely continuous, as specified in Section B.2), regression over two-layer ReLU networks with the DAC regularization generalizes as following:

Theorem 4.3 (Informal result on two-layer neural network with DAC (formally in Theorem B.2))**.** *Conditioned on* $\mathbf{X}$ *and* $\mathbf{\Delta}$*, with* $L(h) = \frac{1}{N}\left\|h(\mathbf{X}) - h^*(\mathbf{X})\right\|_2^2$ *and* $\sqrt{\frac{1}{N}\sum_{i=1}^{N}\left\|\mathbf{P}_{\mathbf{\Delta}}^{\perp}\mathbf{x}_i\right\|_2^2} \leq C_{\mathbb{N}}$*, for any* $\delta \in (0,1)$*, with probability at least* $1-\delta$ *over* $\boldsymbol{\epsilon}$*,*

$$L\left(\widehat{h}^{dac}\right) - L\left(h^*\right) \lesssim \sigma C_w C_{\mathbb{N}}\left(\frac{1}{\sqrt{N}} + \sqrt{\frac{\log(1/\delta)}{N}}\right).$$

Recall $d_{aug} = \mathrm{rank}(\mathbf{\Delta})$. With a sufficiently large N (as specified in Section B.2), we have $C_{\mathbb{N}} \lesssim \sqrt{d - d_{aug}}$ with high probability[3]. Meanwhile, applying DA-ERM directly on the augmented samples achieves no better than $L(\widehat{h}^{da-erm}) - L(h^*) \lesssim \sigma C_w \max\left(\sqrt{\frac{d}{(\alpha+1)N}}, \sqrt{\frac{d-d_{aug}}{N}}\right)$, where the first term corresponds to the generalization bound for a d-dimensional regression with $(\alpha+1)N$ *i.i.d.* samples (in contrast to augmented samples that are potentially dependent); and the second term follows as the augmentations $\widetilde{\mathcal{A}}\left(\mathbf{X}\right)$ keep a $(d - d_{aug})$-dimensional subspace (*i.e.*, the null space of $\mathbf{\Delta} = \widetilde{\mathcal{A}}(\mathbf{X}) - \widetilde{\mathbf{M}}\mathbf{X}$) intact, in which DA-ERM can only rely on the N original samples for learning. In specific, the first term will dominate the $\max$ with limited augmented data (i.e., α being small).

Comparing the two, we see that DAC tends to be more efficient than DA-ERM, and such advantage is enhanced with strong but limited data augmentations (i.e., large d_{aug} and small α). For instance, with $\alpha = 1$ and $d_{aug} = d - 1$, the generalization error of DA-ERM scales as $\sqrt{d/N}$, while DAC yields a dimension-free $\sqrt{1/N}$ error.

As a synopsis for the regression cases in Section 4.4, Section 4.5, and Section 4.6.1 generally, the effect of DAC regularization can be casted as a dimension reduction by d_{aug} – dimension of the subspace perturbed by data augmentations where features contain scarce label information.

4.6.2 Classification with Expansion-based Augmentations

A natural generalization of the dimension reduction viewpoint on DAC regularization in the regression setting is the complexity reduction for general function classes. Here we demonstrate the power of DAC on function class reduction in a DNN-based classification setting.

Concretely, we consider a multi-class classification problem: given a probability space

[3] Here we only account for the randomness in $\mathbf{X}$ but not that in $\mathbf{\Delta}|\mathbf{X}$ which characterizes d_{aug} for conciseness. We refer the readers to Section B.2 for a formal tail bound on $C_{\mathbb{N}}$.

$\mathcal{X}$ with marginal distribution $P(\mathbf{x})$ and K classes $\mathcal{Y} = [K]$, let $h^* : \mathcal{X} \to [K]$ be the ground truth classifier, partitioning $\mathcal{X}$ into K disjoint sets $\{\mathcal{X}_k\}_{k\in[K]}$ such that $P(y|\mathbf{x}) = \mathbf{1}\{y = h^*(\mathbf{x})\} = \mathbf{1}\{\mathbf{x} \in \mathcal{X}_y\}$. In the classification setting, we replace Definition 4.1 with the notion of *expansion-based data augmentations* introduced in [22, 162].

Definition 4.2 (Expansion-based augmentations (formally in Definition B.2))**.** With respect to an augmentation function $\mathcal{A} : \mathcal{X} \to 2^{\mathcal{X}}$, let $NB(S) \triangleq \cup_{\mathbf{x}\in S} \left\{\mathbf{x}' \in \mathcal{X} \,\middle|\, \mathcal{A}(\mathbf{x}) \cap \mathcal{A}(\mathbf{x}') \neq \emptyset\right\}$ be the neighborhood of $S \subseteq \mathcal{X}$. For any $c > 1$, we say that $\mathcal{A}$ induces c-expansion-based data augmentations if (a) $\{\mathbf{x}\} \subsetneq \mathcal{A}(\mathbf{x}) \subseteq \{\mathbf{x}' \in \mathcal{X} \mid h^*(\mathbf{x}) = h^*(\mathbf{x}')\}$ for all $\mathbf{x} \in \mathcal{X}$; and (b) for all $k \in [K]$, given any $S \subseteq \mathcal{X}$ with $P(S \cap \mathcal{X}_k) \leq \frac{1}{2}$, $P(NB(S) \cap \mathcal{X}_k) \geq \min\{c \cdot P(S \cap \mathcal{X}_k), 1\}$.

Particularly, Definition 4.2(a) enforces that the ground truth classifier h^* is invariant throughout each neighborhood. Meanwhile, the expansion factor c in Definition 4.2(b) serves as a quantification of augmentation strength – a larger c implies a stronger augmentation $\mathcal{A}$.

We aim to learn $h(\mathbf{x}) \triangleq \operatorname{argmax}_{k\in[K]} f(\mathbf{x})_k$ with loss $l_{01}(h(\mathbf{x}), y) = \mathbf{1}\{h(\mathbf{x}) \neq y\}$ from $\mathcal{H}$ induced by the class of p-layer fully connected neural networks with maximum width q, $\mathcal{F} = \left\{f : \mathcal{X} \to \mathbb{R}^K \,\middle|\, f = f_{2p-1} \circ \cdots \circ f_1,\right\}$ where $f_{2\iota-1}(\mathbf{x}) = \mathbf{W}_\iota \mathbf{x}$, $f_{2\iota}(\boldsymbol{\epsilon}) = \varphi(\boldsymbol{\epsilon})$, $\mathbf{W}_\iota \in \mathbb{R}^{d_\iota \times d_{\iota-1}}$ $\forall \iota \in [p]$, $q \triangleq \max_{\iota\in[p]} d_\iota$, and φ is the activation function.

Over a general probability space $\mathcal{X}$, DAC with expansion-based augmentations requires stronger conditions than merely consistent classification over $\mathcal{A}(\mathbf{x}_i)$ for all labeled training samples $i \in [N]$. Instead, we enforce a large robust margin $m_{\mathcal{A}}(f, \mathbf{x}^u)$ (adapted from [162], see Section B.3) over an finite set of unlabeled samples $\mathbf{X}^u$ that is independent of $\mathbf{X}$ and drawn *i.i.d.* from $P(\mathbf{x})$. Intuitively, $m_{\mathcal{A}}(f, \mathbf{x}^u)$ measures the maximum allowed perturbation in all parameters of f such that predictions remain consistent throughout $\mathcal{A}(\mathbf{x}^u)$ (*e.g.*, $m_{\mathcal{A}}(f, \mathbf{x}^u) > 0$ is equivalent to enforcing consistent classification outputs). For any $0 < \tau \leq \max_{f\in\mathcal{F}} \inf_{\mathbf{x}^u\in\mathcal{X}} m_{\mathcal{A}}(f, \mathbf{x}^u)$, the DAC regularization reduces the function class $\mathcal{H}$ to

$$\mathcal{H}_{dac} \triangleq \{h \in \mathcal{H} \mid m_{\mathcal{A}}(f, \mathbf{x}^u) > \tau \quad \forall\, \mathbf{x}^u \in \mathbf{X}^u\}.$$

Then for $\widehat{h}^{dac} = \operatorname{argmin}_{h\in\mathcal{H}_{dac}} \frac{1}{N} \sum_{i=1}^N l_{01}(h(\mathbf{x}_i), y_i)$, we have the following.

Theorem 4.4 (Informal result on classification with DAC (formally in Theorem B.7))**.** *Given an augmentation function $\mathcal{A}$ that induces c-expansion-based data augmentations (Definition 4.2) such that*

$$\mu \triangleq \sup_{h\in\mathcal{H}_{dac}} \mathbb{P}_P\left[\exists\, \mathbf{x}' \in \mathcal{A}(\mathbf{x}) : h(\mathbf{x}) \neq h(\mathbf{x}')\right] \leq \frac{c-1}{4},$$

Table 4.1: Testing accuracy of DA-ERM and DAC with different λ's (regularization coeff.).

DA-ERM	DAC Regularization				
	$\lambda = 0$	$\lambda = 1$	$\lambda = 5$	$\lambda = 10$	$\lambda = 20$
69.40 ± 0.05	62.82 ± 0.21	68.63 ± 0.11	$\mathbf{70.56 \pm 0.07}$	$\mathbf{70.52 \pm 0.14}$	68.65 ± 0.27

for any $\delta \in (0, 1)$*, with probability at least* $1 - \delta$*, we have* $\mu \leq \widetilde{O}\left(\frac{\sum_{\iota=1}^{p} \sqrt{q} \|\mathbf{W}_\iota\|_F}{\tau \sqrt{|\mathbf{X}^u|}} + \sqrt{\frac{p \log |\mathbf{X}^u|}{|\mathbf{X}^u|}}\right)$ *such that*

$$L_{01}\left(\widehat{h}^{dac}\right) - L_{01}\left(h^*\right) \lesssim \sqrt{\frac{K \log(N)}{N} + \frac{\mu}{\min\{c-1, 1\}}} + \sqrt{\frac{\log(1/\delta)}{N}}.$$

In particular, DAC regularization leverages the unlabeled samples $\mathbf{X}^u$ and effectively decouples the labeled sample complexity $N = \widetilde{O}(K)$ from the complexity of the function class $\mathcal{H}$ (characterized by $\{\mathbf{W}_\iota\}_{\iota \in [p]}$ and q and encapsulated in μ) via the reduced function class $\mathcal{H}_{dac}$. Notably, Theorem 4.4 is reminiscent of [162] Theorem 3.6, 3.7, and [22] Theorem 2.1, 2.2, 2.3. We unified the existing theories under our function class reduction viewpoint to demonstrate its generality.

4.7 Experiments

In this section, we empirically verify that training with DAC learns more efficiently than DA-ERM. The dataset is derived from CIFAR-100, where we randomly select 10,000 labeled data as the training set (i.e., 100 labeled samples per class). During the training time, given a training batch, we generate augmentations by RandAugment [35]. We set the number of augmentations per sample to 7 unless otherwise mentioned.

The experiments focus on comparisons of 1) training with consistency regularization (DAC), and 2) empirical risk minimization on the augmented dataset (DA-ERM). We use the same network architecture (a WideResNet-28-2 [176]) and the same training settings (e.g., optimizer, learning rate schedule, etc) for both methods. We defer the detailed experiment settings to Section B.6. Our test set is the standard CIFAR-100 test set, and we report the average and standard deviation of the testing accuracy of 5 independent runs. The consistency regularizer is implemented as the l_2 distance of the model's predictions on the original and augmented samples.

Efficacy of DAC regularization. We first show that the DAC regularization learns more efficiently than DA-ERM. The results are listed in Table 4.1. In practice, the augmentations

Table 4.2: Testing accuracy of DA-ERM and DAC with different numbers of augmentations.

Number of Augmentations	1	3	7	15
DA-ERM	67.92 ± 0.08	69.04 ± 0.05	69.25 ± 0.16	69.30 ± 0.11
DAC ($\lambda = 10$)	$\mathbf{70.06 \pm 0.08}$	$\mathbf{70.77 \pm 0.20}$	$\mathbf{70.74 \pm 0.11}$	$\mathbf{70.31 \pm 0.12}$

Table 4.3: Testing accuracy of ERM and DAC regularization with different numbers of labeled data.

Number of Labeled Data	1000	10000	20000
DA-ERM	31.11 ± 0.30	68.89 ± 0.07	$\mathbf{76.79 \pm 0.13}$
DAC ($\lambda = 10$)	$\mathbf{33.59 \pm 0.41}$	$\mathbf{70.71 \pm 0.10}$	$\mathbf{76.86 \pm 0.16}$

Table 4.4: DAC performs well under misspecified augmentations after tuning λ.

No Augmentation	DA-ERM	DAC ($\lambda = 0.1$)	DAC ($\lambda = 1$)	DAC ($\lambda = 10$)
62.82 ± 0.21	61.35 ± 0.27	63.73 ± 0.33	$\mathbf{64.30 \pm 0.20}$	64.00 ± 0.26

almost always alter the label distribution, we therefore follow the discussion in section 4.5 and adopt a finite λ (i.e., the multiplicative coefficient before the DAC regularization, see (4.1)). With proper choice of λ, training with DAC significantly improves over DA-ERM.

DAC regularization helps more with limited augmentations. Our theoretical results suggest that the DAC regularization learns efficiently with a limited number of augmentations. While keeping the number of labeled samples to be 10,000, we evaluate the performance of the DAC regularization and DA-ERM with different numbers of augmentations. The number of augmentations for each training sample ranges from 1 to 15, and the results are listed in Table 4.2. The DAC regularization offers a more significant improvement when the number of augmentations is small. This clearly demonstrates that the DAC regularization learns more efficiently than DA-ERM.

DAC regularization helps more when data is scarce. We conduct experiments with different numbers of labeled samples, ranging from 1,000 (i.e., 10 images per class) to 20,000 samples (i.e., 200 images per class). We generate 3 augmentations for each of the samples during the training time, and the results are presented in Table 4.3. Notice that the DAC regularization gives a bigger improvement over DA-ERM when the labeled samples are scarce. This matches the intuition that when there are sufficient training samples, data augmentation is less necessary. Therefore, the difference between different ways of utilizing the augmented samples becomes diminishing.

DAC performs well under misspecified augmentations. As suggested by Theorem 4.2,

Table 4.5: DAC helps FixMatch when the unlabeled data is scarce.

Number of Unlabeled Data	5000	10000	20000
FixMatch	67.74	69.23	70.76
FixMatch + DAC ($\lambda = 1$)	**71.24**	**72.7**	**74.04**

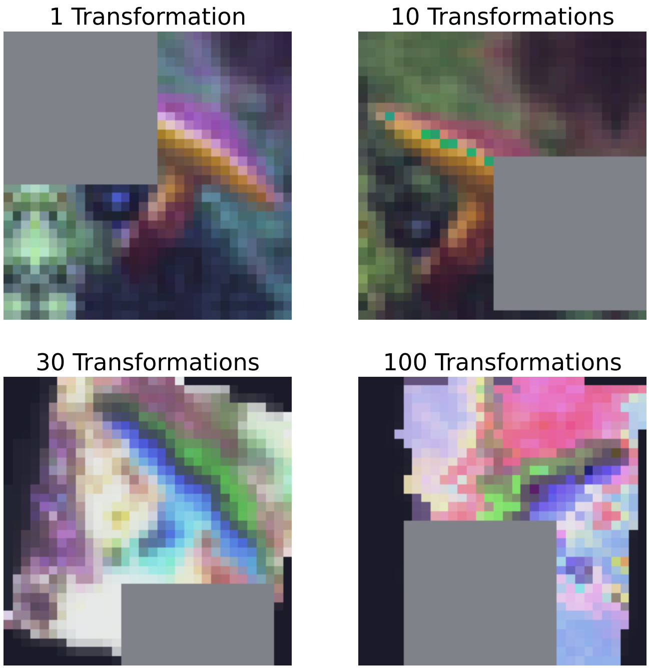


Figure 4.3: Different numbers of transformations.

DAC is more robust to misspecified augmentations with proper λ. We further empirically verify this result with misspecified augmentations - where the augmentations are generated by applying 100 random transformations. When too many transformations are applied (see illustration in Figure 4.3), the augmentation will alter the label distribution and is thus misspecified. The results are presented in Table 4.4. Notice that with $\lambda = 1$, the DAC delivers the best accuracy, which supports our theoretical results.

Further, comparing the results of Table 4.1 and Table 4.4, we see that the optimal λ is different when the augmentations are misspecified. Because of the flexibility in choosing λ, DAC is able to outperform DA-ERM, which matches the result in Theorem 4.2.

Combining with a semi-supervised learning algorithm. Here we show that the DAC regularization can be easily extended to the semi-supervised learning setting. We take the previously established semi-supervised learning method FixMatch [135] as the baseline and adapt the FixMatch by combining it with the DAC regularization. Specifically, besides using FixMatch to learn from the unlabeled data, we additionally generate augmentations for the labeled samples and apply DAC. In particular, we focus on the data-scarce regime by only keeping 10,000 labeled samples and at most 20,000 unlabeled samples. Results are listed in Table 4.5. We see that the DAC regularization also improves the performance of FixMatch when the unlabeled samples are scarce. This again demonstrates the efficiency of learning with DAC.

4.8 Conclusion

In this paper, we take a step toward understanding the statistical efficiency of DAC with limited data augmentations. At the core, DAC is statistically more efficient because it reduces problem dimensions by enforcing consistency regularization.

We demonstrate the benefits of DAC compared to DA-ERM (expanding training set with augmented samples) both theoretically and empirically. Theoretically, we show a strictly smaller generalization error under linear regression, and explicitly characterize the generalization upper bound for two-layer neural networks and expansion-based data augmentations. We further show that DAC better handles the label misspecification caused by strong augmentations. Empirically, we provide apples-to-apples comparisons between DAC and DA-ERM. These together demonstrate the superior efficacy of DAC over DA-ERM.

Chapter 5

Adaptively Weighted Data Augmentation Consistency Regularization: Application in Medical Image Segmentation

Abstract

Concept shift is a prevailing problem in natural tasks like medical image segmentation where samples usually come from different subpopulations with variant correlations between features and labels. One common type of concept shift in medical image segmentation is the "information imbalance" between *label-sparse* samples with few (if any) segmentation labels and *label-dense* samples with plentiful labeled pixels. Existing distributionally robust algorithms have focused on adaptively truncating/down-weighting the "less informative" (*i.e.*, label-sparse in our context) samples. To exploit data features of label-sparse samples more efficiently, we propose an adaptively weighted online optimization algorithm — *AdaWAC*— to incorporate data augmentation consistency regularization in sample reweighting. Our method introduces a set of trainable weights to balance the supervised loss and unsupervised consistency regularization of each sample separately. At the saddle point of the underlying objective, the weights assign label-dense samples to the supervised loss and label-sparse samples to the unsupervised consistency regularization. We provide a convergence guarantee by recasting the optimization as online mirror descent on a saddle point problem. Our empirical results demonstrate that *AdaWAC* not only enhances the segmentation performance and sample efficiency but also improves the robustness to concept shift on various medical image segmentation tasks with different UNet-style backbones.[1]

5.1 Introduction

Modern machine learning is revolutionizing the field of medical imaging, especially in computer-aided diagnosis with computed tomography (CT) and magnetic resonance imaging

[1] This chapter is based on the following published conference paper:
Yijun Dong, Yuege Xie, and Rachel Ward. Adaptively weighted data augmentation consistency regularization for robust optimization under concept shift. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 8296–8316. PMLR, 23–29 Jul 2023 [48].

(MRI) scans. However, classical learning objectives like empirical risk minimization (ERM) generally assume that training samples are independently and identically (*i.i.d.*) distributed, whereas real-world medical image data rarely satisfy this assumption. Figure 5.1 instantiates a common observation in medical image segmentation where the segmentation labels corresponding to different cross-sections of the human body tend to have distinct proportions of labeled (*i.e.*, non-background) pixels, which is accurately reflected by the evaluation of supervised cross-entropy loss during training. We refer to this as the "information imbalance" among samples, as opposed to the well-studied "class imbalance" [141, 164, 174] among the numbers of segmentation labels in different classes. Such information imbalance induces distinct difficulty/paces of learning with the cross-entropy loss for different samples [66, 142, 149, 159]. Specifically, we say a sample is *label-sparse* when it contains very few (if any) segmentation labels; in contrast, a sample is *label-dense* when its segmentation labels are prolific. Motivated by the information imbalance among samples, we explore the following questions:

What is the effect of separation between sparse and dense labels on segmentation?

Can we leverage such information imbalance to improve the segmentation accuracy?

We formulate the mixture of label-sparse and label-dense samples as a concept shift — a type of distribution shift in the conditional distribution of labels given features $P\left(\mathbf{y}|\mathbf{x}\right)$. Coping with concept shifts, prior works have focused on adaptively truncating (hard-thresholding) the empirical loss associated with label-sparse samples. These include the Trimmed Loss Estimator [130], MKL-SGD [127], Ordered SGD [82], and the quantile-based Kacmarz algorithm [67]. Alternatively, another line of works [122, 160] proposes to relax the hard-thresholding operation to soft-thresholding by down-weighting instead of truncating the less informative samples. However, diminishing sample weights reduces the importance of both the features and the labels simultaneously, which is still not ideal as the potentially valuable information in the features of the label-sparse samples may not be fully used.

For further exploitation of the feature of training samples, we propose the incorporation of *data augmentation consistency regularization* on label-sparse samples. As a prevalent strategy for utilizing unlabeled data, consistency regularization [5, 88, 135] encourages data augmentations of the same samples to lie in the vicinity of each other on a proper manifold. For medical imaging segmentation, consistency regularization has been extensively studied in the semi-supervised learning setting [7, 17, 90, 158, 178, 179, 181] as a strategy for overcoming label scarcity. Nevertheless, unlike general vision tasks, for medical image segmentation, the

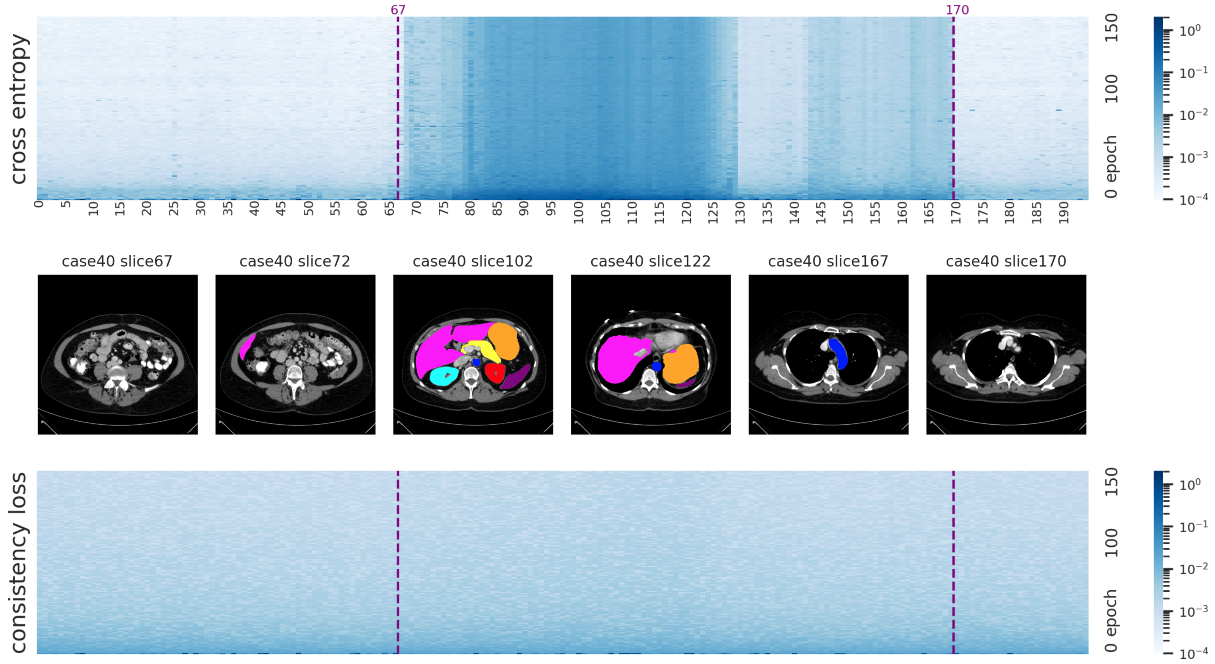


Figure 5.1: Evolution of cross-entropy losses versus consistency regularization terms for slices at different cross-sections of the human body in the Synapse dataset (described in Section 5.5) during training.

scantiness of unlabeled image data can also be a problem due to regulations and privacy considerations [80], which makes it worthwhile to reminisce the more classical supervised learning setting. In contrast to the aforementioned semi-supervised strategies, we explore the potency of consistency regularization in the *supervised learning* setting by leveraging the information in the features of label-sparse samples via data augmentation consistency regularization.

To naturally distinguish the label-sparse and label-dense samples, we make a key observation that the unsupervised consistency regularization on encoder layer outputs (of a UNet-style architecture) is much more uniform across different subpopulations than the supervised cross-entropy loss (as exemplified in Figure 5.1). Since the consistency regularization is characterized by the marginal distribution of features $P(\mathbf{x})$ but not labels, and therefore is less affected by the concept shift in $P(\mathbf{y}|\mathbf{x})$, it serves as a natural reference for separating the label-sparse and label-dense samples. In light of this observation, we present the *weighted data augmentation consistency (WAC) regularization* — a minimax formulation that reweights the cross-entropy loss versus the consistency regularization associated with each sample via a set of trainable weights. At the saddle point of this minimax formulation, the WAC regularization automatically separates samples from different subpopulations by assigning all weights to the consistency regularization for label-sparse samples, and all weights to the cross-entropy terms for label-dense samples.

We further introduce an adaptively weighted online optimization algorithm — *AdaWAC*— for solving the minimax problem posed by the WAC regularization, which is inspired by a mirror-descent-based algorithm for distributionally robust optimization [122]. By adaptively learning the weights between the cross-entropy loss and consistency regularization of different samples, *AdaWAC* comes with both a convergence guarantee and empirical success.

The main contributions are summarized as follows:

- We introduce the *WAC regularization* that leverages the consistency regularization on the encoder layer outputs (of a UNet-style architecture) as a natural reference to distinguish the label-sparse and label-dense samples (Section 5.3).
- We propose an adaptively weighted online optimization algorithm — *AdaWAC*— for solving the WAC regularization problem with a convergence guarantee (Section 5.4).
- Through extensive experiments on different medical image segmentation tasks with different UNet-style backbone architectures, we demonstrate the effectiveness of *AdaWAC* not only for enhancing the segmentation performance and sample efficiency but also for improving the robustness to concept shift (Section 5.5).

5.1.1 Related Work

Sample reweighting. Sample reweighting is a popular strategy for dealing with distribution/-subpopulation shifts in training data where different weights are assigned to samples from different subpopulations. In particular, the distributionally-robust optimization (DRO) framework [10, 53, 54, 122] considers a collection of training sample groups from different distributions. With the explicit grouping of samples, the goal is to minimize the worst-case loss over the groups. Without prior knowledge of sample grouping, importance sampling [2, 60, 81, 94, 110, 180], iterative trimming [82, 130], and empirical-loss-based reweighting [170] are commonly incorporated in the stochastic optimization process for adaptive reweighting and separation of samples from different subpopulations.

Data augmentation consistency regularization. As a popular way of exploiting data augmentations, consistency regularization encourages models to learn the vicinity among augmentations of the same sample based on the assumption that data augmentations generally preserve the semantic information in data and therefore lie closely on proper manifolds. Beyond being a powerful building block in semi-supervised [5, 12, 88, 124, 135] and self-supervised [28, 63, 70, 171] learning, the incorporation of data augmentation and consistency regularization also provably improves generalization and feature learning even in the supervised learning setting [129, 173].

For medical imaging, data augmentation consistency regularization is generally leveraged as a semi-supervised learning tool [7, 17, 90, 158, 178, 179, 181]. In efforts to incorporate consistency regularization in segmentation tasks with augmentation-sensitive labels, [90] encourages transformation consistency between predictions with augmentations applied to the image inputs and the segmentation outputs. [7] penalizes inconsistent segmentation outputs between teacher-student models, with MixUp [177] applied to image inputs of the teacher model and segmentation outputs of the student model. Instead of enforcing consistency in the segmentation output space as above, we leverage the insensitivity of sparse labels to augmentations and encourage consistent encodings (in the latent space of encoder outputs) on label-sparse samples.

5.2 Problem Setup

Notation. For any $K \in \mathbb{N}$, we denote $[K] = \{1, \ldots, K\}$. We represent the elements and subtensors of an arbitrary tensor by adapting the syntax for Python slicing on the subscript (except counting from 1). For example, $\mathbf{x}_{[i,j]}$ denotes the (i, j)-entry of the two-dimensional tensor $\mathbf{x}$, and $\mathbf{x}_{[i,:]}$ denotes the i-th row. Let $\mathbb{I}$ be a function onto $\{0, 1\}$ such that, for any event e, $\mathbb{I}\left\{e\right\} = 1$ if e is true and 0 otherwise. For any distribution P and $n \in \mathbb{N}$, we let P^n denote the joint distribution of n samples drawn *i.i.d.* from P. Finally, we say that an event happens with high probability (*w.h.p.*) if the event takes place with probability $1 - \Omega\left(\text{poly}\left(n\right)\right)^{-1}$.

5.2.1 Pixel-wise Classification with Sparse and Dense Labels

We consider medical image segmentation as a pixel-wise multi-class classification problem where we aim to learn a pixel-wise classifier $h : \mathcal{X} \to [K]^d$ that serves as a good approximation to the ground truth $h^* : \mathcal{X} \to [K]^d$.

Recall the separation of cross-entropy losses between samples with different proportions of background pixels from Figure 5.1. We refer to a sample $(\mathbf{x}, \mathbf{y}) \in \mathcal{X} \times [K]^d$ as *label-sparse* if most pixels in $\mathbf{y}$ are labeled as background; for these samples, the cross-entropy loss on $(\mathbf{x}, \mathbf{y})$

converges rapidly in the early stage of training. Otherwise, we say that $(\mathbf{x}, \mathbf{y})$ is *label-dense*. Formally, we describe such variation as a concept shift in the data distribution.

Definition 5.1 (Mixture of label-sparse and label-dense subpopulations)**.** We assume that *label-sparse and label-dense samples* are drawn from P_0 and P_1 with distinct conditional distributions $P_0(\mathbf{y}|\mathbf{x})$ and $P_1(\mathbf{y}|\mathbf{x})$ but common marginal distribution $P(\mathbf{x})$ such that $P_i(\mathbf{x}, \mathbf{y}) = P_i(\mathbf{y}|\mathbf{x}) P(\mathbf{x})$ $(i = 0, 1)$. For $\xi \in [0, 1]$, we define P_ξ as a data distribution where $(\mathbf{x}, \mathbf{y}) \sim P_\xi$ is drawn either from P_1 with probability ξ or from P_0 with probability $1 - \xi$.

We aim to learn a pixel-wise classifier from a function class $\mathcal{H} \ni h_\theta = \operatorname{argmax}_{k \in [K]} f_\theta(\mathbf{x})_{[j,:]}$ for all $j \in [d]$ where the underlying function $f_\theta \in \mathcal{F}$, parameterized by some $\theta \in \mathcal{F}_\theta$, admits an encoder-decoder structure:

$$\left\{ f_\theta = \phi_\theta \circ \psi_\theta \;\middle|\; \phi_\theta : \mathcal{X} \to \mathcal{Z}, \psi_\theta : \mathcal{Z} \to [0,1]^{d \times K} \right\}. \tag{5.1}$$

Here ϕ_θ, ψ_θ correspond to the encoder and decoder functions, respectively. The parameter space $\mathcal{F}_\theta$ is equipped with the norm $\|\cdot\|_{\mathcal{F}}$ and its dual norm $\|\cdot\|_{\mathcal{F},*}$[2]. $(\mathcal{Z}, \varrho)$ is a latent metric space.

To learn from segmentation labels, we consider the *averaged cross-entropy loss*:

$$\begin{aligned} \ell_{CE}(\theta; (\mathbf{x}, \mathbf{y})) &= -\frac{1}{d} \sum_{j=1}^{d} \sum_{k=1}^{K} \mathbb{I}\left\{ \mathbf{y}_{[j]} = k \right\} \cdot \log\left(f_\theta(\mathbf{x})_{[j,k]} \right) \\ &= -\frac{1}{d} \sum_{j=1}^{d} \log\left(f_\theta(\mathbf{x})_{[j, \mathbf{y}_{[j]}]} \right). \end{aligned} \tag{5.2}$$

We assume the proper learning setting: there exists $\theta^* \in \bigcap_{\xi \in [0,1]} \operatorname{argmin}_{\theta \in \mathcal{F}_\theta} \mathbb{E}_{(\mathbf{x},\mathbf{y}) \sim P_\xi} \left[\ell_{CE}(\theta; (\mathbf{x}, \mathbf{y})) \right]$, which is invariant with respect to ξ.[3]

5.2.2 Augmentation Consistency Regularization

Despite the invariance of f_{θ^*} to P_ξ on the population loss, with a finite number of training samples in practice, the predominance of label-sparse samples would be problematic. As an extreme scenario for the pixel-wise classifier with encoder-decoder structure ((5.1)), when the

[2] For *AdaWAC* (Proposition 5.2 in Section 5.4), $\mathcal{F}_\theta$ is simply a subspace in the Euclidean space with dimension equal to the total number of parameters for each $\theta \in \mathcal{F}_\theta$, with $\|\cdot\|_{\mathcal{F}}$ and $\|\cdot\|_{\mathcal{F},*}$ both being the ℓ_2-norm.

[3] We assume proper learning only to (i) highlight the invariance of the desired ground truth to ξ that can be challenging to learn with finite samples in practice and (ii) provide a natural pivot for the convex and compact neighborhood $\mathcal{F}_{\theta^*}(\gamma)$ of ground truth θ^* in Assumption 1 granted by the pretrained initialization, where θ^* can also be replaced with the pretrained initialization weights $\theta_0 \in \mathcal{F}_{\theta^*}(\gamma)$. In particular, neither our theory nor the *AdaWAC* algorithm requires the function class $\mathcal{F}$ to be expressive enough to truly contain such θ^*.

label-sparse samples are predominant ($\xi \ll 1$), a decoder function ψ_θ that predicts every pixel as background can achieve near-optimal cross-entropy loss, regardless of the encoder function ϕ_θ, considerably compromising the test performance (*cf.* Table 5.1). To encourage legit encoding even in the absence of sufficient dense labels, we leverage the unsupervised consistency regularization on the *encoder function* ϕ_θ based on data augmentations.

Let $\mathcal{A}$ be a distribution over transformations on $\mathcal{X}$ where for any $\mathbf{x} \in \mathcal{X}$, each $A \sim \mathcal{A}$ ($A : \mathcal{X} \to \mathcal{X}$) induces an augmentation $A(\mathbf{x})$ of $\mathbf{x}$ that perturbs low-level information in $\mathbf{x}$. We aim to learn an encoder function $\phi_\theta : \mathcal{X} \to \mathcal{Z}$ that is capable of filtering out low-level information from $\mathbf{x}$ and therefore provides similar encodings for augmentations of the same sample. Recalling the metric ϱ (*e.g.*, the Euclidean distance) on $\mathcal{Z}$, for a given scaling hyperparameter $\lambda_{AC} > 0$, we measure the similarity between augmentations with a consistency regularization term on $\phi_\theta(\cdot)$: for any $A_1, A_2 \sim \mathcal{A}^2$,

$$\ell_{AC}(\theta; \mathbf{x}, A_1, A_2) \triangleq \lambda_{AC} \cdot \varrho\Big(\phi_\theta(A_1(\mathbf{x})), \phi_\theta(A_2(\mathbf{x}))\Big). \tag{5.3}$$

For the n training samples $\{(\mathbf{x}_i, \mathbf{y}_i)\}_{i\in[n]} \sim P_\xi^n$, we consider n pairs of data augmentation transformations $\{(A_{i,1}, A_{i,2})\}_{i\in[n]} \sim \mathcal{A}^{2n}$. In the basic version, we encourage the similar encoding $\phi_\theta(\cdot)$ of the augmentation pairs $(A_{i,1}(\mathbf{x}_i), A_{i,2}(\mathbf{x}_i))$ for all $i \in [n]$ via consistency regularization:

$$\min_{\theta \in \mathcal{F}_{\theta^*}(\gamma)} \frac{1}{n} \sum_{i=1}^{n} \ell_{CE}(\theta; (\mathbf{x}_i, \mathbf{y}_i)) + \ell_{AC}(\theta; \mathbf{x}_i, A_{i,1}, A_{i,2}). \tag{5.4}$$

We enforce consistency on $\phi_\theta(\cdot)$ in light of the encoder-decoder architecture: the encoder is generally designed to abstract essential information and filters out low-level non-semantic perturbations (*e.g.*, those introduced by augmentations), while the decoder recovers the low-level information for the pixel-wise classification. Therefore, with different $A_1, A_2 \sim \mathcal{A}$, the encoder output $\phi_\theta(\cdot)$ tends to be more consistent than the other intermediate layers, especially for label-dense samples.

5.3 Weighted Augmentation Consistency (WAC) Regularization

As the motivation, we begin with a key observation about the averaged cross-entropy:

Remark 5.1 (Separation of averaged cross-entropy loss on P_0 and P_1). As demonstrated in Figure 5.1, the sparse labels from P_0 tend to be much easier to learn than the dense ones from P_1, leading to considerable separation of averaged cross-entropy losses on the sparse and dense labels after a sufficient number of training epochs. In other words, $\ell_{CE}(\theta; (\mathbf{x}, \mathbf{y})) \ll \ell_{CE}(\theta; (\mathbf{x}', \mathbf{y}'))$ for label-sparse samples $(\mathbf{x}, \mathbf{y}) \sim P_0$ and label-dense samples $(\mathbf{x}', \mathbf{y}') \sim P_1$ with high probability.

Although (5.4) with consistency regularization alone can boost the segmentation accuracy during testing (*cf.* Table 5.4), it does not take the separation between label-sparse and label-dense samples into account. In Section 5.5, we will empirically demonstrate that proper exploitation of such separation, like the formulation introduced below, can lead to improved classification performance.

We formalize the notion of separation between P_0 and P_1 with the consistency regularization ((5.3)) as a reference in the following assumption [4].

Assumption 1 (n-separation between P_0 and P_1). Given a sufficiently small $\gamma > 0$, let $\mathcal{F}_{\theta^*}(\gamma) = \{\theta \in \mathcal{F}_\theta \mid \|\theta - \theta^*\|_{\mathcal{F}} \leq \gamma\}$ be a compact and convex neighborhood of well-trained pixel-wise classifiers[5]. We say that P_0 *and* P_1 *are n-separated over* $\mathcal{F}_{\theta^*}(\gamma)$ if there exists $\omega > 0$ such that with probability $1 - \Omega\left(n^{1+\omega}\right)^{-1}$ over $((\mathbf{x}, \mathbf{y}), (A_1, A_2)) \sim P_\xi \times \mathcal{A}^2$, the following hold:

(i) $\ell_{CE}(\theta; (\mathbf{x}, \mathbf{y})) < \ell_{AC}(\theta; \mathbf{x}, A_1, A_2)$ for all $\theta \in \mathcal{F}_{\theta^*}(\gamma)$ given $(\mathbf{x}, \mathbf{y}) \sim P_0$;

(ii) $\ell_{CE}(\theta; (\mathbf{x}, \mathbf{y})) > \ell_{AC}(\theta; \mathbf{x}, A_1, A_2)$ for all $\theta \in \mathcal{F}_{\theta^*}(\gamma)$ given $(\mathbf{x}, \mathbf{y}) \sim P_1$.

