ON SPARSE REPRESENTATIONS AND NEW META-LEARNING PARADIGMS FOR REPRESENTATION LEARNING

A Dissertation
Presented to
The Academic Faculty

by

Nishant A. Mehta

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in
Computer Science

Georgia Institute of Technology
August 2013

ON SPARSE REPRESENTATIONS AND NEW META-LEARNING PARADIGMS FOR REPRESENTATION LEARNING

Approved by:

Charles L. Isbell, Committee Chair
School of Interactive Computing
Georgia Institute of Technology

Alexander G. Gray, Advisor
School of Computational Science
and Engineering
Georgia Institute of Technology

Guy Lebanon
School of Computational Science
and Engineering
Georgia Institute of Technology

Maria-Florina Balcan
School of Computer Science
Georgia Institute of Technology

Tong Zhang
Department of Statistics
Rutgers University

Date Approved: 14 May 2013

To my parents, and the other great teachers I have had ...

ACKNOWLEDGEMENTS

My greatest thanks go to my parents for providing a truly loving and comfortable childhood and for always encouraging my scientific and less-than-scientific pursuits. Also, thanks to my brother Nirav who has served as a stupendous max-entropy paradigm.

I am grateful to my advisor Alex Gray for his initial guidance and for his ability to see connections in seemingly unrelated spaces; this granted me a higher level view of machine learning. I also am in his debt ironically not for his guidance in later years but for granting me an unprecedented amount of freedom to frame my own research plan as well as the time to see it through. I thank Melody Moore Jackson for my entry into the PhD program and early work with brain-computer interfaces; even though I have since pivoted to theoretical machine learning, I hope to one day reconnect to the brain-computer interface world. Also, I thoroughly enjoyed my brief interaction with Thad Starner, and in seeing Thad's research style firsthand, I consider myself incredibly lucky to have been able to collaborate with him.

I owe much to Nina Balcan for introducing me to learning theory and showing how exciting proofs can be. I thoroughly enjoyed the learning reading group and interacting with Yingyu, Stephen, Chris, and Ying. Also, I thank Vladimir Koltchinskii for taking the time to offer a seminar in empirical processes by a true master; the clarity of his lectures are unmatched. Finally, I think it was Krishna Balasubramanian who first proposed the fantastic idea to start a seminar on concentration of measure phenomenon, and I am very happy that we were able to put our differently shaped heads together to make it happen.

Finally, thanks to my friends, namely Dongryeol Lee, Jim Waters, Josh Dillon, Krishna, Parikshit Ram, and Ravi Ganti, for putting up with my half-baked ideas on research and jokes that probably were pretty bad either intentionally or despite my best intentions.

TABLE OF CONTENTS

LIST OF FIGURES

GLOSSARY

Notation	Description	Page List
Spaces		
$\mathcal{X}$	input space, taken as $\mathcal{X} = (B_{\mathbb{R}^d})^k$ for sparse coding	100, 117
$\mathcal{Y}$	output space (space of labels or targets)	14, 100, 117
$\mathcal{Z}$	joint space $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$	117
$\mathscr{P}$	space of probability measures on $\mathcal{Z}$	101, 120
$\mathcal{W}$	space of linear hypotheses, $\mathcal{W} = rB_{\mathbb{R}^d}$	15
$\mathcal{D}$	space of dictionaries $(B_{\mathbb{R}^d})^k$	14
$\mathcal{D}_\mu$	space of μ-incoherent dictionaries, $\mathcal{D}_\mu = \{D \in \mathcal{D} : \mu_s(D) \geq \mu\}$	27
$\mathcal{D}^{(s)}$	space of k_s-atom shared dictionaries, $\mathcal{D}^{(s)} = (B_{\mathbb{R}^d})^{k_s}$	70
$\mathcal{D}^{(e)}$	space of k_e-atom task-exclusive dictionaries, $\mathcal{D}^{(e)} = (B_{\mathbb{R}^d})^{k_e}$	70
$\mathcal{E}_\mu$	subset of $\mathcal{D}^{(s)} \times \left(\mathcal{D}^{(e)}\right)^T$ with μ-incoherent task dictionaries	83
$\mathcal{U}$	isometry class $\mathcal{U} \subset \mathbb{R}^{d \times k}$, in which all $U \in \mathcal{U}$ satisfy $U^T U = I$	38
$\mathcal{S}$	space of k-dimensional dictionaries, $\mathcal{S} = (B_{\mathbb{R}^k})^k$	38
$\mathscr{A}$	set of learning algorithms $\mathscr{A} = \{\mathcal{A}_\mathcal{H} : \mathcal{H} \in \mathbb{H}\}$	118
Probability		
Π	marginal probability measure over input space $\mathcal{X}$	14
P	joint probability measure over $\mathcal{Z}$	6, 14, 119
$\mathsf{P}_\mathbf{z}$	empirical measure with respect to m-sample $\mathbf{z}$	103, 119
P_t	probability measure for t^{th} training task	72, 101, 118
$\tilde{\mathsf{P}}_t$	probability measure for t^{th} test task	121
$\mathsf{P}_{\mathbf{z}^{(t)}}$	empirical measure with respect to m-sample $\mathbf{z}^{(t)}$	103
Q	the environment, a probability measure over $\mathscr{P}$	6, 101, 119
$\mathsf{P}\, f$	$\mathsf{E}_{(x,y)\sim\mathsf{P}}\, f(x)$	20, 119
$\mathsf{P}\, \ell(\cdot, f)$	$\mathsf{E}_{(x,y)}\, \ell\big(y, f(x)\big)$	20
$\mathsf{P}_\mathbf{z}\, f$	$\frac{1}{m}\sum_{i=1}^m f(x_i)$ or $\frac{1}{m}\sum_{i=1}^m f(z_i)$	20, 119
$\mathsf{P}_\mathbf{z}\, \ell(\cdot, f)$	$\sum_{i=1}^m \ell\big(y_i, f(x_i)\big)$	20
Samples		
$\mathbf{z}$	labeled m-sample of training data	14
$\mathbf{z}'$	second labeled m-sample (ghost sample)	24
$\mathbf{x}''$	unlabeled m-sample	21
$\mathbf{x}^{(t)}$	unlabeled m-sample for t^{th} task, drawn from product measure P_t^m, for P_t a probability measure over input space $\mathcal{X}$	72
$\mathbf{z}^{(t)}$	labeled m-sample for t^{th} task, drawn from product measure P_t^m, for P_t a probability measure over joint space $\mathcal{X} \times \mathcal{Y}$	73, 103, 118
$\mathbf{z}'^{(t)}$	labeled ghost m-sample for t^{th} task; independent copy of $\mathbf{z}^{(t)}$	83

Notation	Description	Page List
$\underline{\mathbf{z}}$	meta-sample collecting the m-samples $\mathbf{z}^{(1)}, \ldots, \mathbf{z}^{(T)}$	83, 118
$\underline{\mathbf{z}}'$	ghost meta-sample collecting ghost m-samples $\mathbf{z}'^{(1)}, \ldots, \mathbf{z}'^{(T)}$	83
Function Classes		
$\mathcal{F}$	function class for sparse coding; varies depending on unsupervised vs predictive and single vs multi-task	15, 71, 72, 74
$\mathcal{F}_\mu$	subclass of $\mathcal{F}$ with dictionary (or dictionaries) in $\mathcal{D}_\mu$	27, 85
$\mathcal{F}_{\boldsymbol{\mu}^*}$	$\{f = (D, w) \in \mathcal{F} : (\mu_s(D) \geq \mu_s^*)\,\textbf{and}\,(\mu_{2s}(D) \geq \mu_{2s}^*)$	35
$\mathcal{F}_{\boldsymbol{\mu}^*}(\mathbf{x})$	$\{f \in \mathcal{F}_{\boldsymbol{\mu}^*} : s\text{-sparse}(\varphi_D(\mathbf{x}))\,\textbf{and}\,\big[\mathsf{margin}_s(D, \mathbf{x}) > \tau\big]\}$	36
$\mathcal{F}_{\boldsymbol{\mu}^*,\eta}(\mathbf{x})$	$\{f \in \mathcal{F}_{\boldsymbol{\mu}^*} : \exists\, \tilde{\mathbf{x}} \subseteq_\eta \mathbf{x}\ s\text{-sparse}(\varphi_D(\tilde{\mathbf{x}}))\,\textbf{and}\,\big[\mathsf{margin}_s(D, \tilde{\mathbf{x}}) > \tau\big]\}$	37
Learning		
$\mathcal{H}$	hypothesis space	6, 101, 117
$\mathbf{h}$	$(h_1, \ldots, h_T) \in \mathcal{H}^T$	105
$\mathbb{H}$	family of hypothesis spaces, or meta-hypothesis space	101, 118
$\mathcal{A}_\mathcal{H}$	algorithm mapping m-samples to hypotheses in $\mathcal{H}$	6
$\mathbf{A}$	meta-learner, mapping from meta-samples to algorithms as $\mathbf{A} : (\mathcal{Z}^m)^T \to \mathscr{A}$	118
Ω	regularizer $\Omega : \mathcal{H} \times \mathcal{Z}^m \to \mathbb{R}_+$	117
$\underline{\Omega}(\cdot)$	meta-learning regularizer, defined in Chapter 4 as $\Omega : \mathbb{H} \times \cup_{\mathcal{H}\in\mathbb{H}} \mathcal{H}^T \to \mathbb{R}_+$ and defined in Chapter 5 as $\underline{\Omega}(\cdot) : \mathbb{H} \times (\mathcal{Z}^m)^T \to \mathbb{R}_+$	106, 120
$\mathcal{A}_\mathcal{H}(\cdot)$	Ω-regularized empirical risk minimization algorithm, mapping m-sample $\mathbf{z}$ to $\arg\min_{h\in\mathcal{H}} \frac{1}{m}\sum_{j=1}^m \ell\big(y_j, h(x_j)\big) + \Omega(h, \mathbf{z})$	117
$\mathsf{V}_m(a_1, \ldots, a_m)$	sample variance of $(a_1, \ldots, a_m)$, defined as $\mathsf{V}_m(a_1, \ldots, a_m) = \frac{1}{m(m-1)}\sum_{1\leq i<j\leq m} \frac{(a_i - a_j)^2}{2}$	119
$\mathsf{V}_\mathbf{z}(h)$	sample variance of the loss of hypothesis h with respect to sample $\mathbf{z}$, defined as $\mathsf{V}_\mathbf{z}(h) := \mathsf{V}_m\Big(\ell\big(y_1, h(x_1)\big), \ldots, \ell\big(y_m, h(x_m)\big)\Big)$	119
$\mathsf{V}_{\underline{\mathbf{z}}}(\mathcal{A}_\mathcal{H})$	sample variance of $\mathcal{A}_\mathcal{H}$ on $\underline{\mathbf{z}}$, i.e. the sample variance of empirical risk minimization over $\mathcal{H}$ on meta-sample $\underline{\mathbf{z}}$, defined as $\mathsf{V}_{\underline{\mathbf{z}}}(\mathcal{A}_\mathcal{H}) = \mathsf{V}_T\Big(\mathsf{P}_{\mathbf{z}^{(1)}}\,\ell\big(\cdot, \mathcal{A}_\mathcal{H}(\mathbf{z}^{(1)})\big), \ldots, \mathsf{P}_{\mathbf{z}^{(T)}}\,\ell\big(\cdot, \mathcal{A}_\mathcal{H}(\mathbf{z}^{(T)})\big)\Big)$	119
Parameters		
D	dictionary of k atoms in $\mathbb{R}^d$	2, 12
D_j	j^{th} atom (column) of dictionary (matrix) D	12
$\bar{D}^{(t)}$	dictionary for t^{th} task in multi-task sparse coding model, with $\bar{D}^{(t)} = (D^{(0)}\ D^{(t)})$	70
$D^{(0)}$	shared subdictionary in multi-task sparse coding	70
$D^{(t)}$	exclusive subdictionary for t^{th} task in multi-task sparse coding	70
W_t	linear hypothesis for t^{th} task	73
Losses		
ℓ	loss function $\ell : \mathcal{Y} \times \mathbb{R} \to [0, b]$	15, 117
$\ell(\cdot, f)$	loss-composed function of $\mathcal{Y}\times\mathbb{R}^d$, acting as $(y, x) \mapsto \ell(y, f(x))$	20
$\ell \circ \mathcal{F}$	loss class of functions, $\{\ell(\cdot, f) : f \in \mathcal{F}\}$	20
$\hat{\ell}_t(h_t)$	empirical risk for hypothesis h_t on task $t \in [T]$, defined as $\hat{\ell}_t(h_t) = \sum_{i=1}^m \ell(y_i^{(t)}, h_t(x_i^{(t)}))$	105
$\hat{\boldsymbol{\ell}}(\mathbf{h})$	vector of empirical risks, $\hat{\boldsymbol{\ell}}(\mathbf{h}) := \big(\hat{\ell}_1(h_1), \ldots, \hat{\ell}_T(h_T)\big)$	105
Sparse Coding		

Notation	Description	Page List
φ_D	sparse auto-encoder, $\varphi_D(x) = \arg\min_z \|x - Dz\|_2^2 + \lambda\|z\|_1$	2, 14
$\mu_s(D)$	s-incoherence: minimum $(\sigma_s(D))^2$ among s-atom subdictionaries of D	16
$\mathsf{margin}_s(D, x)$	$\max_{\substack{\mathcal{I}\subseteq[k]\\|\mathcal{I}|=k-s}} \min_{j\in\mathcal{I}} \left\{\lambda - \left\|\langle D_j, x_i - D\varphi_D(x_i)\rangle\right\|\right\}$	17
$\mathsf{margin}_s(D, \mathbf{x})$	$\min_{x_i\in\mathbf{x}} \mathsf{margin}_s(D, x_i)$	17
s-sparse$(\varphi_D(\mathbf{x}))$	for all $x_i \in \mathbf{x}$, $\|\varphi_D(x_i)\|_0 \le s$	17
$\mathbf{f}$	$\mathbf{f} = (f_1, \ldots, f_T) \in \mathcal{F}$ for multi-task (predictive) sparse coding	72
$f_{D,w}$	functions $f_{D,w}(x, y) = \ell(y, \langle w, \varphi_D(x)\rangle)$ for multi-task predictive sparse coding	74
Feature Map		
φ_θ	feature map, or preprocessor, $\varphi_\theta : \mathcal{X} \to \mathbb{R}^k$	123
Θ	metric space indexing elements of $\mathbb{H}$, as $\mathcal{H}_\theta \in \mathbb{H}$ for $\theta \in \Theta$	123
C	Lipschitz constant for φ_θ when viewed as a function of θ	123
$\mathcal{N}(\Theta, \varepsilon)$	ε-covering number for Θ using $\|\cdot\|$	123
Scalars		
m	# training points per task	14
T	# training tasks	70, 101, 118
$\tilde{T}$	# test tasks	121
d	ambient dimension of input space (dimensionality of x)	12
k	# atoms for sparse coding; # learned features in general	12
k_s	# shared atoms for multi-task sparse coding	70
k_e	# exclusive atoms (per task) for multi-task sparse coding	70
b	upper bound on range of ℓ	15, 117
L	Lipschitz constant for ℓ as a function of its second argument	15, 117
r	upper bound on radius of linear hypotheses in $\mathcal{W}$	15
$\iota = \iota(\lambda, \varepsilon)$	permissible radius of perturbation (PRP), $\iota(\lambda, \varepsilon) = \sqrt{243\varepsilon/\lambda}$	27
Complexities		
σ_i	Rademacher random variable, uniform on $\{-1, 1\}$	24, 122
γ_i	standard normal random variable, distributed as $\mathcal{N}(0, 1)$	24
$\mathcal{R}_{m\|\mathbf{x}}(\mathcal{F})$	conditional Rademacher average, $\frac{2}{m}\mathsf{E}_{\boldsymbol{\sigma}} \sup_{f\in\mathcal{F}} \sum_{i=1}^m \sigma_i f(x_i)$	25, 122
$\mathcal{G}_{m\|\mathbf{x}}(\mathcal{F})$	conditional Gaussian average, $\frac{2}{m}\mathsf{E}_{\boldsymbol{\gamma}} \sup_{f\in\mathcal{F}} \sum_{i=1}^m \gamma_i f(x_i)$	25
$\mathcal{R}_m(\mathcal{F})$	Rademacher complexity, $\mathsf{E}\,\mathcal{R}_{m\|\mathbf{x}}(\mathcal{F})$	104
$\overline{\mathcal{R}}_m(\mathcal{F})$	uniform Rademacher complexity, $\overline{\mathcal{R}}_m(\mathcal{F}) = \sup_{\mathbf{x}\in\mathcal{X}^m} \mathcal{R}_{m\|\mathbf{x}}(\mathcal{F})$	122
Miscellaneous		
$[n]$	$\{1, 2, \ldots, n\}$	16
$\rho B_{\mathbb{R}^d}$	ℓ_2-ball in $\mathbb{R}^d$ of radius ρ	14
$\mathsf{supp}(t)$	index support set, $\{i \in [k] : t_i \ne 0\}$ for $t \in \mathbb{R}^k$	16
$\sigma_s(A)$	the s^{th} singular value of A	16
$\|S\|_{2,s}$	The s-restricted 2-norm: $\sup_{\{t\in\mathbb{R}^n:\|t\|=1,\|\mathsf{supp}(t)\|\le s\}} \|St\|_2$	38
transfer risk	$\mathsf{E}_{\mathsf{P}\sim\mathsf{Q}} \mathsf{E}_{\mathbf{z}\sim\mathsf{P}^m} \mathsf{E}_{(x,y)\sim\mathsf{P}} \ell\big(y, \mathcal{A}_{\mathcal{H}}(\mathbf{z})(x)\big)$	6, 120
$\hat{f}_{\mathbf{z}}$	hypothesis returned by learner from $\mathbf{z}$	20
$\tilde{\mathbf{x}} \subseteq_\eta \mathbf{x}$	$\tilde{\mathbf{x}}$ is a subset of $\mathbf{x}$ with at most η elements of $\mathbf{x}$ removed	27
$\underline{\tilde{\mathbf{x}}} \subseteq_\eta \underline{\mathbf{x}}$	$\tilde{\mathbf{x}}$ is a meta-sample collecting samples $\tilde{\mathbf{x}}^{(1)}, \ldots, \tilde{\mathbf{x}}^{(T)}$, with $\tilde{\mathbf{x}}^{(t)} \subset \mathbf{x}^{(t)}$ for each $t \in [T]$, and cumulatively at most η points of $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(T)}$ are not present in $\tilde{\mathbf{x}}^{(1)}, \ldots, \tilde{\mathbf{x}}^{(T)}$	83

SUMMARY

This dissertation is a confluence of representation learning and meta-learning, with a special focus on sparse representations. Meta-learning is fundamental to machine learning, and it translates to learning to learn itself. The presentation unfolds in two parts: Part I concerns sparse representations while Part II studies new multi-task and meta-learning paradigms for representation learning. Our main pursuit with regards to sparse representation learning are learning theoretic bounds to support a supervised dictionary learning model for Lasso-style sparse coding. Such predictive sparse coding algorithms have been applied with much success in the literature, and even more common have been applications of unsupervised sparse coding followed by supervised linear hypothesis learning. The predictive sparse coding generalization error bounds presented in Chapter 2 are the first, and in the process of their derivation, we introduce a new geometric quantity for describing sparse coding dictionaries. Our analysis also leads to a result of independent interest: a fundamental stability result for the Lasso that shows the stability of the solution vector to design matrix perturbations. In the context of sparse coding, this result is dubbed the Sparse Coding Stability Theorem, and it shows the stability of the sparse codes to dictionary perturbations. The generalization bounds for predictive sparse coding handle the overcomplete setting, where there are more learned features than the original dimension, and the high/infinite-dimensional setting in which useful bounds are independent of the ambient dimension. Chapter 3 introduces a new multi-task model for (unsupervised) sparse coding and predictive sparse coding, allowing for one dictionary per task but with some pre-specified sharing of certain atoms by all the tasks. We analyze these new models in the overcomplete setting. For sparse coding, we develop generalization error bounds on the task-average of the reconstruction error, while

for predictive sparse coding we extend the single-task predictive sparse coding bound for the overcomplete setting to a bound on the task-average of the prediction risk.

After looking in-depth at learning sparse representations, in Part II we zoom out to the more general setting of representation learning paradigms. The results in this part focus on obtaining new types of learning guarantees for the future performance of a meta-learner's learned representation on new tasks encountered in an environment. Chapter 4 introduces minimax multi-task learning, as well as a general loss-compositional framework for minimizing a large class of symmetric functions of the empirical risks. This chapter also provides a high probability learning guarantee on the future performance of a meta-learner on individual tasks encountered in the future, the first of its kind; here, the probability of failure of the guarantee decays as $1/T$ for T training tasks. Next, Chapter 5 introduces sample variance penalized meta-learning, in which the meta-learner minimizes the sum of the task average of the training tasks' empirical risks and a penalty on the sample variance of the empirical risks. Controlling this sample variance affords two advantages. First, one potentially can obtain a faster rate of decrease for upper bounds on the transfer risk. Second, the sample variance can be exploited to apply an empirical version of Chebyshev's inequality, ultimately yielding a high probability $\left(1 - O\left(1/\tilde{T}^2\right)\right)$ guarantee on the meta-learner's average performance over a new draw of $\tilde{T}$ test tasks; this guarantee also improves as the sample variance of the empirical risks decreases. In the case when a meta-learner's sole intent is feature selection, a forward step-wise algorithm is presented for direct minimization of the sample variance penalized objective. However, this objective generally is not convex and results in a troublesome bilevel program; in response, we introduce a quite natural convex relaxation for use with more general meta-learning models. Finally, Chapter 6 summarizes the contributions of the thesis and provides a forward looking view of representation learning.

CHAPTER 1

INTRODUCTION

1.1 My Thesis

> The stability of performance of a learned representation can give rise to new generalization guarantees for supervised learners that learn sparse representations and, more generally, can motivate new meta-learning paradigms for representation learning supporting tighter generalization guarantees for a meta-learner's performance on new learning tasks.

Given the "right" features, learning is easy. This statement from machine learning folklore beguiles with its simplicity, but the trick lies in learning to find those features. The classical era of machine learning has concentrated on individual learning tasks such as regression and binary classification, and the past 40 years have seen considerable progress on this front. In the standard setup, features are fixed a priori with respect to the data, preferably with the aid of domain knowledge, by selecting from a variety of possible representations. One popular and tremendously-used class of representations are those arising from kernels: selecting a particular Mercer kernel corresponds to a particular choice of transformation of the original features.

Evidence is gathering that the classical era has reached its apex and that the field is enjoying a transition to a modern era of machine learning. In this modern era, representations, or features, are no longer fixed a priori; they can in fact be learned either by using a large amount of data for a single task, or more commonly, by using a large collective pool of data from multiple tasks. At the time of this writing, the first International Conference

on Learning Representations (ICLR) is about to kick off. Since the year 2000, there have been a number of theoretical results on multi-task and meta-learning[1] of representations (Baxter, 2000; Ben-David and Schuller, 2003; Ando and Zhang, 2005; Maurer, 2005, 2006, 2009). Additionally, the seminal work of Lanckriet et al. (2004) introduced the notion of *learning the kernel* and hence learning a kernel-induced representation itself. These works have started to build a theoretical foundation for representation learning; however, after the fundamental learning to learn work of Baxter (2000), it appears that the only venture into this space that applies to general models is due to Maurer (2005). On this note, one of the goals of this dissertation is to establish new, general learning paradigms for representation learning. In doing so we will answer previously unasked yet very important questions about the kinds of guarantees that can be made about representation learning and meta-learning.

1.2 Sparse representations

Before discussing the new paradigms for multi-task and meta-learning, many concepts will become more concrete by first discussing another goal of this thesis. This goal is to investigate the generalization properties of a particular, very expressive class of representations: that of sparse representations. Sparse representations have long captivated the worlds of neuroscience, signal processing, statistics, and machine learning. Some have suggested that sparse coding might be used in our own neurobiological hardware (Olshausen and Field, 1997), and sparse representations have exhibited remarkable performance on a variety of high-dimensional learning tasks. Nevertheless, fundamental theoretical questions on learning sparse representations remain open. This thesis focuses on a popular and mathematically elegant approach to sparse representations, called sparse coding. In sparse coding, a choice of representation amounts to a choice of a sparse auto-encoder $\varphi_D : \mathbb{R}^d \to \mathbb{R}^k$, where $D \in \mathbb{R}^{d \times k}$ is a dictionary of k columns each lying in the unit ℓ_2-ball of $\mathbb{R}^d$. In particular, we consider the following sparse auto-encoder, induced by the Lasso (Tibshirani, 1996):

$$\varphi_D(x) := \operatorname*{arg\,min}_{z \in \mathbb{R}^k} \frac{1}{2}\|x - Dz\|_2^2 + \lambda\|z\|_1.$$

[1] Meta-learning is synonymous with learning to learn, and a formal definition will come later.

Sparse coding via $\varphi_D(\cdot)$ often is called ℓ_1 sparse coding; the Lasso imposes ℓ_1 regularization on the codes, promoting sparsity for well-studied algebraic and geometric reasons (Tropp, 2006; Donoho and Elad, 2003).

For this sparse auto-encoder, we first and foremost ask what is the sample complexity of learning these sparse representations for supervised tasks? That is, if the ideal objective is the following supervised objective for some loss function ℓ and distribution P over inputs and labels:

$$\mathsf{E}_{(x,y)\sim\mathsf{P}}\,\ell\big(y, \langle w, \varphi_D(x)\rangle\big), \tag{1.1}$$

then we ask at what rate does the empirical version of (1.1) converge to (1.1) itself, for any hypothesis (D, w) in some suitable class. Furthermore, how do properties of the codes, such as sparsity, and properties of the dictionary interact to govern the complexity of the learning problem? The sample complexity of learning these sparse representations in the unsupervised setting recently has been studied by Maurer and Pontil (2010) and Vainsencher et al. (2011). In the unsupervised setting, the ideal objective is simply to minimize the expectation of the ℓ_2 reconstruction error $\|x - D\varphi_D(x)\|_2$. In this setting, when the dictionary has finitely many columns it turns out that obtaining generalization error bounds that decay at the rate $\sqrt{dk/m}$ (for m points) is simple, not requiring use of the sparsity of the codes nor various geometric properties of dictionaries that come into play in the literature for sparse recovery[2]. As we will see in Chapter 2, the predictive setting offers additional challenges not seen in the unsupervised setting, and those challenges can be met by using ideas somewhat similar to the ones from the sparse recovery literature.

Learning a rich representation often does not come cheaply in terms of data. As a result, representation learning naturally may benefit from learning frameworks such as multi-task learning and meta-learning; these frameworks can leverage many low sample size datasets to jointly learn a shared representation of high complexity. Indeed, in many instances, there is not enough data for any single task in order to learn a sparse representation, at least according to the best known upper bounds on the estimation error (established by

[2]Tropp (2006) provides an excellent and eloquent coverage of these properties.

Maurer and Pontil (2010) and Vainsencher et al. (2011) in the unsupervised setting and Mehta and Gray (2013) in the supervised setting). Hence, in Chapter 3 we explore whether sparse representations can be learned, either for unsupervised tasks or supervised tasks, using many low sample size tasks. Rather than using a single sparse representation (or single dictionary) for multiple tasks, we introduce and analyze a more general class of representations wherein each task has some flexibility in adapting a common representation to that task's idiosyncrasies. In particular, the learner will learn a separate dictionary for each task, with all but the last few columns of each dictionary shared by all the tasks. This sharing model, shown in Figure 1.1, allows the last few columns of each task's dictionary to be adapted to that task.

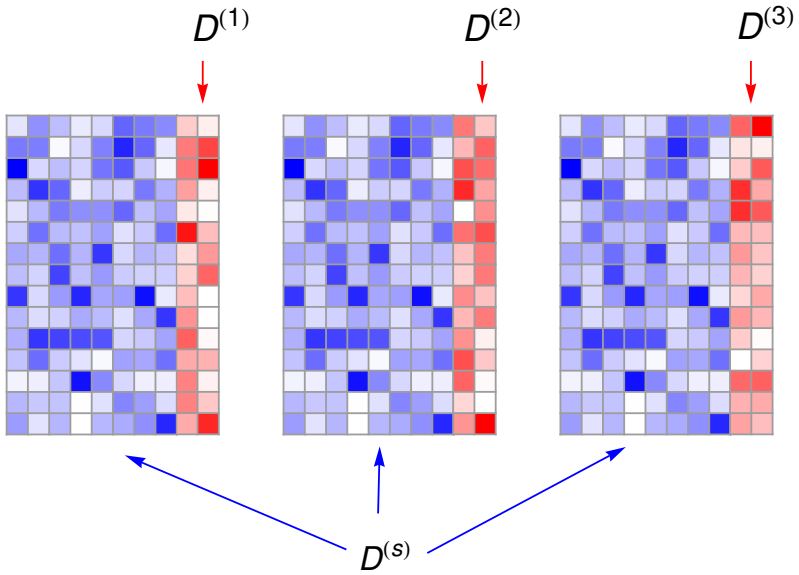


Figure 1.1: Flexible sharing model for multi-task sparse coding. Dictionaries for three tasks are shown. Each task's dictionary shares a common subdictionary $D^{(s)}$ (tinted blue) represented by the leftmost eight atoms. In addition, each task has its own small task-specific subdictionary (tinted red), consisting of either $D^{(1)}$, $D^{(2)}$, or $D^{(3)}$; these subdictionaries are represented by the rightmost two atoms.

This multi-task sparse coding model can operate in both the unsupervised and supervised settings, giving rise to multi-task sparse coding and multi-task predictive sparse coding respectively. While the ideas for the unsupervised multi-task sparse coding model existed, in some sense, in a multi-class classification work of Ramirez et al. (2010), their model never explicitly shares atoms between the different dictionaries, and their multi-class formulation

is different from a typical multi-task formulation. Importantly, the atomic sharing in the multi-task sparse coding model can allow for transfer between representations. It appears that the analysis of this sort of model in Chapter 3 is the first learning theoretic study of learning separate representations for different tasks while reducing the complexity of learning by ensuring that all representation are "close."

In the supervised setting, the shared sparse representation model becomes a multi-task extension of predictive sparse coding in which "similar" sparse coding representations are learned for each task (as per the dictionary sharing model), and a separate linear predictor also is learned for each task. As a result, it is possible to let the estimation error due to the large shared subdictionary decay with the overall number of points *among all the tasks*, while the contribution to the estimation error due to each task's specific subdictionary and linear hypothesis decays with the number of points per task.

1.3 New representation learning paradigms

We now switch to a wide-angle lens and pivot to new multi-task and meta-learning paradigms that lend themselves to unprecedented kinds of generalization guarantees. The history of multi-task learning started with the work of Rich Caruana and his approach toward using inductive transfer to improve a learner's generalization by learning related tasks together (Caruana, 1997). Meta-learning, or learning to learn[3], emanated from Jonathan Baxter's work on learning inductive bias using multi-task learning in order to provide guarantees on a meta-learner's performance on future tasks (Baxter, 2000). There is an important distinction between the multi-task learning and meta-learning settings. In multi-task learning, the set of tasks is fixed, possibly arbitrarily, and identical at training and test time. However, in Baxter's learning to learn setting, the training tasks (from which a representation is learned) are drawn iid from an environment of tasks, and the resulting learner will be tested on new, test tasks drawn iid from the same environment.