This assumption is motivated by the empirical observation that the perturbation in $\phi_\theta(\cdot)$ induced by $\mathcal{A}$ is more uniform across P_0 and P_1 than the averaged cross-entropy losses, as instantiated in Figure 5.3.

Under Assumption 1, up to a proper scaling hyperparameter λ_{AC}, the consistency regularization ((5.3)) can separate the averaged cross-entropy loss ((5.2)) on n label-sparse and label-dense samples with probability $1 - \Omega\left(n^{\omega}\right)^{-1}$ (as explained formally in Section C.1). In particular, the larger n corresponds to the stronger separation between P_0 and P_1.

With Assumption 1, we introduce a minimax formulation that incentivizes the separation of label-sparse and label-dense samples automatically by introducing a flexible weight $\boldsymbol{\beta}_{[i]} \in [0,1]$ that balances $\ell_{CE}(\theta; (\mathbf{x}_i, \mathbf{y}_i))$ and $\ell_{AC}(\theta; \mathbf{x}_i, A_{i,1}, A_{i,2})$ for each of the n samples.

$$\begin{aligned} \widehat{\theta}^{WAC}, \widehat{\boldsymbol{\beta}} &\in \underset{\theta \in \mathcal{F}_{\theta^*}(\gamma)}{\operatorname{argmin}} \ \underset{\boldsymbol{\beta} \in [0,1]^n}{\operatorname{argmax}} \ \widehat{L}^{WAC}(\theta, \boldsymbol{\beta}) \\ \widehat{L}^{WAC}(\theta, \boldsymbol{\beta}) &\triangleq \frac{1}{n} \sum_{i=1}^{n} \boldsymbol{\beta}_{[i]} \cdot \ell_{CE}(\theta; (\mathbf{x}_i, \mathbf{y}_i)) + \left(1 - \boldsymbol{\beta}_{[i]}\right) \cdot \ell_{AC}(\theta; \mathbf{x}_i, A_{i,1}, A_{i,2}) . \end{aligned} \tag{5.5}$$

[4] Although Assumption 1 may seem to be rather strong, it is only required for the separation guarantee of label-sparse and label-dense samples with high probability in Proposition 5.1, but not for the adaptive weighting algorithm introduced in Section 5.4 or in practice for the experiments.

[5] With pretrained initialization, we assume that the optimization algorithm is always probing in $\mathcal{F}_{\theta^*}(\gamma)$.

With convex and continuous loss and regularization terms (formally in Proposition 5.1), (5.5) admits a saddle point corresponding to $\widehat{\boldsymbol{\beta}}$ which separates the label-sparse and label-dense samples under Assumption 1.

Proposition 5.1 (Formal proof in Section C.1)**.** *Assume that* $\ell_{CE}\left(\theta; (\mathbf{x}, \mathbf{y})\right)$ *and* $\ell_{AC}\left(\theta; \mathbf{x}, A_1, A_2\right)$ *are convex and continuous in* θ *for all* $(\mathbf{x}, \mathbf{y}) \in \mathcal{X} \times [K]^d$ *and* $A_1, A_2 \sim \mathcal{A}^2$*;* $\mathcal{F}_{\theta^*}(\gamma) \subset \mathcal{F}_\theta$ *is compact and convex. If* P_0 *and* P_1 *are* n*-separated (Assumption 1), then there exists* $\widehat{\boldsymbol{\beta}} \in \{0,1\}^n$ *and* $\widehat{\theta}^{WAC} \in \operatorname{argmin}_{\theta \in \mathcal{F}_{\theta^*}(\gamma)} \widehat{L}^{WAC}\left(\theta, \widehat{\boldsymbol{\beta}}\right)$ *such that*

$$\min_{\theta \in \mathcal{F}_{\theta^*}(\gamma)} \widehat{L}^{WAC}\left(\theta, \widehat{\boldsymbol{\beta}}\right) = \widehat{L}^{WAC}\left(\widehat{\theta}^{WAC}, \widehat{\boldsymbol{\beta}}\right) = \max_{\boldsymbol{\beta} \in [0,1]^n} \widehat{L}^{WAC}\left(\widehat{\theta}^{WAC}, \boldsymbol{\beta}\right). \tag{5.6}$$

Further, $\widehat{\boldsymbol{\beta}}$ *separates the label-sparse and label-dense samples—*$\widehat{\boldsymbol{\beta}}_{[i]} = \mathbb{I}\left\{(\mathbf{x}_i, \mathbf{y}_i) \sim P_1\right\}$*—*w.h.p..

That is, for n samples drawn from a mixture of n-separated P_0 and P_1, the saddle point of $L_i^{WAC}(\theta, \boldsymbol{\beta})$ in (5.5) corresponds to $\boldsymbol{\beta}_{[i]} = 0$ on label-sparse samples (*i.e.*, learning from the unsupervised consistency regularization), and $\boldsymbol{\beta}_{[i]} = 1$ on label-dense samples (*i.e.*, learning from the supervised averaged cross-entropy loss).

Remark 5.2 (Connection to hard-thresholding algorithms). The saddle point of (5.5) is closely related to hard-thresholding algorithms like Ordered SGD [82] and iterative trimmed loss [130]. In each iteration, these algorithms update the model only on a proper subset of training samples based on the (ranking of) current empirical risks. Compared to hard-thresholding algorithms, (i) (5.5) additionally leverages the unused samples (*e.g.*, label-sparse samples) for unsupervised consistency regularization on data augmentations; (ii) meanwhile, it does not require prior knowledge of the sample subpopulations (*e.g.*, ξ for P_ξ) which is essential for hard-thresholding algorithms.

(5.5) further facilitates the more flexible optimization process. As we will empirically show in Table 5.2, despite the close relation between (5.5) and the hard-thresholding algorithms (Remark 5.2), such updating strategies may be suboptimal for solving (5.5).

5.4 Adaptively Weighted Augmentation Consistency (*AdaWAC*)

Inspired by the breakthrough made by [122] in the distributionally-robust optimization (DRO) setting where gradient updating on weights is shown to enjoy better convergence guarantees than hard thresholding, we introduce an adaptively weighted online optimization algorithm (Algorithm 6) for solving (5.5) based on online mirror descent.

In contrast to the commonly used stochastic gradient descent (SGD), the flexibility of online mirror descent in choosing the associated norm space not only allows gradient updates on sample weights but also grants distinct learning dynamics to sample weights $\boldsymbol{\beta}_t$ and model parameters θ_t, which leads to the following convergence guarantee.

Proposition 5.2 (Formally in Proposition C.1, proof in Section C.2, assumptions instantiated in Example 5)**.** *Assume that $\ell_{CE}(\theta; (\mathbf{x}, \mathbf{y}))$ and $\ell_{AC}(\theta; \mathbf{x}, A_1, A_2)$ are convex and continuous in θ for all $(\mathbf{x}, \mathbf{y}) \in \mathcal{X} \times [K]^d$ and $A_1, A_2 \sim \mathcal{A}^2$. Assume moreover that $\mathcal{F}_{\theta^*}(\gamma) \subset \mathcal{F}_\theta$ is convex and compact. If there exist* [6] *$C_{\theta,*} > 0$ and $C_{\beta,*} > 0$ such that*

$$\frac{1}{n}\sum_{i=1}^{n} \left\| \nabla_\theta \widehat{L}_i^{WAC}(\theta, \boldsymbol{\beta}) \right\|_{\mathcal{F},*}^2 \leq C_{\theta,*}^2$$

$$\frac{1}{n}\sum_{i=1}^{n} \max\left\{\ell_{CE}(\theta; (\mathbf{x}_i, \mathbf{y}_i)), \ell_{AC}(\theta; \mathbf{x}_i, A_{i,1}, A_{i,2})\right\}^2 \leq C_{\beta,*}^2$$

for all $\theta \in \mathcal{F}_{\theta^*}(\gamma)$, $\boldsymbol{\beta} \in [0,1]^n$, then with $\eta_\theta = \eta_\beta = \frac{2}{\sqrt{5T\left(\gamma^2 C_{\theta,*}^2 + 2nC_{\beta,*}^2\right)}}$, Algorithm 6 provides

$$\mathbb{E}\left[\max_{\boldsymbol{\beta}\in[0,1]^n} \widehat{L}^{WAC}\left(\overline{\theta}_T, \boldsymbol{\beta}\right) - \min_{\theta\in\mathcal{F}_{\theta^*}(\gamma)} \widehat{L}^{WAC}\left(\theta, \overline{\boldsymbol{\beta}}_T\right)\right]$$
$$\leq 2\sqrt{5\left(\gamma^2 C_{\theta,*}^2 + 2nC_{\beta,*}^2\right)/T}$$

where $\overline{\theta}_T = \frac{1}{T}\sum_{t=1}^{T}\theta_t$ and $\overline{\boldsymbol{\beta}}_T = \frac{1}{T}\sum_{t=1}^{T}\boldsymbol{\beta}_t$.

In addition to the convergence guarantee, Algorithm 6 also demonstrates superior performance over hard-thresholding algorithms for segmentation problems in practice (Table 5.2). An intuitive explanation is that instead of filtering out all the label-sparse samples via hard thresholding, the adaptive weighting allows the model to learn from some sparse labels at the early epochs, while smoothly down-weighting ℓ_{CE} of these samples since learning sparse labels tends to be easier (Remark 5.1). With the learned model tested on a mixture of label-sparse and label-dense samples, learning sparse labels at the early stage is crucial for accurate segmentation.

5.5 Experiments

In this section, we investigate the proposed *AdaWAC* algorithm (Algorithm 6) on different medical image segmentation tasks with different UNet-style architectures. We first demon-

[6] Following the convention, we use $*$ in subscription to denote the dual spaces. For instance, recalling the parameter space $\mathcal{F}_\theta$ characterized by the norm $\|\cdot\|_{\mathcal{F}}$ from Section 5.2.1, we use $\|\cdot\|_{\mathcal{F},*}$ to denote its dual norm; while $C_{\theta,*}, C_{\beta,*}$ upper bound the dual norms of the gradients with respect to θ and $\boldsymbol{\beta}$.

Algorithm 6 Adaptively Weighted Augmentation Consistency (*AdaWAC*)

```
**Input:** Training samples $\{(\mathbf{x}_i, \mathbf{y}_i)\}_{i\in[n]} \sim P_\xi^n$, augmentations $\{(A_{i,1}, A_{i,2})\}_{i\in[n]} \sim \mathcal{A}^{2n}$, maximum number of iterations $T \in \mathbb{N}$, learning rates $\eta_\theta, \eta_\beta > 0$, pretrained initialization for the pixel-wise classifier $\theta_0 \in \mathcal{F}_{\theta^*}(\gamma)$.
Initialize the sample weights $\boldsymbol{\beta}_0 = \mathbf{1}/2 \in [0,1]^n$.
for $t = 1, \dots, T$ do
    Sample $i_t \sim [n]$ uniformly
    $\mathbf{b} \leftarrow \left[(\boldsymbol{\beta}_{t-1})_{[i_t]}, 1 - (\boldsymbol{\beta}_{t-1})_{[i_t]}\right]$
    $\mathbf{b}_{[1]} \leftarrow \mathbf{b}_{[1]} \cdot \exp\left(\eta_\beta \cdot \ell_{CE}\left(\theta_{t-1}; (\mathbf{x}_{i_t}, \mathbf{y}_{i_t})\right)\right)$
    $\mathbf{b}_{[2]} \leftarrow \mathbf{b}_{[2]} \cdot \exp\left(\eta_\beta \cdot \ell_{AC}\left(\theta_{t-1}; \mathbf{x}_{i_t}, A_{i_t,1}, A_{i_t,2}\right)\right)$
    $\boldsymbol{\beta}_t \leftarrow \boldsymbol{\beta}_{t-1}$, $(\boldsymbol{\beta}_t)_{[i_t]} \leftarrow \mathbf{b}_{[1]} / \|\mathbf{b}\|_1$
    $\theta_t \leftarrow \theta_{t-1} - \eta_\theta \cdot \Big((\boldsymbol{\beta}_t)_{[i_t]} \cdot \nabla_\theta \ell_{CE}\left(\theta_{t-1}; (\mathbf{x}_{i_t}, \mathbf{y}_{i_t})\right) + \left(1 - (\boldsymbol{\beta}_t)_{[i_t]}\right) \cdot \nabla_\theta \ell_{AC}\left(\theta_{t-1}; \mathbf{x}_{i_t}, A_{i_t,1}, A_{i_t,2}\right)\Big)$
```

strate the performance improvements brought by *AdaWAC* in terms of sample efficiency and robustness to concept shift (Table 5.1). Then, we verify the empirical advantage of *AdaWAC* compared to the closely related hard-thresholding algorithms as discussed in Remark 5.2 (Table 5.2). Our ablation study (Table 5.4) further illustrates the indispensability of both sample reweighting and consistency regularization, the deliberate combination of which leads to the superior performance of *AdaWAC*[7].

Experiment setup. We conduct experiments on two medical image segmentation tasks: abdominal CT segmentation for Synapse multi-organ dataset (Synapse)[8] and cine-MRI segmentation for Automated cardiac diagnosis challenge dataset (ACDC)[9], with two UNet-like architectures: TransUNet [26] and UNet [119] (deferred to Section C.5.1). For the main experiments with TransUNet in Section 5.5, we follow the official implementation in [26] and use ERM+SGD as the baseline. We evaluate segmentations with two standard metrics—the average Dice-similarity coefficient (DSC) and the average 95-percentile of Hausdorff distance (HD95). Dataset and implementation details are deferred to Section C.4. Given the sensitivity of medical image semantics to perturbations, our experiments only involve simple augmentations (*i.e.*, rotation and mirroring) adapted from [26].

It is worth highlighting that, in addition to the information imbalance among samples

[7] We release our code anonymously at https://anonymous.4open.science/r/adawac-F5F8.

[8] https://www.synapse.org/#!Synapse:syn3193805/wiki/217789

[9] https://www.creatis.insa-lyon.fr/Challenge/acdc/

caused by the concept shift discussed in this work, the pixel-wise class imbalance (*e.g.*, the predominance of background pixels) is another well-investigated challenge for medical image segmentation, where coupling the dice loss [141, 164, 174] in the objective is a common remedy used in many state-of-the-art methods [23, 26]. The implementation of *AdaWAC* also leverages the dice loss to alleviate pixel-wise class imbalance. We defer the detailed discussion to Section C.3.

5.5.1 Segmentation Performance of *AdaWAC* with TransUNet

Segmentation on Synapse. Figure 5.2 visualizes the segmentation predictions on 6 Synapse test slices given by models trained via *AdaWAC* (ours) and via the baseline (ERM+SGD) with TransUNet [26]. We observe that *AdaWAC* provides more accurate predictions on the segmentation boundaries and captures small organs better than the baseline.

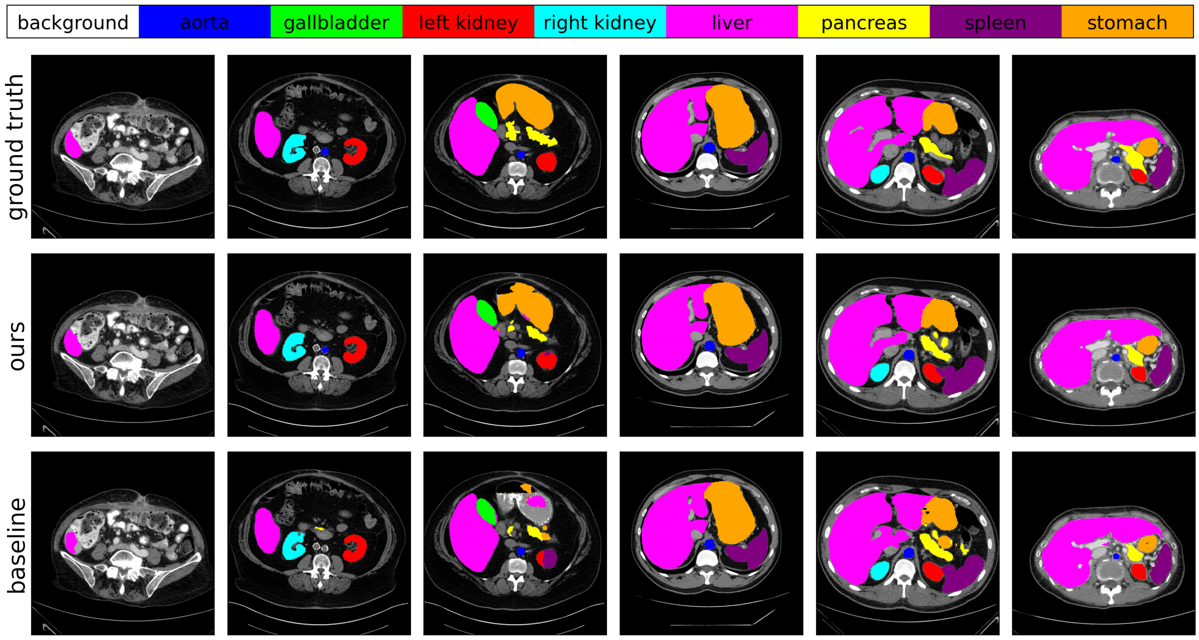


Figure 5.2: Visualization of segmentation predictions with TransUNet [26] on Synapse. Top to bottom: ground truth, ours (*AdaWAC*), baseline.

Visualization of *AdaWAC*. As shown in Figure 5.3, with $\ell_{CE}\left(\theta_t; (\mathbf{x}_i, \mathbf{y}_i)\right)$ ((5.2)) of label-sparse versus label-dense slices weakly separated in the early epochs, the model further learns to distinguish $\ell_{CE}\left(\theta_t; (\mathbf{x}_i, \mathbf{y}_i)\right)$ of label-sparse/label-dense slices during training. By contrast, $\ell_{AC}\left(\theta_t; \mathbf{x}_i, A_{i,1}, A_{i,2}\right)$ ((5.3)) remains mixed for all slices throughout the entire training process.

As a result, the CE weights of label-sparse slices are much smaller than those of label-dense ones, pushing *AdaWAC* to learn more image representations but fewer pixel classifications for slices with sparse labels and learn more pixel classifications for slices with dense labels.

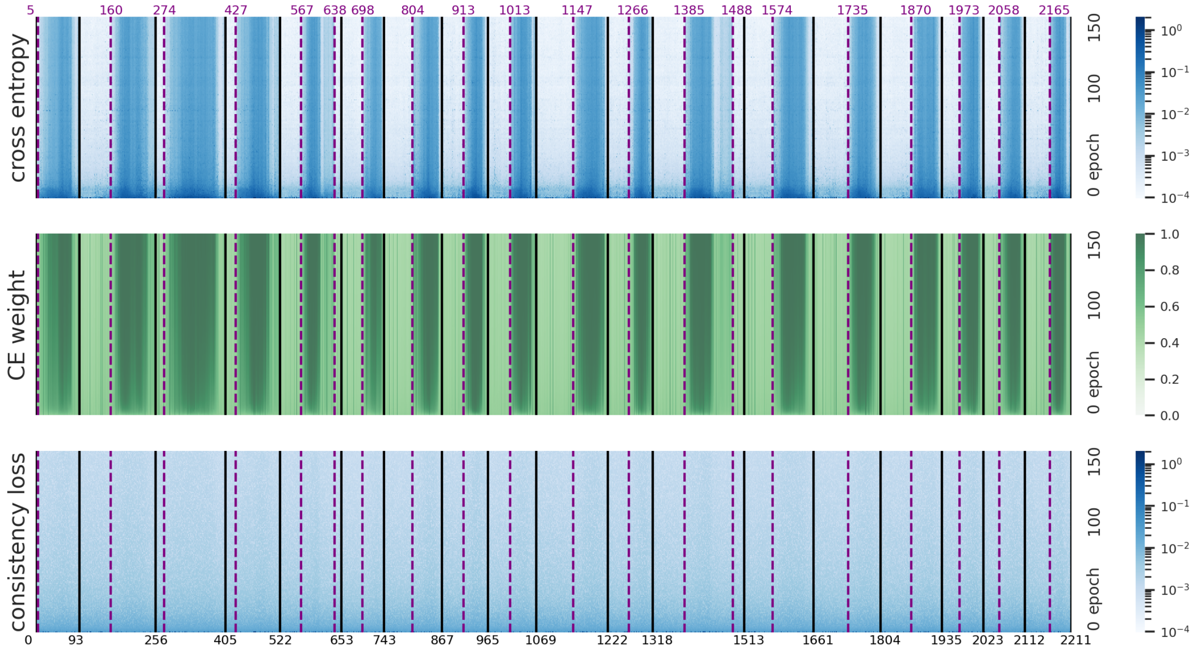


Figure 5.3: $\ell_{CE}\left(\theta_t;(\mathbf{x}_i,\mathbf{y}_i)\right)$ (top), CE weights $\boldsymbol{\beta}_t$ (middle), and $\ell_{AC}\left(\theta_t;\mathbf{x}_i,A_{i,1},A_{i,2}\right)$ (bottom) of the entire Synapse training process. The x-axis indexes slices 0–2211. The y-axis enumerates epochs 0–150. Individual cases (patients) are partitioned by black lines, while purple lines separate slices with/without non-background pixels.

Sample efficiency and robustness. We first demonstrate the *sample efficiency* of *AdaWAC* in comparison to the baseline (ERM+SGD) when training only on different subsets of the full Synapse training set ("**full**" in Table 5.1). Specifically, (i) **half-slice** contains slices with even indices only in each case (patient)[10]; (ii) **half-vol** consists of 9 cases uniformly sampled from the total 18 cases in **full** where different cases tend to have distinct ξs (*i.e.*, ratios of label-dense samples); (iii) **half-sparse** takes the first half slices in each case, most of which tend to be label-sparse (*i.e.*, ξs are made to be small). As shown in Table 5.1, the model trained with *AdaWAC* on **half-slice** generalizes as well as a baseline model trained on **full**, if not better. Moreover, the **half-vol** and **half-sparse** experiments illustrate the *robustness* of *AdaWAC* to concept shift. Furthermore, such sample efficiency and distributional robustness of *AdaWAC*

[10] Such sampling is equivalent to doubling the time interval between two consecutive scans or halving the scanning frequency in practice, resulting in the halving of sample size.

extend to the more widely used UNet architecture. We defer the detailed results and discussions on UNet to Section C.5.1.

Table 5.1: *AdaWAC* with TransUNet trained on the full Synapse and its subsets.

Training	Method	DSC ↑	HD95 ↓	Aorta	Gallbladder	Kidney (L)	Kidney (R)	Liver	Pancreas	Spleen	Stomach
full	baseline	76.66 ± 0.88	29.23 ± 1.90	87.06	55.90	81.95	75.58	94.29	56.30	86.05	76.17
	AdaWAC	**79.04** ± **0.21**	**27.39** ± **1.91**	87.53	56.57	83.23	81.12	94.04	62.05	89.51	78.32
half-slice	baseline	74.62 ± 0.78	31.62 ± 8.37	86.14	44.23	79.09	78.46	93.50	55.78	84.54	75.24
	AdaWAC	**77.37** ± **0.40**	**29.56** ± **1.09**	86.89	55.96	82.15	78.63	94.34	57.36	86.60	77.05
half-vol	baseline	71.08 ± 0.90	46.83 ± 2.91	84.38	46.71	78.19	74.55	92.02	48.03	76.28	68.47
	AdaWAC	**73.81** ± **0.94**	**35.33** ± **0.92**	84.37	48.14	80.32	77.39	93.23	52.78	83.50	70.79
half-sparse	baseline	31.74 ± 2.78	69.72 ± 1.37	65.71	8.33	59.46	51.59	51.18	10.72	6.92	0.00
	AdaWAC	**41.03** ± **2.12**	**59.04** ± **12.32**	71.27	8.33	69.14	63.09	64.29	17.74	30.77	3.57

Comparison with hard-thresholding algorithms. Table 5.2 illustrates the empirical advantage of *AdaWAC* over the hard-thresholding algorithms, as suggested in Remark 5.2. In particular, we consider the following hard-thresholding algorithms: (i) **trim-train** learns only from slices with at least one non-background pixel and trims the rest in each iteration on the fly; (ii) **trim-ratio** ranks the cross-entropy loss $\ell_{CE}\left(\theta_t; (\mathbf{x}_i, \mathbf{y}_i)\right)$ in each iteration (mini-batch) and trims samples with the lowest cross-entropy losses at a fixed ratio – the ratio of all-background slices in the full training set ($1 - \frac{1280}{2211} \approx 0.42$); (iii) **ACR** further incorporates the data augmentation consistency regularization directly via the addition of $\ell_{AC}\left(\theta_t; \mathbf{x}_i, A_{i,1}, A_{i,2}\right)$ without reweighting; (iv) **pseudo-*AdaWAC*** simulates the sample weights $\boldsymbol{\beta}$ at the saddle point and learns via $\ell_{CE}\left(\theta_t; (\mathbf{x}_i, \mathbf{y}_i)\right)$ on slices with at least one non-background pixel while via $\ell_{AC}\left(\theta_t; \mathbf{x}_i, A_{i,1}, A_{i,2}\right)$ otherwise. We see that naive incorporation of **ACR** brings less observable boosts to the hard-thresholding methods. Therefore, the deliberate combination via reweighting in *AdaWAC* is essential for performance improvement.

Table 5.2: *AdaWAC* versus hard-thresholding algorithms with TransUNet on Synapse.

Method	baseline	trim-train		trim-ratio		pseudo-*AdaWAC*	*AdaWAC*
			+ACR		+ACR		
DSC ↑	76.66 ± 0.88	76.80 ± 1.13	78.42 ± 0.17	76.49 ± 0.16	77.71 ± 0.56	77.72 ± 0.65	**79.04** ± **0.21**
HD95 ↓	29.23 ± 1.90	32.05 ± 2.34	27.84 ± 1.16	31.96 ± 2.60	28.51 ± 2.66	28.45 ± 1.18	**27.39** ± **1.91**

Segmentation on ACDC. Performance improvements granted by *AdaWAC* are also observed on the ACDC dataset (Table 5.3). We defer detailed visualization of ACDC segmentation to Section C.5.

Table 5.3: *AdaWAC* with TransUNet trained on ACDC.

Method	DSC ↑	HD95 ↓	RV	Myo	LV
TransUNet	89.40 ± 0.22	2.55 ± 0.37	89.17	83.24	95.78
AdaWAC (ours)	**90.67 ± 0.27**	**1.45 ± 0.55**	90.00	85.94	96.06

5.5.2 Ablation Study

On the influence of consistency regularization. To illustrate the role of consistency regularization in *AdaWAC*, we consider the **reweight-only** scenario with $\lambda_{AC} = 0$ such that $\ell_{AC}(\theta_t; \mathbf{x}_i, A_{i,1}, A_{i,2}) \equiv 0$ and therefore $\mathbf{b}_{[2]}$ (Algorithm 6 line 7) remains intact. With zero consistency regularization in *AdaWAC*, reweighting alone brings little improvement (Table 5.4).

On the influence of sample reweighting. We then investigate the effect of sample reweighting under different reweighting learning rates η_β (recall Algorithm 6): (i) **ACR-only** for $\eta_\beta = 0$ (equivalent to the naive addition of $\ell_{AC}(\theta_t; \mathbf{x}_i, A_{i,1}, A_{i,2})$), (ii) ***AdaWAC*-0.01** for $\eta_\beta = 0.01$, and (iii) ***AdaWAC*-1.0** for $\eta_\beta = 1.0$. As Table 5.4 implies, when removing reweighting from *AdaWAC*, augmentation consistency regularization alone improves DSC slightly from 76.28 (baseline) to 77.89 (ACR-only), whereas *AdaWAC* boosts DSC to 79.12 (*AdaWAC*-1.0) with a proper choice of η_β.

Table 5.4: Ablation study of *AdaWAC* with TransUNet trained on Synapse.

Method	DSC ↑	HD95 ↓	Aorta	Gallbladder	Kidney (L)	Kidney (R)	Liver	Pancreas	Spleen	Stomach
baseline	76.66 ± 0.88	29.23 ± 1.90	87.06	55.90	81.95	75.58	94.29	56.30	86.05	76.17
reweight-only	76.91 ± 0.88	30.92 ± 2.37	87.18	52.89	82.15	77.11	94.15	58.35	86.36	77.08
ACR-only	78.01 ± 0.62	27.78 ± 2.80	87.51	58.79	83.39	79.26	94.70	58.99	86.02	75.43
AdaWAC-0.01	77.75 ± 0.23	28.02 ± 3.50	87.33	56.68	83.35	78.53	94.45	57.02	87.72	76.94
AdaWAC-1.0	**79.04 ± 0.21**	**27.39 ± 1.91**	87.53	56.57	83.23	81.12	94.04	62.05	89.51	78.32

5.6 Discussion

In this paper, we explore the information imbalance commonly observed in medical image segmentation and exploit the information in features of label-sparse samples via *AdaWAC*, an adaptively weighted online optimization algorithm. *AdaWAC* can be viewed as a careful combination of adaptive sample reweighting and data augmentation consistency regularization. By casting the information imbalance among samples as a concept shift in the data distribution, we leverage the unsupervised data augmentation consistency regularization on the encoder layer outputs (of UNet-style architectures) as a natural reference for distinguishing the label-sparse and label-dense samples via the comparisons against the supervised average cross-entropy loss. We formulate such comparisons as a weighted augmentation consistency (WAC) regularization problem and propose *AdaWAC* for iterative and smooth separation of samples from different subpopulations with a convergence guarantee. Our experiments on various medical image segmentation tasks with different UNet-style architectures empirically demonstrate the effectiveness of *AdaWAC* not only in improving the segmentation performance and sample efficiency but also in enhancing the distributional robustness to concept shifts.

Limitations and future directions. From an algorithmic perspective, a limitation of this work is the utilization of the encoder layer outputs $\phi_\theta(\cdot)$ for data augmentation consistency regularization, which resulted in *AdaWAC* being specifically tailored to UNet-style backbones. However, our method can be generalized to other architectures in principle by selecting a representation extractor in the network that (i) well characterizes the marginal distribution of features $P(\mathbf{x})$ (ii) while being robust to the concept shift in $P(\mathbf{y}|\mathbf{x})$. Further investigation into such generalizations is a promising avenue for future research.

Meanwhile, noticing the prevalence of concept shifts in natural data, especially for dense prediction tasks like segmentation and detection, we hope to extend the application/idea of *AdaWAC* beyond medical image segmentation as a potential future direction.

Appendices

Appendix A

Appendix for Chapter 3

A.1 Technical Lemmas

Lemma A.1. *Let* $\mathbf{x} \in \mathbb{R}^d$ *be a random vector with* $\mathbb{E}[\mathbf{x}] = \mathbf{0}$, $\mathbb{E}[\mathbf{x}\mathbf{x}^\top] = \mathbf{\Sigma}$, *and* $\overline{\mathbf{x}} = \mathbf{\Sigma}^{-1/2}\mathbf{x}$ [1] *being* ρ^2*-subgaussian*[2]. *Given a set of* i.i.d. *samples of* $\mathbf{x}$, $\mathbf{X} = [\mathbf{x}_1; \dots ; \mathbf{x}_n]$, *and a set of weights corresponding to the samples,* $\{w_i > 0 \mid i \in [n]\}$, *let* $\mathbf{W} = \mathrm{diag}(w_1, \dots, w_n)$. *If* $n \geq \frac{\mathrm{tr}(\mathbf{W})^2}{\mathrm{tr}(\mathbf{W}^2)} \geq \frac{20736\rho^4 d}{\epsilon^2} + \frac{10368\rho^4 \log(1/\delta)}{\epsilon^2}$, *then with probability at least* $1 - \delta$,

$$(1-\epsilon)\,\mathrm{tr}\,(\mathbf{W})\,\mathbf{\Sigma} \preccurlyeq \mathbf{X}^\top \mathbf{W} \mathbf{X} \preccurlyeq (1+\epsilon)\,\mathrm{tr}\,(\mathbf{W})\,\mathbf{\Sigma}$$

Concretely, with $\mathbf{W} = \mathbf{I}_n$, $n = \frac{\mathrm{tr}(\mathbf{W})^2}{\mathrm{tr}(\mathbf{W}^2)} = \Omega\left(\rho^4 d\right)$, *and* $\epsilon = \Theta\left(\rho^2 \sqrt{\frac{d}{n}}\right)$, $(1-\epsilon)\mathbf{\Sigma} \preccurlyeq \frac{1}{n}\mathbf{X}^\top\mathbf{X} \preccurlyeq (1+\epsilon)\mathbf{\Sigma}$ *with high probability (at least* $1 - e^{-\Theta(d)}$*).*

Proof. We first denote $\mathbf{P}_{\mathcal{X}} \triangleq \mathbf{\Sigma}\mathbf{\Sigma}^\dagger$ as the orthogonal projector onto the subspace $\mathcal{X} \subseteq \mathbb{R}^d$ supported by the distribution of $\mathbf{x}$. With the assumptions $\mathbb{E}[\mathbf{x}] = \mathbf{0}$ and $\mathbb{E}[\mathbf{x}\mathbf{x}^\top] = \mathbf{\Sigma}$, we observe that $\mathbb{E}\left[\overline{\mathbf{x}}\right] = \mathbf{0}$ and $\mathbb{E}\left[\overline{\mathbf{x}}\overline{\mathbf{x}}^\top\right] = \mathbb{E}\left[\left(\mathbf{\Sigma}^{-1/2}\mathbf{x}\right)\left(\mathbf{\Sigma}^{-1/2}\mathbf{x}\right)^\top\right] = \mathbf{P}_{\mathcal{X}}$. Given the sample set $\mathbf{X}$ of size $n \gg \rho^4\left(d + \log(1/\delta)\right)$ for any $\delta \in (0,1)$, we let

$$\mathbf{U} = \frac{1}{\mathrm{tr}\,(\mathbf{W})} \sum_{i=1}^n w_i \left(\mathbf{\Sigma}^{-1/2}\mathbf{x}\right)\left(\mathbf{\Sigma}^{-1/2}\mathbf{x}\right)^\top - \mathbf{P}_{\mathcal{X}}.$$

Then the problem can be reduced to showing that, for any $\epsilon > 0$, with probability at least $1-\delta$, $\left\|\mathbf{U}\right\|_2 \leq \epsilon$. For this, we leverage the ϵ-net argument as follows.

For an arbitrary $\mathbf{v} \in \mathcal{X} \cap \mathbb{S}^{d-1}$, we have

$$\mathbf{v}^\top\mathbf{U}\mathbf{v} = \frac{1}{\mathrm{tr}\,(\mathbf{W})} \sum_{i=1}^n w_i \left(\mathbf{v}^\top\left(\mathbf{\Sigma}^{-1/2}\mathbf{x}\right)\left(\mathbf{\Sigma}^{-1/2}\mathbf{x}\right)^\top \mathbf{v} - 1\right) = \frac{1}{\mathrm{tr}\,(\mathbf{W})} \sum_{i=1}^n w_i \left(\left(\mathbf{v}^\top\overline{\mathbf{x}}_i\right)^2 - 1\right),$$

where, given $\overline{\mathbf{x}}_i$ being ρ^2-subgaussian, $\mathbf{v}^\top\overline{\mathbf{x}}_i$ is ρ^2-subgaussian. Since

$$\mathbb{E}\left[\left(\mathbf{v}^\top\overline{\mathbf{x}}_i\right)^2\right] = \mathbf{v}^\top\mathbb{E}\left[\overline{\mathbf{x}}_i\overline{\mathbf{x}}_i^\top\right]\mathbf{v} = 1,$$

[1] In the case where $\mathbf{\Sigma}$ is rank-deficient, we slightly abuse the notation such that $\mathbf{\Sigma}^{-1/2}$ and $\mathbf{\Sigma}^{-1}$ refer to the respective pseudo-inverses.

[2] A random vector $\mathbf{v} \in \mathbb{R}^d$ is ρ^2-subgaussian if for any unit vector $\mathbf{u} \in \mathbb{S}^{d-1}$, $\mathbf{u}^\top\mathbf{v}$ is ρ^2-subgaussian, $\mathbb{E}\left[\exp(s \cdot \mathbf{u}^\top\mathbf{v})\right] \leq \exp\left(s^2\rho^2/2\right)$ for all $s \in \mathbb{R}$.

we know that $\left(\mathbf{v}^\top \overline{\mathbf{x}}_i\right)^2 - 1$ is $16\rho^2$-subexponential[3]. With $\beta_i \triangleq \frac{w_i}{\operatorname{tr}(\mathbf{W})}$ for all $i \in [n]$ such that $\boldsymbol{\beta} = [\beta_1; \dots; \beta_n]$, we recall Bernstein's inequality [151, Theorem 2.8.2][154, Section 2.1.3],

$$\mathbb{P}\left[\left|\mathbf{v}^\top \mathbf{U} \mathbf{v}\right| = \left|\sum_{i=1}^n \beta_i \cdot \left(\left(\mathbf{v}^\top \overline{\mathbf{x}}_i\right)^2 - 1\right)\right| > t\right] \leq 2 \exp\left(-\frac{1}{2} \min\left(\frac{t^2}{\left(16\rho^2\right)^2 \|\boldsymbol{\beta}\|_2^2}, \frac{t}{16\rho^2 \|\boldsymbol{\beta}\|_\infty}\right)\right),$$

where $\|\boldsymbol{\beta}\|_2^2 = \frac{\operatorname{tr}\left(\mathbf{W}^2\right)}{\operatorname{tr}(\mathbf{W})^2}$ and $\|\boldsymbol{\beta}\|_\infty = \frac{\max_{i\in[n]} w_i}{\operatorname{tr}(\mathbf{W})}$.

Let $N \subset \mathcal{X} \cap \mathbb{S}^{d-1}$ be an ϵ_1-net such that $|N| = \left(1 + \frac{2}{\epsilon_1}\right)^d$. Then for some $0 < \epsilon_2 \leq 16\rho^2 \frac{\|\beta\|_2^2}{\|\beta\|_\infty}$, by the union bound,

$$\begin{aligned} \mathbb{P}\left[\max_{\mathbf{v}\in N} : \left|\mathbf{v}^\top \mathbf{U} \mathbf{v}\right| > \epsilon_2\right] &\leq 2\,|N| \exp\left(-\frac{1}{2} \min\left(\frac{\epsilon_2^2}{\left(16\rho^2\right)^2 \|\boldsymbol{\beta}\|_2^2}, \frac{\epsilon_2}{16\rho^2 \|\boldsymbol{\beta}\|_\infty}\right)\right) \\ &\leq \exp\left(d \log\left(1 + \frac{2}{\epsilon_1}\right) - \frac{1}{2} \cdot \frac{\epsilon_2^2}{\left(16\rho^2\right)^2 \|\boldsymbol{\beta}\|_2^2}\right) = \delta \end{aligned}$$

whenever $\frac{1}{\|\beta\|_2^2} = \frac{\operatorname{tr}(\mathbf{W})^2}{\operatorname{tr}(\mathbf{W}^2)} = 2\left(\frac{16\rho^2}{\epsilon_2}\right)^2 \left(d \log\left(1 + \frac{2}{\epsilon_1}\right) + \log \frac{1}{\delta}\right)$ where $1 < \frac{\operatorname{tr}(\mathbf{W})^2}{\operatorname{tr}(\mathbf{W}^2)} \leq n$.

Now for any $\mathbf{v} \in \mathcal{X} \cap \mathbb{S}^{d-1}$, there exists some $\mathbf{v}' \in N$ such that $\|\mathbf{v} - \mathbf{v}'\|_2 \leq \epsilon_1$. Therefore,

$$\begin{aligned} \left|\mathbf{v}^\top \mathbf{U} \mathbf{v}\right| &= \left|\mathbf{v}'^\top \mathbf{U} \mathbf{v}' + 2\mathbf{v}'^\top \mathbf{U}\left(\mathbf{v} - \mathbf{v}'\right) + \left(\mathbf{v} - \mathbf{v}'\right)^\top \mathbf{U}\left(\mathbf{v} - \mathbf{v}'\right)\right| \\ &\leq \left(\max_{\mathbf{v}\in N} : \left|\mathbf{v}^\top \mathbf{U} \mathbf{v}\right|\right) + 2\|\mathbf{U}\|_2 \|\mathbf{v}'\|_2 \|\mathbf{v} - \mathbf{v}'\|_2 + \|\mathbf{U}\|_2 \|\mathbf{v} - \mathbf{v}'\|_2^2 \\ &\leq \left(\max_{\mathbf{v}\in N} : \left|\mathbf{v}^\top \mathbf{U} \mathbf{v}\right|\right) + \|\mathbf{U}\|_2 \left(2\epsilon_1 + \epsilon_1^2\right). \end{aligned}$$

Taking the supremum over $\mathbf{v} \in \mathbb{S}^{d-1}$, with probability at least $1 - \delta$,

$$\max_{\mathbf{v}\in\mathcal{X}\cap\, \mathbb{S}^{d-1}} : \left|\mathbf{v}^\top \mathbf{U} \mathbf{v}\right| = \|\mathbf{U}\|_2 \leq \epsilon_2 + \|\mathbf{U}\|_2 \left(2\epsilon_1 + \epsilon_1^2\right), \qquad \|\mathbf{U}\|_2 \leq \frac{\epsilon_2}{2 - \left(1 + \epsilon_1\right)^2}.$$

With $\epsilon_1 = \frac{1}{3}$, we have $\epsilon = \frac{9}{2}\epsilon_2$.

Overall, if $n \geq \frac{\operatorname{tr}(\mathbf{W})^2}{\operatorname{tr}(\mathbf{W}^2)} \geq \frac{1024\rho^4 d}{\epsilon_2^2} + \frac{512\rho^4}{\epsilon_2^2} \log \frac{1}{\delta}$, then with probability at least $1 - \delta$, we have $\|\mathbf{U}\|_2 \leq \epsilon$.

As a concrete instance, when $\mathbf{W} = \mathbf{I}_n$ and $n = \frac{\operatorname{tr}(\mathbf{W})^2}{\operatorname{tr}(\mathbf{W}^2)} \geq 9^2 \cdot 1025 \cdot \rho^4 d$, by taking $\epsilon_2 = \sqrt{\frac{1025\rho^4 d}{n}}$, we have $\|\mathbf{U}\|_2 \leq \frac{1}{2}\sqrt{\frac{9^2 \cdot 1025 \cdot \rho^4 d}{n}}$ with high probability (at least $1 - \delta$ where $\delta = \exp\left(-\frac{d}{512}\right)$). ■

[3] We abbreviate (ν, ν)-subexponential (*i.e.*, recall that a random variable X is (ν, α)-subexponential if $\mathbb{E}\left[\exp\left(sX\right)\right] \leq \exp\left(s^2\nu^2/2\right)$ for all $|s| \leq 1/\alpha$) simply as ν-subexponential.

Lemma A.2 (Cauchy interlacing theorem)**.** *Given an arbitrary matrix* $\mathbf{A} \in \mathbb{C}^{m \times n}$ *and an orthogonal projection* $\mathbf{Q} \in \mathbb{C}^{n \times k}$ *with orthonormal columns, for all* $i = 1, \ldots, k$,

$$\sigma_i\left(\mathbf{AQ}\right) \leq \sigma_i\left(\mathbf{A}\right).$$

Proof of Lemma A.2. Let $\left\{\mathbf{v}_j \in \mathbb{C}^k \,\middle|\, j = 1, \ldots, k\right\}$ be the right singular vectors of $\mathbf{AQ}$. By the min-max theorem (*cf.* [59] Theorem 8.6.1),

$$\sigma_i\left(\mathbf{AQ}\right)^2 = \min_{\mathbf{x} \in \mathrm{span}\{\mathbf{v}_1, \ldots, \mathbf{v}_i\} \setminus (\mathbf{0})} \frac{\mathbf{x}^\top \mathbf{Q}^\top \mathbf{A}^\top \mathbf{A} \mathbf{Q} \mathbf{x}}{\mathbf{x}^\top \mathbf{x}} \leq \max_{\dim(\mathcal{V}) = i} \min_{\mathbf{x} \in \mathcal{V} \setminus (\mathbf{0})} \frac{\mathbf{x}^\top \mathbf{A}^\top \mathbf{A} \mathbf{x}}{\mathbf{x}^\top \mathbf{x}} = \sigma_i\left(\mathbf{A}\right)^2.$$

■

Lemma A.3 ([75] (7.3.14))**.** *For arbitrary matrices* $\mathbf{A}, \mathbf{B} \in \mathbb{C}^{m \times n}$,

$$\sigma_i\left(\mathbf{AB}^*\right) \leq \sigma_i\left(\mathbf{A}\right) \sigma_j\left(\mathbf{B}\right) \tag{A.1}$$

for all $i \in [\mathrm{rank}\left(\mathbf{A}\right)]$, $j \in [\mathrm{rank}\left(\mathbf{B}\right)]$ *such that* $i + j - 1 \in [\mathrm{rank}\left(\mathbf{AB}^*\right)]$.

Lemma A.4 ([[75] (7.3.13))**.** *For arbitrary matrices* $\mathbf{A}, \mathbf{B} \in \mathbb{C}^{m \times n}$,

$$\sigma_{i+j-1}\left(\mathbf{A} + \mathbf{B}\right) \leq \sigma_i\left(\mathbf{A}\right) + \sigma_j\left(\mathbf{B}\right) \tag{A.2}$$

for all $i \in [\mathrm{rank}\left(\mathbf{A}\right)]$, $j \in [\mathrm{rank}\left(\mathbf{B}\right)]$ *such that* $i + j - 1 \in [\mathrm{rank}\left(\mathbf{A} + \mathbf{B}\right)]$.

A.2 Supplementary Experiments

A.2.1 Upper and Lower Space-agnostic Bounds

In this section, we visualize and compare the upper and lower bounds in Theorem 3.1 under the sufficient multiplicative oversampling regime (*i.e.*, $l = 4k$. Recall that $k < l < r = \mathrm{rank}\left(\mathbf{A}\right)$ where k is the target rank, l is the oversampled rank, and r is the full rank of the matrix $\mathbf{A}$).

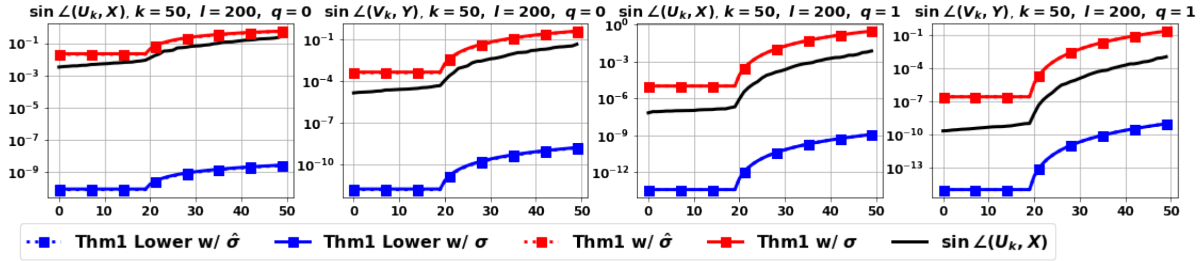


Figure A.1: Synthetic Gaussian with the slower spectral decay. $k = 50$, $l = 200$, $q = 0, 1$.

With the same set of target matrices described in Section 3.6.1, from Figure A.1 to Figure A.5,

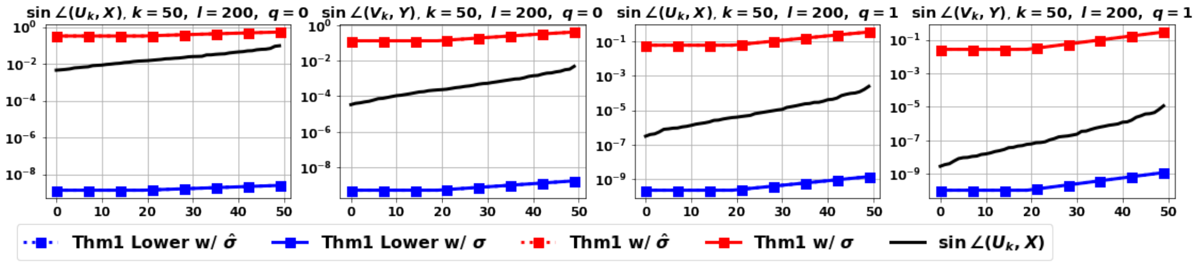


Figure A.2: Synthetic Gaussian with the faster spectral decay. $k = 50$, $l = 200$, $q = 0, 1$.

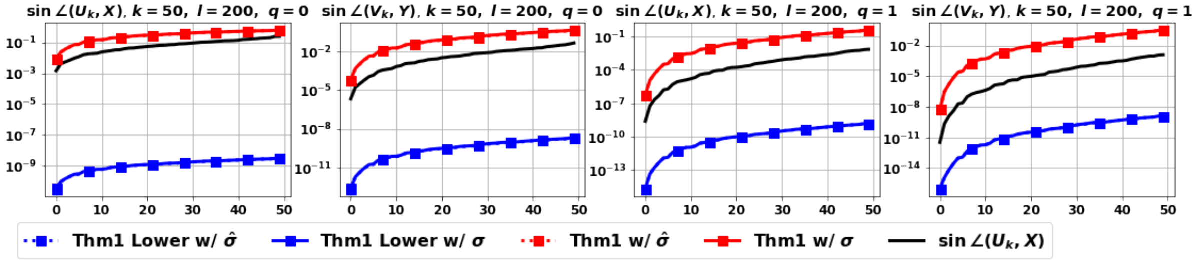


Figure A.3: SNN with $r_1 = 20$, $a = 1$. $k = 50$, $l = 200$, $q = 0, 1$.

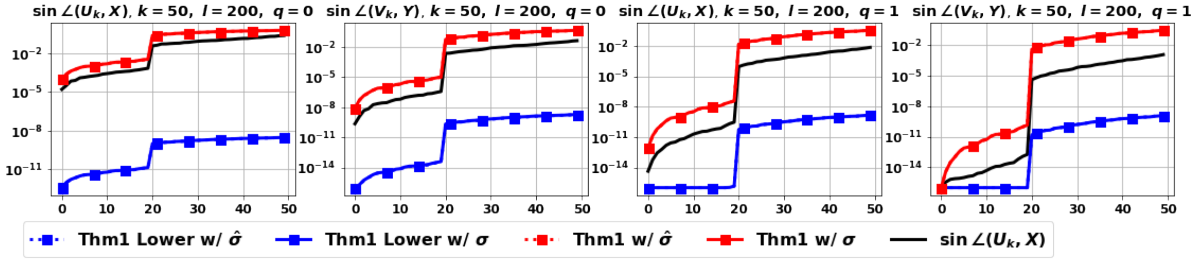


Figure A.4: SNN with $r_1 = 20$, $a = 100$. $k = 50$, $l = 200$, $q = 0, 1$.

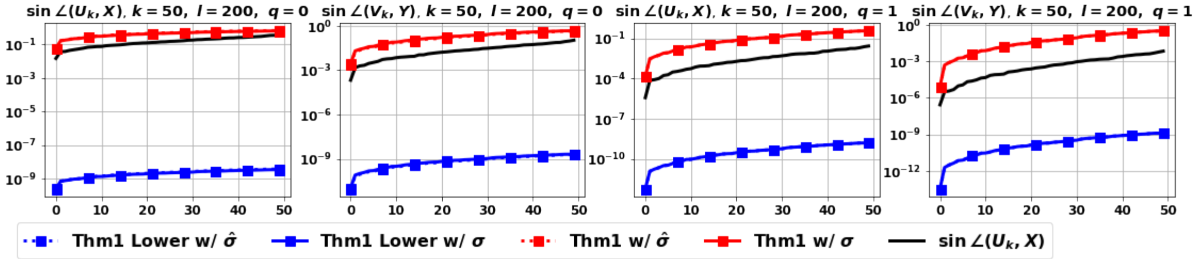


Figure A.5: 800 randomly sampled images from the MNIST training set. $k = 50$, $l = 200$, $q = 0, 1$.

1. Red lines and dashes represent the upper bounds in (3.2) and (3.2) evaluated with the true (lines) and approximated (dashes) singular values, Σ, and $\widetilde{\Sigma}$, respectively, where we

simply ignore tail decay and suppress constants for the distortion factors and set

$$\epsilon_1 = \sqrt{\frac{k}{l}} \quad \text{and} \quad \epsilon_2 = \sqrt{\frac{l}{r-k}}.$$

2. Blue lines and dashes present the lower bounds in (3.4) and (3.5) evaluated with $\boldsymbol{\Sigma}$ and $\widetilde{\boldsymbol{\Sigma}}$, respectively, and slightly larger constants associated with the distortion factors

$$\epsilon_1' = 2\sqrt{\frac{k}{l}} \quad \text{and} \quad \epsilon_2' = 2\sqrt{\frac{l}{r-k}}.$$

The numerical observations imply that the empirical validity of lower bounds requires more aggressive oversampling than that of upper bounds. In particular, we recall from Section 3.6.2 that $l \geq 1.6k$ is usually sufficient for the upper bounds to hold numerically. In contrast, the lower bounds generally require at least $l \geq 4k$, with slightly larger constants associated with the distortion factors $\epsilon_1 = \Theta\left(\sqrt{k/l}\right)$ and $\epsilon_2 = \Theta\left(\sqrt{l/\left(r-k\right)}\right)$.

Appendix B

Appendix for Chapter 4

B.1 Linear Regression Models

In this section, we present formal proofs for the results on linear regression in the fixed design where the training samples $(\mathbf{X}, \mathbf{y})$ and their augmentations $\widetilde{\mathcal{A}}\left(\mathbf{X}\right)$ are considered to be fixed. We discuss two types of augmentations: the label invariant augmentations in Section 4.4 and the misspecified augmentations in Section 4.5.

B.1.1 Linear Regression with Label Invariant Augmentations

For fixed $\widetilde{\mathcal{A}}(\mathbf{X})$, let $\boldsymbol{\Delta} \triangleq \widetilde{\mathcal{A}}(\mathbf{X}) - \widetilde{\mathbf{M}}\mathbf{X}$ in this section. We recall that $d_{aug} = \operatorname{rank}\left(\boldsymbol{\Delta}\right)$ since there is no randomness in $\widetilde{\mathcal{A}}, \mathbf{X}$ in fix design setting. Assuming that $\widetilde{\mathcal{A}}(\mathbf{X})$ admits full column rank, we have the following theorem on the excess risk of DAC and ERM:

Theorem B.1 (Formal restatement of Theorem 4.1 on linear regression.)**.** *Learning with DAC regularization, we have* $\mathbb{E}\left[L(\widehat{\boldsymbol{\theta}}^{dac}) - L(\boldsymbol{\theta}^*)\right] = \frac{(d-d_{aug})\sigma^2}{N}$*, while learning with ERM directly on the augmented dataset, we have* $\mathbb{E}\left[L(\widehat{\boldsymbol{\theta}}^{da-erm}) - L(\boldsymbol{\theta}^*)\right] = \frac{(d-d_{aug}+d')\sigma^2}{N}$*.* d' *is defined as*

$$d' \triangleq \frac{\operatorname{tr}\left(\widetilde{\mathbf{M}}^\top \left(\mathbf{P}_{\widetilde{\mathcal{A}}(\mathbf{X})} - \mathbf{P}_{\mathcal{S}}\right)\widetilde{\mathbf{M}}\right)}{1+\alpha},$$

where $d' \in [0, d_{aug}]$ *with* $\mathbf{P}_{\widetilde{\mathcal{A}}(\mathbf{X})} = \widetilde{\mathcal{A}}(\mathbf{X})\left(\widetilde{\mathcal{A}}(\mathbf{X})^\top \widetilde{\mathcal{A}}(\mathbf{X})\right)^{-1}\widetilde{\mathcal{A}}(\mathbf{X})^\top$ *and* $\mathbf{P}_{\mathcal{S}} \in \mathbb{R}^{(\alpha+1)N\times(\alpha+1)N}$ *is the orthogonal projector onto* $\mathcal{S} \triangleq \left\{\widetilde{\mathbf{M}}\mathbf{X}\boldsymbol{\theta} \mid \forall \boldsymbol{\theta} \in \mathbb{R}^d, s.t. \left(\widetilde{\mathcal{A}}(\mathbf{X}) - \widetilde{\mathbf{M}}\mathbf{X}\right)\boldsymbol{\theta} = \boldsymbol{0}\right\}$*.*

Proof. With $L(\boldsymbol{\theta}) \triangleq \frac{1}{(1+\alpha)N}\|\widetilde{\mathcal{A}}(\mathbf{X})\boldsymbol{\theta} - \widetilde{\mathcal{A}}(\mathbf{X})\boldsymbol{\theta}^*\|_2^2$, the excess risk of ERM on the augmented training set satisfies that:

$$\begin{aligned}
\mathbb{E}\left[L(\widehat{\boldsymbol{\theta}}^{da-erm})\right] &= \frac{1}{(1+\alpha)N}\mathbb{E}\left[\|\widetilde{\mathcal{A}}(\mathbf{X})\widehat{\boldsymbol{\theta}}^{da-erm} - \widetilde{\mathcal{A}}(\mathbf{X})\boldsymbol{\theta}^*\|_2^2\right] \\
&= \frac{1}{(1+\alpha)N}\mathbb{E}\left[\|\widetilde{\mathcal{A}}(\mathbf{X})(\widetilde{\mathcal{A}}(\mathbf{X})^\top\widetilde{\mathcal{A}}(\mathbf{X}))^{-1}\widetilde{\mathcal{A}}(\mathbf{X})^\top(\widetilde{\mathcal{A}}(\mathbf{X})\boldsymbol{\theta}^* + \widetilde{\mathbf{M}}\boldsymbol{\epsilon}) - \widetilde{\mathcal{A}}(\mathbf{X})\boldsymbol{\theta}^*\|_2^2\right] \\
&= \frac{1}{(1+\alpha)N}\mathbb{E}\left[\|\mathbf{P}_{\widetilde{\mathcal{A}}(\mathbf{X})}\widetilde{\mathcal{A}}(\mathbf{X})\boldsymbol{\theta}^* + \mathbf{P}_{\widetilde{\mathcal{A}}(\mathbf{X})}\widetilde{\mathbf{M}}\boldsymbol{\epsilon} - \widetilde{\mathcal{A}}(\mathbf{X})\boldsymbol{\theta}^*\|_2^2\right] \\
&= \frac{1}{(1+\alpha)N}\mathbb{E}\left[\|\mathbf{P}_{\widetilde{\mathcal{A}}(\mathbf{X})}\widetilde{\mathbf{M}}\boldsymbol{\epsilon}\|_2^2\right] \\
&= \frac{1}{(1+\alpha)N}\mathbb{E}\left[\operatorname{tr}(\boldsymbol{\epsilon}^\top\widetilde{\mathbf{M}}^\top\mathbf{P}_{\widetilde{\mathcal{A}}(\mathbf{X})}\widetilde{\mathbf{M}}\boldsymbol{\epsilon})\right] \\
&= \frac{\sigma^2}{(1+\alpha)N}\operatorname{tr}\left(\widetilde{\mathbf{M}}^\top\mathbf{P}_{\widetilde{\mathcal{A}}(\mathbf{X})}\widetilde{\mathbf{M}}\right).
\end{aligned}$$

Let $\mathcal{C}_{\widetilde{\mathcal{A}}(\mathbf{X})}$ and $\mathcal{C}_{\widetilde{\mathbf{M}}}$ denote the column space of $\widetilde{\mathcal{A}}(\mathbf{X})$ and $\widetilde{\mathbf{M}}$, respectively. Notice that $\mathcal{S}$ is a subspace of both $\mathcal{C}_{\widetilde{\mathcal{A}}(\mathbf{X})}$ and $\mathcal{C}_{\widetilde{\mathbf{M}}}$. Observing that $d_{aug} = \operatorname{rank}(\boldsymbol{\Delta}) = \operatorname{rank}(\mathbf{P}_{\mathcal{S}})$, we have

$$\begin{aligned}
\mathbb{E}\left[L(\widehat{\boldsymbol{\theta}}^{da-erm})\right] =& \frac{\sigma^2}{(1+\alpha)N}\operatorname{tr}(\widetilde{\mathbf{M}}^\top\mathbf{P}_{\widetilde{\mathcal{A}}(\mathbf{X})}\widetilde{\mathbf{M}}) \\
=& \frac{\sigma^2}{(1+\alpha)N}\operatorname{tr}(\widetilde{\mathbf{M}}^\top\mathbf{P}_{\mathcal{S}}\widetilde{\mathbf{M}}) + \frac{\sigma^2}{(1+\alpha)N}\operatorname{tr}(\widetilde{\mathbf{M}}^\top(\mathbf{P}_{\widetilde{\mathcal{A}}(\mathbf{X})} - \mathbf{P}_{\mathcal{S}})\widetilde{\mathbf{M}}) \\
=& \frac{\sigma^2}{(1+\alpha)N}\operatorname{tr}(\widetilde{\mathbf{M}}^\top\mathbf{P}_{\mathcal{S}}\widetilde{\mathbf{M}}) + \frac{\sigma^2}{N}\cdot\frac{\operatorname{tr}(\widetilde{\mathbf{M}}^\top(\mathbf{P}_{\widetilde{\mathcal{A}}(\mathbf{X})} - \mathbf{P}_{\mathcal{S}})\widetilde{\mathbf{M}})}{1+\alpha}
\end{aligned}$$

By the data augmentation consistency constraint, we are essentially solving the linear regression on the $(d - d_{aug})$-dimensional space $\{\boldsymbol{\theta} \mid \boldsymbol{\Delta}\boldsymbol{\theta} = 0\}$. The rest of proof is identical to standard regression analysis, with features first projected to $\mathcal{S}$:

$$\begin{aligned}
\mathbb{E}\left[L(\widehat{\boldsymbol{\theta}}^{dac})\right] &= \frac{1}{(1+\alpha)N}\mathbb{E}\left[\|\widetilde{\mathcal{A}}(\mathbf{X})\widehat{\boldsymbol{\theta}}^{dac} - \widetilde{\mathcal{A}}(\mathbf{X})\boldsymbol{\theta}^*\|_2^2\right] \\
&= \frac{1}{(1+\alpha)N}\mathbb{E}\left[\|\widetilde{\mathcal{A}}(\mathbf{X})(\widetilde{\mathcal{A}}(\mathbf{X})^\top\widetilde{\mathcal{A}}(\mathbf{X}))^{-1}\widetilde{\mathcal{A}}(\mathbf{X})^\top\mathbf{P}_{\mathcal{S}}(\widetilde{\mathcal{A}}(\mathbf{X})\boldsymbol{\theta}^* + \widetilde{\mathbf{M}}\boldsymbol{\epsilon}) - \widetilde{\mathcal{A}}(\mathbf{X})\boldsymbol{\theta}^*\|_2^2\right] \\
&= \frac{1}{(1+\alpha)N}\mathbb{E}\left[\|\mathbf{P}_{\widetilde{\mathcal{A}}(\mathbf{X})}\mathbf{P}_{\mathcal{S}}\widetilde{\mathcal{A}}(\mathbf{X})\boldsymbol{\theta}^* + \mathbf{P}_{\widetilde{\mathcal{A}}(\mathbf{X})}\mathbf{P}_{\mathcal{S}}\widetilde{\mathbf{M}}\boldsymbol{\epsilon} - \widetilde{\mathcal{A}}(\mathbf{X})\boldsymbol{\theta}^*\|_2^2\right] \\
&\quad \left(\text{since } \widetilde{\mathcal{A}}(\mathbf{X})\boldsymbol{\theta}^* \in \mathcal{S}, \text{ and } \mathbf{P}_{\widetilde{\mathcal{A}}(\mathbf{X})}\mathbf{P}_{\mathcal{S}} = \mathbf{P}_{\mathcal{S}} \text{ since } \mathcal{S} \subseteq \mathcal{C}_{\widetilde{\mathcal{A}}(\mathbf{X})}\right) \\
&= \frac{1}{(1+\alpha)N}\mathbb{E}\left[\|\mathbf{P}_{\mathcal{S}}\widetilde{\mathbf{M}}\boldsymbol{\epsilon}\|_2^2\right] \\
&= \frac{\sigma^2}{(1+\alpha)N}\operatorname{tr}(\widetilde{\mathbf{M}}^\top\mathbf{P}_{\mathcal{S}}\widetilde{\mathbf{M}}) \\
&= \frac{(d - d_{aug})\sigma^2}{N}.
\end{aligned}$$

■

B.1.2 Linear Regression Beyond Label Invariant Augmentations

Proof of Theorem 4.2. With $L(\boldsymbol{\theta}) \triangleq \frac{1}{N}\|\mathbf{X}\boldsymbol{\theta} - \mathbf{X}\boldsymbol{\theta}^*\|_2^2 = \|\boldsymbol{\theta} - \boldsymbol{\theta}^*\|_{\boldsymbol{\Sigma}_\mathbf{X}}^2$, we start by partitioning the excess risk into two parts – the variance from label noise and the bias from feature-label mismatch due to augmentations (*i.e.*, $\widetilde{\mathcal{A}}\left(\mathbf{X}\right)\boldsymbol{\theta}^* \neq \widetilde{\mathbf{M}}\mathbf{X}\boldsymbol{\theta}^*$):

$$\mathbb{E}_{\boldsymbol{\epsilon}}\left[L\left(\boldsymbol{\theta}\right) - L\left(\boldsymbol{\theta}^*\right)\right] = \mathbb{E}_{\boldsymbol{\epsilon}}\left[\|\boldsymbol{\theta} - \boldsymbol{\theta}^*\|_{\boldsymbol{\Sigma}_\mathbf{X}}^2\right] = \underbrace{\mathbb{E}_{\boldsymbol{\epsilon}}\left[\|\boldsymbol{\theta} - \mathbb{E}_{\boldsymbol{\epsilon}}\left[\boldsymbol{\theta}\right]\|_{\boldsymbol{\Sigma}_\mathbf{X}}^2\right]}_{\text{Variance}} + \underbrace{\|\mathbb{E}_{\boldsymbol{\epsilon}}\left[\boldsymbol{\theta}\right] - \boldsymbol{\theta}^*\|_{\boldsymbol{\Sigma}_\mathbf{X}}^2}_{\text{Bias}}.$$

First, we consider learning with DAC regularization with some finite $0 < \lambda < \infty$,

$$\widehat{\boldsymbol{\theta}}^{dac} = \underset{\boldsymbol{\theta}\in\mathbb{R}^d}{\operatorname{argmin}}\ \frac{1}{N}\|\mathbf{X}\boldsymbol{\theta} - \mathbf{y}\|_2^2 + \frac{\lambda}{\left(1+\alpha\right)N}\|\left(\widetilde{\mathcal{A}}\left(\mathbf{X}\right) - \widetilde{\mathbf{M}}\mathbf{X}\right)\boldsymbol{\theta}\|_2^2.$$

By setting the gradient of (4.4) with respect to $\boldsymbol{\theta}$ to $\mathbf{0}$, with $\mathbf{y} = \mathbf{X}\boldsymbol{\theta}^* + \boldsymbol{\epsilon}$, we have

$$\widehat{\boldsymbol{\theta}}^{dac} = \frac{1}{N}\left(\boldsymbol{\Sigma}_\mathbf{X} + \lambda\boldsymbol{\Sigma}_{\boldsymbol{\Delta}}\right)^\dagger \mathbf{X}^\top\left(\mathbf{X}\boldsymbol{\theta}^* + \boldsymbol{\epsilon}\right),$$

Then with $\mathbb{E}_{\boldsymbol{\epsilon}}\left[\widehat{\boldsymbol{\theta}}^{dac}\right] = \left(\boldsymbol{\Sigma}_\mathbf{X} + \lambda\boldsymbol{\Sigma}_{\boldsymbol{\Delta}}\right)^\dagger\boldsymbol{\Sigma}_\mathbf{X}\boldsymbol{\theta}^*$,

$$\text{Var} = \mathbb{E}_{\boldsymbol{\epsilon}}\left[\left\|\frac{1}{N}\left(\boldsymbol{\Sigma}_\mathbf{X} + \lambda\boldsymbol{\Sigma}_{\boldsymbol{\Delta}}\right)^\dagger\mathbf{X}^\top\boldsymbol{\epsilon}\right\|_{\boldsymbol{\Sigma}_\mathbf{X}}^2\right], \quad \text{Bias} = \left\|\left(\boldsymbol{\Sigma}_\mathbf{X} + \lambda\boldsymbol{\Sigma}_{\boldsymbol{\Delta}}\right)^\dagger\boldsymbol{\Sigma}_\mathbf{X}\boldsymbol{\theta}^* - \boldsymbol{\theta}^*\right\|_{\boldsymbol{\Sigma}_\mathbf{X}}^2.$$

For the variance term, we have

$$\begin{aligned}\text{Var} =& \frac{\sigma^2}{N}\operatorname{tr}\left(\left(\boldsymbol{\Sigma}_\mathbf{X} + \lambda\boldsymbol{\Sigma}_{\boldsymbol{\Delta}}\right)^\dagger\boldsymbol{\Sigma}_\mathbf{X}\left(\boldsymbol{\Sigma}_\mathbf{X} + \lambda\boldsymbol{\Sigma}_{\boldsymbol{\Delta}}\right)^\dagger\boldsymbol{\Sigma}_\mathbf{X}\right)\\ =& \frac{\sigma^2}{N}\operatorname{tr}\left(\left[\boldsymbol{\Sigma}_\mathbf{X}^{1/2}\left(\boldsymbol{\Sigma}_\mathbf{X} + \lambda\boldsymbol{\Sigma}_{\boldsymbol{\Delta}}\right)^\dagger\boldsymbol{\Sigma}_\mathbf{X}^{1/2}\right]^2\right)\\ =& \frac{\sigma^2}{N}\operatorname{tr}\left(\left(\mathbf{I}_d + \lambda\boldsymbol{\Sigma}_\mathbf{X}^{-1/2}\boldsymbol{\Sigma}_{\boldsymbol{\Delta}}\boldsymbol{\Sigma}_\mathbf{X}^{-1/2}\right)^{-2}\right)\end{aligned}$$

For the semi-positive definite matrix $\boldsymbol{\Sigma}_\mathbf{X}^{-1/2}\boldsymbol{\Sigma}_{\boldsymbol{\Delta}}\boldsymbol{\Sigma}_\mathbf{X}^{-1/2}$, we introduce the spectral decomposition:

$$\boldsymbol{\Sigma}_\mathbf{X}^{-1/2}\boldsymbol{\Sigma}_{\boldsymbol{\Delta}}\boldsymbol{\Sigma}_\mathbf{X}^{-1/2} = \underset{d\times d_{aug}}{\mathbf{Q}}\ \underset{d_{aug}\times d_{aug}}{\boldsymbol{\Gamma}}\ \mathbf{Q}^\top, \quad \boldsymbol{\Gamma} = \operatorname{diag}\left(\gamma_1, \dots, \gamma_{d_{aug}}\right),$$

where $\mathbf{Q}$ consists of orthonormal columns and $\gamma_1 \geq \dots \geq \gamma_{d_{aug}} > 0$. Then

$$\text{Var} = \frac{\sigma^2}{N}\operatorname{tr}\left(\left(\mathbf{I}_d - \mathbf{Q}\mathbf{Q}^\top\right) + \mathbf{Q}\left(\mathbf{I}_{d_{aug}} + \lambda\boldsymbol{\Gamma}\right)^{-2}\mathbf{Q}^\top\right) = \frac{\sigma^2\left(d - d_{aug}\right)}{N} + \frac{\sigma^2}{N}\sum_{i=1}^{d_{aug}}\frac{1}{\left(1+\lambda\gamma_i\right)^2}.$$

For the bias term, we observe that

$$\begin{aligned}\text{Bias} &= \left\|\left(\boldsymbol{\Sigma}_\mathbf{X} + \lambda\boldsymbol{\Sigma}_{\boldsymbol{\Delta}}\right)^\dagger\boldsymbol{\Sigma}_\mathbf{X}\boldsymbol{\theta}^* - \boldsymbol{\theta}^*\right\|_{\boldsymbol{\Sigma}_\mathbf{X}}^2\\ &= \left\|\left(\boldsymbol{\Sigma}_\mathbf{X} + \lambda\boldsymbol{\Sigma}_{\boldsymbol{\Delta}}\right)^\dagger\left(-\lambda\boldsymbol{\Sigma}_{\boldsymbol{\Delta}}\right)\boldsymbol{\theta}^*\right\|_{\boldsymbol{\Sigma}_\mathbf{X}}^2\\ &= \left\|\left(\mathbf{I}_d + \lambda\boldsymbol{\Sigma}_\mathbf{X}^{-\frac{1}{2}}\boldsymbol{\Sigma}_{\boldsymbol{\Delta}}\boldsymbol{\Sigma}_\mathbf{X}^{-\frac{1}{2}}\right)^{-1}\left(\lambda\boldsymbol{\Sigma}_\mathbf{X}^{-\frac{1}{2}}\boldsymbol{\Sigma}_{\boldsymbol{\Delta}}\boldsymbol{\Sigma}_\mathbf{X}^{-\frac{1}{2}}\right)\left(\boldsymbol{\Sigma}_\mathbf{X}^{1/2}\mathbf{P}_{\boldsymbol{\Delta}}\boldsymbol{\theta}^*\right)\right\|_2^2.\end{aligned}$$

Then with $\boldsymbol{\vartheta} \triangleq \boldsymbol{\Sigma}_{\mathbf{X}}^{1/2} \mathbf{P}_{\boldsymbol{\Delta}} \boldsymbol{\theta}^*$, we have

$$\text{Bias} = \sum_{i=1}^{d_{aug}} \vartheta_i^2 \left(\frac{\lambda \gamma_i}{1 + \lambda \gamma_i} \right)^2$$

To simply the optimization of regularization parameter λ, we leverage upper bounds of the variance and bias terms:

$$\begin{aligned}
&\text{Var} - \frac{\sigma^2 \left(d - d_{aug} \right)}{N} \leq \frac{\sigma^2}{N} \sum_{i=1}^{d_{aug}} \frac{1}{\left(1 + \lambda \gamma_i\right)^2} \leq \frac{\sigma^2}{2N\lambda} \sum_{i=1}^{d_{aug}} \frac{1}{\gamma_i} \leq \frac{\sigma^2}{2N\lambda} \operatorname{tr}\left(\boldsymbol{\Sigma}_{\mathbf{X}} \boldsymbol{\Sigma}_{\boldsymbol{\Delta}}^{\dagger} \right), \\
&\text{Bias} = \sum_{i=1}^{d_{aug}} \vartheta_i^2 \left(\frac{\lambda \gamma_i}{1 + \lambda \gamma_i} \right)^2 \leq \frac{\lambda}{2} \sum_{i=1}^{d_{aug}} \vartheta_i^2 \gamma_i = \frac{\lambda}{2} \left\| \mathbf{P}_{\boldsymbol{\Delta}} \boldsymbol{\theta}^* \right\|_{\boldsymbol{\Sigma}_{\boldsymbol{\Delta}}}^2 .
\end{aligned}$$

Then with $\lambda = \sqrt{\frac{\sigma^2 \operatorname{tr}\left(\boldsymbol{\Sigma}_{\mathbf{X}} \boldsymbol{\Sigma}_{\boldsymbol{\Delta}}^{\dagger} \right)}{N \left\| \mathbf{P}_{\boldsymbol{\Delta}} \boldsymbol{\theta}^* \right\|_{\boldsymbol{\Sigma}_{\boldsymbol{\Delta}}}^2}}$, we have the generalization bound for $\widehat{\boldsymbol{\theta}}^{dac}$ in Theorem 4.2.

Second, we consider learning with DA-ERM:

$$\widehat{\boldsymbol{\theta}}^{da-erm} = \underset{\boldsymbol{\theta} \in \mathbb{R}^d}{\operatorname{argmin}} \frac{1}{\left(1 + \alpha\right) N} \left\| \widetilde{\mathcal{A}} \left(\mathbf{X} \right) \boldsymbol{\theta} - \widetilde{\mathbf{M}} \mathbf{y} \right\|_2^2 .$$

With

$$\widehat{\boldsymbol{\theta}}^{da-erm} = \frac{1}{(1 + \alpha) N} \boldsymbol{\Sigma}_{\widetilde{\mathcal{A}}(\mathbf{X})}^{-1} \widetilde{\mathcal{A}} \left(\mathbf{X} \right)^{\top} \widetilde{\mathbf{M}} \left(\mathbf{X} \boldsymbol{\theta}^* + \boldsymbol{\epsilon} \right),$$

we again partition the excess risk into the variance and bias terms. For the variance term, with the assumptions $\boldsymbol{\Sigma}_{\widetilde{\mathcal{A}}(\mathbf{X})} \preccurlyeq c_X \boldsymbol{\Sigma}_{\mathbf{X}}$ and $\boldsymbol{\Sigma}_{\widetilde{\mathcal{A}}(\mathbf{X})} \preccurlyeq c_S \boldsymbol{\Sigma}_{\mathbf{S}}$, we have

$$\begin{aligned}
\text{Var} =& \mathbb{E}_{\boldsymbol{\epsilon}} \left[\left\| \frac{1}{(1 + \alpha) N} \boldsymbol{\Sigma}_{\widetilde{\mathcal{A}}(\mathbf{X})}^{-1} \widetilde{\mathcal{A}} \left(\mathbf{X} \right)^{\top} \widetilde{\mathbf{M}} \boldsymbol{\epsilon} \right\|_{\boldsymbol{\Sigma}_{\mathbf{X}}}^2 \right] \\
=& \mathbb{E}_{\boldsymbol{\epsilon}} \left[\left\| \frac{1}{N} \boldsymbol{\Sigma}_{\widetilde{\mathcal{A}}(\mathbf{X})}^{-1} \mathbf{S}^{\top} \boldsymbol{\epsilon} \right\|_{\boldsymbol{\Sigma}_{\mathbf{X}}}^2 \right] \\
=& \frac{\sigma^2}{N} \operatorname{tr}\left(\boldsymbol{\Sigma}_{\mathbf{X}} \boldsymbol{\Sigma}_{\widetilde{\mathcal{A}}(\mathbf{X})}^{-1} \boldsymbol{\Sigma}_{\mathbf{S}} \boldsymbol{\Sigma}_{\widetilde{\mathcal{A}}(\mathbf{X})}^{-1} \right) \\
\geq& \frac{\sigma^2}{N} \operatorname{tr}\left(\frac{1}{c_X c_S} \mathbf{I}_d \right) = \frac{\sigma^2 d}{N c_X c_S} .
\end{aligned}$$

Additionally, for the bias term, we have

$$\begin{aligned}
\text{Bias} =& \left\| \frac{1}{(1 + \alpha) N} \boldsymbol{\Sigma}_{\widetilde{\mathcal{A}}(\mathbf{X})}^{-1} \widetilde{\mathcal{A}} \left(\mathbf{X} \right)^{\top} \widetilde{\mathbf{M}} \mathbf{X} \boldsymbol{\theta}^* - \boldsymbol{\theta}^* \right\|_{\boldsymbol{\Sigma}_{\mathbf{X}}}^2 \\
=& \left\| \left(\widetilde{\mathcal{A}} \left(\mathbf{X} \right)^{\top} \widetilde{\mathcal{A}} \left(\mathbf{X} \right) \right)^{-1} \widetilde{\mathcal{A}} \left(\mathbf{X} \right)^{\top} \boldsymbol{\Delta} \left(\mathbf{P}_{\boldsymbol{\Delta}} \boldsymbol{\theta}^* \right) \right\|_{\boldsymbol{\Sigma}_{\mathbf{X}}}^2 \\
=& \left\| \widetilde{\mathcal{A}} \left(\mathbf{X} \right)^{\dagger} \boldsymbol{\Delta} \left(\mathbf{P}_{\boldsymbol{\Delta}} \boldsymbol{\theta}^* \right) \right\|_{\boldsymbol{\Sigma}_{\mathbf{X}}}^2 = \left\| \mathbf{P}_{\boldsymbol{\Delta}} \boldsymbol{\theta}^* \right\|_{\boldsymbol{\Sigma}_{\widetilde{\boldsymbol{\Delta}}}}^2 .
\end{aligned}$$

Combining the variance and bias leads to the generalization bound for $\widehat{\boldsymbol{\theta}}^{da-erm}$ in Theorem 4.2.