Critically, to date only a few kinds of generalization guarantees have been made in the multi-task and meta-learning settings. In fact, nearly all works in these settings have focused

[3]Meta-learning and learning to learning will be both refer to the same concept in this thesis, the concept itself being the learning to learn setup introduced by Baxter (2000).

on just two types of guarantees: in multi-task learning, the focus has been upper confidence bounds on the the average estimation error, whereas the focus in the meta-learning setting has been upper confidence bounds on the expected true risk, or *transfer risk* (defined below in (1.2)). While obtaining such bounds on average/expected performance across tasks certainly is important, such bounds provide an imprecise view on the performance of a learner on individual tasks.

In multi-task learning, there are situations where one wants to ensure that the true risk the learner suffers on any task will not be large. For example, this is true when the learner is to learn a representation from T tasks, and at test time the learner is tested on only a single one of those tasks, possibly chosen adversarially. In such situations, rather than obtaining a bound on the task-average of the learner's true risk, it is important to obtain *task-wise* learning bounds: that is, a bound on the true risk of the learner for each task. To date, the only general work that has produced task-wise learning bounds operates in a scenario where the average estimation error and task-wise estimation become equivalent (Ben-David and Schuller, 2003). This thesis attempts to fill in this apparent void.

In the meta-learning setting, the typical quantity whose minimum is sought is the transfer risk, defined as

$$\mathsf{E}_{\mathsf{P}\sim\mathsf{Q}}\,\mathsf{E}_{\mathbf{z}\sim\mathsf{P}^m}\,\mathsf{E}_{(x,y)\sim\mathsf{P}}\,\ell\big(y,\mathcal{A}_{\mathcal{H}}(\mathbf{z})(x)\big), \tag{1.2}$$

where:

- Algorithm $\mathcal{A}_{\mathcal{H}}$ takes an m-sample of m labeled points $\mathbf{z} = (x_1, y_1), \ldots, (x_m, y_m)$ and returns a hypothesis in hypothesis space $\mathcal{H}$;
- Q is the environment, a probability measure over task probability measures;
- Each probability measure P is a distribution over labeled examples $z = (x, y)$.

To minimize this objective, a meta-learner attempts to learn a hypothesis space $\mathcal{H}$ (which can result from learning a particular representation) using samples from multiple training tasks such that algorithm $\mathcal{A}_{\mathcal{H}}$ performs well on future tasks. However, if an agent is sufficiently risk-averse, guarantees that the transfer risk of a hypothesis space $\mathcal{H}$ is small are

insufficient. Similar to the single-task learning setting, an agent might require tail bounds on its future performance. That is, it might need guarantees of the form:

> With high probability, the true risk $\mathcal{A}_{\mathcal{H}}$ will suffer when trained on a new task drawn from Q does not exceed some level γ.

Guarantees of this form are about the performance of a meta-learning on a single random task encountered in the future, and such guarantees are unprecedented in the meta-learning literature. Observe that while a simple application of Markov's inequality using a bound on the transfer risk certainly provides some bound on the tail, the level of concentration obtained does not increase with the number of training tasks T.

In Chapter 4, we establish one sort of high probability guarantee on the future performance of a meta-learner's returned learning algorithm. The nature of the bound suggests a new paradigm for multi-task learning, called *minimax multi-task learning*, wherein the goal is to learn a representation capable of minimizing the maximum of the empirical risks of the training tasks. In addition to formulating minimax multi-task learning, we also frame two different spectra of multi-task learning setups. The first spectrum involves the minimization of ℓ_p-norms of the vector of empirical risks, for p in the range $[1, \infty)$. Whereas minimizing the ℓ_1 norm of the empirical risks corresponds to minimizing their mean, minimizing the ℓ_∞ norm corresponds to minimizing their maximum. The second spectrum is a continuous family of relaxations of minimax multi-task learning, where more relaxation translates to a softening of the maximum. When fully relaxed, the learning formulation translates to the classical minimization of the mean of the empirical risks.

Chapter 5 provides another sort of high probability bound on the future performance of a meta-learner on new tasks. This tail bound relies on an empirical Bernstein bound, and it exploits the sample variance of the empirical risks encountered by the selected learning algorithm on the training tasks. As such, when the sample variance of the empirical risks is low, the bound can provide sharper guarantees about the average performance of the learner over sets of test tasks drawn from the environment. The empirical Bernstein bound used to develop the tail bound for the learner's future performance also enables sharper upper confidence bounds on the transfer risk than what have previously been established in a

general meta-learning setting. These dual benefits suggest a new meta-learning framework, *sample variance penalized meta learning*, that implicitly values the stability of a learned representation's performance across tasks drawn from the environment.

Below is a brief summary of the main contributions of this thesis, split up by chapter:

1.4 Summary of contributions

Chapter 2

Sparse Coding Stability Theorem. This is a fundamental stability result for the Lasso. The result shows sufficient conditions under which the optimal solution z^* to the Lasso problem,

$$\arg\min_{z \in \mathbb{R}^k} \frac{1}{2}\|x - Dz\|_2^2 + \lambda\|z\|_1,$$

is stable with respect to perturbations to the dictionary D. This appears to be the first result of this kind. Other ostensibly similar results only apply to the case when the reconstruction error $\|x - Dz^*\|_2^2$ of the optimal solution is small (Herman and Strohmer, 2010)[4], whereas the conditions we use are independent of the magnitude of the reconstruction error. In statistical machine learning applications where the Lasso is being used for denoising, the running assumption is that the residual will be non-trivial, and so this distinction is very important.

Predictive sparse coding learning bound for overcomplete setting. This is the first estimation error bound for predictive sparse coding, and in particular, the bound applies to the overcomplete setting where the number of atoms k in the dictionary exceeds the ambient dimension d. Sparse representations based on ℓ_1 sparse coding have enjoyed widespread use in the machine learning literature, and various techniques, both unsupervised (Raina et al., 2007) and supervised (Mairal et al., 2009, 2012), have been used to learn the dictionary for sparse coding for later use on supervised tasks. The bound presented es-

[4]It should be noted that Herman and Strohmer (2010) actually analyze the ε-error-tolerant basis pursuit problem $\arg\min_{z \in \mathbb{R}^k} \|z\|_1 \quad \text{s.t. } \|x - Dz\|_2 \leq \varepsilon$.

sentially decays at the rate $\sqrt{\frac{dk}{m}}$, with additional dependence on the stability properties of the particular sparse auto-encoder learned on the training sample. Hence, the bound is both data and algorithm dependent.

Predictive sparse coding learning bound for infinite-dimensional setting. This bound is independent of the ambient dimension of the data, and hence it is useful in situations where d is very large or even infinite. The bound essentially decays at the rate $\sqrt{\frac{k^2 s}{m}}$, where s is some notion of sparsity of the codes induced from the training sample and the learned auto-encoder. Similar to the overcomplete setting, this bound also is data and algorithm dependent.

Chapter 3

Multi-task (unsupervised) sparse coding learning bound. A flexible multi-task extension of the sparse coding model for unsupervised tasks, where the goal of each task is to minimize the ℓ_2 reconstruction error. Our results include a multi-task extension of already existing estimation error bounds for for single-task sparse coding in the overcomplete setting.

Multi-task predictive sparse coding learning bound. A similar multi-task extension of the predictive sparse coding model. The main result is a multi-task extension of the single-task predictive sparse coding estimation error bound in the overcomplete setting.

Chapter 4

Tail bounds for future test risk of meta-learner using maximum of empirical risks on training tasks. The first tail bound on the true risk of a meta-learner on future tasks which decays with the number of training tasks. The presented result bounds the probability that the true risk the meta-learner will suffer on a new random task will be much larger than the maximum of the empirical risks on the training tasks; hence, this bound motivates minimax multi-task learning.

Minimax multi-task learning. A new multi-task and meta-learning framework that minimizes the maximum of the (training) tasks' empirical risks.

ℓ_p-multi-task learning. A new multi-task and meta-learning paradigm that minimizes ℓ_p norms of the tasks' empirical risks.

α-relaxed minimax multi-task learning. A relaxation of minimax multi-task learning that allows a softer notion of maximum and interpolates between minimax multi-task learning and classical multi-task learning.

Chapter 5

Better concentration for transfer risk, using sample variance of empirical risks. An upper confidence bound on the transfer risk of a meta-learner, unique in its dependence on the sample variance of the empirical risks on the training tasks. In some instances, this bound can be much tighter than previous bounds. The proof of this bound relies upon an empirical Bernstein bound showing concentration of the empirical risks to the expectation of the empirical risk, where this expectation is with respect to the random task and the random training sample for that task.

Tail bounds for future test risk of meta-learner using sample variance of empirical risks. A tail bound on the average true risk of a meta-learner on a random finite set of future tasks, relying on the sample variance of the empirical risks on the training tasks. This bound arises via an application of an empirical Chebyshev bound — Chebyshev's inequality using a high probability bound on the variance's deviation from the sample variance.

Sample variance penalized meta-learning. A new meta-learning framework that minimizes the sum of the mean of the empirical risks with a penalty on the sample variance of the empirical risks. This problem is non-convex and, for reasons explained in Chapter 5, not amenable to non-convex optimization approaches. We show a suitable convex relaxation with good theoretical qualities.

Part I

SPARSE REPRESENTATIONS

CHAPTER 2

PREDICTIVE SPARSE AUTO-ENCODERS

2.1 Introduction

Learning architectures such as the support vector machine and other linear predictors enjoy strong theoretical properties (Steinwart and Christmann, 2008; Kakade et al., 2009), but a learning-theoretic understanding of many more complex learning architectures is lacking. Predictive methods based on *sparse coding* recently have emerged which simultaneously learn a data representation via a nonlinear encoding scheme and an estimator linear in that representation (Bradley and Bagnell, 2009b; Mairal et al., 2012, 2009). A sparse coding representation $z \in \mathbb{R}^k$ of a data point $x \in \mathbb{R}^d$ is learned by representing x as a sparse linear combination of k *atoms* $D_j \in \mathbb{R}^d$ of a *dictionary* $D = (D_1, \ldots, D_k) \in \mathbb{R}^{d \times k}$. In the coding $x \approx \sum_{j=1}^{k} z_j D_j$, all but a few z_j are zero.

Predictive sparse coding methods such as Mairal et al. (2012)'s *task-driven dictionary learning* recently have achieved state-of-the-art results on many tasks, including the MNIST digits task. Whereas standard sparse coding minimizes an unsupervised, reconstructive ℓ_2 loss, predictive sparse coding seeks to minimize a supervised loss by optimizing a dictionary and a linear predictor that operates on encodings to that dictionary. There is much empirical evidence that sparse coding can provide good abstraction by finding higher-level representations which are useful in predictive tasks (Yu et al., 2009b). Intuitively, the power of *prediction-driven* dictionaries is that they pack more atoms in parts of the representational space where the prediction task is more difficult. However, despite the empirical successes of predictive sparse coding, it is unknown how well it generalize in a theoretical

sense.

In this chapter, we develop what to our knowledge are the first generalization error bounds for predictive sparse coding algorithms; in particular, we focus on ℓ_1-*regularized sparse coding*. Maurer and Pontil (2010) and Vainsencher et al. (2011) previously established generalization bounds for the classical, reconstructive sparse coding setting. Extending their analysis to the predictive setting introduces certain difficulties related to the richness of the class of sparse encoders. Whereas in the reconstructive setting, this complexity can be controlled directly by exploiting the stability of the *reconstruction error* to dictionary perturbations, in the predictive setting it appears that the complexity hinges upon the stability of the *sparse codes themselves* to dictionary perturbations. This latter notion of stability is much harder to prove; moreover, it can be realized only with additional assumptions which depend on the dictionary, the data, and their interaction (see Theorem 2.4). Furthermore, when the assumptions hold for the learned dictionary and data, we also need to guarantee that the assumptions hold on a newly drawn sample.

Contributions We provide learning bounds for two core scenarios in predictive sparse coding: the *overcomplete setting* where the dictionary size, or number of learned features, k exceeds the ambient dimension d; and the infinite-dimensional setting where only dimension-free bounds are acceptable. Both bounds hold provided the size m of the training sample is large enough, where the critical size for the bounds to kick in depends on a certain notion of stability of the learned representation. This chapter's core contributions are:

1. Under mild conditions, a stability bound for the Lasso (Tibshirani, 1996) under dictionary perturbations (Theorem 2.4).

2. In the overcomplete setting, a learning bound that is essentially of order $\sqrt{\frac{dk}{m}} + \frac{\sqrt{s}}{\lambda \mu_s(D)}$, where each sparse code has at most s non-zero components (Theorem 2.5). The term $\frac{1}{\mu_s(D)}$ is the inverse s-incoherence (see Definition 2.1) and is roughly the worst condition number among all linear systems induced by taking s columns of D.

3. In the infinite-dimensional setting, a learning bound that is *independent* of the dimension of the data (Theorem 2.6); this bound is essentially of order $\frac{1}{\mu_{2s}(D)} \sqrt{\frac{k^2 s}{m}}$.

The stability of the sparse codes are absolutely crucial to this work. Proving that the notion of stability of contribution 1 holds is quite difficult because the Lasso objective (see (2.1) below) is not strongly convex in general. Consequently, much of the technical difficulty of this work is owed to finding conditions under which the Lasso is stable under dictionary perturbations and proving that when these conditions hold with respect to the learned hypothesis and the training sample, they also hold with respect to a future sample.

For convenience, we have collected all of the various notation of this chapter in a glossary.

2.1.1 The predictive sparse coding problem

Let P be a probability measure over $B_{\mathbb{R}^d} \times \mathcal{Y}$, the product of an input space $B_{\mathbb{R}^d}$ (the unit ball of $\mathbb{R}^d$) and a space $\mathcal{Y}$ of univariate labels; examples of $\mathcal{Y}$ include a bounded subset of $\mathbb{R}$ for regression and $\{-1, 1\}$ for classification. Let $\mathbf{z} = (z_1, \ldots, z_m)$ be a sample of m points drawn iid from P, where each labeled point z_i equals (x_i, y_i) for $x_i \in B_{\mathbb{R}^d}$ and $y_i \in \mathcal{Y}$. In the reconstructive setting, labels are not of interest and we can just as well consider an unlabeled sample $\mathbf{x}$ of m points drawn iid from the marginal probability measure Π on $B_{\mathbb{R}^d}$.

The sparse coding problem is to represent each point x_i as a sparse linear combination of k basis vectors, or *atoms* $D_1, \ldots, D_k$. The atoms form the columns of a *dictionary* D living in a space of dictionaries $\mathcal{D} := (B_{\mathbb{R}^d})^k$, for $D_i = (D_i^1, \ldots, D_i^d)^T$ in the unit ℓ_2 ball. An encoder φ_D can be used to express ℓ_1 sparse coding:

$$\varphi_D(x) := \arg\min_z \frac{1}{2}\|x - Dz\|_2^2 + \lambda\|z\|_1; \tag{2.1}$$

hence, encoding x as $\varphi_D(x)$ amounts to solving a Lasso problem. The reconstructive ℓ_1 sparse coding objective is then

$$\min_{D \in \mathcal{D}} \mathsf{E}_{x \sim \Pi} \frac{1}{2}\|x - D\varphi_D(x)\|_2^2 + \lambda\|\varphi_D(x)\|_1,$$

Generalization bounds for the empirical risk minimization (ERM) variant of this objective have been established. In the infinite-dimensional setting, Maurer and Pontil (2010) showed[1]

[1]To see this, take Theorem 1.2 of Maurer and Pontil (2010) with $Y = \{y \in \mathbb{R}^k : \|y\|_1 < \frac{1}{\lambda}\}$ and $\mathcal{T} = \{T : \mathbb{R}^k \to \mathbb{R}^d : \|Te_j\| \leq 1, j \in [k]\}$, so that $\|\mathcal{T}\|_Y \leq \frac{1}{\lambda}$.

the following bound:

$$\Pr_{\mathbf{x}}\left\{\sup_{D\in\mathcal{D}} \mathrm{E}_{x\sim\Pi} f_D(x) - \frac{1}{m}\sum_{i=1}^{m} f_D(x_i) \;\geq\; \frac{k}{\sqrt{m}}\left(\frac{14}{\lambda} + \frac{1}{2}\sqrt{\log\left(16m/\lambda^2\right)}\right) + \sqrt{\frac{\log(1/\delta)}{2m}}\right\} \leq \delta. \tag{2.2}$$

where $f_D(x) := \min_{z\in\mathbb{R}^k} \|x - Dz\|_2^2 + \lambda\|z\|_1$. This bound is *independent* of the dimension d and hence useful when $d \gg k$, as in general Hilbert spaces. Vainsencher et al. (2011) handled the overcomplete setting, producing a bound that is $O\left(\sqrt{dk/m}\right)$ as well as fast rates of $O(dk/m)$, with only logarithmic dependence on $\frac{1}{\lambda}$.