■

B.2 Two-layer Neural Network Regression

In the two-layer neural network regression setting with $\mathcal{X} = \mathbb{R}^d$ described in Section 4.6.1, let $\mathbf{X} \sim P^N(\mathbf{x})$ be a set of N *i.i.d.* samples drawn from the marginal distribution $P(\mathbf{x})$ that satifies the following.

Assumption 2 (Regularity of marginal distribution). Let $\mathbf{x} \sim P(\mathbf{x})$ be zero-mean $\mathbb{E}[\mathbf{x}] = \mathbf{0}$, with covairance matrix $\mathbb{E}[\mathbf{x}\mathbf{x}^\top] = \mathbf{\Sigma}_{\mathbf{x}} \succ 0$ whose eigenvalues are bounded by constant factors $\Omega(1) = \sigma_{\min}(\mathbf{\Sigma}_{\mathbf{x}}) \leq \sigma_{\max}(\mathbf{\Sigma}_{\mathbf{x}}) = O(1)$, such that $(\mathbf{\Sigma}_{\mathbf{x}}^{-1/2}\mathbf{x})$ is ρ^2-subgaussian [1].

For the sake of analysis, we isolate the augmented part in $\widetilde{\mathcal{A}}(\mathbf{X})$ and denote the set of these augmentations as

$$\mathcal{A}(\mathbf{X}) = [\mathbf{x}_{1,1}; \cdots; \mathbf{x}_{N,1}; \cdots; \mathbf{x}_{1,\alpha}; \cdots; \mathbf{x}_{N,\alpha}] \in \mathcal{X}^{\alpha N},$$

where for each sample $i \in [N]$, $\{\mathbf{x}_{i,j}\}_{j\in[\alpha]}$ is a set of α augmentations generated from $\mathbf{x}_i$, and $\mathbf{M} \in \mathbb{R}^{\alpha N \times N}$ is the vertical stack of α $N \times N$ identity matrices. Analogous to the notions with respect to $\widetilde{\mathcal{A}}(\mathbf{X})$ in the linear regression cases in Section B.1, in this section, we denote $\mathbf{\Delta} \triangleq \mathcal{A}(\mathbf{X}) - \mathbf{M}\mathbf{X}$ and quantify the augmentation strength as

$$d_{aug} \triangleq \operatorname{rank}(\mathbf{\Delta}) = \operatorname{rank}\left(\widetilde{\mathcal{A}}(\mathbf{X}) - \widetilde{\mathbf{M}}\mathbf{X}\right)$$

such that $0 \leq d_{aug} \leq \min(d, \alpha N)$ can be intuitively interpreted as the number of dimensions in the span of the unlabeled samples, $\operatorname{row}(\mathbf{X})$, perturbed by the augmentations.

Then, to learn the ground truth distribution $\mathbf{y} = h^*(\mathbf{X}) + \boldsymbol{\epsilon} = (\mathbf{X}\mathbf{B}^*)_+ \mathbf{w}^* + \boldsymbol{\epsilon}$ where $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I}_N)$, training with the DAC regularization can be formulated explicitly as

$$\begin{aligned}
\widehat{\mathbf{B}}^{dac}, \widehat{\mathbf{w}}^{dac} \;=\; & \underset{\mathbf{B}\in\mathbb{R}^{d\times q}, \mathbf{w}\in\mathbb{R}^q}{\operatorname{argmin}} \; \frac{1}{N}\|\mathbf{y} - (\mathbf{X}\mathbf{B})_+ \mathbf{w}\|_2^2 \\
& \text{s.t.} \quad \mathbf{B} = \left[\mathbf{b}_1 \dots \mathbf{b}_k \dots \mathbf{b}_q\right], \; \mathbf{b}_k \in \mathbb{S}^{d-1} \; \forall \, k \in [q], \quad \|\mathbf{w}\|_1 \leq C_w \\
& (\mathcal{A}(\mathbf{X})\mathbf{B})_+ = (\mathbf{M}\mathbf{X}\mathbf{B})_+ .
\end{aligned}$$

For the resulted minimizer $\widehat{h}^{dac}(\mathbf{x}) \triangleq (\mathbf{x}^\top \widehat{\mathbf{B}}^{dac})_+ \widehat{\mathbf{w}}^{dac}$, we have the following.

Theorem B.2 (Formal restatement of Theorem 4.3 on two-layer neural network with DAC)**.** *Under Assumption 2, we suppose* $\mathbf{X}$ *and* $\mathbf{\Delta}$ *satisfy that (a)* $\alpha N \geq 4 d_{aug}$*; and (b)* $\mathbf{\Delta}$ *admits an absolutely continuous distribution. Then conditioned on* $\mathbf{X}$ *and* $\mathbf{\Delta}$*, with* $L(h) =$

[1] A random vector $\mathbf{v} \in \mathbb{R}^d$ is ρ^2-subgaussian if for any unit vector $\mathbf{u} \in \mathbb{S}^{d-1}$, $\mathbf{u}^\top\mathbf{v}$ is ρ^2-subgaussian, $\mathbb{E}\left[\exp(s \cdot \mathbf{u}^\top \mathbf{v})\right] \leq \exp\left(s^2\rho^2/2\right)$.

$\frac{1}{N}\left\|h(\mathbf{X})-h^*(\mathbf{X})\right\|_2^2$ *and* $\frac{1}{N}\sum_{i=1}^{N}\left\|\mathbf{P}_{\mathbf{\Delta}}^{\perp}\mathbf{x}_i\right\|_2^2 \leq C_{\mathcal{N}}^2$ *for some* $C_{\mathcal{N}} > 0$*, for all* $\delta \in (0,1)$*, with probability at least* $1-\delta$ *(over* $\boldsymbol{\epsilon}$*),*

$$L\left(\widehat{h}^{dac}\right) - L\left(h^*\right) \lesssim \sigma C_w C_{\mathcal{N}} \left(\frac{1}{\sqrt{N}} + \sqrt{\frac{\log(1/\delta)}{N}}\right).$$

Moreover, to account for randomness in $\mathbf{X}$ and $\mathbf{\Delta}$, we introduce the following notion of augmentation strength.

Definition B.1 (Augmentation strength)**.** For any $\delta \in [0,1)$, let

$$d_{aug}(\delta) \triangleq \underset{d'}{\text{argmax}}\ \mathbb{P}_{\mathbf{\Delta}}\left[\text{rank}\left(\mathbf{\Delta}\right) < d'\right] \leq \delta.$$

Intuitively, the *augmentation strength* d_{aug} ensures that the feature subspace perturbed by the augmentations in $\mathcal{A}(\mathbf{X})$ has a minimum dimension $d_{aug}(\delta)$ with probability at least $1-\delta$. A larger $d_{aug}(\delta)$ corresponds to stronger augmentations. For instance, when $\mathcal{A}(\mathbf{X}) = \mathbf{MX}$ almost surely (*e.g.*, when the augmentations are identical copies of the original samples, corresponding to the weakest augmentation – no augmentations at all), we have $d_{aug}(\delta) = d_{aug} = 0$ for all $\delta < 1$. Whereas for randomly generated augmentations, d_{aug} is likely to be larger (i.e., with more dimensions being perturbed). For example in Example 3, for a given d_{aug}, with random augmentations $\mathcal{A}\left(\mathbf{X}\right) = \mathbf{X}'$ where $\mathbf{X}'_{ij} = \mathbf{X}_{ij} + \mathcal{N}\left(0, 0.1\right)$ for all $i \in [N]$, $d - d_{aug} + 1 \leq j \leq d$, we have $\text{rank}\left(\mathbf{\Delta}\right) = d_{aug}$ with probability 1. That is $d_{aug}(\delta) = d_{aug}$ for all $\delta \geq 0$.

Leveraging the notion of augmentation strength in Definition B.1, we show that the stronger augmentations lead to the better generalization by reducing $C_{\mathcal{N}}$ in Theorem B.2.

Corollary B.3. *When* $N \gg \rho^4 d$ *and* $\alpha N \geq d$*, for any* $\delta \in (0,1)$*, with probability at least* $1-\delta$ *(over* $\mathbf{X}$ *and* $\mathbf{\Delta}$*), we have* $C_{\mathcal{N}} \lesssim \sqrt{d - d_{aug}(\delta)}$.

To prove Theorem B.2, we start by showing that, with sufficient samples ($\alpha N \geq 4d_{aug}$), consistency of the first layer outputs over the samples implies consistency of those over the population.

Lemma B.4. *Under the assumptions in Theorem B.2, every size-*d_{aug} *subset of rows in* $\mathbf{\Delta} = \mathcal{A}(\mathbf{X}) - \mathbf{MX}$ *is linearly independent almost surely.*

Proof of Lemma B.4. Since $\alpha N > d_{aug}$, it is sufficient to show that a random matrix with an absolutely continuous distribution is totally invertible [2] almost surely.

[2] A matrix is totally invertible if all its square submatrices are invertible.

It is known that for any dimension $m \in \mathbb{N}$, an $m \times m$ square matrix $\mathbf{S}$ is singular if $\det(\mathbf{S}) = 0$ where entries of $\mathbf{S}$ lie within the roots of the polynomial equation specified by the determinant. Therefore, the set of all singular matrices in $\mathbb{R}^{m \times m}$ has Lebesgue measure zero,

$$\lambda\left(\left\{\mathbf{S} \in \mathbb{R}^{m \times m} \,\middle|\, \det(\mathbf{S}) = 0\right\}\right) = 0.$$

Then, for an absolutely continuous probability measure μ with respect to λ, we also have

$$\mathbb{P}_{\mu}\left[\mathbf{S} \in \mathbb{R}^{m \times m} \text{ is singular}\right] = \mu\left(\left\{\mathbf{S} \in \mathbb{R}^{m \times m} \,\middle|\, \det(\mathbf{S}) = 0\right\}\right) = 0.$$

Since a general matrix $\mathbf{R}$ contains only finite number of submatrices, when $\mathbf{R}$ is drawn from an absolutely continuous distribution, by the union bound, $\mathbb{P}\left[\mathbf{R} \text{ cotains a singular submatrix}\right] = 0$. That is, $\mathbf{R}$ is totally invertible almost surely. ■

Lemma B.5. *Under the assumptions in Theorem B.2, the hidden layer in the two-layer ReLU network learns* $\operatorname{Null}(\mathbf{\Delta})$*, the invariant subspace under data augmentations : with high probability,*

$$\left(\mathbf{x}^{\top}\widehat{\mathbf{B}}^{dac}\right)_{+} = \left(\mathbf{x}^{\top}\mathbf{P}_{\mathbf{\Delta}}^{\perp}\widehat{\mathbf{B}}^{dac}\right)_{+} \quad \forall\, \mathbf{x} \in \mathcal{X}.$$

Proof of Lemma B.5. We will show that for all $\mathbf{b}_k = \mathbf{P}_{\mathbf{\Delta}}^{\perp}\mathbf{b}_k + \mathbf{P}_{\mathbf{\Delta}}\mathbf{b}_k$, $k \in [q]$, $\mathbf{P}_{\mathbf{\Delta}}\mathbf{b}_k = \mathbf{0}$ with high probability, which then implies that given any $\mathbf{x} \in \mathcal{X}$, $(\mathbf{x}^{\top}\mathbf{b}_k)_+ = (\mathbf{x}^{\top}\mathbf{P}_{\mathbf{\Delta}}^{\perp}\mathbf{b}_k)_+$ for all $k \in [q]$.

For any $k \in [q]$ associated with an arbitrary fixed $\mathbf{b}_k \in \mathbb{S}^{d-1}$, let $\mathbf{X}_k \triangleq \mathbf{X}_k\mathbf{P}_{\mathbf{\Delta}}^{\perp} + \mathbf{X}_k\mathbf{P}_{\mathbf{\Delta}} \in \mathcal{X}^{N_k}$ be the inclusion-wisely maximum row subset of $\mathbf{X}$ such that $\mathbf{X}_k\mathbf{b}_k > \mathbf{0}$ element-wisely. Meanwhile, we denote $\mathcal{A}(\mathbf{X}_k) = \mathbf{M}_k\mathbf{X}_k\mathbf{P}_{\mathbf{\Delta}}^{\perp} + \mathcal{A}(\mathbf{X}_k)\mathbf{P}_{\mathbf{\Delta}} \in \mathcal{X}^{\alpha N_k}$ as the augmentation of $\mathbf{X}_k$ where $\mathbf{M}_k \in \mathbb{R}^{\alpha N_k \times N_k}$ is the vertical stack of α identity matrices with size $N_k \times N_k$. Then the DAC constraint implies that $(\mathcal{A}(\mathbf{X}_k) - \mathbf{M}_k\mathbf{X}_k)\mathbf{P}_{\mathbf{\Delta}}\mathbf{b}_k = \mathbf{0}$.

With Assumption 2, for a fixed $\mathbf{b}_k \in \mathbb{S}^{d-1}$, $\mathbb{P}[\mathbf{x}^{\top}\mathbf{b}_k > 0] = \frac{1}{2}$. Then, with the Chernoff bound,

$$\mathbb{P}\left[N_k < \frac{N}{2} - t\right] \leq e^{-\frac{2t^2}{N}},$$

which implies that, $N_k \geq \frac{N}{4}$ with high probability.

Leveraging the assumptions in Theorem B.2, $\alpha N \geq 4d_{aug}$ implies that $\alpha N_k \geq d_{aug}$. Therefore by Lemma B.4, $\operatorname{row}\left(\mathcal{A}(\mathbf{X}_k) - \mathbf{M}_k\mathbf{X}_k\right) = \operatorname{row}\left(\mathbf{\Delta}\right)$ with probability 1. Thus, $(\mathcal{A}(\mathbf{X}_k) - \mathbf{M}_k\mathbf{X}_k)\mathbf{P}_{\mathbf{\Delta}}\mathbf{b}_k = \mathbf{0}$ enforces that $\mathbf{P}_{\mathbf{\Delta}}\mathbf{b}_k = \mathbf{0}$. ■

Proof of Theorem B.2. Conditioned on $\mathbf{X}$ and $\boldsymbol{\Delta}$, we are interested in the excess risk $L(\widehat{h}^{dac}) - L(h^*) = \frac{1}{N}\|(\mathbf{X}\widehat{\mathbf{B}}^{dac})_+\widehat{\mathbf{w}}^{dac} - (\mathbf{X}\mathbf{B}^*)_+\mathbf{w}^*\|_2^2$ with randomness on $\boldsymbol{\epsilon}$.

We first recall that Lemma B.5 implies $\widehat{h}^{dac} \in \mathcal{H}_{dac} = \left\{h(\mathbf{x}) = \left(\mathbf{x}^\top \mathbf{B}\right)_+ \mathbf{w} \;\middle|\; \mathbf{B} \in \mathcal{B},\ \|\mathbf{w}\|_1 \leq C_w\right\}$ where

$$\mathcal{B} \triangleq \left\{\mathbf{B} = [\mathbf{b}_1 \dots \mathbf{b}_q] \mid \|\mathbf{b}_k\| = 1\ \forall\, k \in [q], (\mathbf{X}\mathbf{B})_+ = (\mathbf{X}\mathbf{P}_{\boldsymbol{\Delta}}^\perp \mathbf{B})_+\right\}.$$

Leveraging Equation (21) and (22) in [52], since $(\mathbf{B}^*, \mathbf{w}^*)$ is feasible under the constraint, by the basic inequality,

$$\|\mathbf{y} - (\mathbf{X}\widehat{\mathbf{B}}^{dac})_+\widehat{\mathbf{w}}^{dac}\|_2^2 \leq \|\mathbf{y} - (\mathbf{X}\mathbf{B}^*)_+\mathbf{w}^*\|_2^2. \tag{B.1}$$

Knowing that $\mathbf{y} = (\mathbf{X}\mathbf{B}^*)_+\mathbf{w}^* + \boldsymbol{\epsilon}$ with $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I}_N)$, we can rewrite (B.1) as

$$\begin{aligned}\frac{1}{N}\|(\mathbf{X}\widehat{\mathbf{B}}^{dac})_+\widehat{\mathbf{w}}^{dac} - (\mathbf{X}\mathbf{B}^*)_+\mathbf{w}^*\|_2^2 \leq & \frac{2}{N}\boldsymbol{\epsilon}^\top \left((\mathbf{X}\widehat{\mathbf{B}}^{dac})_+\widehat{\mathbf{w}}^{dac} - (\mathbf{X}\mathbf{B}^*)_+\mathbf{w}^*\right) \\ \leq & 4 \sup_{h \in \mathcal{H}_{dac}} \frac{1}{N}\boldsymbol{\epsilon}^\top h(\mathbf{X})\end{aligned}$$

First, we observe that $\sigma^{-1}\mathbb{E}_{\boldsymbol{\epsilon}}\left[\sup_{h\in\mathcal{H}_{dac}} \frac{1}{N}\boldsymbol{\epsilon}^\top h(\mathbf{X})\right] = \widehat{\mathfrak{G}}_{\mathbf{X}}\left(\mathcal{H}_{dac}\right)$ measures the empirical Gaussian width of $\mathcal{H}_{dac}$ over $\mathbf{X}$. Moreover, by observing that for any $h \in \mathcal{H}_{dac}$ and $\mathbf{x}_i \in \mathbf{X}$,

$$\begin{aligned}&|h(\mathbf{x}_i)| \leq \left\|\left(\mathbf{B}^\top \mathbf{x}_i\right)_+\right\|_\infty \|\mathbf{w}\|_1 \leq \max_{k\in[q]} \left|\mathbf{b}_k^\top \mathbf{P}_{\boldsymbol{\Delta}}^\perp \mathbf{x}_i\right| \|\mathbf{w}\|_1 \leq \left\|\mathbf{P}_{\boldsymbol{\Delta}}^\perp \mathbf{x}_i\right\|_2 \|\mathbf{w}\|_1, \\ &\frac{1}{N}\|h(\mathbf{X})\|_2^2 = \frac{1}{N}\sum_{i=1}^N |h(\mathbf{x}_i)|^2 \leq \|\mathbf{w}\|_1^2 \cdot \frac{1}{N}\sum_{i=1}^N \left\|\mathbf{P}_{\boldsymbol{\Delta}}^\perp \mathbf{x}_i\right\|_2^2 \leq C_w^2 C_{\mathcal{N}}^2\end{aligned}$$

and

$$\begin{aligned}&\left|\sup_{h\in\mathcal{H}_{dac}} \frac{1}{N}\boldsymbol{\epsilon}_1^\top h(\mathbf{X}) - \sup_{h\in\mathcal{H}_{dac}} \frac{1}{N}\boldsymbol{\epsilon}_2^\top h(\mathbf{X})\right| \\ \leq & \left|\sup_{h\in\mathcal{H}_{dac}} \frac{1}{N} h(\mathbf{X})^\top (\boldsymbol{\epsilon}_1 - \boldsymbol{\epsilon}_2)\right| \\ \leq & \frac{1}{\sqrt{N}}\left\|\frac{1}{\sqrt{N}} h(\mathbf{X})\right\|_2 \|\boldsymbol{\epsilon}_1 - \boldsymbol{\epsilon}_2\|_2 \\ \leq & \frac{C_w C_{\mathcal{N}}}{\sqrt{N}} \|\boldsymbol{\epsilon}_1 - \boldsymbol{\epsilon}_2\|_2,\end{aligned}$$

we know that the function $\boldsymbol{\epsilon} \to \sup_{h\in\mathcal{H}_{dac}} \frac{1}{N}\boldsymbol{\epsilon}^\top h(\mathbf{X})$ is $\frac{C_{\mathcal{N}} C_w}{\sqrt{N}}$-Lipschitz in ℓ_2 norm. Therefore, by [155] Theorem 2.26, we have that with probability at least $1 - \delta$,

$$\sup_{h\in\mathcal{H}_{dac}} \frac{1}{N}\boldsymbol{\epsilon}^\top h(\mathbf{X}) \leq \sigma \cdot \left(\widehat{\mathfrak{G}}_{\mathbf{X}}\left(\mathcal{H}_{dac}\right) + C_w C_{\mathcal{N}} \sqrt{\frac{2\log(1/\delta)}{N}}\right),$$

where the empirical Gaussian complexity is upper bounded by

$$
\begin{aligned}
\widehat{\mathfrak{G}}_{\mathbf{X}}\left(\mathcal{H}_{dac}\right) =& \mathop{\mathbb{E}}_{\mathbf{g}\sim\mathcal{N}(\mathbf{0},\mathbf{I}_N)}\left[\sup_{\mathbf{B}\in\mathcal{B},\|\mathbf{w}\|_1\leq R}\frac{1}{N}\mathbf{g}^\top(\mathbf{XB})_+\mathbf{w}\right]\\
\leq&\frac{C_w}{N}\mathop{\mathbb{E}}_{\mathbf{g}}\left[\sup_{\mathbf{B}\in\mathcal{B}}\|(\mathbf{XB})_+^\top\mathbf{g}\|_\infty\right]\\
=&\frac{C_w}{N}\mathop{\mathbb{E}}_{\mathbf{g}}\left[\sup_{\mathbf{b}\in\mathbb{S}^{d-1}}\mathbf{g}^\top\left(\mathbf{X}\mathbf{P}_{\mathbf{\Delta}}^\perp\mathbf{b}\right)_+\right] \qquad \text{(Lemma B.13, $(\cdot)_+$ is 1-Lipschitz)}\\
\leq&\frac{C_w}{N}\mathop{\mathbb{E}}_{\mathbf{g}}\left[\sup_{\mathbf{b}\in\mathbb{S}^{d-1}}\mathbf{g}^\top\mathbf{X}\mathbf{P}_{\mathbf{\Delta}}^\perp\mathbf{b}\right]\\
=&\frac{C_w}{N}\mathop{\mathbb{E}}_{\mathbf{g}}\left[\|\mathbf{P}_{\mathbf{\Delta}}^\perp\mathbf{X}^\top\mathbf{g}\|_2\right]\\
\leq&\frac{C_w}{N}\left(\mathop{\mathbb{E}}_{\mathbf{g}}\left[\|\mathbf{P}_{\mathbf{\Delta}}^\perp\mathbf{X}^\top\mathbf{g}\|_2^2\right]\right)^{1/2}\\
=&\frac{C_w}{N}\sqrt{\mathrm{tr}(\mathbf{P}_{\mathbf{\Delta}}^\perp\mathbf{X}^\top\mathbf{X}\mathbf{P}_{\mathbf{\Delta}}^\perp)}\\
=&\frac{C_wC_{\mathcal{N}}}{\sqrt{N}}.
\end{aligned}
$$

■

Proof of Corollary B.3. By Definition B.1, we have with probability at least $1-\delta$ that $d_{aug}=\mathrm{rank}(\mathbf{P}_{\mathbf{\Delta}})\geq d_{aug}(\delta)$ and $\mathrm{rank}(\mathbf{P}_{\mathbf{\Delta}}^\perp)\leq d-d_{aug}(\delta)$. Meanwhile, leveraging Lemma B.12, we have that under Assumption 2 and with $N\gg\rho^4 d$, with high probability,

$$
\|\frac{1}{N}\mathbf{P}_{\mathbf{\Delta}}^\perp\mathbf{X}^\top\mathbf{X}\mathbf{P}_{\mathbf{\Delta}}^\perp\|_2\leq\|\frac{1}{N}\mathbf{X}^\top\mathbf{X}\|_2\leq 1.1C\lesssim 1.
$$

Therefore, there exists $C_{\mathcal{N}}>0$ with $\frac{1}{N}\sum_{i=1}^n\|\mathbf{P}_{\mathbf{\Delta}}^\perp\mathbf{x}_i\|_2^2\leq C_{\mathcal{N}}^2$ such that, with probability at least $1-\delta$,

$$
C_{\mathcal{N}}^2\leq(d-d_{aug})\cdot\|\frac{1}{N}\mathbf{P}_{\mathbf{\Delta}}^\perp\mathbf{X}^\top\mathbf{X}\mathbf{P}_{\mathbf{\Delta}}^\perp\|_2\lesssim d-d_{aug}(\delta).
$$

■

B.3 Classification with Expansion-based Augmentations

We first recall the multi-class classification problem setup in Section 4.6.2, while introducing some helpful notions. For an arbitrary set $\mathcal{X}$, let $\mathcal{Y}=[K]$, and $h^*:\mathcal{X}\to[K]$ be the ground truth classifier that partitions $\mathcal{X}$: for each $k\in[K]$, let $\mathcal{X}_k\triangleq\{\mathbf{x}\in\mathcal{X}\mid h^*(\mathbf{x})=k\}$, with $\mathcal{X}_i\cap\mathcal{X}_j=\emptyset,\forall i\neq j$. In addition, for an arbitrary classifier $h:\mathcal{X}\to[K]$, we denote the majority label with respect to h for each class,

$$\widehat{y}_k \triangleq \underset{y \in [K]}{\mathrm{argmax}}\ \mathbb{P}_P \left[h(\mathbf{x}) = y \,\middle|\, \mathbf{x} \in \mathcal{X}_k \right] \quad \forall\, k \in [K],$$

along with the respective class-wise local and global minority sets,

$$M_k \triangleq \left\{ \mathbf{x} \in \mathcal{X}_k \,\middle|\, h(\mathbf{x}) \neq \widehat{y}_k \right\} \subsetneq \mathcal{X}_k \quad \forall\, k \in [K], \quad M \triangleq \bigcup_{k=1}^{K} M_k.$$

Given the marginal distribution $P(\mathbf{x})$, we introduce the *expansion-based data augmentations* that concretizes Definition 4.1 in the classification setting:

Definition B.2 (Expansion-based data augmentations, [22]). We call $\mathcal{A} : \mathcal{X} \to 2^{\mathcal{X}}$ an augmentation function that induces expansion-based data augmentations if $\mathcal{A}$ is class invariant: $\{\mathbf{x}\} \subsetneq \mathcal{A}(\mathbf{x}) \subseteq \{\mathbf{x}' \in \mathcal{X} \mid h^*(\mathbf{x}) = h^*(\mathbf{x}')\}$ for all $\mathbf{x} \in \mathcal{X}$. Let

$$NB(\mathbf{x}) \triangleq \left\{ \mathbf{x}' \in \mathcal{X} \,\middle|\, \mathcal{A}(\mathbf{x}) \cap \mathcal{A}(\mathbf{x}') \neq \emptyset \right\}, \quad NB(S) \triangleq \cup_{\mathbf{x} \in S} NB(\mathbf{x})$$

be the neighborhoods of $\mathbf{x} \in \mathcal{X}$ and $S \subseteq \mathcal{X}$ with respect to $\mathcal{A}$. Then, $\mathcal{A}$ satisfies

(a) $\underline{(q, \xi)\text{-constant expansion}}$ if given any $S \subseteq \mathcal{X}$ with $P(S) \geq q$ and $P(S \cap \mathcal{X}_k) \leq \frac{1}{2}$ for all $k \in [K]$, $P(NB(S)) \geq \min\{P(S), \xi\} + P(S)$;

(b) $\underline{(a, c)\text{-multiplicative expansion}}$ if for all $k \in [K]$, given any $S \subseteq \mathcal{X}$ with $P(S \cap \mathcal{X}_k) \leq a$, $P(NB(S) \cap \mathcal{X}_k) \geq \min\{c \cdot P(S \cap \mathcal{X}_k), 1\}$.

On Definition B.2, we first point out that the ground truth classifier is invariant throughout the neighborhood: given any $\mathbf{x} \in \mathcal{X}$, $h^*(\mathbf{x}) = h^*(\mathbf{x}')$ for all $\mathbf{x}' \in NB(\mathbf{x})$. Second, in contrast to the linear regression and two-layer neural network cases where we assume $\mathcal{X} \subseteq R^d$, with the expansion-based data augmentation over a general $\mathcal{X}$, the notion of d_{aug} in Definition B.1 is not well-established. Alternatively, we leverage the concept of constant / multiplicative expansion from [22], and quantify the augmentation strength with parameters (q, ξ) or (a, c). Intuitively, the strength of expansion-based data augmentations is characterized by expansion capability of $\mathcal{A}$: for a neighborhood $S \subseteq \mathcal{X}$ of proper size (characterized by q or a under measure P), the stronger augmentation $\mathcal{A}$ leads to more expansion in $NB(S)$, and therefore larger ξ or c. For example in Definition 4.2, we use an expansion-based augmentation function $\mathcal{A}$ that satisfies $\left(\frac{1}{2}, c\right)$-multiplicative expansion.

Adapting the existing setting in [22, 162], we concretize the classifier class $\mathcal{H}$ with a function class $\mathcal{F} \subseteq \left\{ f : \mathcal{X} \to \mathbb{R}^K \right\}$ of fully connected neural networks such that $\mathcal{H} =$

$\left\{h(\mathbf{x}) \triangleq \mathrm{argmax}_{k\in[K]}\ f(\mathbf{x})_k \ \middle|\ f \in \mathcal{F}\right\}$. To constrain the feasible hypothesis class through the DAC regularization with finite unlabeled samples, we recall the notion of all-layer-margin, $m : \mathcal{F} \times \mathcal{X} \times \mathcal{Y} \to \mathbb{R}_{\geq 0}$ (from [162]) that measures the maximum possible perturbation in all layers of f while maintaining the prediction y. Precisely, given any $f \in \mathcal{F}$ such that $f(\mathbf{x}) = \mathbf{W}_p \varphi(\ldots \varphi(\mathbf{W}_1 \mathbf{x}) \ldots)$ for some activation function $\varphi : \mathbb{R} \to \mathbb{R}$ and parameters $\left\{\mathbf{W}_\iota \in \mathbb{R}^{d_\iota \times d_{\iota-1}}\right\}_{\iota=1}^{p}$, we can write $f = f_{2p-1} \circ \cdots \circ f_1$ where $f_{2\iota-1}(\mathbf{x}) = \mathbf{W}_\iota \mathbf{x}$ for all $\iota \in [p]$ and $f_{2\iota}(\mathbf{z}) = \varphi(\mathbf{z})$ for $\iota \in [p-1]$. For an arbitrary set of perturbation vectors $\boldsymbol{\delta} = (\boldsymbol{\delta}_1, \ldots, \boldsymbol{\delta}_{2p-1})$ such that $\boldsymbol{\delta}_{2\iota-1}, \boldsymbol{\delta}_{2\iota} \in \mathbb{R}^{d_\iota}$ for all ι, let $f(\mathbf{x}, \boldsymbol{\delta})$ be the perturbed neural network defined recursively such that

$$\begin{aligned}
&\widetilde{\mathbf{z}}_1 = f_1(\mathbf{x}) + \|\mathbf{x}\|_2 \boldsymbol{\delta}_1, \\
&\widetilde{\mathbf{z}}_\iota = f_\iota(\widetilde{\mathbf{z}}_{\iota-1}) + \|\widetilde{\mathbf{z}}_{\iota-1}\|_2 \boldsymbol{\delta}_\iota \quad \forall\, \iota = 2, \ldots, 2p-1, \\
&f(\mathbf{x}, \boldsymbol{\delta}) = \widetilde{\mathbf{z}}_{2p-1}.
\end{aligned}$$

The all-layer-margin $m(f, \mathbf{x}, y)$ measures the minimum norm of the perturbation $\boldsymbol{\delta}$ such that $f(\mathbf{x}, \boldsymbol{\delta})$ fails to provide the classification y,

$$m(f, \mathbf{x}, y) \triangleq \min_{\boldsymbol{\delta}=(\boldsymbol{\delta}_1,\ldots,\boldsymbol{\delta}_{2p-1})} \sqrt{\sum_{\iota=1}^{2p-1} \|\boldsymbol{\delta}_\iota\|_2^2} \quad \text{s.t.} \quad \underset{k\in[K]}{\mathrm{argmax}}\ f(\mathbf{x}, \boldsymbol{\delta})_k \neq y. \tag{B.2}$$

With the notion of all-layer-margin established, for any $\mathcal{A} : \mathcal{X} \to 2^{\mathcal{X}}$ that satisfies conditions in Definition B.2, the robust margin is defined as

$$m_{\mathcal{A}}(f, \mathbf{x}) \triangleq \sup_{\mathbf{x}' \in \mathcal{A}(\mathbf{x})} m\left(f, \mathbf{x}', \underset{k\in[K]}{\mathrm{argmax}}\ f(\mathbf{x})_k\right).$$

Intuitively, the robust margin measures the maximum possible perturbation in all-layer weights of f such that predictions on all data augmentations of $\mathbf{x}$ remain consistent. For instance, $m_{\mathcal{A}}(f, \mathbf{x}) > 0$ is equivalent to enforcing $h(\mathbf{x}) = h(\mathbf{x}')$ for all $\mathbf{x}' \in \mathcal{A}(\mathbf{x})$.

To achieve finite sample guarantees, DAC regularization requires stronger consistency conditions than merely consistent classification outputs (*i.e.*, $m_{\mathcal{A}}(f, \mathbf{x}) > 0$). Instead, we enforce $m_{\mathcal{A}}(f, \mathbf{x}) > \tau$ for any $0 < \tau < \max_{f\in\mathcal{F}} \inf_{\mathbf{x}\in\mathcal{X}} m_{\mathcal{A}}(f, \mathbf{x})$[3] over an finite set of unlabeled samples $\mathbf{X}^u$ that is independent of $\mathbf{X}$ and drawn *i.i.d.* from $P(\mathbf{x})$. Then, learning the classifier

[3] The upper bound on τ ensures the proper learning setting, *i.e.*, there exists $f \in \mathcal{F}$ such that $m_{\mathcal{A}}(f, \mathbf{x}) > \tau$ for all $\mathbf{x} \in \mathcal{X}$.

with zero-one loss $l_{01}\left(h(\mathbf{x}), y\right) = \mathbf{1}\left\{h(\mathbf{x}) \neq y\right\}$ from a class of p-layer fully connected neural networks with maximum width q,

$$\mathcal{F} = \left\{ f : \mathcal{X} \to \mathbb{R}^K \;\middle|\; f = f_{2p-1} \circ \cdots \circ f_1,\; f_{2\iota-1}(\mathbf{x}) = \mathbf{W}_\iota \mathbf{x},\; f_{2\iota}(\mathbf{z}) = \varphi(\mathbf{z}) \right\},$$

where $\mathbf{W}_\iota \in \mathbb{R}^{d_\iota \times d_{\iota-1}}$ for all $\iota \in [p]$, and $q \triangleq \max_{\iota=0,\dots,p} d_\iota$, we solve

$$\begin{aligned} \widehat{h}^{dac} \triangleq & \operatorname*{argmin}_{h \in \mathcal{H}} \widehat{L}^{dac}_{01}(h) = \frac{1}{N} \sum_{i=1}^{N} \mathbf{1}\left\{h\left(\mathbf{x}_i\right) \neq h^*\left(\mathbf{x}_i\right)\right\} \\ & \text{s.t.} \quad m_{\mathcal{A}}(f, \mathbf{x}^u_i) > \tau \quad \forall\, i \in [|\mathbf{X}^u|] \end{aligned} \tag{B.3}$$

for any $0 < \tau < \max_{f \in \mathcal{F}} \inf_{\mathbf{x} \in \mathcal{X}} m_{\mathcal{A}}(f, \mathbf{x})$. The corresponding reduced function class is given by

$$\mathcal{H}_{dac} \triangleq \left\{h \in \mathcal{H} \mid m_{\mathcal{A}}(f, \mathbf{x}^u_i) > \tau \quad \forall\, i \in [|\mathbf{X}^u|]\right\}.$$

Specifically, with $\mu \triangleq \sup_{h \in \mathcal{H}_{dac}} \mathbb{P}_P\left[\exists\, \mathbf{x}' \in \mathcal{A}(\mathbf{x}) : h(\mathbf{x}) \neq h(\mathbf{x}')\right]$, [22, 162] demonstrate the following for $\mathcal{H}_{dac}$:

Proposition B.6 ([162] Theorem 3.7, [22] Proposition 2.2)**.** *For any $\delta \in (0,1)$, with probability at least $1 - \delta/2$ (over $\mathbf{X}^u$),*

$$\mu \leq \widetilde{O}\left(\frac{\sum_{\iota=1}^{p} \sqrt{q} \|\mathbf{W}_\iota\|_F}{\tau \sqrt{|\mathbf{X}^u|}} + \sqrt{\frac{\log\left(1/\delta\right) + p \log |\mathbf{X}^u|}{|\mathbf{X}^u|}} \right),$$

where $\widetilde{O}\left(\cdot\right)$ hides polylogarithmic factors in $|\mathbf{X}^u|$ and d.

Leveraging the existing theory above on finite sample guarantee of the maximum possible inconsistency, we have the following.