Predictive sparse coding, introduced by Mairal et al. (2012), minimizes a supervised loss with respect to a representation and an estimator linear in the representation. Let $\mathcal{W}$ be a space of linear hypotheses with $\mathcal{W} := rB_{\mathbb{R}^k}$, the ball in $\mathbb{R}^k$ scaled to radius r. A predictive sparse coding hypothesis function f is identified by $f = (D, w) \in \mathcal{D}\times\mathcal{W}$ and defined as $f(x) = \langle w, \varphi_D(x)\rangle$. The function class $\mathcal{F}$ is the set of such hypotheses. The loss will be measured via $\ell : \mathcal{Y}\times\mathbb{R} \to [0, b]$, $b > 0$, a bounded loss function that is L-Lipschitz in its second argument.

The predictive sparse coding objective is[2]

$$\min_{D\in\mathcal{D}, w\in\mathcal{W}} \mathrm{E}_{(x,y)\sim\mathrm{P}}\, \ell(y, \langle w, \varphi_D(x)\rangle) + \frac{1}{r}\|w\|_2^2; \tag{2.3}$$

In this work, we analyze the ERM variant of (2.3):

$$\min_{D\in\mathcal{D}, w\in\mathcal{W}} \frac{1}{m}\sum_{i=1}^{m} \ell(y_i, \langle w, \varphi_D(x_i)\rangle) + \frac{1}{r}\|w\|_2^2. \tag{2.4}$$

This objective is not convex, and it is unclear how to find global minima, so a priori we cannot say whether an optimal or nearly optimal hypothesis will be returned by any learning algorithm. However, we can and will bet on certain sparsity-related stability properties holding with respect to the learned hypothesis and the training sample. Consequently, all the presented learning bounds will hold uniformly *not over the set of all hypotheses*

[2] While the focus of this chapter is (2.3), formally the predictive sparse coding framework admits swapping out the squared ℓ_2 norm regularizer on w for any other regularizer.

but rather potentially much smaller random subclasses of hypotheses. Additionally, the presented bounds will be algorithm-independent[3], although algorithm design can influence the learned hypothesis's stability and hence the best a posteriori learning bound.

Encoder stability Defining the encoder (2.1) via the ℓ_1 sparsity-inducing regularizer (or *sparsifier*) is just one choice for the encoder. The choice of sparsifier seems to be pivotal both from an empirical perspective and a theoretical one. Bradley and Bagnell (2009b) used a differentiable *approximate* sparsifier based on the Kullback-Leibler divergence (true sparsity may not result). The ℓ_1 sparsifier $\|\cdot\|_1$ is the most popular and notably is the tightest convex lower bound for the ℓ_0 "norm": $\|x\|_0 := |\{i : x_i \neq 0\}|$ (Fazel, 2002). Regrettably, from a stability perspective the ℓ_1 sparsifier is not well-behaved in general. Indeed, due to the lack of strict convexity, each x need not have a unique image under φ_D. It also is unclear how to analyze the class of mappings φ_D, parameterized by D, if the map changes drastically under small perturbations to D. Hence, we will begin by establishing sufficient conditions under which φ_D is stable under perturbations to D.

2.2 Conditions and main results

In this section, we develop several quantities that are central to the statement of the main results. Throughout this chapter, let $[n] := \{1, \ldots, n\}$ for $n \in \mathbb{N}$. Also, for $t \in \mathbb{R}^k$, define $\mathsf{supp}(t) := \{i \in [k] : t_i \neq 0\}$.

Definition 2.1 (s-incoherence). *For $s \in [k]$ and $D \in \mathcal{D}$, the* s-incoherence $\mu_s(D)$ *is defined as the square of the minimum singular value among s-atom subdictionaries of D. Formally,*

$$\mu_s(D) = \left(\min\left\{\sigma_s(D_\Lambda) : \Lambda \subseteq [k], |\Lambda| = s\right\}\right)^2,$$

where $\sigma_s(A)$ is the s^{th} singular value of A.

The s-incoherence can used to guarantee that sparse codes are stable in a certain sense.

[3]Empirically we have observed that stochastic gradient approaches like the one in Mairal et al. (2012) perform very well.

We also introduce some key parameter-and-data-dependent properties. The first property regards the sparsity of the encoder on a sample $\mathbf{x} = (x_1, \ldots, x_m)$.

Definition 2.2 (s-sparsity). *If every point x_i in the set of points $\mathbf{x}$ satisfies $\|\varphi_D(x_i)\|_0 \leq s$, then φ_D is s-sparse on $\mathbf{x}$. More concisely, the Boolean expression $s\text{-}\mathsf{sparse}(\varphi_D(\mathbf{x}))$ is true.*

This property is critical as the learning bounds will exploit the observed sparsity level over the training sample. Finally, we require some margin properties.

Definition 2.3 (s-margin). *Given a dictionary D and a point $x_i \in B_{\mathbb{R}^d}$, the* s-margin of D on x_i *is*

$$\mathsf{margin}_s(D, x_i) := \max_{\substack{\mathcal{I} \subseteq [k] \\ |\mathcal{I}| = k-s}} \min_{j \in \mathcal{I}} \Big\{ \lambda - \big| \langle D_j, x_i - D\varphi_D(x_i) \rangle \big| \Big\}.$$

The sample version of the s-margin is the maximum s-margin that holds for all points in $\mathbf{x}$, or the s-margin of D on $\mathbf{x}$*:*

$$\mathsf{margin}_s(D, \mathbf{x}) := \min_{x_i \in \mathbf{x}} \mathsf{margin}_s(D, x_i).$$

The importance of these s-margin properties flows directly from the upcoming Sparse Coding Stability Theorem (Theorem 2.4). Intuitively, if the s-margin of D on x is high, then there is a set of $(k - s)$ inactive atoms that are poorly correlated with the optimal residual $x - D\varphi_D(x)$; hence these are far from being included in the set of active atoms. More formally, $\mathsf{margin}_s(D, x_i)$ is equal to the $(s+1)^{\text{th}}$ smallest element of the set of k elements $\{\lambda - |\langle D_j, x_i - D\varphi_D(x_i) \rangle|\}_{j \in [k]}$. Note that if $\|\varphi_D(x_i)\|_0 = s$, we can use the $(s + \rho)$-margin for any integer $\rho \geq 0$. Indeed, $\rho > 0$ is justified when $\varphi_D(x_i)$ has only s non-zero dimensions but for precisely one index j^* outside the support set $|\langle D_{j^*}, x_i - D\varphi_D(x_i) \rangle|$ is arbitrarily close to λ. In this scenario, the s-margin of D on x_i is trivially small; however, the $(s+1)$-margin is non-trivial because the $\max$ in the definition of the margin will remove j^* from the $\min$'s choices $\mathcal{I}$. Empirical evidence shown in Section 2.6 suggests that even when ρ is small, the $(s + \rho)$-margin is not too small.

Sparse coding stability The first result of this chapter is a fundamental stability result for the Lasso. In addition to being critical in motivating the presented conditions, the result may be of interest in its own right.

Theorem 2.4 (Sparse Coding Stability). *Let $D, \tilde{D} \in \mathcal{D}$ satisfy $\mu_s(D), \mu_s(\tilde{D}) \geq \mu$ and $\|D - \tilde{D}\|_2 \leq \varepsilon$, and let $x \in B_{\mathbb{R}^d}$. Suppose that there exists an index set $\mathcal{I} \subseteq [k]$ of $k - s$ indices such that for all $i \in \mathcal{I}$:*

$$|\langle D_i, x - D\varphi_D(x)\rangle| < \lambda - \tau \tag{2.5}$$

with

$$\varepsilon \leq \frac{\tau^2 \lambda}{27}. \tag{2.6}$$

Then the following stability bound holds:

$$\|\varphi_D(x) - \varphi_{\tilde{D}}(x)\|_2 \leq \frac{3}{2}\frac{\varepsilon\sqrt{s}}{\lambda\mu}.$$

Moreover, if $\varepsilon = \frac{\tau'^2\lambda}{27}$ for $\tau' < \tau$, then for all $i \in \mathcal{I}$:

$$\left|\langle \tilde{D}_i, x - \tilde{D}\varphi_{\tilde{D}}(x)\rangle\right| \leq \lambda - (\tau - \tau').$$

Thus, some margin, and hence sparsity, is retained after perturbation.

Condition (2.5) means that at least $k-s$ inactive atoms in the coding $\varphi_D(x)$ do not have too high absolute correlation with the residual $x - D\varphi_D(x)$. We refer to the right-hand side of (2.6) as the permissible radius of perturbation (PRP) because it indicates the maximum amount of perturbation for which the theorem can guarantee encoder stability. In short, the theorem says that if problem (2.1) admits a stable sparse solution, then a small perturbation to the dictionary will not change the fact that a certain set of $k - s$ atoms remains inactive in the new solution. The theorem further states that the perturbation to the solution will be bounded by a constant factor times the size of the perturbation, where the constant depends on the s-incoherence, the amount of ℓ_1-regularization, and the sparsity level.

The proof of Theorem 2.4 is quite long, and so we leave all but the following very high-level summary to Section 2.8.1.

Proof sketch First, we show that the solution $\varphi_{\tilde{D}}(x)$ is s-sparse and, in particular, has support contained in the complement of $\mathcal{I}$. Second, we reframe the Lasso as a quadratic program (QP). By exploiting the convexity of the QP and the fact that both solutions have their support contained in a set of s atoms, simple linear algebra yields the desired stability bound. The first step appears much more difficult than the second. The quartet below is our strategy for the first step:

1. OPTIMAL VALUE STABILITY: The two problems' optimal objective values are close; this is an easy consequence of the closeness of D and $\tilde{D}$.

2. STABILITY OF NORM OF RECONSTRUCTOR: The *norms* of the optimal reconstructors ($D\varphi_D(x)$ and $\tilde{D}\varphi_{\tilde{D}}(x)$) of the two problems are close. We show this using OPTIMAL VALUE STABILITY and

$$(x - D\varphi_D(x))^T D\varphi_D(x) = \lambda \|\varphi_D(x)\|_1, \tag{2.7}$$

   the latter of which holds due to the subgradient of (2.1) with respect to z (Osborne et al., 2000).

3. RECONSTRUCTOR STABILITY: The optimal reconstructors of the two problems are close. This fact is a consequence of STABILITY OF NORM OF RECONSTRUCTOR, using the ℓ_1 norm's convexity and the equality (2.7).

4. PRESERVATION OF SPARSITY: The solution to the perturbed problem also is supported on the complement of $\mathcal{I}$. To show this, it is sufficient to show that the absolute correlation of each atom $\tilde{D}_i$ ($i \in \mathcal{I}$) with the residual in the perturbed problem is less than λ. This last claim is a relatively easy consequence of RECONSTRUCTOR STABILITY.

■

2.2.1 Main results

The following notation will aid and abet the below results and the subsequent analysis. Recall that the loss ℓ is bounded by b and L-Lipschitz in its second argument. Also recall that $\mathcal{F}$ is the set of predictive sparse coding hypothesis functions $f(x) = \langle w, \varphi_D(x) \rangle$ indexed by $D \in \mathcal{D}$ and $w \in \mathcal{W}$. For $f \in \mathcal{F}$, define $\ell(\cdot, f) : \mathcal{Y} \times \mathbb{R}^d \to [0, b]$ as the loss-composed function $(y, x) \mapsto \ell(y, f(x))$. Let $\ell \circ \mathcal{F}$ be the class of such functions induced by the choice of $\mathcal{F}$ and ℓ. A probability measure P operates on functions and loss-composed functions as:

$$\mathsf{P}\, f = \mathsf{E}_{(x,y)\sim\mathsf{P}}\, f(x) \qquad\qquad \mathsf{P}\, \ell(\cdot, f) = \mathsf{E}_{(x,y)\sim\mathsf{P}}\, \ell(y, f(x)).$$

Similarly, an empirical measure $\mathsf{P}_{\mathbf{z}}$ associated with sample $\mathbf{z}$ operates on functions and loss-composed functions as:

$$\mathsf{P}_{\mathbf{z}}\, f = \frac{1}{m} \sum_{i=1}^{m} f(x_i) \qquad\qquad \mathsf{P}_{\mathbf{z}}\, \ell(\cdot, f) = \frac{1}{m} \sum_{i=1}^{m} \ell(y_i, f(x_i)).$$

Finally, when provided a training sample $\mathbf{z}$, the hypothesis returned by the learner will be referred to as $\hat{f}_{\mathbf{z}}$. Note that $\hat{f}_{\mathbf{z}}$ is random, but $\hat{f}_{\mathbf{z}}$ becomes a fixed function upon conditioning on $\mathbf{z}$.

Classically speaking, the overcomplete setting is the modus operandi in sparse coding. In this setting, an overcomplete basis is learned which will be used parsimoniously in coding individual points. The next result bounds the generalization error in the overcomplete setting. The $\tilde{O}(\cdot)$ notation hides $\mathsf{log}(\mathsf{log}(\cdot))$ terms and assumes that $r \leq m^{\mathsf{min}\{d,k\}}$.

Theorem 2.5 (Overcomplete Learning Bound). *With probability at least* $1 - \delta$ *over* $\mathbf{z} \sim \mathsf{P}^m$*, for any* $s \in [k]$ *and any* $f = (D, w) \in \mathcal{F}$ *satisfying* s-$\mathsf{sparse}(\varphi_D(\mathbf{x}))$ *and*

$$m > \frac{243}{\mathsf{margin}_s(D, \mathbf{x})^2 \lambda},$$

the generalization error $(\mathsf{P} - \mathsf{P}_{\mathbf{z}})\ell(\cdot, f)$ *is*

$$\tilde{O}\left(b\sqrt{\frac{dk\log m + \log\frac{1}{\delta}}{m}} + \frac{b}{m}\left(dk\log\frac{1}{\mathsf{margin}_s^2(D,\mathbf{x})\cdot\lambda}\right) + \frac{L}{m}\left(\frac{r\sqrt{s}}{\lambda\mu_s(D)}\right)\right). \tag{2.8}$$

Note that this bound also applies to the particular hypothesis $\hat{f}_{\mathbf{z}} = (\hat{D}_{\mathbf{z}}, \hat{w}_{\mathbf{z}})$ learned from the training sample.

Often in learning problems, we first map the data implicitly to a space of very high dimension or even infinite dimension and use kernels for efficient computations. In these cases where $d \gg k$ or d is infinite, it is unacceptable for any learning bound to exhibit dependence on d. It is possible to untether the analysis from d by using the s-margin of the learned dictionary $\hat{D}_{\mathbf{z}}$ on a second, unlabeled sample. In the infinite-dimensional setting, the following dimension-free learning bound holds.