Theorem B.7 (Formal restatement of Theorem 4.4 on classification with DAC)**.** *Learning the classifier with DAC regularization in (B.3) provides that, for any $\delta \in (0,1)$, with probability at least $1 - \delta$,*

$$L_{01}\left(\widehat{h}^{dac}\right) - L_{01}\left(h^*\right) \leq 4\mathfrak{R} + \sqrt{\frac{2\log(4/\delta)}{N}}, \tag{B.4}$$

where with $0 < \mu < 1$ defined in Proposition B.6, for any $0 \leq q < \frac{1}{2}$ and $c > 1 + 4\mu$,

(a) when $\mathcal{A}$ satisfies $(q, 2\mu)$-constant expansion, $\mathfrak{R} \leq \sqrt{\frac{2K\log(2N)}{N}} + 2\max\left\{q, 2\mu\right\}$;

(b) when $\mathcal{A}$ satisfies $(\frac{1}{2}, c)$-multiplicative expansion, $\mathfrak{R} \leq \sqrt{\frac{2K\log(2N)}{N}} + \frac{4\mu}{\min\{c-1,1\}}$.

First, to quantify the function class complexity and relate it to the generalization error, we leverage the notion of Rademacher complexity and the associated standard generalization bound.

Lemma B.8. *Given a fixed function class* $\mathcal{H}_{dac}$ (i.e., *conditioned on* $\mathbf{X}^u$) *and a* B*-bounded and* C_l*-Lipschitz loss function* l*, let* $\widehat{L}(h) = \frac{1}{N}\sum_{i=1}^N l(h(\mathbf{x}_i), y_i)$*,* $L(h) = \mathbb{E}\left[l(h(\mathbf{x}_i), y_i)\right]$*, and* $\widehat{h}^{dac} = \mathrm{argmin}_{h\in\mathcal{H}_{dac}} \widehat{L}(h)$*. Then for any* $\delta \in (0,1)$*, with probability at least* $1-\delta$ *over* $\mathbf{X}$*,*

$$L(\widehat{h}^{dac}) - L(h^*) \leq 4C_l \cdot \mathfrak{R}_N\left(\mathcal{H}_{dac}\right) + \sqrt{\frac{2B^2\log(4/\delta)}{N}}.$$

Proof of Lemma B.8. We first decompose the expected excess risk as

$$L(\widehat{h}^{dac}) - L(h^*) = \left(L(\widehat{h}^{dac}) - \widehat{L}(\widehat{h}^{dac})\right) + \left(\widehat{L}(\widehat{h}^{dac}) - \widehat{L}(h^*)\right) + \left(\widehat{L}(h^*) - L(h^*)\right),$$

where $\widehat{L}(\widehat{h}^{dac}) - \widehat{L}(h^*) \leq 0$ by the basic inequality. Since both $\widehat{h}^{dac}, h^* \in \mathcal{H}_{dac}$, we then have

$$L(\widehat{h}^{dac}) - L(h^*) \leq 2 \sup_{h\in\mathcal{H}_{dac}} \left|L(h) - \widehat{L}(h)\right|.$$

Let $g^+(\mathbf{X},\mathbf{y}) = \sup_{h\in\mathcal{H}_{dac}} : L(h) - \widehat{L}(h)$ and $g^-(\mathbf{X},\mathbf{y}) = \sup_{h\in\mathcal{H}_{dac}} : -L(h) + \widehat{L}(h)$. Then,

$$\mathbb{P}\left[L(\widehat{h}^{dac}) - L(h^*) \geq \epsilon\right] \leq \mathbb{P}\left[g^+(\mathbf{X},\mathbf{y}) \geq \frac{\epsilon}{2}\right] + \mathbb{P}\left[g^-(\mathbf{X},\mathbf{y}) \geq \frac{\epsilon}{2}\right].$$

We will derive a tail bound for $g^+(\mathbf{X},\mathbf{y})$ with the standard inequalities and symmetrization argument [6, 155], while the analogous statement holds for $g^-(\mathbf{X},\mathbf{y})$.

Let $(\mathbf{X}^{(1)}, \mathbf{y}^{(1)})$ be a sample set generated by replacing an arbitrary sample in $(\mathbf{X},\mathbf{y})$ with an independent sample $(\mathbf{x}, y) \sim P(\mathbf{x}, y)$. Since l is B-bounded, we have $\left|g^+(\mathbf{X},\mathbf{y}) - g^+(\mathbf{X}^{(1)},\mathbf{y}^{(1)})\right| \leq B/N$. Then, via McDiarmid's inequality [6],

$$\mathbb{P}\left[g^+(\mathbf{X},\mathbf{y}) \geq \mathbb{E}[g^+(\mathbf{X},\mathbf{y})] + t\right] \leq \exp\left(-\frac{2Nt^2}{B^2}\right).$$

For an arbitrary sample set $(\mathbf{X},\mathbf{y})$, let $\widehat{L}_{(\mathbf{X},\mathbf{y})}\left(h\right) = \frac{1}{N}\sum_{i=1}^N l\left(h(\mathbf{x}_i), y_i\right)$ be the empirical risk of h with respect to $(\mathbf{X},\mathbf{y})$. Then, by a classical symmetrization argument (e.g., proof of [155] Theorem 4.10), we can bound the expectation: for an independent sample set $(\mathbf{X}',\mathbf{y}') \in$

$\mathcal{X}^N \times \mathcal{Y}^N$ drawn *i.i.d.* from P,

$$\begin{aligned}
\mathbb{E}\left[g^+(\mathbf{X},\mathbf{y})\right] =& \mathbb{E}_{(\mathbf{X},\mathbf{y})}\left[\sup_{h\in\mathcal{H}_{dac}} L(h) - \widehat{L}_{(\mathbf{X},\mathbf{y})}(h)\right] \\
=& \mathbb{E}_{(\mathbf{X},\mathbf{y})}\left[\sup_{h\in\mathcal{H}_{dac}} \mathbb{E}_{(\mathbf{X}',\mathbf{y}')}\left[\widehat{L}_{(\mathbf{X}',\mathbf{y}')}(h)\right] - \widehat{L}_{(\mathbf{X},\mathbf{y})}(h)\right] \\
=& \mathbb{E}_{(\mathbf{X},\mathbf{y})}\left[\sup_{h\in\mathcal{H}_{dac}} \mathbb{E}_{(\mathbf{X}',\mathbf{y}')}\left[\widehat{L}_{(\mathbf{X}',\mathbf{y}')}(h) - \widehat{L}_{(\mathbf{X},\mathbf{y})}(h) \,\middle|\, (\mathbf{X},\mathbf{y})\right]\right] \\
\leq& \mathbb{E}_{(\mathbf{X},\mathbf{y})}\left[\mathbb{E}_{(\mathbf{X}',\mathbf{y}')}\left[\sup_{h\in\mathcal{H}_{dac}} \widehat{L}_{(\mathbf{X}',\mathbf{y}')}(h) - \widehat{L}_{(\mathbf{X},\mathbf{y})}(h) \,\middle|\, (\mathbf{X},\mathbf{y})\right]\right] \\
& \text{(Law of iterated conditional expectation)} \\
=& \mathbb{E}_{(\mathbf{X},\mathbf{y},\mathbf{X}',\mathbf{y}')}\left[\sup_{h\in\mathcal{H}_{dac}} \widehat{L}_{(\mathbf{X}',\mathbf{y}')}(h) - \widehat{L}_{(\mathbf{X},\mathbf{y})}(h)\right]
\end{aligned}$$

Since $(\mathbf{X},\mathbf{y}), (\mathbf{X}',\mathbf{y}')$ are drawn *i.i.d.* from P, we can introduce *i.i.d.* Rademacher random variables $\mathbf{r} = \{r_i \in \{-1,+1\} \mid i \in [N]\}$ (independent of both $(\mathbf{X},\mathbf{y})$ and $(\mathbf{X}',\mathbf{y}')$) such that

$$\begin{aligned}
\mathbb{E}\left[g^+(\mathbf{X},\mathbf{y})\right] \leq& \mathbb{E}_{(\mathbf{X},\mathbf{y},\mathbf{X}',\mathbf{y}',\mathbf{r})}\left[\sup_{h\in\mathcal{H}_{dac}} \frac{1}{N}\sum_{i=1}^{N} r_i \cdot \left(l\left(h\left(\mathbf{x}_i'\right), y_i'\right) - l\left(h\left(\mathbf{x}_i\right), y_i\right)\right)\right] \\
\leq& 2\, E_{(\mathbf{X},\mathbf{y},\mathbf{r})}\left[\sup_{h\in\mathcal{H}_{dac}} \frac{1}{N}\sum_{i=1}^{N} r_i \cdot l\left(h\left(\mathbf{x}_i\right), y_i\right)\right] \\
\leq& 2\, \mathfrak{R}_N\left(l \circ \mathcal{H}_{dac}\right)
\end{aligned}$$

where $l \circ \mathcal{H}_{dac} = \{l(h(\cdot),\cdot) : \mathcal{X} \times \mathcal{Y} \to \mathbb{R} : h \in \mathcal{H}_{dac}\}$ is the loss function class, and

$$\mathfrak{R}_N\left(\mathcal{F}\right) \triangleq E_{(\mathbf{X},\mathbf{y},\mathbf{r})}\left[\sup_{f\in\mathcal{F}} \frac{1}{N}\sum_{i=1}^{N} r_i \cdot f\left(\mathbf{x}_i, y_i\right)\right]$$

denotes the Rademacher complexity. Analogously, $\mathbb{E}[g^-(\mathbf{X},\mathbf{y})] \leq 2\mathfrak{R}_N\left(l \circ \mathcal{H}_{dac}\right)$. Therefore, assuming that $T^{dac}_{\widetilde{\mathcal{A}},\mathbf{X}}(\mathcal{H}) \subseteq \mathcal{H}_{dac}(\mathcal{H})$ holds, with probability at least $1-\delta/2$,

$$L(\widehat{h}^{dac}) - L(h^*) \leq 4\mathfrak{R}_N\left(l \circ \mathcal{H}_{dac}\right) + \sqrt{\frac{2B^2\log(4/\delta)}{N}}$$

Finally, since $l(\cdot, y)$ is C_l-Lipschitz for all $y \in \mathcal{Y}$, by [89] Theorem 4.12, we have $\mathfrak{R}_N\left(l \circ \mathcal{H}_{dac}\right) \leq C_l \cdot \mathfrak{R}_N\left(\mathcal{H}_{dac}\right)$. ■

Lemma B.9 ([22], Lemma A.1)**.** *For any $h \in \mathcal{H}_{dac}$, when P satisfies*

(a) $(q, 2\mu)$-constant expansion with $q < \frac{1}{2}$, $P\left(M\right) \leq \max\left\{q, 2\mu\right\}$;

(b) $\left(\frac{1}{2}, c\right)$-multiplicative expansion with $c > 1 + 4\mu$, $P\left(M\right) \leq \max\left\{\frac{2\mu}{c-1}, 2\mu\right\}$.

Proof of Lemma B.9. We start with the proof for Lemma B.9 (a). By definition of M_k and $\widehat{y}_k$, we know that $M_k = M \cap \mathcal{X}_k \leq \frac{1}{2}$. Therefore, for any $0 < q < \frac{1}{2}$, one of the following two cases holds:

(i) $P(M) < q$;

(ii) $P(M) \geq q$. Since $P(M \cap \mathcal{X}_k) < \frac{1}{2}$ for all $k \in [K]$ holds by construction, with the $(q, 2\mu)$-constant expansion, $P(NB(M)) \geq \min\{P(M), 2\mu\} + P(M)$.

Meanwhile, since the ground truth classifier h^* is invariant throughout the neighborhoods, $NB(M_k) \cap NB(M_{k'}) = \emptyset$ for $k \neq k'$, and therefore $NB(M) \backslash M = \bigcup_{k=1}^{K} NB(M_k) \backslash M_k$ with each $NB(M_k) \backslash M_k$ disjoint. Then, we observe that for each $\mathbf{x} \in NB(M) \backslash M$, here exists some $k = h^*(\mathbf{x})$ such that $\mathbf{x} \in NB(M_k) \backslash M_k$. $\mathbf{x} \in \mathcal{X}_k \backslash M_k$ implies that $h(\mathbf{x}) = \widehat{y}_k$, while $\mathbf{x} \in NB(M_k)$ suggests that there exists some $\mathbf{x}' \in \mathcal{A}(\mathbf{x}) \cap \mathcal{A}(\mathbf{x}'')$ where $\mathbf{x}'' \in M_k$ such that either $h(\mathbf{x}') = \widehat{y}_k$ and $h(\mathbf{x}') \neq h(\mathbf{x}'')$ for $\mathbf{x}' \in \mathcal{A}(\mathbf{x}'')$, or $h(\mathbf{x}') \neq \widehat{y}_k$ and $h(\mathbf{x}') \neq h(\mathbf{x})$ for $\mathbf{x}' \in \mathcal{A}(\mathbf{x})$. Therefore, we have

$$P(NB(M) \backslash M) \leq 2\mathbb{P}_P[\exists\, \mathbf{x}' \in \mathcal{A}(\mathbf{x}) \text{ s.t. } h(\mathbf{x}) \neq h(\mathbf{x}')] \leq 2\mu.$$

Moreover, since $P(NB(M)) - P(M) \leq P(NB(M) \backslash M) \leq 2\mu$, we know that

$$\min\{P(M), 2\mu\} + P(M) \leq P(NB(M)) \leq P(M) + 2\mu.$$

That is, $P(M) \leq 2\mu$.

Overall, we have $P(M) \leq \max\{q, 2\mu\}$.

To show Lemma B.9 (b), we recall from [162] Lemma B.6 that for any $c > 1 + 4\mu$, $\left(\frac{1}{2}, c\right)$-multiplicative expansion implies $\left(\frac{2\mu}{c-1}, 2\mu\right)$-constant expansion. Then leveraging the proof for Lemma B.9 (a), with $q = \frac{2\mu}{c-1}$, we have $P(M) \leq \max\left\{\frac{2\mu}{c-1}, 2\mu\right\}$. ■

Proof of Theorem B.7. To show (B.4), we leverage Lemma B.8 and observe that $B = 1$ with the zero-one loss. Therefore, conditioned on $\mathcal{H}_{dac}$ (which depends only on $\mathbf{X}^u$ but not on $\mathbf{X}$), for any $\delta \in (0, 1)$, with probability at least $1 - \delta/2$,

$$L_{01}\left(\widehat{h}^{dac}\right) - L_{01}(h^*) \leq 4\mathfrak{R}_N(l_{01} \circ \mathcal{H}_{dac}) + \sqrt{\frac{2\log(4/\delta)}{N}}.$$

For the upper bounds of the Rademacher complexity, let $\widetilde{\mu} \triangleq \sup_{h \in \mathcal{H}_{dac}} P(M)$ where M denotes the global minority set with respect to $h \in \mathcal{H}_{dac}$. Lemma B.9 suggests that

(a) when P satisfies $(q, 2\mu)$-constant expansion for some $q < \frac{1}{2}$, $\widetilde{\mu} \leq \max\{q, 2\mu\}$; while

(b) when P satisfies $(\frac{1}{2}, c)$-multiplicative expansion for some $c > 1 + 4\mu$, $\widetilde{\mu} \leq \frac{2\mu}{\min\{c-1,1\}}$. Then, it is sufficient to show that, conditioned on $\mathcal{H}_{dac}$,

$$\mathfrak{R}_N \left(l_{01} \circ \mathcal{H}_{dac}\right) \leq \sqrt{\frac{2K \log(2N)}{N}} + 2\widetilde{\mu}. \tag{B.5}$$

To show this, we first consider a fixed set of n observations in $\mathcal{X}$, $\mathbf{X} = \left[\mathbf{x}_1, \ldots, \mathbf{x}_N\right]^\top \in \mathcal{X}^N$. Let the number of distinct behaviors over $\mathbf{X}$ in $\mathcal{H}_{dac}$ be

$$\mathfrak{s}\left(l_{01} \circ \mathcal{H}_{dac}, \mathbf{X}\right) \triangleq \left|\left\{\left[l_{01} \circ h\left(\mathbf{x}_1\right), \ldots, l_{01} \circ h\left(\mathbf{x}_N\right)\right] \;\middle|\; h \in \mathcal{H}_{dac}\right\}\right|.$$

Then, by the Massart's finite lemma, the empirical rademacher complexity with respect to $\mathbf{X}$ is upper bounded by

$$\widehat{\mathfrak{R}}_{\mathbf{X}} \left(l_{01} \circ \mathcal{H}_{dac}\right) \leq \sqrt{\frac{2 \log \mathfrak{s}\left(l_{01} \circ \mathcal{H}_{dac}, \mathbf{X}\right)}{N}}.$$

By the concavity of $\sqrt{\log\left(\cdot\right)}$, we know that,

$$\begin{aligned}
\mathfrak{R}_N \left(l_{01} \circ \mathcal{H}_{dac}\right) =& \mathbb{E}_{\mathbf{X}} \left[\widehat{\mathfrak{R}}_{\mathbf{X}} \left(l_{01} \circ \mathcal{H}_{dac}\right)\right] \leq \mathbb{E}_{\mathbf{X}} \left[\sqrt{\frac{2 \log \mathfrak{s}\left(l_{01} \circ \mathcal{H}_{dac}, \mathbf{X}\right)}{N}}\right] \\
\leq& \sqrt{\frac{2 \log \mathbb{E}_{\mathbf{X}} \left[\mathfrak{s}\left(l_{01} \circ \mathcal{H}_{dac}, \mathbf{X}\right)\right]}{N}}.
\end{aligned} \tag{B.6}$$

Since $P\left(M\right) \leq \widetilde{\mu} \leq \frac{1}{2}$ for all $h \in \mathcal{H}_{dac}$, we have that, conditioned on $\mathcal{H}_{dac}$,

$$\begin{aligned}
\mathbb{E}_{\mathbf{X}} \left[\mathfrak{s}\left(l_{01} \circ \mathcal{H}_{dac}, \mathbf{X}\right)\right] \leq& \sum_{r=0}^{N} \binom{N}{r} \widetilde{\mu}^r \left(1 - \widetilde{\mu}\right)^{N-r} \cdot \binom{N-r-1}{\min(K, N-r) - 1} 2^{K+r} \\
\leq& (2N)^K \sum_{r=0}^{N} \binom{N}{r} \left(2\widetilde{\mu}\right)^r \left(1 - \widetilde{\mu}\right)^{N-r} \\
=& (2N)^K \left(1 - \widetilde{\mu} + 2\widetilde{\mu}\right)^N \\
\leq& (2N)^K \cdot e^{N\widetilde{\mu}}.
\end{aligned}$$

Plugging this into (B.6) yields (B.5). Finally, the randomness in $\mathcal{H}_{dac}$ is quantified by $\widetilde{\mu}$, μ, and upper bounded by Proposition B.6. ■

B.4 Supplementary Application: Domain Adaptation

As a supplementary example, we demonstrate the possible failure of DA-ERM, and alternatively how DAC regularization can serve as a remedy. Concretely, we consider an illustrative linear regression problem in the domain adaptation setting: with training samples drawn from a source distribution P^s and generalization (in terms of excess risk) evaluated over a related but different target distribution P^t. With distinct $\mathbb{E}_{P^s}[y|\mathbf{x}]$ and $\mathbb{E}_{P^t}[y|\mathbf{x}]$, we assume the existence of an unknown but unique inclusionwisely maximal invariant feature subspace $\mathcal{X}_r \subset \mathcal{X}$ such that $P^s[y|\mathbf{x} \in \mathcal{X}_r] = P^t[y|\mathbf{x} \in \mathcal{X}_r]$, we aim to demonstrate the advantage of the DAC regularization over the ERM on augmented training set, with a provable separation in the respective excess risks.

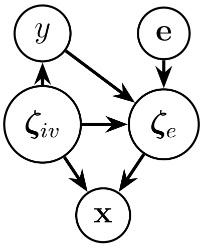


Figure B.1: Causal graph shared by P^s and P^t.

Source and target distributions. Formally, the source and target distributions are concretized with the causal graph in Figure B.1. For both P^s and P^t, the observable feature $\mathbf{x}$ is described via a linear generative model in terms of two latent features, the 'invariant' feature $\boldsymbol{\zeta}_{iv} \in \mathbb{R}^{d_{iv}}$ and the 'environmental' feature $\boldsymbol{\zeta}_e \in \mathbb{R}^{d_e}$:

$$\mathbf{x} = g(\boldsymbol{\zeta}_{iv}, \boldsymbol{\zeta}_e) \triangleq \mathbf{S}\left[\boldsymbol{\zeta}_{iv}; \boldsymbol{\zeta}_e\right] = \mathbf{S}_{iv}\boldsymbol{\zeta}_{iv} + \mathbf{S}_e\boldsymbol{\zeta}_e,$$

where $\mathbf{S} = \left[\mathbf{S}_{iv}, \mathbf{S}_e\right] \in \mathbb{R}^{d \times (d_{iv}+d_e)}$ ($d_{iv} + d_e \leq d$) consists of orthonormal columns. Let the label y depends only on the invariant feature $\boldsymbol{\zeta}_{iv}$ for both domains,

$$y = (\boldsymbol{\theta}^*)^\top \mathbf{x} + z = (\boldsymbol{\theta}^*)^\top \mathbf{S}_{iv}\boldsymbol{\zeta}_{iv} + z, \quad z \sim \mathcal{N}\left(0, \sigma^2\right), \quad z \perp \boldsymbol{\zeta}_{iv},$$

for some $\boldsymbol{\theta}^* \in \mathrm{Range}\left(\mathbf{S}_{iv}\right)$ such that $P^s[y|\boldsymbol{\zeta}_{iv}] = P^t[y|\boldsymbol{\zeta}_{iv}]$, while the environmental feature $\boldsymbol{\zeta}_e$ is conditioned on y, $\boldsymbol{\zeta}_{iv}$, (along with the Gaussian noise z), and varies across different domains $\mathbf{e}$ with $\mathbb{E}_{P^s}[y|\mathbf{x}] \neq \mathbb{E}_{P^t}[y|\mathbf{x}]$. In other words, with the square loss $l(h(\mathbf{x}), y) = \frac{1}{2}(h(\mathbf{x}) - y)^2$, the optimal hypotheses that minimize the expected excess risk over the source and target distributions are distinct. Therefore, learning via the ERM with training samples from P^s can overfit the source distribution, in which scenario identifying the invariant feature subspace $\mathrm{Range}\left(\mathbf{S}_{iv}\right)$ becomes indispensable for achieving good generalization in the target domain.

For P^s and P^t, we assume the following regularity conditions:

Assumption 3 (Regularity conditions for P^s and P^t). Let P^s satisfy Assumption 2. While P^t satisfies that $\mathbb{E}_{P^t}[\mathbf{x}\mathbf{x}^\top] \succ 0$, and

(a) for the invariant feature, $c_{t,iv}\mathbf{I}_{d_{iv}} \preccurlyeq \mathbb{E}_{P^t}[\boldsymbol{\zeta}_{iv}\boldsymbol{\zeta}_{iv}^\top] \preccurlyeq C_{t,iv}\mathbf{I}_{d_{iv}}$ for some $C_{t,iv} \geq c_{t,iv} = \Theta(1)$;

(b) for the environmental feature, $\mathbb{E}_{P^t}[\boldsymbol{\zeta}_e\boldsymbol{\zeta}_e^\top] \succcurlyeq c_{t,e}\mathbf{I}_{d_e}$ for some $c_{t,e} > 0$, and $\mathbb{E}_{P^t}\left[z \cdot \boldsymbol{\zeta}_e\right] = \mathbf{0}$.

Training samples and data augmentations. Let $\mathbf{X} = [\mathbf{x}_1; \ldots; \mathbf{x}_N]$ be a set of N samples drawn *i.i.d.* from $P^s(\mathbf{x})$ such that $\mathbf{y} = \mathbf{X}\boldsymbol{\theta}^* + \mathbf{z}$ where $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I}_N)$. Recall that we denote the augmented training sets, including/excluding the original samples, respectively, with

$$\widetilde{\mathcal{A}}(\mathbf{X}) = [\mathbf{x}_1; \cdots; \mathbf{x}_N; \mathbf{x}_{1,1}; \cdots; \mathbf{x}_{N,1}; \cdots; \mathbf{x}_{1,\alpha}; \cdots; \mathbf{x}_{N,\alpha}] \in \mathcal{X}^{(1+\alpha)N},$$
$$\mathcal{A}(\mathbf{X}) = [\mathbf{x}_{1,1}; \cdots; \mathbf{x}_{N,1}; \cdots; \mathbf{x}_{1,\alpha}; \cdots; \mathbf{x}_{N,\alpha}] \in \mathcal{X}^{\alpha N}.$$

In particular, we consider a set of augmentations that only perturb the environmental feature $\boldsymbol{\zeta}_e$, while keep the invariant feature $\boldsymbol{\zeta}_{iv}$ intact:

$$\mathbf{S}_{iv}^\top \mathbf{x}_i = \mathbf{S}_{iv}^\top \mathbf{x}_{i,j}, \quad \mathbf{S}_e^\top \mathbf{x}_i \neq \mathbf{S}_e^\top \mathbf{x}_{i,j} \quad \forall\, i \in [n],\ j \in [\alpha]. \tag{B.7}$$

We recall the notion $\boldsymbol{\Delta} \triangleq \mathcal{A}\left(\mathbf{X}\right) - \mathbf{M}\mathbf{X}$ such that $d_{aug} \triangleq \operatorname{rank}\left(\boldsymbol{\Delta}\right) = \operatorname{rank}\left(\widetilde{\mathcal{A}}\left(\mathbf{X}\right) - \widetilde{\mathbf{M}}\mathbf{X}\right)$ ($0 \leq d_{aug} \leq d_e$), and assume that $\mathbf{X}$ and $\mathcal{A}(\mathbf{X})$ are representative enough:

Assumption 4 (Diversity of $\mathbf{X}$ and $\mathcal{A}(\mathbf{X})$). $(\mathbf{X}, \mathbf{y}) \in \mathcal{X}^n \times \mathcal{Y}^n$ is sufficiently large with $n \gg \rho^4 d$, $\boldsymbol{\theta}^* \in \operatorname{row}(\mathbf{X})$, and $d_{aug} = d_e$.

Excess risks in target domain. Learning from the linear hypothesis class $\mathcal{H} = \left\{h(\mathbf{x}) = \mathbf{x}^\top\boldsymbol{\theta} \;\middle|\; \boldsymbol{\theta} \in \mathbb{R}^d\right\}$, with the DAC regularization on $h\left(\mathbf{x}_i\right) = h\left(\mathbf{x}_{i,j}\right)$, we have

$$\widehat{\boldsymbol{\theta}}^{dac} = \underset{\boldsymbol{\theta} \in \mathcal{H}_{dac}}{\operatorname{argmin}} \frac{1}{2N}\|\mathbf{y} - \mathbf{X}\boldsymbol{\theta}\|_2^2, \quad \mathcal{H}_{dac} = \left\{h\left(\mathbf{x}\right) = \boldsymbol{\theta}^\top\mathbf{x} \;\middle|\; \boldsymbol{\Delta}\boldsymbol{\theta} = \mathbf{0}\right\},$$

while with the ERM on augmented training set,

$$\widehat{\boldsymbol{\theta}}^{da-erm} = \underset{\boldsymbol{\theta} \in \mathbb{R}^d}{\operatorname{argmin}} \frac{1}{2(1+\alpha)N}\|\widetilde{\mathbf{M}}\mathbf{y} - \widetilde{\mathcal{A}}(\mathbf{X})\boldsymbol{\theta}\|_2^2,$$

where $\mathbf{M}$ and $\widetilde{\mathbf{M}}$ denote the vertical stacks of α and $1+\alpha$ identity matrices of size $n \times n$, respectively as denoted earlier.

We are interested in the excess risk on P^t: $L_t\left(\boldsymbol{\theta}\right) - L_t\left(\boldsymbol{\theta}^*\right)$ where $L_t\left(\boldsymbol{\theta}\right) \triangleq \mathbb{E}_{P^t(\mathbf{x},y)}\left[\frac{1}{2}(y - \mathbf{x}^\top\boldsymbol{\theta})^2\right]$.

Theorem B.10 (Domain adaptation with DAC)**.** *Under Assumption 3(a) and Assumption 4, $\widehat{\boldsymbol{\theta}}^{dac}$ satisfies that, with constant probability,*

$$\mathbb{E}_{P^s}\left[L_t(\widehat{\boldsymbol{\theta}}^{dac}) - L_t(\boldsymbol{\theta}^*)\right] \lesssim \frac{\sigma^2 d_{iv}}{N}. \tag{B.8}$$

Theorem B.11 (Domain adaptation with ERM on augmented samples)**.** *Under Assumption 3 and Assumption 4,* $\widehat{\boldsymbol{\theta}}^{dac}$ *and* $\widehat{\boldsymbol{\theta}}^{da-erm}$ *satisfies that,*

$$\mathbb{E}_{P^s}\left[L_t(\widehat{\boldsymbol{\theta}}^{da-erm}) - L_t(\boldsymbol{\theta}^*)\right] \geq \mathbb{E}_{P^s}\left[L_t(\widehat{\boldsymbol{\theta}}^{dac}) - L_t(\boldsymbol{\theta}^*)\right] + c_{t,e} \cdot EER_e, \tag{B.9}$$

for some $EER_e > 0$*.*

In contrast to $\widehat{\boldsymbol{\theta}}^{dac}$ where the DAC constraints enforce $\mathbf{S}_e^\top \widehat{\boldsymbol{\theta}}^{dac} = \mathbf{0}$ with a sufficiently diverse $\mathcal{A}(\mathbf{X})$ (Assumption 4), the ERM on augmented training set fails to filter out the environmental feature in $\widehat{\boldsymbol{\theta}}^{da-erm}$: $\mathbf{S}_e^\top \widehat{\boldsymbol{\theta}}^{da-erm} \neq \mathbf{0}$. As a consequence, the expected excess risk of $\widehat{\boldsymbol{\theta}}^{da-erm}$ in the target domain can be catastrophic when $c_{t,e} \to \infty$, as instantiated by Example 4.

Proofs and instantiation. Recall that for $\boldsymbol{\Delta} \triangleq \mathcal{A}(\mathbf{X}) - \mathbf{M}\mathbf{X}$, $\mathbf{P}_{\boldsymbol{\Delta}}^{\perp} \triangleq \mathbf{I}_d - \boldsymbol{\Delta}^\dagger \boldsymbol{\Delta}$ denotes the orthogonal projector onto the dimension-$(d - d_{aug})$ null space of $\boldsymbol{\Delta}$. Furthermore, let $\mathbf{P}_{iv} \triangleq \mathbf{S}_{iv}\mathbf{S}_{iv}^\top$ and $\mathbf{P}_e \triangleq \mathbf{S}_e\mathbf{S}_e^\top$ be the orthogonal projectors onto the invariant and environmental feature subspaces, respectively, such that $\mathbf{x} = \mathbf{S}_{iv}\boldsymbol{\zeta}_{iv} + \mathbf{S}_e\boldsymbol{\zeta}_e = (\mathbf{P}_{iv} + \mathbf{P}_e)\,\mathbf{x}$ for all $\mathbf{x}$.

Proof of Theorem B.10. By construction (B.7), $\boldsymbol{\Delta}\mathbf{P}_{iv} = \mathbf{0}$, and it follows that $\mathbf{P}_{iv} \preccurlyeq \mathbf{P}_{\boldsymbol{\Delta}}^{\perp}$. Meanwhile from Assumption 4, $d_{aug} = d_e$ implies that $\dim\left(\mathbf{P}_{\boldsymbol{\Delta}}^{\perp}\right) = d_{iv}$. Therefore, $\mathbf{P}_{iv} = \mathbf{P}_{\boldsymbol{\Delta}}^{\perp}$, and the data augmentation consistency constraints can be restated as

$$\mathcal{H}_{dac} = \left\{h(\mathbf{x}) = \boldsymbol{\theta}^\top \mathbf{x} \;\middle|\; \mathbf{P}_{\boldsymbol{\Delta}}^{\perp}\boldsymbol{\theta} = \boldsymbol{\theta}\right\} = \left\{h(\mathbf{x}) = \boldsymbol{\theta}^\top \mathbf{x} \;\middle|\; \mathbf{P}_{iv}\boldsymbol{\theta} = \boldsymbol{\theta}\right\}$$

Then with $\boldsymbol{\theta}^* \in \mathrm{row}(\mathbf{X})$ from Assumption 4,

$$\widehat{\boldsymbol{\theta}}^{dac} - \boldsymbol{\theta}^* = \frac{1}{N}\widehat{\boldsymbol{\Sigma}}_{\mathbf{X}_{iv}}^\dagger \mathbf{P}_{iv}\mathbf{X}^\top(\mathbf{X}\mathbf{P}_{iv}\boldsymbol{\theta}^* + \mathbf{z}) - \boldsymbol{\theta}^* = \frac{1}{N}\widehat{\boldsymbol{\Sigma}}_{\mathbf{X}_{iv}}^\dagger \mathbf{P}_{iv}\mathbf{X}^\top\mathbf{z},$$

where $\widehat{\boldsymbol{\Sigma}}_{\mathbf{X}_{iv}} \triangleq \frac{1}{N}\mathbf{P}_{iv}\mathbf{X}^\top\mathbf{X}\mathbf{P}_{iv}$. Since $\widehat{\boldsymbol{\theta}}^{dac} - \boldsymbol{\theta}^* \in \mathrm{col}\,(\mathbf{S}_{iv})$, we have $\mathbb{E}_{P^t}\left[z \cdot \mathbf{x}^\top \mathbf{P}_e(\widehat{\boldsymbol{\theta}}^{dac} - \boldsymbol{\theta}^*)\right] =$

0. Therefore, let $\mathbf{\Sigma}_{\mathbf{x},t} \triangleq \mathbb{E}_{P^t}[\mathbf{x}\mathbf{x}^\top]$, with high probability,

$$
\begin{aligned}
E_{P^s}\left[L_t(\widehat{\boldsymbol{\theta}}^{dac}) - L_t(\boldsymbol{\theta}^*)\right] &= E_{P^s}\left[\frac{1}{2}\|\widehat{\boldsymbol{\theta}}^{dac} - \boldsymbol{\theta}^*\|_{\mathbf{\Sigma}_{\mathbf{x},t}}^2\right] \\
&= \mathrm{tr}\left(\frac{1}{2N}\mathbb{E}_{P^s}\left[\mathbf{z}\mathbf{z}^\top\right]\ \mathbb{E}_{P^s}\left[\left(\frac{1}{N}\mathbf{P}_{iv}\mathbf{X}^\top\mathbf{X}\mathbf{P}_{iv}\right)^\dagger\right]\ \mathbf{\Sigma}_{\mathbf{x},t}\right) \\
&= \mathrm{tr}\left(\frac{\sigma^2}{2N}\ \mathbb{E}_{P^s}\left[\widehat{\mathbf{\Sigma}}_{\mathbf{X}_{iv}}^\dagger\right]\ \mathbf{\Sigma}_{\mathbf{x},t}\right) \\
&\leq C_{t,iv}\ \frac{\sigma^2}{2N}\ tr\left(\mathbb{E}_{P^s}\left[\widehat{\mathbf{\Sigma}}_{\mathbf{X}_{iv}}^\dagger\right]\right) \qquad (\text{Lemma B.12},\ \textit{w.h.p.}) \\
&\lesssim \frac{\sigma^2}{2N}\ \mathrm{tr}\left(\left(\mathbb{E}_{P^s}\left[\mathbf{P}_{iv}\mathbf{x}\mathbf{x}^\top\mathbf{P}_{iv}\right]\right)^\dagger\right) \\
&\leq \frac{\sigma^2 d_{iv}}{2Nc} \lesssim \frac{\sigma^2 d_{iv}}{2N}.
\end{aligned}
$$

■

Proof of Theorem B.11. Let $\widehat{\mathbf{\Sigma}}_{\widetilde{\mathcal{A}}(\mathbf{X})} \triangleq \frac{1}{(1+\alpha)N}\widetilde{\mathcal{A}}\left(\mathbf{X}\right)^\top \widetilde{\mathcal{A}}\left(\mathbf{X}\right)$. Then with $\boldsymbol{\theta}^* \in \mathrm{row}(\mathbf{X})$ from Assumption 4, we have $\boldsymbol{\theta}^* = \widehat{\mathbf{\Sigma}}_{\widetilde{\mathcal{A}}(\mathbf{X})}^\dagger \widehat{\mathbf{\Sigma}}_{\widetilde{\mathcal{A}}(\mathbf{X})}\boldsymbol{\theta}^*$. Since $\boldsymbol{\theta}^* \in \mathrm{col}\left(\mathbf{S}_{iv}\right)$, $\widetilde{\mathbf{M}}\mathbf{X}\boldsymbol{\theta}^* = \widetilde{\mathbf{M}}\mathbf{X}\mathbf{P}_{iv}\boldsymbol{\theta}^* = \widetilde{\mathcal{A}}(\mathbf{X})\boldsymbol{\theta}^*$. Then, the ERM on the augmented training set yields

$$
\begin{aligned}
\widehat{\boldsymbol{\theta}}^{da-erm} - \boldsymbol{\theta}^* &= \frac{1}{(1+\alpha)N}\widehat{\mathbf{\Sigma}}_{\widetilde{\mathcal{A}}(\mathbf{X})}^\dagger \widetilde{\mathcal{A}}(\mathbf{X})^\top \widetilde{\mathbf{M}}(\mathbf{X}\boldsymbol{\theta}^* + \mathbf{z}) - \widehat{\mathbf{\Sigma}}_{\widetilde{\mathcal{A}}(\mathbf{X})}^\dagger \widehat{\mathbf{\Sigma}}_{\widetilde{\mathcal{A}}(\mathbf{X})}\boldsymbol{\theta}^* \\
&= \frac{1}{(1+\alpha)N}\widehat{\mathbf{\Sigma}}_{\widetilde{\mathcal{A}}(\mathbf{X})}^\dagger \widetilde{\mathcal{A}}(\mathbf{X})^\top \widetilde{\mathbf{M}}\mathbf{z}.
\end{aligned}
$$