Theorem 2.6 (Infinite-Dimensional Learning Bound). *With probability at least* $1-\delta$ *over a labeled* m*-sample* $\mathbf{z} \sim \mathsf{P}^m$ *and a second, unlabeled sample* $\mathbf{x}'' \sim \Pi^m$*, if an algorithm learns hypothesis* $\hat{f}_{\mathbf{z}} = (\hat{D}_{\mathbf{z}}, \hat{w}_{\mathbf{z}})$ *such that* $\varphi_{\hat{D}_{\mathbf{z}}}$ *is* s*-sparse on* $(\mathbf{x}\cup\mathbf{x}'')$*,* $\mu_{2s}(\hat{f}_{\mathbf{z}}) > 0$*, and*

$$m \geq \frac{27}{\mathsf{margin}_s^2(\hat{D}_{\mathbf{z}}, \mathbf{x}\cup\mathbf{x}'')\cdot\lambda},$$

then the generalization error $(\mathsf{P} - \mathsf{P}_{\mathbf{z}})\ell(\cdot, \hat{f}_{\mathbf{z}})$ *is*

$$\tilde{O}\left(\frac{L}{\sqrt{m}}\left(\frac{rk\sqrt{s}}{\mu_{2s}(\hat{f}_{\mathbf{z}})}\right) + b\sqrt{\frac{(k^2+\log\frac{1}{\delta})\log m}{m}} + \frac{L}{m}\left(\frac{r\sqrt{s}}{\lambda\mu_s(\hat{f}_{\mathbf{z}})}\right)\right). \tag{2.9}$$

2.2.2 Discussion of Theorems 2.5 and 2.6

The results highlight the central role of the stability of the sparse encoder. The presented bounds are data-dependent and exploit properties relating to the training sample and the learned hypothesis. Since $k \geq d$ in the overcomplete setting, an ideal learning bound has minimal dependence on k. The $\frac{1}{m}$ term of the learning bound for the overcomplete setting (2.8) exhibits square root dependence on both the size of the dictionary k and the ambient dimension d. It is unclear whether further improvement is possible, even in the

reconstructive setting. The two known results in the reconstructive setting were established first by Maurer and Pontil (2010) and later by Vainsencher et al. (2011), as mentioned in the Introduction. The infinite-dimensional setting learning bound (2.9) is dimension free, with linear dependence on k, square root dependence on s, and inverse dependence on the $2s$-incoherence $\mu_{2s}(\hat{f}_{\mathbf{z}})$. While both bounds exhibit dependence on the sparsity level s, the sparsity level appears to be much more significant in the infinite-dimensional setting.

Let us compare these bounds to the reconstructive setting, starting with the overcomplete regime. The first term of (2.8) matches the slower of the rates shown by Vainsencher et al. (2011) for the unsupervised case. Vainsencher et al. also showed fast rates of $\frac{dk}{m}$ (plus a small fraction of the observed empirical risk), but in the predictive setting it is an open question whether similar fast rates are possible. The second term of (2.8) represents the error in approximating the estimator via an ($\varepsilon = \frac{1}{m}$)-cover of the space of dictionaries. This term reflects the stability of the sparse codes with respect to dictionary perturbations, as quantified by the Sparse Coding Stability Theorem (Theorem 2.4). The reason for the lower bound on m is that the ε-net used to approximate the space of dictionaries needs to be fine enough to satisfy the PRP condition (2.6) of the Sparse Coding Stability Theorem. Hence, both this lower bound and the second term are determined primarily by the Sparse Coding Stability Theorem, and so with this proof strategy the extent to which the Sparse Coding Stability Theorem cannot be improved also indicates the extent to which Theorem 2.5 cannot be improved.

Shifting to the infinite-dimensional setting, Maurer and Pontil (2010) previously showed the generalization bound (2.2) for unsupervised (ℓ_1-regularized) sparse coding. Comparing their result to (2.9) and neglecting regularization parameters, the dimension-free bound in the predictive case is larger by a factor of $\frac{\sqrt{s}}{\mu_{2s}(\hat{f}_z)}$. It is unclear whether either of the terms in this factor are avoidable in the predictive setting. At least from our analysis, it appears that the $\frac{\sqrt{s}}{\mu_{2s}(\hat{f}_z)}$ factor is the price one pays for encoder stability. Critically, encoder stability is not necessary in the reconstructive setting because stability in loss (reconstruction error) requires only stability in the *norm of the residual* to the Lasso problem rather than stability in the *value of the solution* to the problem. Stability of the norm of the residual is readily obtainable without any of the incoherence, sparsity, and margin conditions used here.

Remarks on conditions One may wonder about typical values for the various hypothesis-and-data-dependent properties in Theorems 2.5 and 2.6. In practical applications of reconstructive and predictive sparse coding, the regularization parameter λ is set to ensure that s is small relative to the dimension d. As a result, both incoherences $\mu_s(D)$ and $\mu_{2s}(D)$ for the learned dictionary can be expected to be bounded away from zero. A sufficiently large s-incoherence certainly is necessary if one hopes for any amount of stability of the class of sparse coders with respect to dictionary perturbations. Since our path to reaching Theorems 2.5 and 2.6 passes through the Sparse Coding Stability Theorem (Theorem 2.4), it seems that a drastically different strategy needs to be used if it is possible to avoid dependence on $\mu_s(D)$ in the learning bounds.

A curious aspect of both learning bounds is their dependence on the s-margin term $\mathsf{margin}_s(D, \mathbf{x})$. Suppose that a dictionary is learned which is s-sparse on the training sample $\mathbf{x}$, and s is the lowest such integer for which this holds. It may not always be the case that the s-margin is bounded away from zero because for some points a small collection of inactive atoms may be very close to being brought into the optimal solution (the code); however, we can instead use the $(s+\rho)$-margin for some small positive integer ρ for which the $(s+\rho)$-margin is non-trivial. In Section 2.6, we gather empirical evidence that such a non-trivial $(s+\rho)$-margin does exist, for small ρ, when learning predictive sparse codes on real data. Hence, there is evidence that predictive sparse coding learns a dictionary with high s-incoherence $\mu_s(D)$ and non-trivial s-margin $\mathsf{margin}_s(D, \mathbf{x})$ on the training sample, for low s.

If one entertains a mixture of ℓ_1 and ℓ_2 norm regularization, $\lambda_1\|\cdot\|_1 + \frac{1}{2}\lambda_2\|\cdot\|_2^2$, as in the elastic net (Zou and Hastie, 2005), fall-back guarantees are possible in both scenarios. For small values of λ_2, this regularizer induces true sparsity similar to the ℓ_1 regularizer. A considerably simpler, data-independent analysis is possible in the overcomplete setting with a final bound that essentially just trades $\mu_s(D)$ for the ℓ_2 norm regularization parameter λ_2. In the infinite-dimensional setting, a simpler non-data-dependent analysis using our approach would only attain a bound of the larger order $\frac{k^{3/2}}{\lambda_2\sqrt{m}}$.

2.3 Tools

As before, let $\mathbf{z}$ be a labeled sample of m points (an m-sample) drawn iid from P. In addition, let $\mathbf{z}'$ be a second labeled m-sample drawn iid from P. In the infinite-dimensional setting, we also will make use of an unlabeled m-sample $\mathbf{x}''$ drawn iid from the marginal Π. All epsilon-nets of spaces of dictionaries use the metric induced by the operator norm $\|\cdot\|_2$.

2.3.1 Symmetrization by ghost sample for random subclasses

The next result is essentially due to Mendelson and Philips (2004); it applies symmetrization by a ghost sample for random subclasses. Our main departure is that we allow the random subclass to depend on a second, unlabeled sample $\mathbf{x}''$.

Lemma 2.7 (Symmetrization by Ghost Sample). *Let $\mathcal{F}(\mathbf{z}, \mathbf{x}'') \subset \mathcal{F}$ be a random subclass which can depend on both a labeled sample $\mathbf{z}$ and an unlabeled sample $\mathbf{x}''$. Recall that $\mathbf{z}'$ is a ghost sample of m points. If $m \geq \left(\frac{b}{t}\right)^2$, then*

$$\mathsf{Pr}_{\mathbf{z}\,\mathbf{x}''}\left\{\exists f \in \mathcal{F}(\mathbf{z}, \mathbf{x}''),\ (\mathsf{P} - \mathsf{P}_{\mathbf{z}})\ell(\cdot, f) \geq t\right\} \leq 2\mathsf{Pr}_{\mathbf{z}\,\mathbf{z}'\mathbf{x}''}\left\{\exists f \in \mathcal{F}(\mathbf{z}, \mathbf{x}''),\ (\mathsf{P}_{\mathbf{z}'} - \mathsf{P}_{\mathbf{z}})\ell(\cdot, f) \geq \frac{t}{2}\right\}.$$

For completeness, this lemma is proved in Section 2.8.2. This symmetrization lemma will be applied in both the overcomplete and infinite-dimensional settings to shift the analysis from large deviations of the empirical risk from the expected risk to large deviations of two independent empirical risks: in the overcomplete setting the lemma will be specialized as Proposition 2.12, and in the infinite-dimensional setting the lemma will be adapted to Proposition 2.15.

2.3.2 Rademacher and Gaussian averages and related results

Let $\sigma_1, \ldots, \sigma_m$ be independent Rademacher random variables distributed uniformly on $\{-1, 1\}$, and let $\gamma_1, \ldots \gamma_m$ be independent Gaussian random variables distributed as $\mathcal{N}(0, 1)$. Denote the collections by $\boldsymbol{\sigma} = (\sigma_1, \ldots, \sigma_m)$ and $\boldsymbol{\gamma} = (\gamma_1, \ldots, \gamma_m)$. Given a sample of m

points $\mathbf{x}$, define the conditional Rademacher and Gaussian averages of a function class as

$$\mathcal{R}_{m|\mathbf{x}}(\mathcal{F}) = \frac{2}{m}\,\mathsf{E}_{\boldsymbol{\sigma}} \sup_{f\in\mathcal{F}} \sum_{i=1}^{m} \sigma_i f(x_i) \qquad \text{and} \qquad \mathcal{G}_{m|\mathbf{x}}(\mathcal{F}) = \frac{2}{m}\,\mathsf{E}_{\boldsymbol{\gamma}} \sup_{f\in\mathcal{F}} \sum_{i=1}^{m} \gamma_i f(x_i).$$

respectively.

Lemmas 2.8 and 2.9 below are used near the end of the proof of Theorem 2.20 of the infinite-dimensional setting, when shifting the analysis from the Gaussian complexity of a loss-composed function class to the Rademacher complexity of the original function class. From Meir and Zhang (2003, Theorem 7), the loss-composed conditional Rademacher average of a function class $\mathcal{F}$ is bounded by the scaled conditional Rademacher average:

Lemma 2.8 (Rademacher Loss Comparison Lemma). *For every function class $\mathcal{F}$, m-sample $\mathbf{x}$, and ℓ which is L-Lipschitz continuous in its second argument:*

$$\mathcal{R}_{m|\mathbf{z}}(\ell \circ \mathcal{F}) \leq L\mathcal{R}_{m|\mathbf{x}}(\mathcal{F}).$$

Additionally, from Ledoux and Talagrand (1991, a brief argument following Lemma 4.5), the conditional Rademacher average of a function class $\mathcal{F}$ is bounded up to a constant by the conditional Gaussian average of $\mathcal{F}$:

Lemma 2.9 (Rademacher-Gaussian Average Comparison Lemma). *For every function class $\mathcal{F}$ and sample of m points $\mathbf{x}$:*

$$\mathcal{R}_{m|\mathbf{x}}(\mathcal{F}) \leq \sqrt{\frac{\pi}{2}}\mathcal{G}_{m|\mathbf{x}}(\mathcal{F}).$$

The next relation is due to Slepian (1962):

Lemma 2.10 (Slepian's Lemma). *Let Ω and Γ be mean zero, separable Gaussian processes*[4] *indexed by a set T such that $\mathsf{E}\,(\Omega_{t_1} - \Omega_{t_2})^2 \leq \mathsf{E}\,(\Gamma_{t_1} - \Gamma_{t_2})^2$ for all $t_1, t_2 \in T$. Then $\mathsf{E}\sup_{t\in T} \Omega_t \leq \mathsf{E}\sup_{t\in T} \Gamma_t$.*

Slepian's Lemma essentially says that if the variance of one Gaussian process is bounded by the variance of another, then the expected maximum of the first is bounded by the

[4] $\{\Omega_t\}_{t\in T}$ is a Gaussian process with index set T if the collection is jointly Gaussian in the sense that every finite linear combination of the variables is Gaussian.

expected maximum of the second. This lemma will be used in the proof of Theorem 2.18 to bound the Gaussian complexity of an analytically difficult function class via a bound on the Gaussian complexity of a related but analytically easier function class.

We also will make use of the bounded differences inequality (McDiarmid, 1989), in order to shift the analysis in the proof of Theorem 2.20 to the Rademacher complexity of a certain function class:

Theorem 2.11 (Bounded Differences Inequality). *Let $X_1, \ldots, X_m$ be random variables drawn iid according to a probability measure μ over a space $\mathcal{X}$. Suppose that a function $f : \mathcal{X}^m \to \mathbb{R}$ satisfies*

$$\sup_{x_1,\ldots,x_m,x_i' \in \mathcal{X}} \left| f(x_1, \ldots, x_n) - f(x_1, \ldots, x_{i-1}, x_i', x_{i+1}, \ldots, x_m \right| \leq c_i$$

for any $i \in [m]$. Then

$$\Pr_{X_1,\ldots,X_n}\left\{ f(X_1, \ldots, X_n) - \mathsf{E}\, f(X_1, \ldots, X_n) \geq t \right\} \leq \exp\left(-2t^2 / \sum_{i=1}^{m} c_i^2 \right).$$

2.4 Overcomplete setting

The overcomplete setting is classically the more popular regime, and in this setting useful learning bounds may exhibit dependence on both the dimension d and the dictionary size k. At a high level, our strategy for the overcomplete case learning bound is to construct an epsilon-net over a subclass of the space of functions $\mathcal{F} := \{f = (D, w) : D \in \mathcal{D}, w \in \mathcal{W}\}$ and to show that the metric entropy of this subclass is of order dk. The main difficulty is that an epsilon-net over $\mathcal{D}$ need not approximate $\mathcal{F}$ to any degree, *unless* one has a notion of encoder stability. Our analysis effectively will be concerned only with a training sample and a ghost sample, and it is similar to in style to the luckiness framework of Shawe-Taylor et al. (1998). If we observe that the sufficient conditions for encoder stability hold true on the training sample, then it is enough to guarantee that most points in a ghost sample also satisfy these conditions (at a weaker level). Figure 2.1 exhibits the high-level flow of the proof of Theorem 2.5.