Meanwhile with $\mathbb{E}_{P^t}\left[z \cdot \boldsymbol{\zeta}_e\right] = \mathbf{0}$ from Assumption 3, we have $\mathbb{E}_{P^t}\left[z \cdot \mathbf{P}_e\mathbf{x}\right] = \mathbf{0}$. Therefore, by recalling that $\mathbf{\Sigma}_{\mathbf{x},t} \triangleq \mathbb{E}_{P^t}[\mathbf{x}\mathbf{x}^\top]$,

$$
L_t(\boldsymbol{\theta}) - L_t(\boldsymbol{\theta}^*) \ = \ \mathbb{E}_{P^t(\mathbf{x})}\left[\frac{1}{2}\left(\mathbf{x}^\top(\boldsymbol{\theta} - \boldsymbol{\theta}^*)\right)^2 + z \cdot \mathbf{x}^\top\mathbf{P}_e(\boldsymbol{\theta} - \boldsymbol{\theta}^*)\right] \ = \ \frac{1}{2}\|\boldsymbol{\theta}^* - \boldsymbol{\theta}\|_{\mathbf{\Sigma}_{\mathbf{x},t}}^2,
$$

such that the expected excess risk can be expressed as

$$
\mathbb{E}_{P^s}\left[L_t(\widehat{\boldsymbol{\theta}}^{da-erm}) - L_t(\boldsymbol{\theta}^*)\right] = \frac{1}{2(1+\alpha)^2N^2}\ \mathrm{tr}\left(\mathbb{E}_{P^s}\left[\widehat{\mathbf{\Sigma}}_{\widetilde{\mathcal{A}}(\mathbf{X})}^\dagger\left(\widetilde{\mathcal{A}}(\mathbf{X})^\top\widetilde{\mathbf{M}}\mathbf{z}\mathbf{z}^\top\widetilde{\mathbf{M}}^\top\widetilde{\mathcal{A}}(\mathbf{X})\right)\widehat{\mathbf{\Sigma}}_{\widetilde{\mathcal{A}}(\mathbf{X})}^\dagger\right]\mathbf{\Sigma}_{\mathbf{x},t}\right),
$$

where let $\widehat{\mathbf{\Sigma}}_{\widetilde{\mathcal{A}}(\mathbf{X}_e)} \triangleq \mathbf{P}_e\widehat{\mathbf{\Sigma}}_{\widetilde{\mathcal{A}}(\mathbf{X})}\mathbf{P}_e$,

$$
\begin{aligned}
&\mathbb{E}_{P^s}\left[\widehat{\mathbf{\Sigma}}_{\widetilde{\mathcal{A}}(\mathbf{X})}^\dagger\left(\widetilde{\mathcal{A}}(\mathbf{X})^\top\widetilde{\mathbf{M}}\mathbf{z}\mathbf{z}^\top\widetilde{\mathbf{M}}^\top\widetilde{\mathcal{A}}(\mathbf{X})\right)\widehat{\mathbf{\Sigma}}_{\widetilde{\mathcal{A}}(\mathbf{X})}^\dagger\right] \\
\succcurlyeq\ &\mathbb{E}_{P^s}\left[\left(\mathbf{P}_{iv}\widehat{\mathbf{\Sigma}}_{\widetilde{\mathcal{A}}(\mathbf{X})}^\dagger\mathbf{P}_{iv} + \mathbf{P}_e\widehat{\mathbf{\Sigma}}_{\widetilde{\mathcal{A}}(\mathbf{X})}^\dagger\mathbf{P}_e\right)\widetilde{\mathcal{A}}(\mathbf{X})^\top\widetilde{\mathbf{M}}\mathbf{z}\mathbf{z}^\top\widetilde{\mathbf{M}}^\top\widetilde{\mathcal{A}}(\mathbf{X})\left(\mathbf{P}_{iv}\widehat{\mathbf{\Sigma}}_{\widetilde{\mathcal{A}}(\mathbf{X})}^\dagger\mathbf{P}_{iv} + \mathbf{P}_e\widehat{\mathbf{\Sigma}}_{\widetilde{\mathcal{A}}(\mathbf{X})}^\dagger\mathbf{P}_e\right)\right] \\
\succcurlyeq\ &\sigma^2(1+\alpha)^2N \cdot \mathbb{E}_{P^s}\left[\widehat{\mathbf{\Sigma}}_{\mathbf{X}_{iv}}^\dagger\right] + \mathbb{E}_{P^s}\left[\widehat{\mathbf{\Sigma}}_{\widetilde{\mathcal{A}}(\mathbf{X}_e)}^\dagger\widetilde{\mathcal{A}}(\mathbf{X}_e)^\top\widetilde{\mathbf{M}}\mathbf{z}\mathbf{z}^\top\widetilde{\mathbf{M}}^\top\widetilde{\mathcal{A}}(\mathbf{X}_e)\widehat{\mathbf{\Sigma}}_{\widetilde{\mathcal{A}}(\mathbf{X}_e)}^\dagger\right].
\end{aligned}
$$

We denote

$$\mathrm{EER}_e \triangleq \operatorname{tr}\left(\mathbb{E}_{P^s}\left[\frac{1}{2(1+\alpha)^2 N^2}\widehat{\mathbf{\Sigma}}^{\dagger}_{\widetilde{\mathcal{A}}(\mathbf{X}_e)}\widetilde{\mathcal{A}}(\mathbf{X}_e)^{\top}\widetilde{\mathbf{M}}\mathbf{z}\mathbf{z}^{\top}\widetilde{\mathbf{M}}^{\top}\widetilde{\mathcal{A}}(\mathbf{X}_e)\widehat{\mathbf{\Sigma}}^{\dagger}_{\widetilde{\mathcal{A}}(\mathbf{X}_e)}\right]\right),$$

and observe that

$$\mathrm{EER}_e = \mathbb{E}_{P^s}\left[\frac{1}{2}\|\frac{1}{(1+\alpha)N}\widehat{\mathbf{\Sigma}}^{\dagger}_{\widetilde{\mathcal{A}}(\mathbf{X}_e)}\widetilde{\mathcal{A}}(\mathbf{X}_e)^{\top}\widetilde{\mathbf{M}}\mathbf{z}\|_2^2\right] > 0.$$

Finally, we complete the proof by partitioning the lower bound for the target expected excess risk of $\widehat{\boldsymbol{\theta}}^{da-erm}$ into the invariantand environmental parts such that

$$\begin{aligned}
&\mathbb{E}_{P^s}\left[L_t(\widehat{\boldsymbol{\theta}}^{da-erm}) - L_t(\boldsymbol{\theta}^*)\right] \\
\geq\ &\underbrace{\operatorname{tr}\left(\frac{\sigma^2}{2N}\,\mathbb{E}_{P^s}\left[\widehat{\mathbf{\Sigma}}^{\dagger}_{\mathbf{X}_{iv}}\right]\mathbf{\Sigma}_{\mathbf{x},t}\right)}_{=\mathbb{E}\left[L_t(\widehat{\boldsymbol{\theta}}^{dac})-L_t(\boldsymbol{\theta}^*)\right]} \\
&+\underbrace{\operatorname{tr}\left(\mathbb{E}_{P^s}\left[\frac{1}{2(1+\alpha)^2N^2}\widehat{\mathbf{\Sigma}}^{\dagger}_{\widetilde{\mathcal{A}}(\mathbf{X}_e)}\widetilde{\mathcal{A}}(\mathbf{X}_e)^{\top}\widetilde{\mathbf{M}}\mathbf{z}\mathbf{z}^{\top}\widetilde{\mathbf{M}}^{\top}\widetilde{\mathcal{A}}(\mathbf{X}_e)\widehat{\mathbf{\Sigma}}^{\dagger}_{\widetilde{\mathcal{A}}(\mathbf{X}_e)}\right]\mathbf{\Sigma}_{\mathbf{x},t}\right)}_{\text{expected excess risk from environmental feature subspace}\geq c_{t,e}\cdot\mathrm{EER}_e} \\
\geq\ &\mathbb{E}_{P^s}\left[L_t(\widehat{\boldsymbol{\theta}}^{dac}) - L_t(\boldsymbol{\theta}^*)\right] + c_{t,e}\cdot\mathrm{EER}_e.
\end{aligned}$$

■

Now we construct a specific domain adaptation example with a large separation (*i.e.*, proportional to d_e) in the target excess risk between learning with the DAC regularization (*i.e.*, $\widehat{\boldsymbol{\theta}}^{dac}$) and with the ERM on augmented training set (*i.e.*, $\widehat{\boldsymbol{\theta}}^{da-erm}$).

Example 4. We consider P^s and P^t that follow the same set of relations in Figure B.1, except for the distributions over $\mathbf{e}$ where $P^s(\mathbf{e}) \neq P^t(\mathbf{e})$. Precisely, let the environmental feature $\boldsymbol{\zeta}_e$ depend on $(\boldsymbol{\zeta}_{iv}, y, \mathbf{e})$:

$$\boldsymbol{\zeta}_e = \operatorname{sign}\left(y - (\boldsymbol{\theta}^*)^{\top}\mathbf{S}_{iv}\boldsymbol{\zeta}_{iv}\right)\mathbf{e} = \operatorname{sign}(z)\mathbf{e}, \quad z \sim \mathcal{N}(0,\sigma^2), \quad z \perp \mathbf{e},$$

where $\mathbf{e} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_{d_e})$ for $P^s(\mathbf{e})$ and $\mathbf{e} \sim \mathcal{N}(\mathbf{0}, \sigma_t^2\mathbf{I}_{d_e})$ for $P^t(\mathbf{e})$, $\sigma_t \geq c_{t,e}$ (recall $c_{t,e}$ from Assumption 3). Assume that the training set $\mathbf{X}$ is sufficiently large, $n \gg d_e + \log(1/\delta)$ for some given $\delta \in (0,1)$. Augmenting $\mathbf{X}$ with a simple by common type of data augmentations – the linear transforms, we let

$$\widetilde{\mathcal{A}}(\mathbf{X}) = [\mathbf{X}; (\mathbf{X}\mathbf{A}_1); \dots; (\mathbf{X}\mathbf{A}_\alpha)], \quad \mathbf{A}_j = \mathbf{P}_{iv} + \mathbf{u}_j\mathbf{v}_j^{\top}, \quad \mathbf{u}_j, \mathbf{v}_j \in \operatorname{col}(\mathbf{S}_e) \quad \forall\, j \in [\alpha],$$

and define

$$\nu_1 \triangleq \max\left\{1\right\} \cup \left\{\sigma_{\max}(\mathbf{A}_j) \mid j \in [\alpha]\right\} \quad \text{and} \quad \nu_2 \triangleq \sigma_{\min}\left(\frac{1}{1+\alpha}\left(\mathbf{I}_d + \sum_{j=1}^{\alpha} \mathbf{A}_k\right)\right),$$

where $\sigma_{\min}(\cdot)$ and $\sigma_{\max}(\cdot)$ refer to the minimum and maximum singular values, respectively. Then under Assumption 3 and Assumption 4, with constant probability,

$$\mathbb{E}_{P^s}\left[L_t(\widehat{\boldsymbol{\theta}}^{da-erm}) - L_t(\boldsymbol{\theta}^*)\right] \gtrsim \mathbb{E}_{P^s}\left[L_t(\widehat{\boldsymbol{\theta}}^{dac}) - L_t(\boldsymbol{\theta}^*)\right] + c_{t,e} \cdot \frac{\sigma^2 d_e}{2N}.$$

Proof of Example 4. With the specified distribution, for $\mathbf{E} = [\mathbf{e}_1; \dots; \mathbf{e}_N] \in \mathbb{R}^{N \times d_e}$,

$$\widehat{\boldsymbol{\Sigma}}_{\widetilde{\mathcal{A}}(\mathbf{X}_e)} = \frac{1}{(1+\alpha)N}\mathbf{S}_e\left(\mathbf{E}^\top\mathbf{E} + \sum_{j=1}^{\alpha}\mathbf{A}_j^\top\mathbf{E}^\top\mathbf{E}\mathbf{A}_j\right)\mathbf{S}_e^\top \preccurlyeq \frac{\nu_1^2}{N}\mathbf{S}_e\mathbf{E}^\top\mathbf{E}\mathbf{S}_e^\top,$$

$$\frac{1}{(1+\alpha)N}\widetilde{\mathcal{A}}(\mathbf{X}_e)^\top\widetilde{\mathbf{M}}\mathbf{z} = \left(\frac{1}{1+\alpha}\left(\mathbf{I}_d + \sum_{j=1}^{\alpha}\mathbf{A}_j\right)\right)^\top \frac{1}{N}\mathbf{S}_e\mathbf{E}^\top\left|\mathbf{z}\right|.$$

By Lemma B.12, under Assumption 3 and Assumption 4, we have that with high probability, $0.9\mathbf{I}_{d_e} \preccurlyeq \frac{1}{N}\mathbf{E}^\top\mathbf{E} \preccurlyeq 1.1\mathbf{I}_{d_e}$. Therefore with $\mathbf{E}$ and $\mathbf{z}$ being independent,

$$\begin{aligned}
\mathrm{EER}_e \;&=\; \mathbb{E}_{P^s}\left[\frac{1}{2}\|\frac{1}{(1+\alpha)N}\widehat{\boldsymbol{\Sigma}}^\dagger_{\widetilde{\mathcal{A}}(\mathbf{X}_e)}\widetilde{\mathcal{A}}(\mathbf{X}_e)^\top\widetilde{\mathbf{M}}\mathbf{z}\|_2^2\right] \\
&\geq \frac{\sigma^2}{2N}\frac{\nu_2^2}{\nu_1^4}\operatorname{tr}\left(\mathbb{E}_{P^s}\left[\left(\frac{1}{N}\mathbf{S}_e\mathbf{E}^\top\mathbf{E}\mathbf{S}_e^\top\right)^\dagger\right]\right) \\
&\gtrsim \frac{\sigma^2}{2N}\frac{\nu_2^2}{\nu_1^4}d_e \\
&\gtrsim \frac{\sigma^2 d_e}{2N},
\end{aligned}$$

and the rest follows from Theorem B.11. ■

B.5 Technical Lemmas

Lemma B.12. *We consider a random vector* $\mathbf{x} \in \mathbb{R}^d$ *with* $\mathbb{E}[\mathbf{x}] = \mathbf{0}$, $\mathbb{E}[\mathbf{x}\mathbf{x}^\top] = \boldsymbol{\Sigma}$, *and* $\overline{\mathbf{x}} = \boldsymbol{\Sigma}^{-1/2}\mathbf{x}$ [4] *being* ρ^2*-subgaussian. Given an* i.i.d. *sample of* $\mathbf{x}$, $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_n]^\top$, *for any* $\delta \in (0,1)$, *if* $n \gg \rho^4 d$, *then* $0.9\boldsymbol{\Sigma} \preccurlyeq \frac{1}{n}\mathbf{X}^\top\mathbf{X} \preccurlyeq 1.1\boldsymbol{\Sigma}$ *with high probability.*

Proof. We first denote $\mathbf{P}_{\mathcal{X}} \triangleq \boldsymbol{\Sigma}\boldsymbol{\Sigma}^\dagger$ as the orthogonal projector onto the subspace $\mathcal{X} \subseteq \mathbb{R}^d$ supported by the distribution of $\mathbf{x}$. With the assumptions $\mathbb{E}[\mathbf{x}] = \mathbf{0}$ and $\mathbb{E}[\mathbf{x}\mathbf{x}^\top] = \boldsymbol{\Sigma}$, we

[4] In the case where $\boldsymbol{\Sigma}$ is rank-deficient, we slightly abuse the notation such that $\boldsymbol{\Sigma}^{-1/2}$ and $\boldsymbol{\Sigma}^{-1}$ refer to the respective pseudo-inverses.

observe that $\mathbb{E}\left[\overline{\mathbf{x}}\right] = \mathbf{0}$ and $\mathbb{E}\left[\overline{\mathbf{x}}\overline{\mathbf{x}}^\top\right] = \mathbb{E}\left[\mathbf{x}\boldsymbol{\Sigma}^{-1}\mathbf{x}^\top\right] = \mathbf{P}_{\mathcal{X}}$. Given the sample set $\mathbf{X}$ of size $n \gg \rho^4\left(d + \log(1/\delta)\right)$ for some $\delta \in (0,1)$, we let $\mathbf{U} = \frac{1}{n}\sum_{i=1}^n \mathbf{x}_i\boldsymbol{\Sigma}^{-1}\mathbf{x}_i^\top - \mathbf{P}_{\mathcal{X}}$. Then the problem can be reduced to showing that, with probability at least $1-\delta$, $\|\mathbf{U}\|_2 \leq 0.1$. For this, we leverage the ϵ-net argument as following.

For an arbitrary $\mathbf{v} \in \mathcal{X} \cap \mathbb{S}^{d-1}$, we have

$$\mathbf{v}^\top\mathbf{U}\mathbf{v} = \frac{1}{n}\sum_{i=1}^n \left(\mathbf{v}^\top\mathbf{x}_i\boldsymbol{\Sigma}^{-1}\mathbf{x}_i^\top\mathbf{v} - 1\right) = \frac{1}{n}\sum_{i=1}^n \left(\left(\mathbf{v}^\top\overline{\mathbf{x}}_i\right)^2 - 1\right),$$

where, given $\overline{\mathbf{x}}_i$ being ρ^2-subgaussian, $\mathbf{v}^\top\overline{\mathbf{x}}_i$ is ρ^2-subgaussian. Since

$$\mathbb{E}\left[\left(\mathbf{v}^\top\overline{\mathbf{x}}_i\right)^2\right] = \mathbf{v}^\top\mathbb{E}\left[\overline{\mathbf{x}}_i\overline{\mathbf{x}}_i^\top\right]\mathbf{v} = 1,$$

we know that $\left(\mathbf{v}^\top\overline{\mathbf{x}}_i\right)^2 - 1$ is $16\rho^2$-subexponential. Then, we recall the Bernstein's inequality,

$$\mathbb{P}\left[\left|\mathbf{v}^\top\mathbf{U}\mathbf{v}\right| > \epsilon\right] \leq 2\exp\left(-\frac{n}{2}\min\left(\frac{\epsilon^2}{\left(16\rho^2\right)^2}, \frac{\epsilon}{16\rho^2}\right)\right).$$

Let $N \subset \mathcal{X} \cap \mathbb{S}^{d-1}$ be an ϵ_1-net such that $|N| = e^{O(d)}$. Then for some $0 < \epsilon_2 \leq 16\rho^2$, by the union bound,

$$\begin{aligned}\mathbb{P}\left[\max_{\mathbf{v}\in N} : \left|\mathbf{v}^\top\mathbf{U}\mathbf{v}\right| > \epsilon_2\right] &\leq 2\left|N\right|\exp\left(-\frac{n}{2}\min\left(\frac{\epsilon_2^2}{\left(16\rho^2\right)^2}, \frac{\epsilon_2}{16\rho^2}\right)\right)\\ &\leq \exp\left(O\left(d\right) - \frac{n}{2}\cdot\frac{\epsilon_2^2}{\left(16\rho^2\right)^2}\right) \leq \delta\end{aligned}$$

whenever $n > \frac{2\left(16\rho^2\right)^2}{\epsilon_2^2}\left(\Theta\left(d\right) + \log\frac{1}{\delta}\right)$. By taking $\delta = \exp\left(-\frac{1}{4}\left(\frac{\epsilon_2}{16\rho^2}\right)^2 n\right)$, we have that $\max_{\mathbf{v}\in N}\left|\mathbf{v}^\top\mathbf{U}\mathbf{v}\right| \leq \epsilon_2$ with high probability when $n > 4\left(\frac{16\rho^2}{\epsilon_2}\right)^2\Theta\left(d\right)$, and taking $n \gg \rho^4 d$ is sufficient.

Now for any $\mathbf{v} \in \mathcal{X} \cap \mathbb{S}^{d-1}$, there exists some $\mathbf{v}' \in N$ such that $\|\mathbf{v} - \mathbf{v}'\|_2 \leq \epsilon_1$. Therefore,

$$\begin{aligned}\left|\mathbf{v}^\top\mathbf{U}\mathbf{v}\right| &= \left|\mathbf{v}'^\top\mathbf{U}\mathbf{v}' + 2\mathbf{v}'^\top\mathbf{U}\left(\mathbf{v} - \mathbf{v}'\right) + \left(\mathbf{v} - \mathbf{v}'\right)^\top\mathbf{U}\left(\mathbf{v} - \mathbf{v}'\right)\right|\\ &\leq \left(\max_{\mathbf{v}\in N} : \left|\mathbf{v}^\top\mathbf{U}\mathbf{v}\right|\right) + 2\|\mathbf{U}\|_2\|\mathbf{v}'\|_2\|\mathbf{v} - \mathbf{v}'\|_2 + \|\mathbf{U}\|_2\|\mathbf{v} - \mathbf{v}'\|_2^2\\ &\leq \left(\max_{\mathbf{v}\in N} : \left|\mathbf{v}^\top\mathbf{U}\mathbf{v}\right|\right) + \|\mathbf{U}\|_2\left(2\epsilon_1 + \epsilon_1^2\right).\end{aligned}$$

Taking the supremum over $\mathbf{v} \in \mathbb{S}^{d-1}$, with probability at least $1-\delta$,

$$\max_{\mathbf{v}\in\mathcal{X}\cap\,\mathbb{S}^{d-1}} : \left|\mathbf{v}^\top\mathbf{U}\mathbf{v}\right| = \|\mathbf{U}\|_2 \leq \epsilon_2 + \|\mathbf{U}\|_2\left(2\epsilon_1 + \epsilon_1^2\right), \qquad \|\mathbf{U}\|_2 \leq \frac{\epsilon_2}{2 - \left(1 + \epsilon_1\right)^2}.$$

With $\epsilon_1 = \frac{1}{3}$ and $\epsilon_2 = \frac{1}{45}$, we have $\frac{\epsilon_2}{2-(1+\epsilon_1)^2} = \frac{1}{10}$.

Overall, if $n \gg \rho^4 d$, then with high probability, we have $\|\mathbf{U}\|_2 \leq 0.1$. ■

Lemma B.13. *Let $U \subseteq \mathbb{R}^d$ be an arbitrary subspace in $\mathbb{R}^d$, and $\mathbf{g} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_d)$ be a Gaussian random vector. Then for any continuous and C_l-Lipschitz function $\varphi : \mathbb{R} \to \mathbb{R}$ (i.e., $|\varphi(u) - \varphi(u')| \leq C_l \cdot |u - u'|$ for all $u, u' \in \mathbb{R}$),*

$$\mathbb{E}_{\mathbf{g}}\left[\sup_{\mathbf{u}\in U} \mathbf{g}^\top \varphi(\mathbf{u})\right] \leq C_l \cdot \mathbb{E}_{\mathbf{g}}\left[\sup_{\mathbf{u}\in U} \mathbf{g}^\top \mathbf{u}\right],$$

where φ acts on $\mathbf{u}$ entry-wisely, $(\varphi(\mathbf{u}))_j = \varphi(u_j)$. In other words, the Gaussian width of the image set $\varphi(U) \triangleq \left\{\varphi(\mathbf{u}) \in \mathbb{R}^d \mid \mathbf{u} \in U\right\}$ is upper bounded by that of U scaled by the Lipschitz constant.

Proof.

$$\begin{aligned}
\mathbb{E}_{\mathbf{g}}\left[\sup_{\mathbf{u}\in U} \mathbf{g}^\top \varphi(\mathbf{u})\right] =& \frac{1}{2}\mathbb{E}_{\mathbf{g}}\left[\sup_{\mathbf{u}\in U} \mathbf{g}^\top \varphi(\mathbf{u}) + \sup_{\mathbf{u}'\in U} \mathbf{g}^\top \varphi(\mathbf{u})\right] \\
=& \frac{1}{2}\mathbb{E}_{\mathbf{g}}\left[\sup_{\mathbf{u},\mathbf{u}'\in U} \mathbf{g}^\top \left(\varphi(\mathbf{u}) - \varphi(\mathbf{u}')\right)\right] \\
\leq& \frac{1}{2}\mathbb{E}_{\mathbf{g}}\left[\sup_{\mathbf{u},\mathbf{u}'\in U} \sum_{j=1}^{d} |g_j| \left|\varphi(u_j) - \varphi(u'_j)\right|\right] \qquad \text{(since } \varphi \text{ is } C_l\text{-Lipschitz)} \\
\leq& \frac{C_l}{2}\mathbb{E}_{\mathbf{g}}\left[\sup_{\mathbf{u},\mathbf{u}'\in U} \sum_{j=1}^{d} |g_j| \left|u_j - u'_j\right|\right] \\
=& \frac{C_l}{2}\mathbb{E}_{\mathbf{g}}\left[\sup_{\mathbf{u},\mathbf{u}'\in U} \mathbf{g}^\top \left(\mathbf{u} - \mathbf{u}'\right)\right] \\
=& \frac{C_l}{2}\mathbb{E}_{\mathbf{g}}\left[\sup_{\mathbf{u}\in U} \mathbf{g}^\top \mathbf{u} + \sup_{\mathbf{u}'\in U} \mathbf{g}^\top \left(-\mathbf{u}'\right)\right] \\
=& C_l \cdot \mathbb{E}_{\mathbf{g}}\left[\sup_{\mathbf{u}\in U} \mathbf{g}^\top \mathbf{u}\right]
\end{aligned}$$

■

B.6 Experiment Details

In this section, we provide the details of our experiments. Our code is adapted from the publicly released repo: https://github.com/kekmodel/FixMatch-pytorch.

Dataset: Our training dataset is derived from CIFAR-100, where the original dataset contains 50,000 training samples of 100 different classes. Out of the original 50,000 samples, we randomly select 10,000 labeled data as training set (i.e., 100 labeled samples per class). To see the impact of different training samples, we also trained our model with dataset that contains 1,000 and 20,000 samples. Evaluations are done on standard test set of CIFAR-100, which contains 10,000 testing samples.

Data Augmentation: During the training time, given a training batch, we generate corresponding augmented samples by RandAugment [35]. We set the number of augmentations per sample to 7, unless otherwise mentioned.

To generate an augmented image, the RandAugment draws n transformations uniformaly at random from 14 different augmentations, namely {identity, autoContrast, equalize, rotate, solarize, color, posterize, contrast, brightness, sharpness, shear-x, shear-y, translate-x, translate-y}. The RandAugment provides each transformation with a single scalar (1 to 10) to control the strength of each of them, which we always set to 10 for all transformations. By default, we set $n = 2$ (i.e., using 2 random transformations to generate an augmented sample). To see the impact of different augmentation strength, we choose $n \in \{1, 2, 5, 10\}$. Examples of augmented samples are shown in Figure 4.3.

Parameter Setting: The batch size is set to 64 and the entire training process takes 2^{15} steps. During the training, we adopt the SGD optimizer with momentum set to 0.9, with learning rate for step i being $0.03 \times \cos\left(\frac{i \times 7\pi}{2^{15} \times 16}\right)$.

Additional Settings for the semi-supervised learning results: For the results on semi-supervised learning, besides the 10,000 labeled samples, we also draw additionally samples (ranging from 5,000 to 20,000) from the training set of the original CIFAR-100. We remove the labels of those additionally sampled images, as they serve as “unlabeled" samples in the semi-supervised learning setting. The FixMatch implementation follows the publicly available on in https://github.com/kekmodel/FixMatch-pytorch.

Appendix C

Appendix for Chapter 5

C.1 Separation of Label-sparse and Label-dense Samples

Proof of Proposition 5.1. We first observe that, since $\ell_{CE}\left(\theta; (\mathbf{x}, \mathbf{y})\right)$ and $\ell_{AC}\left(\theta; \mathbf{x}, A_1, A_2\right)$ are convex and continuous in θ for all $(\mathbf{x}, \mathbf{y}) \in \mathcal{X} \times \mathcal{Y}$ and $A_1, A_2 \sim \mathcal{A}^2$, for all $i \in [n]$, $\widehat{L}^{WAC}_i\left(\theta, \boldsymbol{\beta}\right)$ is continuous, convex in θ, and affine (thus concave) in $\boldsymbol{\beta}$; and therefore so is $\widehat{L}^{WAC}\left(\theta, \boldsymbol{\beta}\right)$. Then with the compact and convex domains $\theta \in \mathcal{F}_{\theta^*}\left(\gamma\right)$ and $\boldsymbol{\beta} \in [0,1]^n$, Sion's minimax theorem [134] suggests the minimax equality,

$$\min_{\theta \in \mathcal{F}_{\theta^*}(\gamma)} \max_{\boldsymbol{\beta} \in [0,1]^n} \widehat{L}^{WAC}\left(\theta, \boldsymbol{\beta}\right) = \max_{\boldsymbol{\beta} \in [0,1]^n} \min_{\theta \in \mathcal{F}_{\theta^*}(\gamma)} \widehat{L}^{WAC}\left(\theta, \boldsymbol{\beta}\right), \tag{C.1}$$

where $\inf, \sup$ can be replaced by $\min, \max$ respectively due to compactness of the domains.

Further, by the continuity and convexity-concavity of $\widehat{L}^{WAC}\left(\theta, \boldsymbol{\beta}\right)$, the pointwise maximum $\max_{\boldsymbol{\beta} \in [0,1]^n} \widehat{L}^{WAC}\left(\theta, \boldsymbol{\beta}\right)$ is lower semi-continuous and convex in θ while the pointwise minimum $\min_{\theta \in \mathcal{F}_{\theta^*}(\gamma)} \widehat{L}^{WAC}\left(\theta, \boldsymbol{\beta}\right)$ is upper semi-continuous and concave in $\boldsymbol{\beta}$. Then via Weierstrass' theorem ([14], Proposition 3.2.1), there exist $\widehat{\theta}^{WAC} \in \mathcal{F}_{\theta^*}\left(\gamma\right)$ and $\widehat{\boldsymbol{\beta}} \in [0,1]^n$ that achieve the minimax optimal by minimizing $\max_{\boldsymbol{\beta} \in [0,1]^n} \widehat{L}^{WAC}\left(\theta, \boldsymbol{\beta}\right)$ and maximizing $\min_{\theta \in \mathcal{F}_{\theta^*}(\gamma)} \widehat{L}^{WAC}\left(\theta, \boldsymbol{\beta}\right)$. Along with (C.1), such $\left(\widehat{\theta}^{WAC}, \widehat{\boldsymbol{\beta}}\right)$ provides a saddle point for (5.5) ([14], Proposition 3.4.1).

Next, we show via contradiction that there exists a saddle point with $\widehat{\boldsymbol{\beta}}$ attained on a vertex $\widehat{\boldsymbol{\beta}} \in \{0,1\}^n$. Suppose the opposite, then for any saddle point $\left(\widehat{\theta}^{WAC}, \widehat{\boldsymbol{\beta}}\right)$, there must be an $i \in [n]$ with $\widehat{\boldsymbol{\beta}}_{[i]} \in (0,1)$, where we have the following contradictions:

(i) If $\ell_{CE}\left(\widehat{\theta}^{WAC}; (\mathbf{x}_i, \mathbf{y}_i)\right) < \ell_{AC}\left(\widehat{\theta}^{WAC}; \mathbf{x}_i, A_{i,1}, A_{i,2}\right)$, decreasing $\widehat{\boldsymbol{\beta}}_{[i]} > 0$ to $\widehat{\boldsymbol{\beta}}'_{[i]} = 0$ leads to $\widehat{L}^{WAC}\left(\widehat{\theta}^{WAC}, \widehat{\boldsymbol{\beta}}'\right) > \widehat{L}^{WAC}\left(\widehat{\theta}^{WAC}, \widehat{\boldsymbol{\beta}}\right)$, contradicting (5.6).

(ii) If $\ell_{CE}\left(\widehat{\theta}^{WAC}; (\mathbf{x}_i, \mathbf{y}_i)\right) > \ell_{AC}\left(\widehat{\theta}^{WAC}; \mathbf{x}_i, A_{i,1}, A_{i,2}\right)$, increasing $\widehat{\boldsymbol{\beta}}_{[i]} < 1$ to $\widehat{\boldsymbol{\beta}}'_{[i]} = 1$ again leads to $\widehat{L}^{WAC}\left(\widehat{\theta}^{WAC}, \widehat{\boldsymbol{\beta}}'\right) > \widehat{L}^{WAC}\left(\widehat{\theta}^{WAC}, \widehat{\boldsymbol{\beta}}\right)$, contradicting (5.6).

(iii) If $\ell_{CE}\left(\widehat{\theta}^{WAC}; (\mathbf{x}_i, \mathbf{y}_i)\right) = \ell_{AC}\left(\widehat{\theta}^{WAC}; \mathbf{x}_i, A_{i,1}, A_{i,2}\right)$, $\widehat{\boldsymbol{\beta}}_{[i]}$ can be replaced with any value in $[0,1]$, including $0, 1$.

Therefore, there must be a saddle point $\left(\widehat{\theta}^{WAC}, \widehat{\boldsymbol{\beta}}\right)$ with $\widehat{\boldsymbol{\beta}} \in \{0,1\}^n$ such that

$$\widehat{\boldsymbol{\beta}}_{[i]} = \mathbb{I}\left\{\ell_{CE}\left(\widehat{\theta}^{WAC}; (\mathbf{x}_i, \mathbf{y}_i)\right) > \ell_{AC}\left(\widehat{\theta}^{WAC}; \mathbf{x}_i, A_{i,1}, A_{i,2}\right)\right\}.$$

Finally, it remains to show that *w.h.p.* over $\left\{(\mathbf{x}_i, \mathbf{y}_i)\right\}_{i \in [n]} \sim P_\xi^n$ and $\left\{(A_{i,1}, A_{i,2})\right\}_{i \in [n]} \sim \mathcal{A}^{2n}$,

(i) $\ell_{CE}\left(\widehat{\theta}^{WAC}; (\mathbf{x}_i, \mathbf{y}_i)\right) \leq \ell_{AC}\left(\widehat{\theta}^{WAC}; \mathbf{x}_i, A_{i,1}, A_{i,2}\right)$ for all $(\mathbf{x}_i, \mathbf{y}_i) \sim P_0$; and

(ii) $\ell_{CE}\left(\widehat{\theta}^{WAC}; (\mathbf{x}_i, \mathbf{y}_i)\right) > \ell_{AC}\left(\widehat{\theta}^{WAC}; \mathbf{x}_i, A_{i,1}, A_{i,2}\right)$ for all $(\mathbf{x}_i, \mathbf{y}_i) \sim P_1$;

which leads to $\boldsymbol{\beta}_{[i]} = \mathbb{I}\left\{(\mathbf{x}_i, \mathbf{y}_i) \sim P_1\right\}$ *w.h.p.* as desired. To illustrate this, we begin by observing that when P_0 and P_1 are n-separated (Assumption 1), since $\widehat{\theta}^{WAC} \in \mathcal{F}_{\theta^*}(\gamma)$, there exists some $\omega > 0$ such that for each $i \in [n]$,

$$\mathbb{P}\left[\ell_{CE}\left(\widehat{\theta}^{WAC}; (\mathbf{x}_i, \mathbf{y}_i)\right) < \ell_{AC}\left(\widehat{\theta}^{WAC}; \mathbf{x}_i, A_{i,1}, A_{i,2}\right) \,\middle|\, (\mathbf{x}_i, \mathbf{y}_i) \sim P_0\right] \geq 1 - \frac{1}{\Omega\left(n^{1+\omega}\right)},$$

and

$$\mathbb{P}\left[\ell_{CE}\left(\widehat{\theta}^{WAC}; (\mathbf{x}_i, \mathbf{y}_i)\right) > \ell_{AC}\left(\widehat{\theta}^{WAC}; \mathbf{x}_i, A_{i,1}, A_{i,2}\right) \,\middle|\, (\mathbf{x}_i, \mathbf{y}_i) \sim P_1\right] \geq 1 - \frac{1}{\Omega\left(n^{1+\omega}\right)}.$$

Therefore by the union bound over the set of n samples $\left\{(\mathbf{x}_i, \mathbf{y}_i)\right\}_{i \in [n]} \sim P_\xi^n$,

$$\mathbb{P}\left[\ell_{CE}\left(\widehat{\theta}^{WAC}; (\mathbf{x}_i, \mathbf{y}_i)\right) < \ell_{AC}\left(\widehat{\theta}^{WAC}; \mathbf{x}_i, A_{i,1}, A_{i,2}\right) \;\forall\; (\mathbf{x}_i, \mathbf{y}_i) \sim P_0\right] \geq 1 - \frac{1}{\Omega\left(n^{\omega}\right)}, \tag{C.2}$$

and

$$\mathbb{P}\left[\ell_{CE}\left(\widehat{\theta}^{WAC}; (\mathbf{x}_i, \mathbf{y}_i)\right) > \ell_{AC}\left(\widehat{\theta}^{WAC}; \mathbf{x}_i, A_{i,1}, A_{i,2}\right) \;\forall\; (\mathbf{x}_i, \mathbf{y}_i) \sim P_1\right] \geq 1 - \frac{1}{\Omega\left(n^{\omega}\right)}. \tag{C.3}$$

Applying the union bound again on (C.2) and (C.3), we have the desired condition holds with probability $1 - \Omega\left(n^{\omega}\right)^{-1}$, *i.e.*, *w.h.p.*. ■

C.2 Convergence of *AdaWAC*

Recall the underlying function class $\mathcal{F} \ni f_\theta$ parameterized by some $\theta \in \mathcal{F}_\theta$ that we aim to learn for the pixel-wise classifier $h_\theta = \mathrm{argmax}_{k \in [K]} f_\theta(\mathbf{x})_{[j,:]}$, $j \in [d]$:

$$\mathcal{F} = \left\{ f_\theta = \phi_\theta \circ \psi_\theta \,\middle|\, \phi_\theta : \mathcal{X} \to \mathcal{Z},\; \psi_\theta : \mathcal{Z} \to [0,1]^{d \times K} \right\}, \tag{C.4}$$

where ϕ_θ, ψ_θ correspond to the encoder and decoder functions. Formally, we consider an inner product space of parameters $(\mathcal{F}_\theta, \langle \cdot, \cdot \rangle_{\mathcal{F}})$ with the induced norm $\|\cdot\|_{\mathcal{F}}$ and dual norm $\|\cdot\|_{\mathcal{F},*}$.

For any $d \in \mathbb{N}$, let $\Delta_d^n \triangleq \left\{ [\boldsymbol{\beta}_1; \dots; \boldsymbol{\beta}_n] \in [0,1]^{n \times d} \,\middle|\, \|\boldsymbol{\beta}_i\|_1 = 1 \;\forall\; i \in [n] \right\}$. Then (5.5) can be reformulated as:

$$\begin{aligned} \widehat{\theta}^{WAC}, \widehat{\boldsymbol{\beta}} &= \underset{\theta \in \mathcal{F}_{\theta^*}(\gamma)}{\mathrm{argmin}} \; \underset{\mathbf{B} \in \Delta_2^n}{\mathrm{argmax}} \left\{ \widehat{L}^{WAC}(\theta, \mathbf{B}) \triangleq \frac{1}{n} \sum_{i=1}^{n} \widehat{L}_i^{WAC}(\theta, \mathbf{B}) \right\}, \\ \widehat{L}_i^{WAC}(\theta, \mathbf{B}) &\triangleq \mathbf{B}_{[i,1]} \cdot \ell_{CE}(\theta; (\mathbf{x}_i, \mathbf{y}_i)) + \mathbf{B}_{[i,2]} \cdot \ell_{AC}(\theta; \mathbf{x}_i, A_{i,1}, A_{i,2}). \end{aligned} \tag{C.5}$$

Proposition C.1 (Convergence (formal restatement of Proposition 5.2))**.** *Assume that* $\ell_{CE}(\theta; (\mathbf{x}, \mathbf{y}))$ *and* $\ell_{AC}(\theta; \mathbf{x}, A_1, A_2)$ *are convex and continuous in* θ *for all* $(\mathbf{x}, \mathbf{y}) \in \mathcal{X} \times \mathcal{Y}$ *and* $A_1, A_2 \sim \mathcal{A}^2$*, and that* $\mathcal{F}_{\theta^*}(\gamma) \subset \mathcal{F}_\theta$ *is convex and compact. If there exist*

(i) $C_{\theta,*} > 0$ *such that* $\frac{1}{n}\sum_{i=1}^{n} \left\|\nabla_\theta \widehat{L}_i^{WAC}(\theta, \mathbf{B})\right\|_{\mathcal{F},*}^2 \le C_{\theta,*}^2$ *for all* $\theta \in \mathcal{F}_{\theta^*}(\gamma)$*,* $\mathbf{B} \in \Delta_2^n$ *and*

(ii) $C_{\mathbf{B},*} > 0$ *such that* $\frac{1}{n}\sum_{i=1}^{n} \max\left\{\ell_{CE}(\theta; (\mathbf{x}_i, \mathbf{y}_i)), \ell_{AC}(\theta; \mathbf{x}_i, A_{i,1}, A_{i,2})\right\}^2 \le C_{\mathbf{B},*}^2$ *for all* $\theta \in \mathcal{F}_{\theta^*}(\gamma)$*,*

then with $\eta_\theta = \eta_{\mathbf{B}} = 2 \Big/ \sqrt{5T\left(\gamma^2 C_{\theta,*}^2 + 2nC_{\mathbf{B},*}^2\right)}$*, Algorithm 6 provides the convergence guarantee for the duality gap* $\mathcal{E}\left(\overline{\theta}_T, \overline{\mathbf{B}}_T\right) \triangleq \max_{\mathbf{B}\in\Delta_2^n} \widehat{L}^{WAC}\left(\overline{\theta}_T, \mathbf{B}\right) - \min_{\theta\in\mathcal{F}_{\theta^*}(\gamma)} \widehat{L}^{WAC}\left(\theta, \overline{\mathbf{B}}_T\right)$*:*

$$\mathbb{E}\left[\mathcal{E}\left(\overline{\theta}_T, \overline{\mathbf{B}}_T\right)\right] \le 2\sqrt{\frac{5\left(\gamma^2 C_{\theta,*}^2 + 2nC_{\mathbf{B},*}^2\right)}{T}},$$

where $\overline{\theta}_T = \frac{1}{T}\sum_{t=1}^{T} \theta_t$ *and* $\overline{\mathbf{B}}_T = \frac{1}{T}\sum_{t=1}^{T} \mathbf{B}_t$*.*

Proof of Proposition C.1. The proof is an application of the standard convergence guarantee for the online mirror descent on saddle point problems, as recapitulated in Lemma C.2.

Specifically, for $\mathbf{B} \in \Delta_2^n$, we use the norm $\|\mathbf{B}\|_{1,2} \triangleq \sqrt{\sum_{i=1}^{n}\left(\sum_{j=1}^{2}\left|\mathbf{B}_{[i,j]}\right|\right)^2}$ with its dual norm $\|\mathbf{B}\|_{1,2,*} \triangleq \sqrt{\sum_{i=1}^{n}\left(\max_{j\in[2]}\left|\mathbf{B}_{[i,j]}\right|\right)^2}$. We consider a mirror map $\varphi_{\mathbf{B}} : [0,1]^{n\times 2} \to \mathbb{R}$ such that $\varphi_{\mathbf{B}}(\mathbf{B}) = \sum_{i=1}^{n}\sum_{j=1}^{2} \mathbf{B}_{[i,j]} \log \mathbf{B}_{[i,j]}$. We observe that, since $\mathbf{B}_{[i,:]}, \mathbf{B}'_{[i,:]} \in \Delta_2$ for all $i \in [n]$,

$$D_{\varphi_{\mathbf{B}}}(\mathbf{B}, \mathbf{B}') = \sum_{i=1}^{n}\sum_{j=1}^{2} \mathbf{B}_{[i,j]} \log \frac{\mathbf{B}_{[i,j]}}{\mathbf{B}'_{[i,j]}} \ge \frac{1}{2}\sum_{i=1}^{n}\left(\sum_{j=1}^{2}\left|\mathbf{B}_{[i,j]} - \mathbf{B}'_{[i,j]}\right|\right)^2 = \frac{1}{2}\|\mathbf{B} - \mathbf{B}'\|_{1,2}^2,$$

and therefore $\varphi_{\mathbf{B}}$ is 1-strongly convex with respect to $\|\cdot\|_{1,2}$. With such $\varphi_{\mathbf{B}}$, we have the associated Fenchel dual $\varphi_{\mathbf{B}}^*(\mathbf{G}) = \sum_{i=1}^{n} \log\left(\sum_{j=1}^{2} \exp\left(\mathbf{G}_{[i,j]}\right)\right)$, along with the gradients

$$\nabla\varphi_{\mathbf{B}}(\mathbf{B})_{[i,j]} = 1 + \log \mathbf{B}_{[i,j]}, \quad \nabla\varphi_{\mathbf{B}}^*(\mathbf{G})_{[i,j]} = \frac{\exp\left(\mathbf{G}_{[i,j]}\right)}{\sum_{j=1}^{2}\exp\left(\mathbf{G}_{[i,j]}\right)},$$

such that the mirror descent update on $\mathbf{B}$ is given by

$$\begin{aligned}\left(\mathbf{B}_{t+1}\right)_{[i,j]} =& \nabla\varphi_{\mathbf{B}}^*\left(\nabla\varphi_{\mathbf{B}}(\mathbf{B}_t) - \eta_{\mathbf{B}} \cdot \nabla_{\mathbf{B}} \widehat{L}_{i_t}^{WAC}(\theta_t, \mathbf{B}_t)\right) \\ =& \frac{\left(\mathbf{B}_t\right)_{[i,j]} \exp\left(\eta_{\mathbf{B}} \cdot \left(\nabla_{\mathbf{B}} \widehat{L}_{i_t}^{WAC}(\theta_t, \mathbf{B}_t)\right)_{[i,j]}\right)}{\sum_{j=1}^{2}\left(\mathbf{B}_t\right)_{[i,j]} \exp\left(\eta_{\mathbf{B}} \cdot \left(\nabla_{\mathbf{B}} \widehat{L}_{i_t}^{WAC}(\theta_t, \mathbf{B}_t)\right)_{[i,j]}\right)}.\end{aligned}$$

For $i_t \sim [n]$ uniformly, the stochastic gradient with respect to $\mathbf{B}$ satisfies

$$\begin{aligned}
&\mathbb{E}_{i_t \sim [n]}\left[\left\|\nabla_{\mathbf{B}} \widehat{L}_{i_t}^{WAC}\left(\theta_t, \mathbf{B}_t\right)\right\|_{1,2,*}^2\right] \\
=&\frac{1}{n} \sum_{i_t=1}^{n} \max\left\{\ell_{CE}\left(\theta_t; \left(\mathbf{x}_{i_t}, \mathbf{y}_{i_t}\right)\right), \ell_{AC}\left(\theta_t; \mathbf{x}_{i_t}, A_{i_t,1}, A_{i_t,2}\right)\right\}^2 \leq C_{\mathbf{B},*}^2.
\end{aligned}$$

Further, in the distance induced by $\varphi_{\mathbf{B}}$, we have

$$R_{\Delta_2^n}^2 \triangleq \max_{\mathbf{B} \in \Delta_2^n} \varphi_{\mathbf{B}}\left(\mathbf{B}\right) - \min_{\mathbf{B} \in \Delta_2^n} \varphi_{\mathbf{B}}\left(\mathbf{B}\right) = 0 - \sum_{i=1}^{n} \sum_{j=1}^{2} \frac{1}{2} \log \frac{1}{2} = n.$$

Meanwhile, for $\theta \in \mathcal{F}_{\theta^*}\left(\gamma\right)$, we consider the norm $\left\|\theta\right\|_{\mathcal{F}} \triangleq \sqrt{\left\langle \theta, \theta \right\rangle_{\mathcal{F}}}$ induced by the inner product that characterizes $\mathcal{F}_\theta$, with the associated dual norm $\left\|\cdot\right\|_{\mathcal{F},*}$. We use a mirror map $\varphi_\theta : \mathcal{F}_\theta \to \mathbb{R}$ such that $\varphi_\theta\left(\theta\right) = \frac{1}{2}\left\|\theta - \theta^*\right\|_{\mathcal{F}}^2$. By observing that

$$D_{\varphi_\theta}\left(\theta, \theta'\right) = \frac{1}{2}\left\|\theta - \theta'\right\|_{\mathcal{F}}^2 \quad \forall\, \theta, \theta' \in \mathcal{F}.$$

we have φ_θ being 1-strongly convex with respect to $\left\|\cdot\right\|_{\mathcal{F}}$. With the gradient of φ_θ, $\nabla\varphi_\theta(\theta) = \theta - \theta^*$, and that of its Fenchel dual $\nabla\varphi_\theta^*(g) = g + \theta^*$, at the $(t+1)$-th iteration, we have

$$\theta_{t+1} = \nabla\varphi_\theta^*\left(\nabla\varphi_\theta\left(\theta_t\right) - \eta_\theta \cdot \nabla_\theta \widehat{L}_{i_t}^{WAC}\left(\theta_t, \mathbf{B}_{t+1}\right)\right) = \theta_t - \eta_\theta \cdot \nabla_\theta \widehat{L}_{i_t}^{WAC}\left(\theta_t, \mathbf{B}_{t+1}\right).$$

For $i_t \sim [n]$ uniformly, the stochastic gradient with respect to f satisfies that

$$\mathbb{E}_{i_t \sim [n]}\left[\left\|\nabla_\theta \widehat{L}_{i_t}^{WAC}\left(\theta_t, \mathbf{B}_{t+1}\right)\right\|_{\mathcal{F},*}^2\right] = \frac{1}{n} \sum_{i_t=1}^{n} \left\|\nabla_\theta \widehat{L}_{i_t}^{WAC}\left(\theta_t, \mathbf{B}_{t+1}\right)\right\|_{\mathcal{F},*}^2 \leq C_{\theta,*}^2.$$

Further, in light of the definition of $\mathcal{F}_{\theta^*}\left(\gamma\right)$, since $\theta^* \in \mathcal{F}_{\theta^*}\left(\gamma\right)$, with $\theta^* = \operatorname{argmin}_{\theta \in \mathcal{F}_{\theta^*}(\gamma)} \varphi_\theta(\theta)$ and $\theta' = \operatorname{argmax}_{\theta \in \mathcal{F}_{\theta^*}(\gamma)} \varphi_\theta(\theta)$, we have

$$R_{\mathcal{F}_{\theta^*}(\gamma)}^2 \triangleq \max_{\theta \in \mathcal{F}_{\theta^*}(\gamma)} \varphi_\theta\left(\theta\right) - \min_{\theta \in \mathcal{F}_{\theta^*}(\gamma)} \varphi_\theta\left(\theta\right) = \frac{1}{2}\left\|\theta' - \theta^*\right\|_{\mathcal{F}}^2 \leq \frac{\gamma^2}{2}.$$

Finally, leveraging Lemma C.2 completes the proof. ■

We recall the standard convergence guarantee for online mirror descent on saddle point problems. In general, we consider a stochastic function $F : \mathcal{U} \times \mathcal{V} \times \mathcal{I} \to \mathbb{R}$ with the randomness of $F\left(u, v; i\right)$ on $i \in \mathcal{I}$. Overloading notation $\mathcal{I}$ both as the distribution of i and as the support, we are interested in solving the saddle point problem on the expectation function

$$\min_{u \in \mathcal{U}} \max_{v \in \mathcal{V}} f\left(u, v\right) \quad \text{where} \quad f\left(u, v\right) \triangleq \mathbb{E}_{i \sim \mathcal{I}}\left[F\left(u, v; i\right)\right]. \tag{C.6}$$

Assumption 5. Assume that the stochastic objective satisfies the following:

(i) For every $i \in \mathcal{I}$, $F(\cdot, v, i)$ is convex for all $v \in \mathcal{V}$ and $F(u, \cdot, i)$ is concave for all $u \in \mathcal{U}$.

(ii) The stochastic subgradients $G_u(u, v; i) \in \partial_u F(u, v; i)$ and $G_v(u, v; i) \in \partial_v F(u, v; i)$ with respect to u and v evaluated at any $(u, v) \in \mathcal{U} \times \mathcal{V}$ provide unbiased estimators for some respective subgradients of the expectation function: for any $(u, v) \in \mathcal{U} \times \mathcal{V}$, there exist some $g_u(u, v) \triangleq \mathbb{E}_{i\sim\mathcal{I}}[G_u(u, v; i)] \in \partial_u f(u, v)$ and $g_v(u, v) \triangleq \mathbb{E}_{i\sim\mathcal{I}}[G_v(u, v; i)] \in \partial_v f(u, v)$.

(iii) Let $\|\cdot\|_{\mathcal{U}}$ and $\|\cdot\|_{\mathcal{V}}$ be arbitrary norms that are well-defined on $\mathcal{U}$ and $\mathcal{V}$, while $\|\cdot\|_{\mathcal{U},*}$ and $\|\cdot\|_{\mathcal{V},*}$ be their respective dual norms. There exist constants $C_{u,*}, C_{v,*} > 0$ such that

$$\mathbb{E}_{i\sim\mathcal{I}}\left[\|G_u(u, v; i)\|_{\mathcal{U},*}^2\right] \le C_{u,*}^2 \quad \text{and} \quad \mathbb{E}_{i\sim\mathcal{I}}\left[\|G_v(u, v; i)\|_{\mathcal{V},*}^2\right] \le C_{v,*}^2 \quad \forall\ (u, v) \in \mathcal{U} \times \mathcal{V}.$$

For online mirror descent, we further introduce two mirror maps that induce distances on $\mathcal{U}$ and $\mathcal{V}$, respectively.

Assumption 6. Let $\varphi_u : \mathcal{D}_u \to \mathbb{R}$ and $\varphi_v : \mathcal{D}_v \to \mathbb{R}$ satisfy the following:

(i) $\mathcal{U} \subseteq \mathcal{D}_u \cup \partial\mathcal{D}_u$, $\mathcal{U} \cap \mathcal{D}_u \neq \emptyset$ and $\mathcal{V} \subseteq \mathcal{D}_v \cup \partial\mathcal{D}_v$, $\mathcal{V} \cap \mathcal{D}_v \neq \emptyset$.

(ii) φ_u is ρ_u-strongly convex with respect to $\|\cdot\|_{\mathcal{U}}$; φ_v is ρ_v-strongly convex with respect to $\|\cdot\|_{\mathcal{V}}$.

(iii) $\lim_{u\to\partial\mathcal{D}_u} \|\nabla\varphi_u(u)\|_{\mathcal{U},*} = \lim_{v\to\partial\mathcal{D}_v} \|\nabla\varphi_v(v)\|_{\mathcal{V},*} = +\infty$.

Given the learning rates η_u, η_v, in each iteration $t = 1, \dots, T$, the online mirror descent samples $i_t \sim \mathcal{I}$ and updates

$$\begin{aligned} v_{t+1} &= \underset{v\in\mathcal{V}}{\operatorname{argmin}}\ -\eta_v \cdot G_v(u_t, v_t; i_t)^\top v + D_{\varphi_v}(v, v_t), \\ u_{t+1} &= \underset{u\in\mathcal{U}}{\operatorname{argmin}}\ \eta_u \cdot G_u(u_t, v_{t+1}; i_t)^\top u + D_{\varphi_u}(u, u_t), \end{aligned} \tag{C.7}$$

where $D_\varphi(w, w') = \varphi(w) - \varphi(w') - \nabla\varphi(w')^\top(w - w')$ denotes the Bregman divergence.

We measure the convergence of the saddle point problem in the duality gap:

$$\mathcal{E}(\overline{u}_T, \overline{v}_T) \triangleq \max_{v\in\mathcal{V}} f(\overline{u}_T, v) - \min_{u\in\mathcal{U}} f(u, \overline{v}_T)$$

such that, with

$$R_{\mathcal{U}} \triangleq \sqrt{\max_{u\in\mathcal{U}\cap\mathcal{D}_u} \varphi_u(u) - \min_{u\in\mathcal{U}\cap\mathcal{D}_u} \varphi_u(u)} \quad \text{and} \quad R_{\mathcal{V}} \triangleq \sqrt{\max_{v\in\mathcal{V}\cap\mathcal{D}_v} \varphi_v(v) - \min_{v\in\mathcal{V}\cap\mathcal{D}_v} \varphi_v(v)},$$

the online mirror descent converges as follows.

Lemma C.2 ([112] (3.11))**.** *Under Assumption 5 and Assumption 6, when taking constant learning rates* $\eta_u = \eta_v = 2\Big/\sqrt{5T\left(\frac{2R_{\mathcal{U}}^2}{\rho_u}C_{u,*}^2 + \frac{2R_{\mathcal{V}}^2}{\rho_v}C_{v,*}^2\right)}$*, with* $\overline{u}_T = \frac{1}{T}\sum_{t=1}^{T} u_t$ *and* $\overline{v}_T = \frac{1}{T}\sum_{t=1}^{T} v_t$*,*

$$\mathbb{E}\left[\mathcal{E}\left(\overline{u}_T, \overline{v}_T\right)\right] \leq 2\sqrt{\frac{10\left(\rho_v R_{\mathcal{U}}^2 C_{u,*}^2 + \rho_u R_{\mathcal{V}}^2 C_{v,*}^2\right)}{\rho_u \rho_v \cdot T}}.$$

Example 5 (Binary linear pixel-wise classifiers with convex and continuous objectives). We consider a pixel-wise binary classification problem with $\mathcal{X} = [0,1]^d$, augmentations $A : \mathcal{X} \to \mathcal{X}$ for all $A \sim \mathcal{A}$, and a class of linear "UNets",

$$\mathcal{F} = \left\{ f_\theta : \mathcal{X} \to [0,1]^d \;\middle|\; f_\theta\left(\mathbf{x}\right) = \sigma\left(\boldsymbol{\theta}_d \boldsymbol{\theta}_e^\top \mathbf{x}\right) = \psi_\theta\left(\phi_\theta\left(\mathbf{x}\right)\right),\ \phi_\theta\left(\mathbf{x}\right) = \frac{1}{\sqrt{d}}\boldsymbol{\theta}_e^\top \mathbf{x} \right\},$$

where the parameter space $\theta = (\boldsymbol{\theta}_e, \boldsymbol{\theta}_d) \in \mathcal{F}_\theta = \mathbb{S}^{d-1} \times \mathbb{S}^{d-1}$ is equipped with the ℓ_2 norm $\|\theta\|_{\mathcal{F}} = \left(\|\boldsymbol{\theta}_e\|_2^2 + \|\boldsymbol{\theta}_d\|_2^2\right)^{1/2}$; $\sigma : \mathbb{R}^d \to [0,1]^d$ denotes entry-wise application of the sigmoid function $\sigma(z) = (1+e^{-z})^{-1}$; and the latent space of encoder outputs $(\mathcal{Z}, \varrho)$ is simply the real line. Given the data distribution P_ξ, we recall that $\theta^* = \operatorname{argmin}_{\theta \in \mathcal{F}_\theta} \mathbb{E}_{(\mathbf{x},\mathbf{y}) \sim P_\xi}\left[\ell_{CE}\left(\theta; (\mathbf{x}, \mathbf{y})\right)\right]$ for all $\xi \in [0,1]$ and let $\mathcal{F}_{\theta^*}\left(\gamma\right) = \left\{\theta \in \mathcal{F}_\theta \mid \|\theta - \theta^*\|_{\mathcal{F}} \leq \gamma\right\}$ for some $\gamma = O\left(1/\sqrt{d}\right)$. We assume that $\left|\mathbf{x}^\top \boldsymbol{\theta}_e^*\right| = O(1)$ for all $\mathbf{x} \in \mathcal{X}$. Then, $\ell_{CE}\left(\theta; (\mathbf{x}, \mathbf{y})\right)$ and $\ell_{AC}\left(\theta; \mathbf{x}, A_1, A_2\right)$ are convex and continuous in θ for all $(\mathbf{x}, \mathbf{y}) \in \mathcal{X} \times [K]^d$, $A_1, A_2 \sim \mathcal{A}^2$; while $C_{\theta,*} \leq \max\left(2\sqrt{2}, 2\lambda_{AC}\right)$ and $C_{\beta,*} \leq \max\left(O(1), 2\lambda_{AC}\right)$.

Rationale for Example 5. Let $\mathbf{y}_k = \mathbb{I}\left\{\mathbf{y} = k\right\}$ entry-wise for $k = 0, 1$. We would like to show that, for any given $(\mathbf{x}, \mathbf{y}) \in \mathcal{X} \times [K]^d$, $A_1, A_2 \sim \mathcal{A}^2$,

$$\begin{aligned}
\ell_{CE}\left(\theta\right) &= -\frac{1}{d}\left(\mathbf{y}_1^\top \log \sigma\left(\boldsymbol{\theta}_d \boldsymbol{\theta}_e^\top \mathbf{x}\right) + \mathbf{y}_0^\top \log \sigma\left(-\boldsymbol{\theta}_d \boldsymbol{\theta}_e^\top \mathbf{x}\right)\right), \\
\ell_{AC}\left(\theta\right) &= \frac{\lambda_{AC}}{\sqrt{d}} \cdot \left(A_1(\mathbf{x}) - A_2(\mathbf{x})\right)^\top \boldsymbol{\theta}_e
\end{aligned}$$

are convex and continuous in $\theta = (\boldsymbol{\theta}_e, \boldsymbol{\theta}_d)$.

First, we observe that $\ell_{AC}\left(\theta\right)$ is linear (and therefore convex and continuous) in θ for all $\mathbf{x} \in \mathcal{X}$, $A_1, A_2 \sim \mathcal{A}^2$, with

$$\nabla_{\boldsymbol{\theta}_e} \ell_{AC}\left(\theta\right) = \frac{\lambda_{AC}}{\sqrt{d}} \cdot \left(A_1(\mathbf{x}) - A_2(\mathbf{x})\right), \quad \nabla_{\boldsymbol{\theta}_d} \ell_{AC}\left(\theta\right) = \mathbf{0}$$

such that $\|\nabla_\theta \ell_{AC}\left(\theta\right)\|_{\mathcal{F},*} \leq 2\lambda_{AC}$.

Meanwhile, with $\mathbf{z}(\theta) = \boldsymbol{\theta}_d \boldsymbol{\theta}_e^\top \mathbf{x}$, we have $\ell_{CE}(\theta) = -\frac{1}{d}\left(\mathbf{y}_1^\top \log\sigma(\mathbf{z}(\theta)) + \mathbf{y}_0^\top \log\sigma(-\mathbf{z}(\theta))\right)$ being convex and continuous in $\mathbf{z}(\theta)$:

$$\nabla^2_{\mathbf{z}} \ell_{CE}(\theta) = \frac{1}{d}\operatorname{diag}(\sigma(\mathbf{z}(\theta)))\operatorname{diag}(1-\sigma(\mathbf{z}(\theta))) \succcurlyeq 0.$$

Therefore, $\ell_{CE}(\theta)$ is convex and continuous in θ for all $(\mathbf{x},\mathbf{y}) \in \mathcal{X}\times[K]^d$:

$$\underbrace{\nabla^2_\theta \ell_{CE}(\theta)}_{2d\times 2d} = \begin{bmatrix} \mathbf{x}\boldsymbol{\theta}_d^\top \\ \left(\boldsymbol{\theta}_e^\top\mathbf{x}\right)\mathbf{I}_d\end{bmatrix} \left(\frac{1}{d}\operatorname{diag}(\sigma(\mathbf{z}(\theta)))\operatorname{diag}(1-\sigma(\mathbf{z}(\theta)))\right)\begin{bmatrix}\mathbf{x}\boldsymbol{\theta}_d^\top & \left(\boldsymbol{\theta}_e^\top\mathbf{x}\right)\mathbf{I}_d\end{bmatrix} \succcurlyeq 0,$$

where $\mathbf{I}_d$ denotes the $d \times d$ identity matrix. Further, from the derivation, we have

$$\nabla_{\boldsymbol{\theta}_e}\ell_{CE}(\theta) = \frac{1}{d}\boldsymbol{\theta}_d^\top\left(\sigma\left(\boldsymbol{\theta}_d\boldsymbol{\theta}_e^\top\mathbf{x}\right) - \mathbf{y}\right)\mathbf{x}, \quad \nabla_{\boldsymbol{\theta}_d}\ell_{CE}(\theta) = \frac{\boldsymbol{\theta}_e^\top\mathbf{x}}{d}\left(\sigma\left(\boldsymbol{\theta}_d\boldsymbol{\theta}_e^\top\mathbf{x}\right) - \mathbf{y}\right)$$

such that $\|\nabla_\theta\ell_{CE}(\theta)\|_{\mathcal{F},*} = \sqrt{\|\nabla_{\boldsymbol{\theta}_e}\ell_{CE}(\theta)\|_2^2 + \|\nabla_{\boldsymbol{\theta}_d}\ell_{CE}(\theta)\|_2^2} \leq 2\sqrt{2}$.

Finally, knowing $\|\nabla_\theta\ell_{CE}(\theta)\|_{\mathcal{F},*} \leq 2\sqrt{2}$ and $\|\nabla_\theta\ell_{AC}(\theta)\|_{\mathcal{F},*} \leq 2\lambda_{AC}$, we have

$$\left\|\nabla_\theta \widehat{L}_i^{WAC}(\theta,\boldsymbol{\beta})\right\|_{\mathcal{F},*} \leq \boldsymbol{\beta}_{[i]}\|\nabla_\theta\ell_{CE}(\theta)\|_{\mathcal{F},*} + (1-\boldsymbol{\beta}_{[i]})\|\nabla_\theta\ell_{AC}(\theta)\|_{\mathcal{F},*} \leq \max\left(2\sqrt{2}, 2\lambda_{AC}\right)$$

for all $i \in [n]$, and therefore,

$$C_{\theta,*} \leq \max\left(2\sqrt{2}, 2\lambda_{AC}\right).$$

Besides, with

$$\ell_{AC}(\theta) \leq \frac{\lambda_{AC}}{\sqrt{d}}\|A_1(\mathbf{x}) - A_2(\mathbf{x})\|_2\|\boldsymbol{\theta}_e\|_2 \leq 2\lambda_{AC},$$

and since

$$\begin{aligned}\left(\boldsymbol{\theta}_d\boldsymbol{\theta}_e^\top\mathbf{x}\right)_{[j]} \leq& \left|\mathbf{x}^\top\boldsymbol{\theta}_e\right| \leq \left|\mathbf{x}^\top(\boldsymbol{\theta}_e - \boldsymbol{\theta}_e^*)\right| + \left|\mathbf{x}^\top\boldsymbol{\theta}_e^*\right| \leq \|\mathbf{x}\|_2\|\boldsymbol{\theta}_e - \boldsymbol{\theta}_e^*\|_2 + O(1) \\ \leq& \gamma\sqrt{d} + O(1) = O(1)\end{aligned}$$

for all $j \in [d]$, $\ell_{CE}(\theta) \leq \log\left(1 + e^{O(1)}\right) = O(1)$, we have

$$C_{\boldsymbol{\beta},*} \leq \max\left(O(1), 2\lambda_{AC}\right).$$

■

C.3 Dice Loss for Pixel-wise Class Imbalance

With finite samples in practice, since the averaged cross-entropy loss ((5.2)) weights each pixel in the image label equally, the pixel-wise class imbalance can become a problem. For example, the background pixels can be dominant in most of the segmentation labels, making the classifier prone to predict pixels as background.

To cope with such vulnerability, [23, 26, 141, 164, 174] propose to combine the cross-entropy loss with the *dice loss*—a popular segmentation loss based on the overlap between true labels and their corresponding predictions in each class:

$$\ell_{DICE}\left(\theta; (\mathbf{x}, \mathbf{y})\right) = 1 - \frac{1}{K} \sum_{k=1}^{K} DSC\left(f_\theta\left(\mathbf{x}\right)_{[:,k]}, \mathbb{I}\left\{\mathbf{y} = k\right\}\right), \tag{C.8}$$

where for any $\mathbf{p} \in [0,1]^d$, $\mathbf{q} \in \{0,1\}^d$, $DSC\left(\mathbf{p}, \mathbf{q}\right) = \frac{2\mathbf{p}^\top \mathbf{q}}{\|\mathbf{p}\|_1 + \|\mathbf{q}\|_1} \in [0,1]$ denotes the dice coefficient [106, 140]. Notice that by measuring the bounded dice coefficient for each of the K classes individually, the dice loss tends to be robust to class imbalance.

[141] merges both dice and averaged cross-entropy losses via a convex combination. It is also a common practice to add a smoothing term in both the nominator and denominator of the DSC [121].

Combining the dice loss ((C.8)) with the weighted augmentation consistency regularization formulation ((5.5)), in practice, we solve

$$\widehat{\theta}^{WAC}, \widehat{\boldsymbol{\beta}} \in \underset{\theta \in \mathcal{F}_{\theta^*}(\gamma)}{\operatorname{argmin}} \ \underset{\boldsymbol{\beta} \in [0,1]^n}{\operatorname{argmax}} \left\{ \widehat{L}^{WAC}\left(\theta, \boldsymbol{\beta}\right) \triangleq \frac{1}{n} \sum_{i=1}^{n} \widehat{L}_i^{WAC}\left(\theta, \boldsymbol{\beta}\right) \right\} \tag{C.9}$$

$$\widehat{L}_i^{WAC}\left(\theta, \boldsymbol{\beta}\right) \triangleq \ell_{DICE}\left(\theta; (\mathbf{x}_i, \mathbf{y}_i)\right) + \boldsymbol{\beta}_{[i]} \cdot \ell_{CE}\left(\theta; (\mathbf{x}_i, \mathbf{y}_i)\right) + \left(1 - \boldsymbol{\beta}_{[i]}\right) \cdot \ell_{AC}\left(\theta; \mathbf{x}_i, A_{i,1}, A_{i,2}\right)$$

with a slight modification in Algorithm 6 line 9:

$$\begin{aligned} \theta_t \leftarrow \theta_{t-1} - \eta_\theta \cdot \Big(& \nabla_\theta \ell_{DICE}\left(\theta_{t-1}; (\mathbf{x}_{i_t}, \mathbf{y}_{i_t})\right) + \left(\boldsymbol{\beta}_t\right)_{[i_t]} \cdot \nabla_\theta \ell_{CE}\left(\theta_{t-1}; (\mathbf{x}_{i_t}, \mathbf{y}_{i_t})\right) \\ & + \left(1 - \left(\boldsymbol{\beta}_t\right)_{[i_t]}\right) \cdot \nabla_\theta \ell_{AC}\left(\theta_{t-1}; \mathbf{x}_{i_t}, A_{i_t,1}, A_{i_t,2}\right) \Big). \end{aligned}$$

On the influence of incorporating dice loss in experiments. We note that, in the experiments, the dice loss ℓ_{DICE} is treated independently of *AdaWAC* in Algorithm 6 via standard stochastic gradient descent. In particular for the comparison with hard-thresholding algorithms in Table 5.2, we keep the updating on ℓ_{DICE} of the original untrimmed batch intact for both **trim-train** and **trim-ratio** to exclude the potential effect of ℓ_{DICE} that is not involved in reweighting.

C.4 Implementation Details and Datasets

We follow the official implementation of TransUNet[1] for model training. We use the same optimizer (SGD with learning rate 0.01, momentum 0.9, and weight decay 1e-4). For the Synapse dataset, we train TransUNet for 150 epochs on the training dataset and evaluate the last-iteration model on the test dataset. For the ACDC dataset, we train TransUNet for 360 epochs in total, while validating models on the ACDC validation dataset for every 10 epochs and testing on the best model selected by the validation. The total number of training iterations (*i.e.*, total number of batches) is set to be the same as that in the vanilla TransUNet [26] experiments. In particular, the results in Table 5.1 are averages (and standard deviations) over 3 arbitrary random seeds. The results in Table 5.2, Table 5.3, and Table 5.4 are given by the original random seed used in the TransUNet experiments.

Synapse multi-organ segmentation dataset (Synapse). The Synapse dataset[2] is multi-organ abdominal CT scans for medical image segmentation in the MICCAI 2015 Multi-Atlas Abdomen Labelling Challenge [26]. There are 30 cases of CT scans with variable sizes $(512 \times 512 \times 85 - 512 \times 512 \times 198)$, and slice thickness ranges from 2.5mm to 5.0mm. We use the pre-processed data provided by [26] and follow their train/test split to use 18 cases for training and 12 cases for testing on 8 abdominal organs—aorta, gallbladder, left kidney (L), right kidney (R), liver, pancreas, spleen, and stomach. The abdominal organs were labeled by experience undergraduates and verified by a radiologist using MIPAV software according to the information from Synapse wiki page.

Automated cardiac diagnosis challenge dataset (ACDC). The ACDC dataset[3] is cine-MRI scans in the MICCAI 2017 Automated Cardiac Diagnosis Challenge. There are 200 scans from 100 patients, and each patient has two frames with slice thickness from 5mm to 8mm. We use the pre-processed data also provided by [26] and follow their train/validate/test split to use 70 patients' scans for training, 10 patients' scans for validation, and 20 patients' scans for testing on three cardiac structures—left ventricle (LV), myocardium (MYO), and right ventricle (RV). The data were labeled by one clinical expert according to the description on ACDC dataset website.

[1] https://github.com/Beckschen/TransUNet

[2] See detailed description at https://www.synapse.org/#!Synapse:syn3193805/wiki/217789

[3] See detailed description at https://www.creatis.insa-lyon.fr/Challenge/acdc/

C.5 Additional Experimental Results

C.5.1 Sample Efficiency and Robustness of *AdaWAC* with UNet

In addition to the empirical evidence on TransUNet presented in Table 5.1, here, we demonstrate that the sample efficiency and distributional robustness of *AdaWAC* extend to the more widely used UNet architecture. In Table C.1, analogous to Table 5.1, the experiments on the **full** and **half-slice** datasets provide evidence for the *sample efficiency* of *AdaWAC* compared to the baseline (ERM+SGD) on UNet. Meanwhile, the *distributional robustness* of *AdaWAC* with UNet is well illustrated by the **half-vol** and **half-sparse** experiments.

Table C.1: *AdaWAC* with UNet trained on the full Synapse and its subsets

Training	Method	DSC ↑	HD95 ↓	Aorta	Gallbladder	Kidney (L)	Kidney (R)	Liver	Pancreas	Spleen	Stomach
full	baseline	74.04 ± 1.52	36.65 ± 0.33	84.93	55.59	77.59	70.92	92.21	55.01	82.87	73.21
	AdaWAC	**76.71** ± **0.62**	**30.67** ± **2.85**	85.68	55.19	80.15	75.45	94.11	56.19	87.54	81.39
half-slice	baseline	73.09 ± 0.10	40.05 ± 4.99	83.23	53.18	74.69	71.51	92.74	52.81	83.85	72.71
	AdaWAC	**75.12** ± **0.78**	**29.26** ± **2.16**	85.15	55.77	79.29	72.47	93.71	54.93	86.09	73.53
half-vol	baseline	63.21 ± 2.53	64.20 ± 4.46	79.46	45.79	55.79	54.91	88.65	41.61	71.68	67.77
	AdaWAC	**71.09** ± **1.14**	**39.95** ± **7.76**	83.15	49.14	75.74	70.33	90.47	44.81	82.34	72.75
half-sparse	baseline	37.30 ± 1.32	69.67 ± 2.89	61.57	8.33	57.45	50.44	60.28	23.51	17.83	18.99
	AdaWAC	**44.85** ± **1.03**	**62.40** ± **5.17**	71.56	8.40	65.42	62.73	74.02	24.16	36.65	15.88

Implementation details of UNet experiments. For the backbone architecture of experiments in Table C.1, we use a UNet with a ResNet-34 encoder initialized with ImageNet pre-trained weights. We leverage the implementation of UNet and load the pre-trained model via the PyTorch API for segmentation models [77]. For training, we use the same optimizer (SGD with learning rate 0.01, momentum 0.9, and weight decay 1e-4) and the total number of epochs (150 epochs on Synapse training set) as the TransUNet experiments, evaluating the last-iteration model on the test dataset. As before, the results in Table C.1 are averages (and standard deviations) over 3 arbitrary random seeds.

C.5.2 Visualization of Segmentation on ACDC dataset

As shown in Figure C.1, the model trained by *AdaWAC* segments cardiac structures with more accurate shapes (column 1), identifies organs missed by baseline TransUNet (column 2-3) and circumvents the false-positive pixel classifications (*i.e.*, fake predictions of background pixels as organs) suffered by the TransUNet baseline (column 4-6).

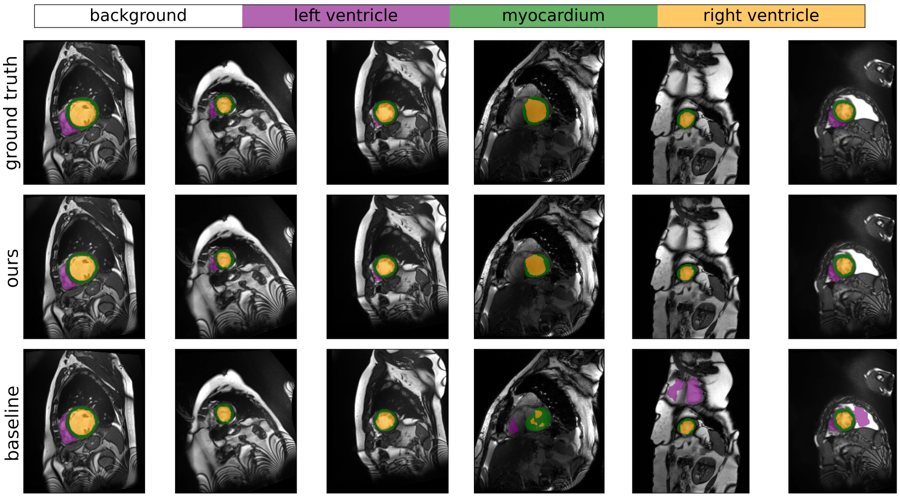


Figure C.1: Visualization of segmentation results on ACDC dataset. From top to bottom: ground truth, ours, and baseline method.

C.5.3 Visualization of Segmentation on Synapse with Distributional Shift

Figure C.2 visualizes the segmentation predictions on 6 Synapse test slices made by models trained via *AdaWAC* (ours) and via the baseline (ERM+SGD) with TransUNet [26] on the **half-sparse** subset of the Synapse training set. We observe that, although the segmentation performances of both the baseline and *AdaWAC* are compromised by the extreme scarcity of label-dense samples and the severe distributional shift, *AdaWAC* provides more accurate predictions on the relative positions of organs, as well as less misclassification of organs (*e.g.*, the baseline tends to misclassify other organs and the background as the left kidney). Nevertheless, due to the scarcity of labels, both the model trained with *AdaWAC* and that trained with the baseline fail to make good predictions on the segmentation boundaries.

C.5.4 Experimental Results on Previous Metrics

In this section, we include the results of experiments on Synapse[4] dataset with metrics defined in TransUNet [26] for reference. In TransUNet [26], DSC is 1 when the sum of ground truth labels is zero (i.e., gt.sum() == 0) while the sum of predicted labels is nonzero (i.e., pred.sum() $>$ 0). However, according to the definition of dice scores, $DSC = 2|A \cap B|/(|A| + |B|), \forall A, B$, the DSC for the above case should be 0 since the intersection is 0 and the denominator is non-zero. In our evaluation, we change the special condition for DSC as 1 to pred.sum == 0 and gt.sum() == 0 instead, in which case the denominator is 0.

[4] Note that the numbers of correct metrics and metrics in TransUNet [26] on ACDC dataset are the same.