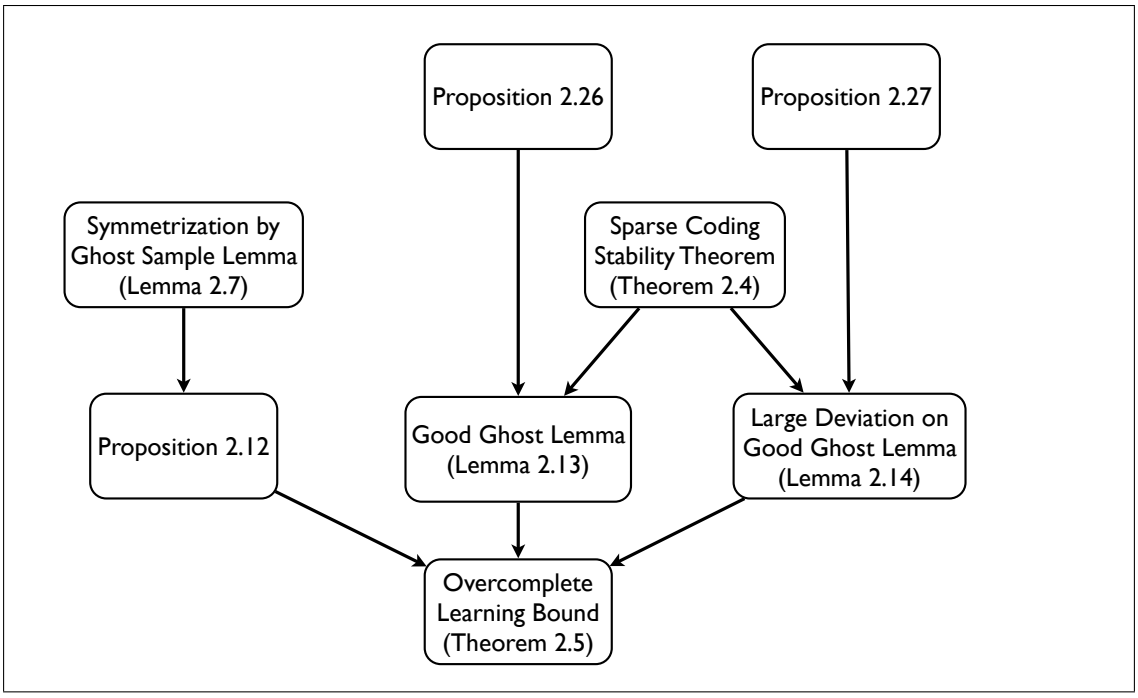


Figure 2.1: Proof flowchart for the Overcomplete Learning Bound (Theorem 2.5).

2.4.1 Useful conditions and subclasses

Let $\tilde{\mathbf{x}} \subseteq_\eta \mathbf{x}$ indicate that $\tilde{\mathbf{x}}$ is a subset of $\mathbf{x}$ with at most η elements of $\mathbf{x}$ removed. This notation is identical to Shawe-Taylor et al. (1998)'s notation from the luckiness framework.

Our bounds will require a crucial PRP-based condition that depends on both the learned dictionary and the training sample:

$$\mathsf{margin}_s(D, \mathbf{x}) \geq \iota(\lambda, \varepsilon) \qquad \text{for} \quad \iota(\lambda, \varepsilon) = \sqrt{\frac{243\varepsilon}{\lambda}}.$$

For brevity we will refer to ι with its parameters implicit; the dependence on ε and λ will not be an issue because we first develop bounds with these quantities fixed a priori. Lastly, for $\mu > 0$ define $\mathcal{D}_\mu := \{D \in \mathcal{D} : \mu_s(D) \geq \mu\}$ and $\mathcal{F}_\mu := \{f = (D, w) \in \mathcal{F} : D \in \mathcal{D}_\mu\}$.

2.4.2 Learning bound

The following proposition is simply a specialization of Lemma 2.7 with $\mathbf{x}''$ taken as the empty set and $\mathcal{F}(\mathbf{z}, \mathbf{x}'') := \{f \in \mathcal{F}_\mu : \big[\mathsf{margin}_s(D, \mathbf{x}) > \iota\big]\}$.

Proposition 2.12. *If* $m \geq \left(\frac{b}{t}\right)^2$, *then*

$$\begin{aligned}&\mathsf{Pr}_{\mathbf{z}}\left\{\exists f \in \mathcal{F}_\mu,\ \left[\mathsf{margin}_s(D,\mathbf{x})) > \iota\right] \textbf{ and } ((\mathsf{P}-\mathsf{P}_{\mathbf{z}})\ell(\cdot,f) > t)\right\}\\ &\leq 2\mathsf{Pr}_{\mathbf{z}\,\mathbf{z}'}\left\{\exists f \in \mathcal{F}_\mu,\ \left[\mathsf{margin}_s(D,\mathbf{x}) > \iota\right] \textbf{ and } ((\mathsf{P}_{\mathbf{z}'}-\mathsf{P}_{\mathbf{z}})\ell(\cdot,f) > t/2)\right\}.\end{aligned}$$

In the RHS of the above, let the event whose probability is being measured be

$$J := \left\{\mathbf{z}\,\mathbf{z}' : \exists f \in \mathcal{F}_\mu,\ \left[\mathsf{margin}_s(D,\mathbf{x}) > \iota\right] \textbf{ and } (\mathsf{P}_{\mathbf{z}'}-\mathsf{P}_{\mathbf{z}})\ell(\cdot,f) > t/2)\right\}.$$

Define Z as the event that there exists a hypothesis with stable codes on the original sample, in the sense of the Sparse Coding Stability Theorem (Theorem 2.4), but more than $\eta = \eta(m,d,k,D,\mathbf{x},\delta)$ points[5] of the ghost sample whose codes are not guaranteed stable by the Sparse Coding Stability Theorem:

$$Z := \left\{ \mathbf{z}\,\mathbf{z}' : \begin{array}{l} \exists f \in \mathcal{F}_\mu,\ \left[\mathsf{margin}_s(D,\mathbf{x}) > \iota\right] \\ \qquad \textbf{and } \left(\nexists\, \tilde{\mathbf{x}} \subseteq_\eta \mathbf{x}'\ \left[\mathsf{margin}_s(D,\tilde{\mathbf{x}}) > \frac{1}{3}\mathsf{margin}_s(D,\mathbf{x})\right]\right) \end{array} \right\}.$$

Our strategy will be to show that $\mathsf{Pr}(J)$ is small by use of the fact that

$$\mathsf{Pr}(J) = \mathsf{Pr}(J \cap \bar{Z}) + \mathsf{Pr}(J \cap Z) \leq \mathsf{Pr}(J \cap \bar{Z}) + \mathsf{Pr}(Z),$$

a strategy which thus far is similar to the beginning of Shawe-Taylor et al.'s proof of the main luckiness framework learning bound (see Shawe-Taylor et al., 1998, Theorem 5.22). We now show that each of $\mathsf{Pr}(Z)$ and $\mathsf{Pr}(J \cap \bar{Z})$ is small in turn.

The imminent Good Ghost Lemma shadows Shawe-Taylor et al. (1998)'s notion of probable smoothness and provides a bound on $\mathsf{Pr}(Z)$.

Lemma 2.13 (Good Ghost). *Fix* $\mu, \lambda > 0$ *and* $s \in [k]$. *With probability at least* $1-\delta$ *over an* m*-sample* $\mathbf{x} \sim \mathsf{P}^m$ *and a second* m*-sample* $\mathbf{x}' \sim \mathsf{P}^m$, *for any* $D \in \mathcal{D}_\mu$ *for which* φ_D *is* s-sparse *on* $\mathbf{x}$, *at least* $m - \eta(m,d,k,D,\mathbf{x},\delta)$ *points* $\tilde{\mathbf{x}} \subseteq \mathbf{x}'$ *satisfy* $\left[\mathsf{margin}_s(D,\tilde{\mathbf{x}}) >\right.$

[5] We use the shorthand $\eta = \eta(m,d,k,D,\mathbf{x},\delta)$ for conciseness.

$\frac{1}{3}\mathsf{margin}_s(D, \mathbf{x})\big]$, *for*

$$\eta(m, d, k, D, \mathbf{x}, \delta) := dk \log \frac{1944}{\mathsf{margin}_s^2(D, \mathbf{x}) \cdot \lambda} + \log(2m + 1) + \log \frac{1}{\delta}.$$

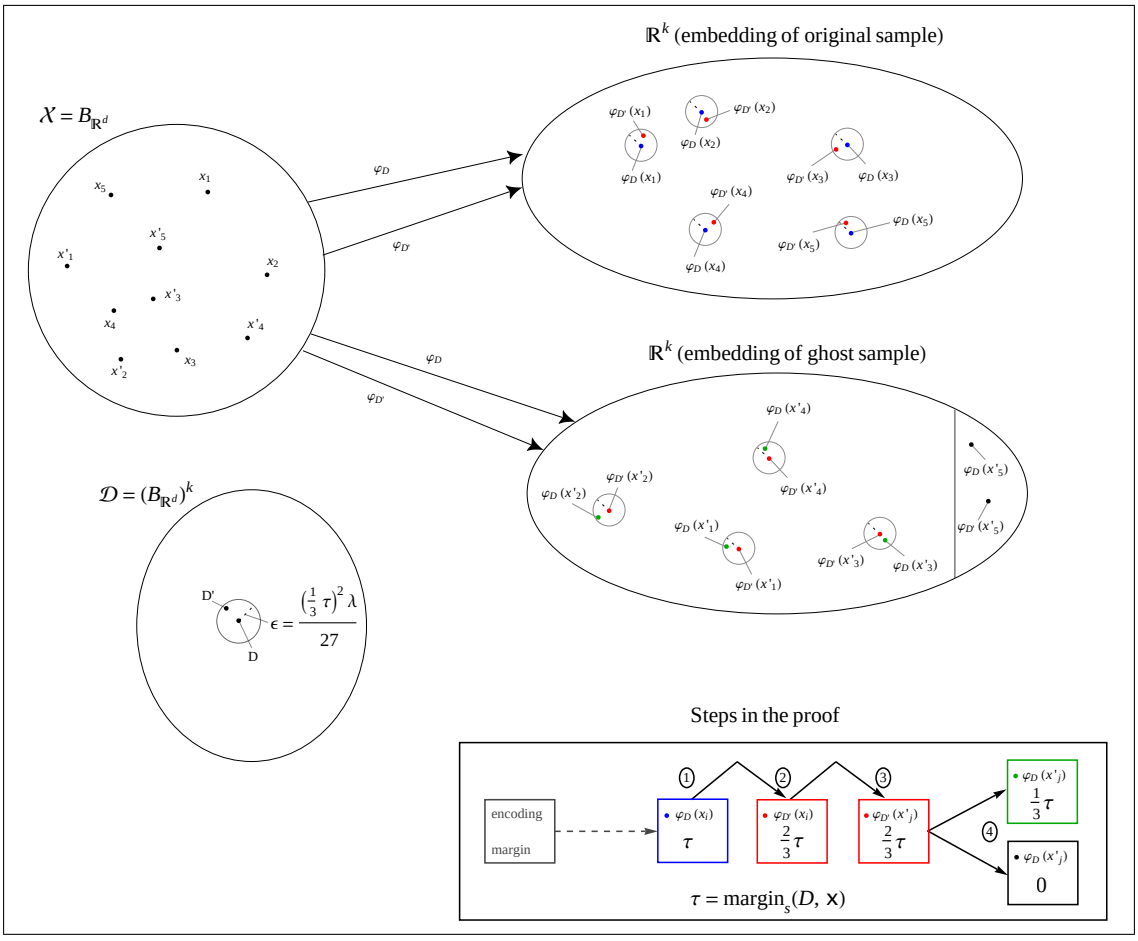


Figure 2.2: Visualization of the proof of the Good Ghost Lemma (Lemma 2.13). Best seen in color.

Proof. Figure 2.2 illustrates the proof. By the assumptions of the lemma, consider an arbitrary dictionary D satisfying $\mu_s(D) \geq \mu$ and s-$\mathsf{sparse}(\varphi_D(\mathbf{x}))$. The goal is to guarantee with high probability that all but η points of the ghost sample are coded by φ_D with s-margin of at least $\frac{1}{3}\mathsf{margin}_s(D, \mathbf{x})$.

Let $\varepsilon = \frac{(\frac{1}{3}\mathsf{margin}_s(D,\mathbf{x}))^2 \cdot \lambda}{27}$, and consider a minimum-cardinality proper ε-cover $\mathcal{D}'$ of $\mathcal{D}_\mu$. Let D' be a candidate element of $\mathcal{D}'$ satisfying $\|D - D'\|_2 \leq \varepsilon$. Then the Sparse Coding Stability Theorem (Theorem 2.4) implies that the coding margin of D' on $\mathbf{x}$ retains over

two-thirds the coding margin of D on $\mathbf{x}$; that is, $\left[\mathsf{margin}_s(D', \mathbf{x}) > \frac{2}{3}\mathsf{margin}_s(D, \mathbf{x})\right]$.

Furthermore, most points from the *ghost* sample satisfy $\left[\mathsf{margin}_s(D', \cdot) > \frac{2}{3}\mathsf{margin}_s(D, \mathbf{x})\right]$. To see this, let $\mathcal{F}_D^{\mathsf{marg}} := \{f_{D,\tau}^{\mathsf{marg}} | \tau \in \mathbb{R}_+\}$ be the class of threshold functions defined via

$$f_{D,\tau}^{\mathsf{marg}}(x) := \begin{cases} 1; & \text{if } \mathsf{margin}_s(D, x) > \tau, \\ 0; & \text{otherwise.} \end{cases}$$

The VC dimension of the one-dimensional threshold functions is 1, and so it follows that $\mathsf{VC}(\mathcal{F}_D^{\mathsf{marg}}) = 1$. By using the VC dimension of $\mathcal{F}_D^{\mathsf{marg}}$ and the standard permutation argument of Vapnik and Chervonenkis (1968, Proof of Theorem 2), it follows that for a single, *fixed* element of $\mathcal{D}'$, with probability at least $1 - \delta$ at most $\mathsf{log}(2m+1) + \mathsf{log}\,\frac{1}{\delta}$ points from a ghost sample will violate the margin inequality in question. Hence, by the bound on the proper covering numbers provided by Proposition 2.26 (see Section 2.8.5), we can we can guarantee for all candidate members $D' \in \mathcal{D}'$ that with probability $1 - \delta$ at most

$$\eta = dk \log \frac{1944}{\mathsf{margin}_s^2(D, \mathbf{x}) \cdot \lambda} + \mathsf{log}(2m+1) + \log \frac{1}{\delta}$$

points from the ghost sample violate the s-margin inequality. Thus, for arbitrary $D' \in \mathcal{D}'$ satisfying the conditions of the lemma, with probability $1 - \delta$ at most $\eta(m, d, k, D, \mathbf{x}, \delta)$ points from the ghost sample violate $\left[\mathsf{margin}_s(D', \cdot) > \frac{2}{3}\mathsf{margin}_s(D, \mathbf{x})\right]$.

Finally, consider the at least $m - \eta$ points in the ghost sample that satisfy $\left[\mathsf{margin}_s(D', \cdot) > \frac{2}{3}\mathsf{margin}_s(D, \mathbf{x})\right]$. Since $\|D' - D\|_2 \leq \frac{(\frac{1}{3}\mathsf{margin}_s(D,\mathbf{x}))^2 \cdot \lambda}{27}$, the Sparse Coding Stability Theorem (Theorem 2.4) implies that these points satisfy $\left[\mathsf{margin}_s(D, \cdot) > \frac{1}{3}\mathsf{margin}_s(D, \mathbf{x})\right]$. □

It remains to bound $\mathsf{Pr}(J \cap \bar{Z})$.

Lemma 2.14 (Large Deviation on Good Ghost). *Define* $\varpi := t/2 - \left(2L\beta + \frac{b\eta}{m}\right)$ *and* $\beta := \frac{\varepsilon}{2\lambda}\left(1 + \frac{3r\sqrt{s}}{\mu}\right)$. *Then*

$$\mathsf{Pr}(J \cap \bar{Z}) \leq \left(\frac{8(r/2)^{1/(d+1)}}{\varepsilon}\right)^{(d+1)k} \exp(-m\varpi^2/(2b^2)).$$

Equivalently, the difference between the loss on $\mathbf{z}$ *and the loss on* $\mathbf{z}'$ *is greater than*

$\varpi + 2L\beta + \frac{b\eta}{m}$ *with probability at most* $\left(\frac{8(r/2)^{1/(d+1)}}{\varepsilon}\right)^{(d+1)k} \exp(-m\varpi^2/(2b^2))$.