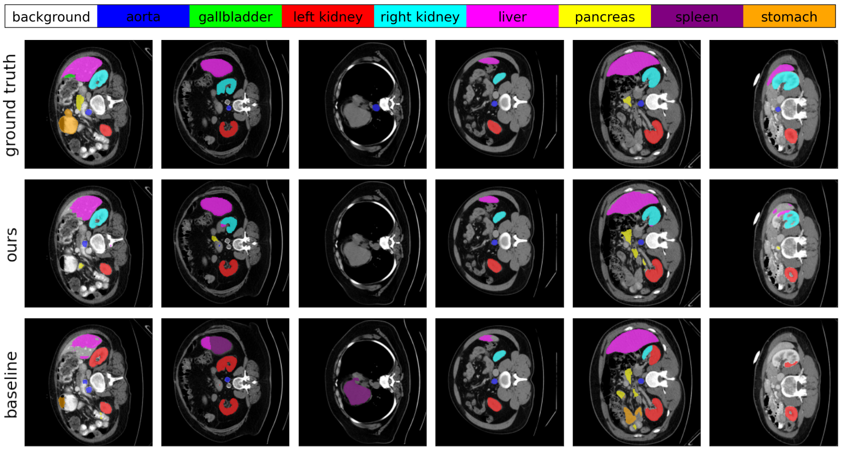


Figure C.2: Visualization of segmentation predictions made by models trained via *AdaWAC* (ours) and via the baseline (ERM+SGD) with TransUNet [26] on the **half-sparse** subset of the Synapse training set. Top to bottom: ground truth, ours (*AdaWAC*), baseline.

Table C.2: *AdaWAC* with TransUNet trained on the full Synapse and its subsets, measured by metrics in TransUNet [26].

Training	Method	DSC ↑	HD95 ↓	Aorta	Gallbladder	Kidney (L)	Kidney (R)	Liver	Pancreas	Spleen	Stomach
full	baseline	77.32	29.23	87.46	63.54	82.06	77.76	94.10	54.06	85.07	74.54
	AdaWAC	80.16	25.79	87.23	63.27	84.58	81.69	94.62	58.29	90.63	81.01
half-slice	baseline	76.24	24.66	86.26	57.61	79.32	76.55	94.34	54.04	86.20	75.57
	AdaWAC	78.14	29.75	86.66	62.28	81.36	78.84	94.60	57.95	85.38	78.01
half-vol	baseline	72.65	35.86	83.29	43.70	78.25	77.25	92.92	51.32	83.80	70.66
	AdaWAC	75.93	34.95	84.45	60.40	79.59	76.06	93.19	54.46	84.91	74.37
half-sparse	baseline	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	AdaWAC	39.68	80.93	76.59	0.00	66.53	62.11	49.69	31.09	12.30	19.11

Table C.3: AdaWAC versus hard-thresholding algorithms with TransUNet on Synapse, measured by metrics in TransUNet [26].

Method	DSC ↑	HD95↓	Aorta	Gallbladder	Kidney (L)	Kidney (R)	Liver	Pancreas	Spleen	Stomach
baseline	77.32	29.23	87.46	63.54	82.06	77.76	94.10	54.06	85.07	74.54
trim-train	77.05	26.94	86.70	60.65	80.02	76.64	94.25	54.20	86.44	77.49
trim-ratio	75.30	28.59	87.35	57.29	78.70	72.22	94.18	52.32	86.31	74.03
trim-train+ACR	76.70	35.06	87.11	62.22	74.19	75.25	92.19	57.16	88.21	77.30
trim-ratio+ACR	79.02	33.59	86.82	61.67	83.52	81.22	94.07	59.06	88.08	77.71
AdaWAC (ours)	80.16	25.79	87.23	63.27	84.58	81.69	94.62	58.29	90.63	81.01

Table C.4: Ablation study of AdaWAC with TransUNet trained on Synapse, measured by metrics in TransUNet [26].

Method	DSC ↑	HD95↓	Aorta	Gallbladder	Kidney (L)	Kidney (R)	Liver	Pancreas	Spleen	Stomach
baseline	77.32	29.23	87.46	63.54	82.06	77.76	94.10	54.06	85.07	74.54
reweight-only	77.72	29.24	86.15	62.31	82.96	80.28	93.42	55.86	85.29	75.49
ACR-only	78.93	31.65	87.96	62.67	81.79	80.21	94.52	60.41	88.07	75.83
AdaWAC-0.01	78.98	27.81	87.58	61.09	82.29	80.22	94.90	55.92	91.63	78.23
AdaWAC-1.0	80.16	25.79	87.23	63.27	84.58	81.69	94.62	58.29	90.63	81.01

Bibliography

[1] Yariv Aizenbud, Gil Shabat, and Amir Averbuch. Randomized lu decomposition using sparse projections. *Computers & Mathematics with Applications*, 72(9):2525–2534, 2016.

[2] Guillaume Alain, Alex Lamb, Chinnadhurai Sankar, Aaron Courville, and Yoshua Bengio. Variance reduction in sgd by distributed importance sampling. *arXiv preprint arXiv:1511.06481*, 2015.

[3] David Anderson, Simon Du, Michael Mahoney, Christopher Melgaard, Kunming Wu, and Ming Gu. Spectral Gap Error Bounds for Improving CUR Matrix Decomposition and the Nyström Method. In Guy Lebanon and S. V. N. Vishwanathan, editors, *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics*, volume 38 of *Proceedings of Machine Learning Research*, pages 19–27, San Diego, California, USA, 09–12 May 2015. PMLR.

[4] David Anderson and Ming Gu. An efficient, sparsity-preserving, online algorithm for low-rank approximation. In *International Conference on Machine Learning*, pages 156–165. PMLR, 2017.

[5] Philip Bachman, Ouais Alsharif, and Doina Precup. Learning with pseudo-ensembles. *Advances in neural information processing systems*, 27:3365–3373, 2014.

[6] Peter L. Bartlett and Shahar Mendelson. Rademacher and gaussian complexities: Risk bounds and structural results. *J. Mach. Learn. Res.*, 3(null):463–482, March 2003.

[7] Hritam Basak, Rajarshi Bhattacharya, Rukhshanda Hussain, and Agniv Chatterjee. An embarrassingly simple consistency regularization method for semi-supervised medical image segmentation. *arXiv preprint arXiv:2202.00677*, 2022.

[8] Joshua D. Batson, Daniel A. Spielman, and Nikhil Srivastava. Twice-ramanujan sparsifiers. In *Proceedings of the Forty-First Annual ACM Symposium on Theory of Computing*, STOC ’09, pages 255–262, New York, NY, USA, 2009. Association for Computing Machinery.

[9] Mario Bebendorf. Approximation of boundary element matrices. *Numerische Mathematik*, 86:565–589, 2000.

[10] Aharon Ben-Tal, Dick den Hertog, Anja De Waegenaere, Bertrand Melenberg, and Gijs Rennen. Robust solutions of optimization problems affected by uncertain probabilities. *Management Science*, 59(2):341–357, 2013.

[11] Gregory Benton, Marc Finzi, Pavel Izmailov, and Andrew Gordon Wilson. Learning invariances in neural networks. *arXiv preprint arXiv:2010.11882*, 2020.

[12] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin A Raffel. Mixmatch: A holistic approach to semi-supervised learning. *Advances in neural information processing systems*, 32, 2019.

[13] David Berthelot, Rebecca Roelofs, Kihyuk Sohn, Nicholas Carlini, and Alex Kurakin. Adamatch: A unified approach to semi-supervised learning and domain adaptation. *arXiv preprint arXiv:2106.04732*, 2021.

[14] Dimitri Bertsekas. *Convex optimization theory*, volume 1. Athena Scientific, 2009.

[15] Jacob Bien, Ya Xu, and Michael Mahoney. Cur from a sparse optimization viewpoint. *Annual Advances in Neural Information Processing Systems 24: Proceedings of the 2010 Conference*, 11 2010.

[16] Alberto Bietti, Luca Venturi, and Joan Bruna. On the sample complexity of learning with geometric stability. *arXiv preprint arXiv:2106.07148*, 2021.

[17] Gerda Bortsova, Florian Dubost, Laurens Hogeweg, Ioannis Katramados, and Marleen de Bruijne. Semi-supervised medical image segmentation via learning consistency under transformations. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 810–818. Springer, 2019.

[18] Christos Boutsidis, Petros Drineas, and Malik Magdon-Ismail. Near-optimal column-based matrix reconstruction. *SIAM Journal on Computing*, 43(2):687–717, 2014.

[19] Christos Boutsidis and Alex Gittens. Improved matrix algorithms via the subsampled randomized hadamard transform. *SIAM Journal on Matrix Analysis and Applications*, 34(3):1301–1340, 2013.

[20] Christos Boutsidis, Prabhanjan Kambadur, and Alex Gittens. Spectral clustering via the power method-provably. In *International conference on machine learning*, pages 40–48. PMLR, 2015.

[21] Christos Boutsidis and David P. Woodruff. Optimal cur matrix decompositions. In *Proceedings of the Forty-Sixth Annual ACM Symposium on Theory of Computing*, STOC '14, pages 353–362, New York, NY, USA, 2014. Association for Computing Machinery.

[22] Tianle Cai, Ruiqi Gao, Jason D. Lee, and Qi Lei. A theory of label propagation for subpopulation shift, 2021.

[23] Hu Cao, Yueyue Wang, Joy Chen, Dongsheng Jiang, Xiaopeng Zhang, Qi Tian, and Manning Wang. Swin-unet: Unet-like pure transformer for medical image segmentation. *arXiv preprint arXiv:2105.05537*, 2021.

[24] Tony F. Chan. Rank revealing qr factorizations. *Linear Algebra and its Applications*, 88-89:67–82, 1987.

[25] Cheng Chen, Ming Gu, Zhihua Zhang, Weinan Zhang, and Yong Yu. Efficient spectrum-revealing cur matrix decomposition. In *International Conference on Artificial Intelligence and Statistics*, pages 766–775. PMLR, 2020.

[26] Jieneng Chen, Yongyi Lu, Qihang Yu, Xiangde Luo, Ehsan Adeli, Yan Wang, Le Lu, Alan L Yuille, and Yuyin Zhou. Transunet: Transformers make strong encoders for medical image segmentation. *arXiv preprint arXiv:2102.04306*, 2021.

[27] Shuxiao Chen, Edgar Dobriban, and Jane H Lee. A group-theoretic framework for data augmentation. *Journal of Machine Learning Research*, 21(245):1–71, 2020.

[28] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

[29] Ali Civril and Malik Magdon-Ismail. Exponential inapproximability of selecting a maximum volume sub-matrix. *Algorithmica*, 65(1):159–176, 2013.

[30] Kenneth L. Clarkson and David P. Woodruff. Low-rank approximation and regression in input sparsity time. *J. ACM*, 63(6), January 2017.

[31] Michael B. Cohen, Yin Tat Lee, Cameron Musco, Christopher Musco, Richard Peng, and Aaron Sidford. Uniform sampling for matrix approximation. In *Proceedings of the 2015 Conference on Innovations in Theoretical Computer Science*, ITCS '15, pages 181–190, New York, NY, USA, 2015. Association for Computing Machinery.

[32] Taco Cohen and Max Welling. Group equivariant convolutional networks. In *International conference on machine learning*, pages 2990–2999. PMLR, 2016.

[33] Alice Cortinovis and Daniel Kressner. Low-rank approximation in the frobenius norm by column and row subset selection. *SIAM Journal on Matrix Analysis and Applications*, 41(4):1651–1673, 2020.

[34] Ekin D Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V Le. Autoaugment: Learning augmentation strategies from data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 113–123, 2019.

[35] Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 702–703, 2020.

[36] Tri Dao, Albert Gu, Alexander Ratner, Virginia Smith, Chris De Sa, and Christopher Ré. A kernel theory of modern data augmentation. In *International Conference on Machine Learning*, pages 1528–1537. PMLR, 2019.

[37] Kenneth R Davidson and Stanislaw J Szarek. Local operator theory, random matrices and banach spaces. *Handbook of the geometry of Banach spaces*, 1(317-366):131, 2001.

[38] C. Davis and W. M. Kahan. The rotation of eigenvectors by a perturbation. III. *sinum*, 7(1):1–46, 1970.

[39] Michal Derezinski, Rajiv Khanna, and Michael W. Mahoney. Improved guarantees and a multiple-descent curve for the column subset selection problem and the nyström method. *CoRR*, abs/2002.09073, 2020.

[40] Michał Dereziński and Michael Mahoney. Determinantal point processes in randomized numerical linear algebra. *Notices of the American Mathematical Society*, 68:1, 01 2021.

[41] Amit Deshpande and Luis Rademacher. Efficient volume sampling for row/column subset selection. In *2010 IEEE 51st Annual Symposium on Foundations of Computer Science*, pages 329–338, 2010.

[42] Amit Deshpande, Luis Rademacher, Santosh Vempala, and Grant Wang. Matrix approximation and projective clustering via volume sampling. *Theory of Computing*, 2(12):225–247, 2006.

[43] Terrance DeVries and Graham W Taylor. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, 2017.

[44] Yijun Dong, Chao Chen, Per-Gunnar Martinsson, and Katherine Pearce. Robust blockwise random pivoting: Fast and accurate adaptive interpolative decomposition. *arXiv preprint arXiv:2309.16002*, 2023.

[45] Yijun Dong and Per-Gunnar Martinsson. Simpler is better: a comparative study of randomized pivoting algorithms for cur and interpolative decompositions. *Advances in Computational Mathematics*, 49(4):66, 2023.

[46] Yijun Dong, Per-Gunnar Martinsson, and Yuji Nakatsukasa. Efficient bounds and estimates for canonical angles in randomized subspace approximations. *arXiv preprint arXiv:2211.04676*, 2022.

[47] Yijun Dong, Kevin Miller, Qi Lei, and Rachel Ward. Cluster-aware semi-supervised learning: Relational knowledge distillation provably learns clustering. *arXiv preprint arXiv:2307.11030*, 2023.

[48] Yijun Dong, Yuege Xie, and Rachel Ward. Adaptively weighted data augmentation consistency regularization for robust optimization under concept shift. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 8296–8316. PMLR, 23–29 Jul 2023.

[49] Petros Drineas, Malik Magdon-Ismail, Michael W Mahoney, and David P Woodruff. Fast approximation of matrix coherence and statistical leverage. *The Journal of Machine Learning Research*, 13(1):3475–3506, 2012.

[50] Petros Drineas, Michael W Mahoney, and Shan Muthukrishnan. Relative-error cur matrix decompositions. *SIAM Journal on Matrix Analysis and Applications*, 30(2):844–881, 2008.

[51] Zlatko Drmač and Serkan Gugercin. A new selection operator for the discrete empirical interpolation method—improved a priori error bound and extensions. *SIAM Journal on Scientific Computing*, 38(2):A631–A648, 2016.

[52] Simon S. Du, Wei Hu, Sham M. Kakade, Jason D. Lee, and Qi Lei. Few-shot learning via learning the representation, provably, 2020.

[53] John Duchi, Peter Glynn, and Hongseok Namkoong. Statistics of robust optimization: A generalized empirical likelihood approach. *arXiv preprint arXiv:1610.03425*, 2016.

[54] John Duchi and Hongseok Namkoong. Learning models with uniform performance via distributionally robust optimization. *arXiv preprint arXiv:1810.08750*, 2018.

[55] Carl Eckart and Gale Young. The approximation of one matrix by another of lower rank. *Psychometrika*, 1(3):211–218, Sep 1936.

[56] Geoffrey French, Michal Mackiewicz, and Mark Fisher. Self-ensembling for visual domain adaptation. *arXiv preprint arXiv:1706.05208*, 2017.

[57] Siddhant Garg and Yingyu Liang. Functional regularization for representation learning: A unified theoretical perspective. *arXiv preprint arXiv:2008.02447*, 2020.

[58] George A. Geist and Charles H. Romine. \$lu\$ factorization algorithms on distributed-memory multiprocessor architectures. *SIAM Journal on Scientific and Statistical Computing*, 9(4):639–649, 1988.

[59] Gene H Golub and Charles F Van Loan. *Matrix computations*. JHU press, Baltimore, MD, USA, 2013.

[60] Siddharth Gopal. Adaptive sampling for sgd by exploiting side information. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48*, ICML'16, pages 364–372. JMLR.org, 2016.

[61] S.A. Goreinov, E.E. Tyrtyshnikov, and N.L. Zamarashkin. A theory of pseudoskeleton approximations. *Linear Algebra and its Applications*, 261(1):1–21, 1997.

[62] Laura Grigori, James W. Demmel, and Hua Xiang. Calu: A communication optimal lu factorization algorithm. *SIAM Journal on Matrix Analysis and Applications*, 32(4):1317–1350, 2011.

[63] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in neural information processing systems*, 33:21271–21284, 2020.

[64] M. Gu. Subspace iteration randomization and singular value problems. *SIAM Journal on Scientific Computing*, 37(3):A1139–A1173, 2015.

[65] Ming Gu and Stanley C. Eisenstat. Efficient algorithms for computing a strong rank-revealing qr factorization. *SIAM Journal on Scientific Computing*, 17(4):848–869, 1996.

[66] Guy Hacohen and Daphna Weinshall. On the power of curriculum learning in training deep networks. In *International Conference on Machine Learning*, pages 2535–2544. PMLR, 2019.

[67] Jamie Haddock, Deanna Needell, Elizaveta Rebrova, and William Swartworth. Quantile-based iterative methods for corrupted systems of linear equations. *SIAM Journal on Matrix Analysis and Applications*, 43(2):605–637, 2022.

[68] N. Halko, P. G. Martinsson, and J. A. Tropp. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM Review*, 53(2):217–288, 2011.

[69] Jeff Z HaoChen, Colin Wei, Adrien Gaidon, and Tengyu Ma. Provable guarantees for self-supervised deep learning with spectral contrastive loss. *arXiv preprint arXiv:2106.04156*, 2021.

[70] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738, 2020.

[71] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[72] Zhuoxun He, Lingxi Xie, Xin Chen, Ya Zhang, Yanfeng Wang, and Qi Tian. Data augmentation revisited: Rethinking the distribution gap between clean and augmented data. *arXiv preprint arXiv:1909.09148*, 2019.

[73] Ken Ho, Sheehan Olver, Tony Kelman, Elias Jarlebring, Julia TagBot, and Mikael Slevinsky. Lowrankapprox,jl: v0.4.3. https://github.com/JuliaMatrices/LowRankApprox.jl, 2020.

[74] Y. P. Hong and C.-T. Pan. Rank-revealing qr factorizations and the singular value decomposition. *Mathematics of Computation*, 58(197):213–232, 1992.

[75] Roger A. Horn and Charles R. Johnson. *Matrix Analysis*. Cambridge University Press, 2 edition, 2012.

[76] Alston S. Householder. Unitary triangularization of a nonsymmetric matrix. *J. ACM*, 5(4):339–342, October 1958.

[77] Pavel Iakubovskii. Segmentation models pytorch. https://github.com/qubvel/segmentation_models.pytorch, 2019.

[78] Piotr Indyk and Rajeev Motwani. Approximate nearest neighbors: Towards removing the curse of dimensionality. In *Proceedings of the Thirtieth Annual ACM Symposium on Theory of Computing*, STOC '98, page 604–613, New York, NY, USA, 1998. Association for Computing Machinery.

[79] W. Kahan. Numerical linear algebra. *Canadian Mathematical Bulletin*, 9(5):757–801, 1966.

[80] Davood Karimi, Haoran Dou, Simon K Warfield, and Ali Gholipour. Deep learning with noisy labels: Exploring techniques and remedies in medical image analysis. *Medical Image Analysis*, 65:101759, 2020.

[81] Angelos Katharopoulos and François Fleuret. Not all samples are created equal: Deep learning with importance sampling. In *International conference on machine learning*, pages 2525–2534. PMLR, 2018.

[82] Kenji Kawaguchi and Haihao Lu. Ordered sgd: A new stochastic optimization framework for empirical risk minimization. In *International Conference on Artificial Intelligence and Statistics*, pages 669–679. PMLR, 2020.

[83] Kezhong Zhao, M. N. Vouvakis, and Jin-Fa Lee. The adaptive cross approximation algorithm for accelerated method of moments computations of emc problems. *IEEE Transactions on Electromagnetic Compatibility*, 47(4):763–773, 2005.

[84] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25:1097–1105, 2012.

[85] Michael Kuchnik and Virginia Smith. Efficient augmentation via data subsampling. *arXiv preprint arXiv:1810.05222*, 2018.

[86] J. Kuczyński and H. Woźniakowski. Estimating the largest eigenvalue by the power and lanczos algorithms with a random start. *SIAM Journal on Matrix Analysis and Applications*, 13(4):1094–1122, 1992.

[87] J. Kurzak, P. Luszczek, M. Faverge, and J. Dongarra. Lu factorization with partial pivoting for a multicore system with accelerators. *IEEE Transactions on Parallel and Distributed Systems*, 24(8):1613–1621, 2013.

[88] Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. *arXiv preprint arXiv:1610.02242*, 2016.

[89] Michel Ledoux and Michel Talagrand. *Probability in Banach Spaces: isoperimetry and processes*. Springer Science & Business Media, 2013.

[90] Xiaomeng Li, Lequan Yu, Hao Chen, Chi-Wing Fu, Lei Xing, and Pheng-Ann Heng. Transformation-consistent self-ensembling model for semisupervised medical image segmentation. *IEEE Transactions on Neural Networks and Learning Systems*, 32(2):523–534, 2020.

[91] Zhiyuan Li, Ruosong Wang, Dingli Yu, Simon S Du, Wei Hu, Ruslan Salakhutdinov, and Sanjeev Arora. Enhanced convolutional neural tangent kernels. *arXiv preprint arXiv:1911.00809*, 2019.

[92] Edo Liberty, Franco Woolfe, Per-Gunnar Martinsson, Vladimir Rokhlin, and Mark Tygert. Randomized algorithms for the low-rank approximation of matrices. *Proceedings of the National Academy of Sciences*, 104(51):20167–20172, 2007.

[93] Sungbin Lim, Ildoo Kim, Taesup Kim, Chiheon Kim, and Sungwoong Kim. Fast autoaugment. *Advances in Neural Information Processing Systems*, 32:6665–6675, 2019.

[94] Ilya Loshchilov and Frank Hutter. Online batch selection for faster training of neural networks. *arXiv preprint arXiv:1511.06343*, 2015.

[95] Clare Lyle, Marta Kwiatkowksa, and Yarin Gal. An analysis of the effect of invariance on generalization in neural networks. In *International conference on machine learning Workshop on Understanding and Improving Generalization in Deep Learning*, 2019.

[96] Michael W Mahoney and Petros Drineas. Cur matrix decompositions for improved data analysis. *Proceedings of the National Academy of Sciences*, 106(3):697–702, 2009.

[97] Diego Marcos, Michele Volpi, Nikos Komodakis, and Devis Tuia. Rotation equivariant vector field networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5048–5057, 2017.

[98] Per-Gunnar Martinsson. Randomized methods for matrix computations. *The Mathematics of Data*, 25(4):187–231, 2019.

[99] Per-Gunnar Martinsson, Vladimir Rokhlin, and Mark Tygert. A randomized algorithm for the decomposition of matrices. *Applied and Computational Harmonic Analysis*, 30(1):47–68, 2011.

[100] Per-Gunnar Martinsson and Joel A. Tropp. Randomized numerical linear algebra: Foundations and algorithms. *Acta Numerica*, 29:403–572, 2020.

[101] Per-Gunnar Martinsson and Joel A. Tropp. Randomized numerical linear algebra: Foundations and algorithms. *Acta Numerica*, 29:403–572, 2020.

[102] Song Mei, Theodor Misiakiewicz, and Andrea Montanari. Learning with invariances in random features and kernel models. *arXiv preprint arXiv:2102.13219*, 2021.

[103] Xiangrui Meng and Michael W. Mahoney. Low-distortion subspace embeddings in input-sparsity time and applications to robust linear regression. In *Proceedings of the Forty-Fifth Annual ACM Symposium on Theory of Computing*, STOC '13, pages 91–100, New York, NY, USA, 2013. Association for Computing Machinery.

[104] C. Meszaros. Meszaros/large, lp sequence: large000 to large036. http://old.sztaki.hu/ meszaros/public_ftp/lptestset/, 2004.

[105] Raphael A Meyer, Cameron Musco, Christopher Musco, and David P Woodruff. Hutch++: Optimal stochastic trace estimation. In *Symposium on Simplicity in Algorithms (SOSA)*, pages 142–155. SIAM, 2021.

[106] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *2016 Fourth International Conference on 3D Vision (3DV)*, pages 565–571, 2016.

[107] L Miranian and Ming Gu. Strong rank revealing lu factorizations. *Linear Algebra and its Applications*, 367:1–16, 07 2003.

[108] Cameron Musco and Christopher Musco. Randomized block krylov methods for stronger and faster approximate singular value decomposition. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'15, pages 1396–1404, Cambridge, MA, USA, 2015. MIT Press.

[109] Yuji Nakatsukasa. Sharp error bounds for ritz vectors and approximate singular vectors. *Math. Comput.*, 89:1843–1866, 2020.

[110] Deanna Needell, Rachel Ward, and Nati Srebro. Stochastic gradient descent, weighted sampling, and the randomized kaczmarz algorithm. *Advances in neural information processing systems*, 27, 2014.

[111] J. Nelson and H. L. Nguyên. Osnap: Faster numerical linear algebra algorithms via sparser subspace embeddings. In *2013 IEEE 54th Annual Symposium on Foundations of Computer Science*, pages 117–126, 2013.

[112] A. Nemirovski, A. Juditsky, G. Lan, and A. Shapiro. Robust stochastic approximation approach to stochastic programming. *SIAM Journal on Optimization*, 19(4):1574–1609, 2009.

[113] C.-T. Pan. On the existence and computation of rank-revealing lu factorizations. *Linear Algebra and its Applications*, 316(1):199–222, 2000. Special Issue: Conference celebrating the 60th birthday of Robert J. Plemmons.

[114] Victor Y. Pan, Guoliang Qian, and Xiaodong Yan. Random multipliers numerically stabilize gaussian and block gaussian elimination: Proofs and an extension to low-rank approximation. *Linear Algebra and its Applications*, 481:202–234, 2015.

[115] Victor Y. Pan and Liang Zhao. Numerically safe gaussian elimination with no pivoting. *Linear Algebra and its Applications*, 527:349–383, 2017.

[116] G. Peters and J. H. Wilkinson. On the stability of gauss-jordan elimination with pivoting. *Commun. ACM*, 18(1):20–24, January 1975.

[117] Shashank Rajput, Zhili Feng, Zachary Charles, Po-Ling Loh, and Dimitris Papailiopoulos. Does data augmentation lead to positive margin? In *International Conference on Machine Learning*, pages 5321–5330. PMLR, 2019.

[118] Vladimir Rokhlin and Mark Tygert. A fast randomized algorithm for overdetermined linear least-squares regression. *Proceedings of the National Academy of Sciences*, 105(36):13212–13217, 2008.

[119] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[120] Mark Rudelson. Invertibility of random matrices: Norm of the inverse. *Annals of Mathematics*, 168(2):575–600, 2008.

[121] Stuart J Russell and Peter Norvig. Artificial intelligence: a modern approach. malaysia, 2016.

[122] Shiori Sagawa, Pang Wei Koh, Tatsunori B Hashimoto, and Percy Liang. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. *arXiv preprint arXiv:1911.08731*, 2019.

[123] Arvind K Saibaba. Randomized subspace iteration: Analysis of canonical angles and unitarily invariant norms. *SIAM Journal on Matrix Analysis and Applications*, 40(1):23–48, 2019.

[124] Mehdi Sajjadi, Mehran Javanmardi, and Tolga Tasdizen. Regularization with stochastic transformations and perturbations for deep semi-supervised learning. *Advances in neural information processing systems*, 29:1163–1171, 2016.

[125] Rico Sennrich, Barry Haddow, and Alexandra Birch. Improving neural machine translation models with monolingual data. *arXiv preprint arXiv:1511.06709*, 2015.

[126] Gil Shabat, Yaniv Shmueli, Yariv Aizenbud, and Amir Averbuch. Randomized lu decomposition. *Applied and Computational Harmonic Analysis*, 44(2):246–272, 2018.

[127] Vatsal Shah, Xiaoxia Wu, and Sujay Sanghavi. Choosing the sample with lowest loss makes sgd robust. In *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*, pages 2120–2130, Online, 26–28 Aug 2020. PMLR.

[128] Han Shao, Omar Montasser, and Avrim Blum. A theory of pac learnability under transformation invariances. *arXiv preprint arXiv:2202.07552*, 2022.

[129] Ruoqi Shen, Sebastien Bubeck, and Suriya Gunasekar. Data augmentation as feature manipulation. In *International Conference on Machine Learning*, pages 19773–19808. PMLR, 2022.

[130] Yanyao Shen and Sujay Sanghavi. Learning with bad training data via iterative trimmed loss minimization. In *International Conference on Machine Learning*, pages 5739–5748. PMLR, 2019.

[131] Connor Shorten and Taghi M Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of Big Data*, 6(1):1–48, 2019.

[132] Patrice Y Simard, Yann A LeCun, John S Denker, and Bernard Victorri. Transformation invariance in pattern recognition—tangent distance and tangent propagation. In *Neural networks: tricks of the trade*, pages 239–274. Springer, 1998.

[133] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[134] Maurice Sion. On general minimax theorems. *Pacific Journal of Mathematics*, 8(1):171 – 176, 1958.

[135] Kihyuk Sohn, David Berthelot, Chun-Liang Li, Zizhao Zhang, Nicholas Carlini, Ekin D Cubuk, Alex Kurakin, Han Zhang, and Colin Raffel. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. *arXiv preprint arXiv:2001.07685*, 2020.

[136] Edgar Solomonik and James Demmel. Communication-optimal parallel 2.5d matrix multiplication and lu factorization algorithms. In Emmanuel Jeannot, Raymond Namyst, and Jean Roman, editors, *Euro-Par 2011 Parallel Processing*, pages 90–109, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.

[137] D. C. Sorensen and Mark Embree. A deim induced cur factorization. *SIAM Journal on Scientific Computing*, 38(3):A1454–A1482, 2016.

[138] Danny C Sorensen and Mark Embree. A deim induced cur factorization. *SIAM Journal on Scientific Computing*, 38(3):A1454–A1482, 2016.

[139] Ivan Sosnovik, Michał Szmaja, and Arnold Smeulders. Scale-equivariant steerable networks. *arXiv preprint arXiv:1910.11093*, 2019.

[140] Saeid Asgari Taghanaki, Kumar Abhishek, Joseph Paul Cohen, Julien Cohen-Adad, and Ghassan Hamarneh. Deep semantic segmentation of natural and medical images: A review, 2019.

[141] Saeid Asgari Taghanaki, Yefeng Zheng, S Kevin Zhou, Bogdan Georgescu, Puneet Sharma, Daguang Xu, Dorin Comaniciu, and Ghassan Hamarneh. Combo loss: Handling input and output imbalance in multi-organ segmentation. *Computerized Medical Imaging and Graphics*, 75:24–33, 2019.

[142] Yuxing Tang, Xiaosong Wang, Adam P Harrison, Le Lu, Jing Xiao, and Ronald M Summers. Attention-guided curriculum learning for weakly supervised classification and localization of thoracic diseases on chest radiographs. In *International Workshop on Machine Learning in Medical Imaging*, pages 249–258. Springer, 2018.

[143] Christopher Tensmeyer and Tony Martinez. Improving invariance and equivariance properties of convolutional neural networks, 2017.

[144] Lloyd N Trefethen and David Bau. *Numerical linear algebra*, volume 181. Siam, Philadelphia, PA, USA, 2022.

[145] Lloyd N. Trefethen and Robert S. Schreiber. Average-case stability of gaussian elimination. *SIAM Journal on Matrix Analysis and Applications*, 11(3):335–360, 1990.

[146] Joel A. Tropp. Improved analysis of the subsampled randomized hadamard transform. *Advances in Adaptive Data Analysis*, 03(01n02):115–126, 2011.

[147] Joel A Tropp, Alp Yurtsever, Madeleine Udell, and Volkan Cevher. Fixed-rank approximation of a positive-semidefinite matrix from streaming data. *Advances in Neural Information Processing Systems*, 30, 2017.

[148] Joel A. Tropp, Alp Yurtsever, Madeleine Udell, and Volkan Cevher. Streaming low-rank matrix approximation with an application to scientific simulation. *SIAM Journal on Scientific Computing*, 41(4):A2430–A2463, 2019.

[149] Jonathan G Tullis and Aaron S Benjamin. On the effectiveness of self-paced learning. *Journal of memory and language*, 64(2):109–118, 2011.

[150] E. Tyrtyshnikov. Incomplete cross approximation in the mosaic-skeleton method. *Computing*, 64(4):367–380, Jun 2000.

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

[152] Julius von Kügelgen, Yash Sharma, Luigi Gresele, Wieland Brendel, Bernhard Schölkopf, Michel Besserve, and Francesco Locatello. Self-supervised learning with data augmentations provably isolates content from style. *arXiv preprint arXiv:2106.04619*, 2021.

[153] Sergey Voronin and Per-Gunnar Martinsson. Efficient algorithms for cur and interpolative matrix decompositions. *Advances in Computational Mathematics*, 43(3):495–516, Jun 2017.

[154] Martin J Wainwright. *High-dimensional statistics: A non-asymptotic viewpoint*, volume 48. Cambridge University Press, 2019.

[155] Martin J. Wainwright. *High-Dimensional Statistics: A Non-Asymptotic Viewpoint*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 2019.

[156] Haohan Wang, Zeyi Huang, Xindi Wu, and Eric P Xing. Squared ell_2 norm as consistency loss for leveraging augmented data to learn robust and invariant representations. *arXiv preprint arXiv:2011.13052*, 2020.

[157] Shusen Wang and Zhihua Zhang. A scalable cur matrix decomposition algorithm: Lower time complexity and tighter bound. In *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'12, pages 647–655, Red Hook, NY, USA, 2012. Curran Associates Inc.

[158] Xi Wang, Hao Chen, Huiling Xiang, Huangjing Lin, Xi Lin, and Pheng-Ann Heng. Deep virtual adversarial self-training with consistency regularization for semi-supervised medical image classification. *Medical image analysis*, 70:102010, 2021.

[159] Xin Wang, Yudong Chen, and Wenwu Zhu. A survey on curriculum learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

[160] Yisen Wang, Weiyang Liu, Xingjun Ma, James Bailey, Hongyuan Zha, Le Song, and Shu-Tao Xia. Iterative learning with open-set noisy labels. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8688–8696, 2018.

[161] Per-Åke Wedin. Perturbation theory for pseudo-inverses. *bit*, 13(2):217–232, 1973.

[162] Colin Wei, Kendrick Shen, Yining Chen, and Tengyu Ma. Theoretical analysis of self-training with deep networks on unlabeled data, 2021.

[163] Zixin Wen and Yuanzhi Li. Toward understanding the feature learning process of self-supervised contrastive learning. *arXiv preprint arXiv:2105.15134*, 2021.

[164] Ken C. L. Wong, Mehdi Moradi, Hui Tang, and Tanveer Syeda-Mahmood. 3d segmentation with exponential logarithmic loss for highly unbalanced object sizes. In Alejandro F. Frangi, Julia A. Schnabel, Christos Davatzikos, Carlos Alberola-López, and Gabor Fichtinger, editors, *Medical Image Computing and Computer Assisted Intervention – MICCAI 2018*, pages 612–619, Cham, 2018. Springer International Publishing.

[165] David P. Woodruff. Sketching as a tool for numerical linear algebra. *Found. Trends Theor. Comput. Sci.*, 10(1–2):1–157, October 2014.

[166] Franco Woolfe, Edo Liberty, Vladimir Rokhlin, and Mark Tygert. A fast randomized algorithm for the approximation of matrices. *Applied and Computational Harmonic Analysis*, 25(3):335–366, 2008.

[167] Daniel E Worrall, Stephan J Garbin, Daniyar Turmukhambetov, and Gabriel J Brostow. Harmonic networks: Deep translation and rotation equivariance. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5028–5037, 2017.

[168] Daniel E Worrall and Max Welling. Deep scale-spaces: Equivariance over scale. *arXiv preprint arXiv:1905.11697*, 2019.

[169] Sen Wu, Hongyang Zhang, Gregory Valiant, and Christopher Ré. On the generalization effects of linear transformations in data augmentation. In *International Conference on Machine Learning*, pages 10410–10420. PMLR, 2020.

[170] Xiaoxia Wu, Yuege Xie, Simon Shaolei Du, and Rachel Ward. Adaloss: A computationally-efficient and provably convergent adaptive gradient method. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(8):8691–8699, Jun. 2022.

[171] Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3733–3742, 2018.

[172] Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V Le. Self-training with noisy student improves imagenet classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10687–10698, 2020.

[173] Shuo Yang, Yijun Dong, Rachel Ward, Inderjit S Dhillon, Sujay Sanghavi, and Qi Lei. Sample efficiency of data augmentation consistency regularization. In *International Conference on Artificial Intelligence and Statistics*, pages 3825–3853. PMLR, 2023.

[174] Michael Yeung, Evis Sala, Carola-Bibiane Schönlieb, and Leonardo Rundo. Unified focal loss: Generalising dice and cross entropy-based losses to handle class imbalanced medical image segmentation. *Computerized Medical Imaging and Graphics*, 95:102026, 2022.

[175] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6023–6032, 2019.

[176] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. *arXiv preprint arXiv:1605.07146*, 2016.

[177] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017.

[178] Yi Zhang, Bin Zhou, Lei Chen, Yulin Wu, and Hongchao Zhou. Multi-transformation consistency regularization for semi-supervised medical image segmentation. In *2021 4th International Conference on Artificial Intelligence and Big Data (ICAIBD)*, pages 485–489. IEEE, 2021.

[179] Amy Zhao, Guha Balakrishnan, Fredo Durand, John V Guttag, and Adrian V Dalca. Data augmentation using learned transformations for one-shot medical image segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8543–8553, 2019.

[180] Peilin Zhao and Tong Zhang. Stochastic optimization with importance sampling for regularized loss minimization. In *international conference on machine learning*, pages 1–9. PMLR, 2015.

[181] Hong-Yu Zhou, Chengdi Wang, Haofeng Li, Gang Wang, Shu Zhang, Weimin Li, and Yizhou Yu. Ssmd: semi-supervised medical image detection with adaptive consistency and heterogeneous perturbation. *Medical Image Analysis*, 72:102117, 2021.

[182] Yanzhao Zhou, Qixiang Ye, Qiang Qiu, and Jianbin Jiao. Oriented response networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 519–528, 2017.

[183] Åke Björck and Gene H. Golub. Numerical methods for computing angles between linear subspaces. *Mathematics of Computation*, 27(123):579–594, 1973.

[184] Ali Çivril and Malik Magdon-Ismail. On selecting a maximum volume sub-matrix of a matrix and related problems. *Theoretical Computer Science*, 410(47):4801–4811, 2009.

Vita

Yijun Dong was born in Shanghai, China in October 1995 and grew up there before college. She obtained a Bachelor of Science in Applied Mathematics and Engineering Science from Emory University in May 2018 where she got her first taste of research in biophysics and soft matter physics. In August 2018, she began her doctoral study in Computational Science, Engineering, and Mathematics at the Oden Institute of the University of Texas at Austin. Her primary research interests are randomized numerical linear algebra and statistical learning theory. In 2021 and 2022, she spent two summers at Dell Technologies working on edge computing and semi-supervised tabular learning. Upon completing her graduate study in May 2023, she expects to continue her postdoctoral research as a Courant Instructor at the Courant Institute of New York University.

Address: ydong@utexas.edu

This dissertation was typeset with LaTeX by the author.