Proof. First, note that the event $J \cap \bar{Z}$ is a subset of the event

$$R := \left\{ \mathbf{z}\mathbf{z}' : \begin{array}{c} \exists f \in \mathcal{F}_\mu,\ \left[\mathsf{margin}_s(D, \mathbf{x}) > \iota\right] \textbf{ and} \\ \left(\exists \tilde{\mathbf{x}} \subseteq_\eta \mathbf{x}',\ \left[\mathsf{margin}_s(D, \tilde{\mathbf{x}}) > \frac{1}{3}\mathsf{margin}_s(D, \mathbf{x})\right]\right) \\ \textbf{and}\ ((\mathsf{P}_{\mathbf{z}'} - \mathsf{P}_{\mathbf{z}})\ell(\cdot, f) > t/2) \end{array} \right\}.$$

Bounding the probability of the event R is equivalent to bounding the probability of a large deviation (i.e. $((\mathsf{P}_{\mathbf{z}'} - \mathsf{P}_{\mathbf{z}})\ell(\cdot, f) > t/2)$) for the random subclass:

$$\tilde{\mathcal{F}}(\mathbf{x}, \mathbf{x}') := \left\{ \begin{array}{c} f \in \mathcal{F}_\mu : \left[\mathsf{margin}_s(D, \mathbf{x}) > \iota\right] \textbf{ and} \\ \left(\exists \tilde{\mathbf{x}} \subseteq_\eta \mathbf{x}',\ \left[\mathsf{margin}_s(D, \tilde{\mathbf{x}}) > \frac{1}{3}\mathsf{margin}_s(D, \mathbf{x})\right]\right) \end{array} \right\}.$$

Let $\mathcal{F}_\varepsilon = \mathcal{D}_\varepsilon \times \mathcal{W}_\varepsilon$, where $\mathcal{D}_\varepsilon$ is a minimum-cardinality proper ε-cover of $\mathcal{D}_\mu$ and $\mathcal{W}_\varepsilon$ is a minimum-cardinality ε-cover of $\mathcal{W}$. It is sufficient to bound the probability of a large deviation for all of $\mathcal{F}_\varepsilon$ and to then consider the maximum difference between an element of $\tilde{\mathcal{F}}(\mathbf{x}, \mathbf{x}')$ and its closest representative in $\mathcal{F}_\varepsilon$. Clearly, for each $f = (D, w) \in \tilde{\mathcal{F}}(\mathbf{x}, \mathbf{x}')$, there is a $f' = (D', w') \in \mathcal{F}_\varepsilon$ satisfying $\|D - D'\|_2 \leq \varepsilon$ and $\|w - w'\|_2 \leq \varepsilon$. If ε is sufficiently small, then for all but η of the points x_i in the ghost sample (and for all points x_i of the original sample) it is guaranteed that

$$\begin{aligned} |\langle w, \varphi_D(x_i)\rangle - \langle w', \varphi_{D'}(x_i)\rangle| &\leq \left|\langle w - w', \varphi_D(x_i)\rangle\right| + \left|\langle w', \varphi_D(x_i) - \varphi_{D'}(x_i)\rangle\right| \\ &\leq \frac{\varepsilon}{2\lambda} + r\frac{3}{2}\frac{\varepsilon\sqrt{s}}{\lambda\mu} \\ &= \frac{\varepsilon}{2\lambda}\left(1 + \frac{3r\sqrt{s}}{\mu}\right) = \beta, \end{aligned}$$

where the second inequality follows from the Sparse Coding Stability Theorem (Theorem 2.4). Trivially, for the rest of the points x_i in the ghost sample each loss is bounded by b. Hence, on the original sample:

$$\frac{1}{m}\sum_{i=1}^{m} \left|\ell(y_i, \langle w, \varphi_D(x_i)\rangle) - \ell(y_i, \langle w', \varphi_{D'}(x_i)\rangle)\right| \leq L\beta,$$

and on the ghost sample:

$$\begin{aligned}
&\frac{1}{m}\sum_{i=1}^{m}\left|\ell(y_i', \langle w, \varphi_D(x_i')\rangle) - \ell(y_i', \langle w', \varphi_{D'}(x_i')\rangle)\right| \\
&\quad \le \frac{L}{m}\sum_{i\ \text{GOOD}}\left|\langle w, \varphi_D(x_i)\rangle - \langle w', \varphi_{D'}(x_i)\rangle\right| + \frac{1}{m}\sum_{i\ \text{BAD}}\left|\ell(y_i', \langle w, \varphi_D(x_i')\rangle) - \ell(y_i', \langle w', \varphi_{D'}(x_i')\rangle)\right| \\
&\quad \le L\beta + \frac{b\eta}{m},
\end{aligned}$$

where GOOD denotes the (at least $m-\eta$) points of the ghost sample for which the Sparse Coding Stability Theorem applies, and BAD denotes the complement thereof.

Concluding the above argument, the difference between the losses of f and f' on the double sample is at most $2L\beta + \frac{b\eta}{m}$. Consequently, if $(\mathsf{P}_{\mathbf{z}'} - \mathsf{P}_{\mathbf{z}})\ell(\cdot, f) > t/2$, then the absolute deviation between the loss of f' on the original sample and the loss of f' on the ghost sample must be at least $t/2 - \left(2L\beta + \frac{b\eta}{m}\right)$. To bound the probability of R it therefore is sufficient to control

$$\mathsf{Pr}_{\mathbf{z}\,\mathbf{z}'}\left\{\exists f = (D', w') \in \mathcal{D}_\varepsilon \times \mathcal{W}_\varepsilon,\ (\mathsf{P}_{\mathbf{z}'} - \mathsf{P}_{\mathbf{z}})\ell(\cdot, f) > t/2 - \left(2L\beta + \frac{b\eta}{m}\right)\right\}.$$

We first handle the case of a fixed $f = (D', w') \in \mathcal{D}_\varepsilon \times \mathcal{W}_\varepsilon$. Applying Hoeffding's inequality to the random variable $\ell(y_i, f(x_i)) - \ell(y_i', f(x_i'))$, with range in $[-b, b]$, yields:

$$\mathsf{Pr}_{\mathbf{z}\,\mathbf{z}'}\left\{(\mathsf{P}_{\mathbf{z}'} - \mathsf{P}_{\mathbf{z}})\ell(\cdot, f) > \varpi\right\} \le \exp(-m\varpi^2/(2b^2)),$$

for $\varpi := t/2 - \left(2L\beta + \frac{b\eta}{m}\right)$. By way of a proper covering number bound of $\mathcal{D}_\varepsilon \times \mathcal{W}_\varepsilon$ (see Proposition 2.27) and the union bound, this result extends over all of $\mathcal{D}_\varepsilon \times \mathcal{W}_\varepsilon$:

$$\begin{aligned}
&\mathsf{Pr}_{\mathbf{z}\,\mathbf{z}'}\left\{\exists f = (D', w') \in \mathcal{D}_\varepsilon \times \mathcal{W}_\varepsilon,\ (\mathsf{P}_{\mathbf{z}'} - \mathsf{P}_{\mathbf{z}})\ell(\cdot, f) > \varpi\right\} \\
&\qquad\qquad \le \left(\frac{8(r/2)^{1/(d+1)}}{\varepsilon}\right)^{(d+1)k} \exp(-m\varpi^2/(2b^2)).
\end{aligned}$$

The bound on $\mathsf{Pr}(J \cap \bar{Z})$ now follows. □

The stage is now set to prove Theorem 2.5; the full proof is in Section 2.8.3.

Proof sketch (of Theorem 2.5) Proposition 2.12 and Lemmas 2.13 and 2.14 imply that

$$\begin{aligned}&\mathsf{Pr}_{\mathbf{z}}\left\{\exists f \in \mathcal{F}_{\mu},\ \left[\mathsf{margin}_s(D,\mathbf{x}) > \iota\right] \textbf{ and } ((\mathsf{P}-\mathsf{P}_{\mathbf{z}})\ell(\cdot,f) > t)\right\}\\ &\leq 2\left(\left(\frac{8(r/2)^{1/(d+1)}}{\varepsilon}\right)^{(d+1)k}\exp(-m\varpi^2/(2b^2)) + \delta\right).\end{aligned}$$

Fix $s \in [k]$ and $\mu > 0$ a priori. Setting $\varepsilon = \frac{1}{m}$ in the above, elementary manipulations show that provided $m > \frac{243}{\mathsf{margin}_s(D,\mathbf{x})^2\lambda}$, with probability at least $1-\delta$ over $\mathbf{z} \sim \mathsf{P}^m$, for any $f = (D,w) \in \mathcal{F}$ satisfying $\mu_s(D) \geq \mu$ and $\left[\mathsf{margin}_s(D,\mathbf{x}) > \iota\right]$, the generalization error $(\mathsf{P}-\mathsf{P}_{\mathbf{z}})\ell(\cdot,f)$ is bounded by:

$$\begin{aligned}&2b\sqrt{\frac{2((d+1)k\log(8m) + k\log\frac{r}{2} + \log\frac{4}{\delta})}{m}}\\ &+\frac{2L}{m}\left(\frac{1}{\lambda}\left(1+\frac{3r\sqrt{s}}{\mu}\right)\right) + \frac{2b}{m}\left(dk\log\frac{1944}{\mathsf{margin}_s^2(D,\mathbf{x})\cdot\lambda} + \log(2m+1) + \log\frac{4}{\delta}\right).\end{aligned}$$

It remains to distribute a prior across the bounds for each choice of s and μ. To each choice of $s \in [k]$ assign prior probability $\frac{1}{k}$. To each choice of $i \in \mathbb{N}\cup\{0\}$ for $2^{-i} \leq \mu$ assign prior probability $(i+1)^{-2}$. For a given choice of $s \in [k]$ and $2^{-i} \leq \mu$ we use $\delta(s,i) := \frac{6}{\pi^2}\frac{1}{(i+1)^2}\frac{1}{k}\delta$ (since $\sum_{i=1}^{\infty}\frac{1}{i^2} = \frac{\pi^2}{6}$). The theorem now follows. ■

2.5 Infinite-dimensional setting

In the infinite-dimensional setting learning bounds with dependence on d are useless. Unfortunately, the strategy of the previous section breaks down in the infinite-dimensional setting because the straightforward construction of any epsilon-net over the space of dictionaries had cardinality that depends on d. Even worse, epsilon-nets actually were used both to approximate the function class $\mathcal{F}$ in $\|\cdot\|_\infty$ norm and to guarantee that most points of the ghost sample are good provided that all points of the training sample were good (the Good Ghost Lemma (Lemma 2.13)).

These issues can be overcome by requiring an additional, *unlabeled* sample — a device

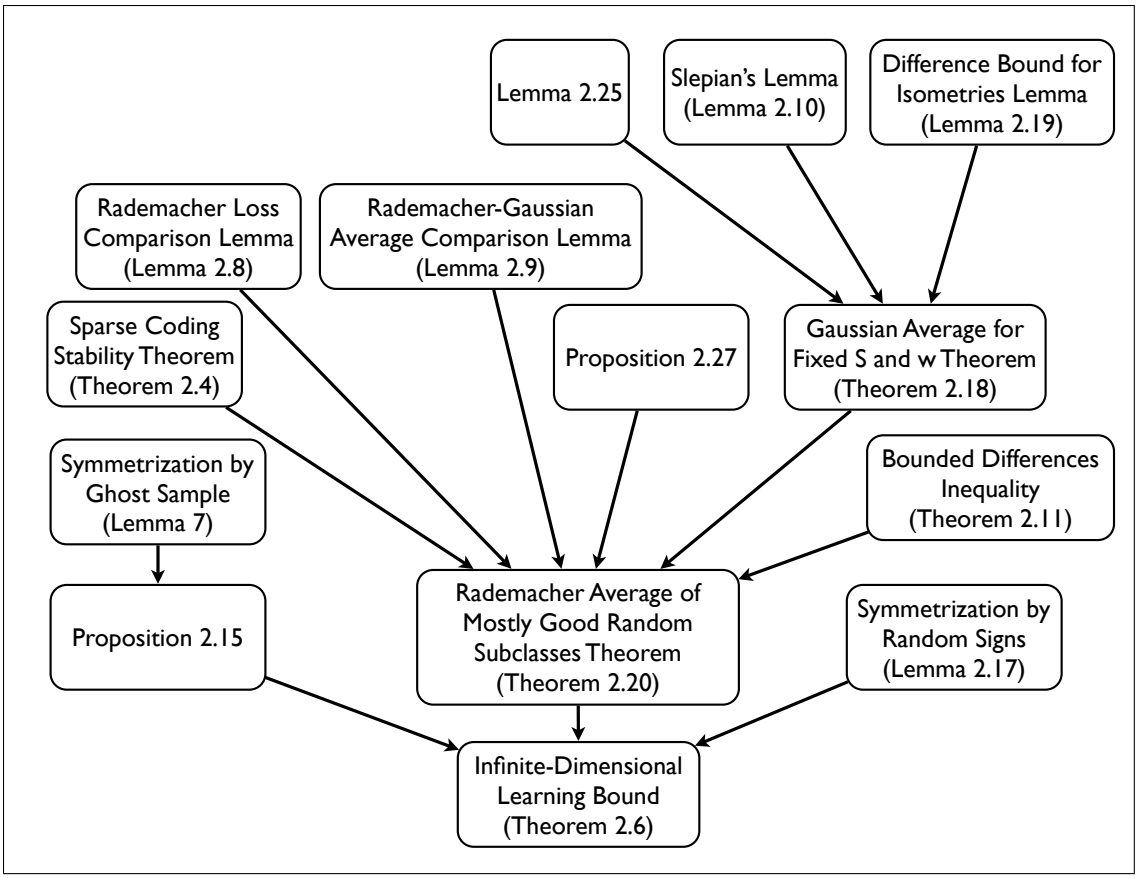


Figure 2.3: Proof flowchart for the Infinite-Dimensional Learning Bound (Theorem 2.6).

often justified in supervised learning problems because unlabeled data may be inexpensive and yet quite helpful — and by switching to more sophisticated techniques based on conditional Rademacher and Gaussian averages. After learning a hypothesis $\hat{f}_{\mathbf{z}}$ from a predictive sparse coding algorithm, the sparsity level and coding margin are measured on a second, unlabeled sample $\mathbf{x}''$ of m points[6]. Since this sample is independent of the choice of $\hat{f}_{\mathbf{z}}$, it is possible to guarantee that all but a very small fraction ($\frac{\eta}{m} = \frac{\log \frac{1}{\delta}}{m}$) of points of a ghost sample $\mathbf{z}$ are good with probability $1 - \delta$. In the likely case of this good event, and for a fixed sample, we then consider all possible choices of a set of η bad indices in the ghost sample; each of the $\binom{m}{\eta}$ cases corresponds to a subclass of functions. We then approximate each subclass by a special ε-cover that is a disjoint union of a finite number of special subclasses; for each of these smaller subclasses, we bound the conditional Rademacher average by exploiting a sparsity property. The proof flowchart in Figure 2.3 shows the structure of

[6]The cardinality matches the size of the training sample $\mathbf{z}$ purely for simplicity.

the proof of Theorem 2.6.

2.5.1 Symmetrization and decomposition

The proof of the infinite-dimensional setting learning bound Theorem 2.6 depends critically on Lemma 2.19, a lemma which is non-trivial only for dictionaries with non-zero $2s$-incoherence. The s-incoherence also will continue to play an important role, as it did in the overcomplete setting. Therefore, rather than wielding the deterministic subclass $\mathcal{F}_\mu$ of the previous section, we will work with a deterministic subclass with lower bounded s-incoherence *and* lower bounded $2s$-incoherence.

Let $\boldsymbol{\mu^*} = (\mu_s^*, \mu_{2s}^*) \in \mathbb{R}_+^2$ and define the deterministic subclass

$$\mathcal{F}_{\boldsymbol{\mu^*}} = \big\{ f = (D, w) \in \mathcal{F} : (\mu_s(D) \geq \mu_s^*) \,\mathbf{and}\, (\mu_{2s}(D) \geq \mu_{2s}^*) \big\}.$$

The next result is immediate from Lemma 2.7, taking the random subclass $\mathcal{F}(\cdot)$ to be

$$\mathcal{F}(\mathbf{z}, \mathbf{x}'') := \Big\{ \{\hat{f}_{\mathbf{z}}\} \cap \{ f \in \mathcal{F}_{\boldsymbol{\mu^*}} : \big[\mathsf{margin}_s(D, \mathbf{x} \cup \mathbf{x}'') > \tau\big] \Big\}.$$

Proposition 2.15. *If* $m \geq \left(\frac{b}{t}\right)^2$, *then*

$$\begin{aligned} \mathsf{Pr}_{\mathbf{z}\,\mathbf{x}''} \Big\{ \; \hat{f}_{\mathbf{z}} \in \mathcal{F}_{\boldsymbol{\mu^*}} \; \big[\mathsf{margin}_s(\hat{D}_{\mathbf{z}}, \mathbf{x} \cup \mathbf{x}'') > \tau\big] \,\mathbf{and}\, \Big((\mathsf{P} - \mathsf{P}_{\mathbf{z}}) \ell(\cdot, \hat{f}_{\mathbf{z}}) \geq t \Big) \; \Big\} \qquad (2.10) \\ \leq 2\mathsf{Pr}_{\mathbf{z}\,\mathbf{z}'\mathbf{x}''} \Big\{ \; \hat{f}_{\mathbf{z}} \in \mathcal{F}_{\boldsymbol{\mu^*}} \; \big[\mathsf{margin}_s(\hat{D}_{\mathbf{z}}, (\mathbf{x} \cup \mathbf{x}'')) > \tau\big] \,\mathbf{and}\, \Big((\mathsf{P}_{\mathbf{z}'} - \mathsf{P}_{\mathbf{z}}) \ell(\cdot, \hat{f}_{\mathbf{z}}) \geq \tfrac{t}{2} \Big) \; \Big\}. \end{aligned}$$

Now, observe that the probability of interest can be split into the probability of a large

deviation happening under a "good" event and the probability of a "bad" event occurring:

$$
\begin{aligned}
&\mathsf{Pr}_{\mathbf{z}\,\mathbf{z}'\mathbf{x}''}\left\{ \ \hat{f}_{\mathbf{z}} \in \mathcal{F}_{\boldsymbol{\mu}^*} \ \left[\mathsf{margin}_s(\hat{D}_{\mathbf{z}}, \mathbf{x}\cup\mathbf{x}'') > \tau\right] \textbf{ and } \Big((\mathsf{P}_{\mathbf{z}'} - \mathsf{P}_{\mathbf{z}})\ell(\cdot, \hat{f}_{\mathbf{z}}) \geq \tfrac{t}{2}\Big) \ \right\} \\
&\quad = \quad \mathsf{Pr}_{\mathbf{z}\,\mathbf{z}'\mathbf{x}''}\left\{ \begin{array}{l} \hat{f}_{\mathbf{z}} \in \mathcal{F}_{\boldsymbol{\mu}^*} \ \left[\mathsf{margin}_s(\hat{D}_{\mathbf{z}}, \mathbf{x}\cup\mathbf{x}'') > \tau\right] \textbf{ and } \Big(\exists\, \tilde{\mathbf{x}} \subseteq_\eta \mathbf{x}' \ \left[\mathsf{margin}_s(\hat{D}_{\mathbf{z}}, \tilde{\mathbf{x}}) > \tau\right]\Big) \\ \qquad \textbf{and } \Big((\mathsf{P}_{\mathbf{z}'} - \mathsf{P}_{\mathbf{z}})\ell(\cdot, \hat{f}_{\mathbf{z}}) \geq \tfrac{t}{2}\Big) \end{array} \right\} \\
&\qquad + \mathsf{Pr}_{\mathbf{z}\,\mathbf{z}'\mathbf{x}''}\left\{ \begin{array}{l} \hat{f}_{\mathbf{z}} \in \mathcal{F}_{\boldsymbol{\mu}^*} \ \left[\mathsf{margin}_s(\hat{D}_{\mathbf{z}}, \mathbf{x}\cup\mathbf{x}'') > \tau\right] \textbf{ and } \Big(\nexists\, \tilde{\mathbf{x}} \subseteq_\eta \mathbf{x}' \ \left[\mathsf{margin}_s(\hat{D}_{\mathbf{z}}, \tilde{\mathbf{x}}) > \tau\right]\Big) \\ \qquad \textbf{and } \Big((\mathsf{P}_{\mathbf{z}'} - \mathsf{P}_{\mathbf{z}})\ell(\cdot, \hat{f}_{\mathbf{z}}) \geq \tfrac{t}{2}\Big) \end{array} \right\} \\
&\quad \leq \quad \mathsf{Pr}_{\mathbf{z}\,\mathbf{z}'}\left\{ \begin{array}{l} \exists f \in \mathcal{F}_{\boldsymbol{\mu}^*}, \ \left[\mathsf{margin}_s(\hat{D}_{\mathbf{z}}, \mathbf{x}) > \tau\right] \textbf{ and } \Big(\exists\, \tilde{\mathbf{x}} \subseteq_\eta \mathbf{x}' \ \left[\mathsf{margin}_s(\hat{D}_{\mathbf{z}}, \tilde{\mathbf{x}}) > \tau\right]\Big) \\ \qquad \textbf{and } \big((\mathsf{P}_{\mathbf{z}'} - \mathsf{P}_{\mathbf{z}})\ell(\cdot, f) \geq \tfrac{t}{2}\big) \end{array} \right\} \\
&\qquad + \mathsf{Pr}_{\mathbf{x}'\mathbf{x}''}\left\{ \ \hat{f}_{\mathbf{z}} \in \mathcal{F}_{\boldsymbol{\mu}^*} \ \left[\mathsf{margin}_s(\hat{D}_{\mathbf{z}}, \mathbf{x}'') > \tau\right] \textbf{ and } \Big(\nexists\, \tilde{\mathbf{x}} \subseteq_\eta \mathbf{x}' \ \left[\mathsf{margin}_s(\hat{D}_{\mathbf{z}}, \tilde{\mathbf{x}}) > \tau\right]\Big) \ \right\}.
\end{aligned}
$$

Of the two probabilities summed in the last line, we treat the first in the next subsection. To bound the second one, note that for each choice of $\mathbf{x}$, $\hat{f}_{\mathbf{z}}$ is a fixed function. Hence, it is sufficient to select η such that, for *any* fixed function $f = (D, w) \in \mathcal{F}$, this second probability is bounded by δ. The next lemma accomplishes this bound:

Lemma 2.16 (Unlikely Bad Ghost). *Let $f = (D, w) \in \mathcal{F}$ be fixed. If $\eta = \log \frac{1}{\delta}$, then*

$$
\mathsf{Pr}_{\mathbf{x}'\mathbf{x}''}\left\{ \ \left[\mathsf{margin}_s(D, \mathbf{x}'') > \tau\right] \textbf{ and} \Big(\nexists\, \tilde{\mathbf{x}} \subseteq_\eta \mathbf{x}' \ \left[\mathsf{margin}_s(D, \tilde{\mathbf{x}}) > \tau\right]\Big) \ \right\} \leq \delta.
$$

Proof sketch The proof just uses the same standard permutation argument as in the proof of the Good Ghost Lemma (Lemma 2.13). ■

2.5.2 Rademacher bound in the case of the good event

We now bound the probability of a large deviation in the (likely) case of the good event. Denote by $\mathcal{F}_{\boldsymbol{\mu}^*}(\mathbf{x})$ the intersection of the deterministic subclass $\mathcal{F}_{\boldsymbol{\mu}^*}$ with the random subclass of functions for which the Sparse Coding Stability Theorem (Theorem 2.4) kicks

in with constants (μ_s^*, s, τ):

$$\mathcal{F}_{\boldsymbol{\mu}^*}(\mathbf{x}) := \big\{ f \in \mathcal{F}_{\boldsymbol{\mu}^*} : \big[\mathsf{margin}_s(D, \mathbf{x}) > \tau\big] \big\}.$$

This is the "good" random subclass. Similarly, let $\mathcal{F}_{\boldsymbol{\mu}^*,\eta}(\mathbf{x})$ denote the "mostly good" (or "all-but-η-good") random subclass:

$$\mathcal{F}_{\boldsymbol{\mu}^*,\eta}(\mathbf{x}) := \Big\{ f \in \mathcal{F}_{\boldsymbol{\mu}^*} : \exists\, \tilde{\mathbf{x}} \subseteq_\eta \mathbf{x} \ \big[\mathsf{margin}_s(D, \tilde{\mathbf{x}}) > \tau\big] \Big\}.$$

Recall that $\sigma_1, \ldots, \sigma_m$ are independent Rademacher random variables.

Lemma 2.17 (Symmetrization by Random Signs).

$$\mathsf{Pr}_{\mathbf{z}\,\mathbf{z}'} \left\{ \begin{array}{l} \exists f \in \mathcal{F}_{\boldsymbol{\mu}^*},\ \big[\mathsf{margin}_s(D, \mathbf{x}) > \tau\big] \textbf{ and } \big(\exists\, \tilde{\mathbf{x}} \subseteq_\eta \mathbf{x}'\ \big[\mathsf{margin}_s(D, \tilde{\mathbf{x}}) > \tau\big]\big) \\ \qquad \textbf{and } \big((\mathsf{P}_{\mathbf{z}'} - \mathsf{P}_{\mathbf{z}})\ell(\cdot, f) \geq \tfrac{t}{2}\big) \end{array} \right\}$$
$$\leq \mathsf{Pr}_{\mathbf{z},\boldsymbol{\sigma}} \left\{ \sup_{f \in \mathcal{F}_{\boldsymbol{\mu}^*}(\mathbf{x})} \frac{1}{m} \sum_{i=1}^m \sigma_i \ell(y_i, f(x_i)) \geq \frac{t}{4} \right\} + \mathsf{Pr}_{\mathbf{z},\boldsymbol{\sigma}} \left\{ \sup_{f \in \mathcal{F}_{\boldsymbol{\mu}^*,\eta}(\mathbf{x})} \frac{1}{m} \sum_{i=1}^m \sigma_i \ell(y_i, f(x_i)) \geq \frac{t}{4} \right\}.$$

Proof. From the definitions of the random subclasses $\mathcal{F}_{\boldsymbol{\mu}^*}(\cdot)$ and $\mathcal{F}_{\boldsymbol{\mu}^*,\eta}(\cdot)$, the left hand side in the lemma is equal to

$$\mathsf{Pr}_{\mathbf{z}\,\mathbf{z}'} \left\{ \sup_{f \in \mathcal{F}_{\boldsymbol{\mu}^*}(\mathbf{x}) \cap \mathcal{F}_{\boldsymbol{\mu}^*,\eta}(\mathbf{x}')} \frac{1}{m} \sum_{i=1}^m \big(\ell(y_i', f(x_i')) - \ell(y_i, f(x_i))\big) \geq \frac{t}{2} \right\}.$$

Now, by a routine application of symmetrization by random signs this is equal to

$$\mathsf{Pr}_{\mathbf{z}\mathbf{z}',\boldsymbol{\sigma}} \left\{ \sup_{f \in \mathcal{F}_{\boldsymbol{\mu}^*}(\mathbf{x}) \cap \mathcal{F}_{\boldsymbol{\mu}^*,\eta}(\mathbf{x}')} \frac{1}{m} \sum_{i=1}^m \sigma_i(\ell(y_i', f(x_i')) - \ell(y_i, f(x_i))) \geq \frac{t}{2} \right\}$$
$$\leq \mathsf{Pr}_{\mathbf{z},\boldsymbol{\sigma}} \left\{ \sup_{f \in \mathcal{F}_{\boldsymbol{\mu}^*}(\mathbf{x})} \frac{1}{m} \sum_{i=1}^m \sigma_i \ell(y_i, f(x_i)) \geq \frac{t}{4} \right\} + \mathsf{Pr}_{\mathbf{z},\boldsymbol{\sigma}} \left\{ \sup_{f \in \mathcal{F}_{\boldsymbol{\mu}^*,\eta}(\mathbf{x})} \frac{1}{m} \sum_{i=1}^m \sigma_i \ell(y_i, f(x_i)) \geq \frac{t}{4} \right\} \square$$

Since the good random subclass $\mathcal{F}_{\boldsymbol{\mu}^*}(\mathbf{x})$ is just the all-but-0-good random subclass $\mathcal{F}_{\boldsymbol{\mu}^*,0}(\mathbf{x})$, it is sufficient to bound the second term of the last line above for arbitrary $\eta \in [m]$. For fixed $\mathbf{z}$, the randomness of the subclass is annihilated and the above supremum over $\mathcal{F}_{\boldsymbol{\mu}^*,\eta}(\mathbf{x})$ is a conditional Rademacher average. Bounding this conditional Rademacher

average will call for a few results on the *Gaussian* average of a related function class.

First, note that for any $D \in \mathcal{D}$, the dictionary D can be factorized as $D = US$, where all $U \in \mathcal{U} \subset \mathbb{R}^{d\times k}$ satisfy the isometry property $U^T U = I$, and S lives in a space $\mathcal{S} := (B_{\mathbb{R}^k})^k$ of lower-dimensional dictionaries (Maurer and Pontil, 2010). Consider a particular choice of $S \in \mathcal{S}$, linear hypothesis $w \in \mathcal{W}$, and m-sample $\mathbf{x}$. The subclass of interest will be those functions corresponding to $U \in \mathcal{U}$ such that the encoder φ_{US} is s-sparse on $\mathbf{x}$. It turns out that the Gaussian average of this subclass is well-behaved.

Recall that $\boldsymbol{\gamma} = (\gamma_1, \ldots \gamma_m)$ where the γ_i are iid standard normals.

Theorem 2.18 (Gaussian Average for Fixed S and w). *Let $S \in \mathcal{S}$, $s \in [k]$, and $\mathbf{x}$ be a fixed m-sample. Denote by $\mathcal{U}_{\mathbf{x}}$ the particular subclass of $\mathcal{U}$ defined as:*

$$\mathcal{U}_{\mathbf{x}} := \big\{U \in \mathcal{U} : s\text{-sparse}(\varphi_{US}(\mathbf{x}))\big\}.$$

Then

$$\mathsf{E}_{\boldsymbol{\gamma}} \sup_{U\in\mathcal{U}_{\mathbf{x}}} \frac{2}{m}\sum_{i=1}^{m} \gamma_i \langle w, \varphi_{US}(x_i)\rangle \le \frac{4rk\sqrt{2s}}{\mu_{2s}(S)\sqrt{m}}. \tag{2.11}$$

The proof of this result uses the following lemma that shows how the difference between the feature maps φ_{US} and $\varphi_{U'S}$ can be characterized by the difference between U and U'. Define the *s-restricted 2-norm* of S as $\|S\|_{2,s} := \sup_{\{t\in\mathbb{R}^k : \|t\|=1, |\operatorname{supp}(t)|\le s\}} \|St\|_2$.

Lemma 2.19 (Difference Bound for Isometries). *Let $U, U' \in \mathcal{U}$ be isometries as above, $S \in \mathcal{S}$, and $x \in B_{\mathbb{R}^d}$. If $\|\varphi_{US}(x)\|_0 \le s$ and $\|\varphi_{U'S}(x)\|_0 \le s$, then*

$$\|\varphi_{US}(x) - \varphi_{U'S}(x)\|_2 \le \frac{2\|S\|_{2,2s}}{\mu_{2s}(S)} \|(U'^T - U^T)x\|_2.$$

Proof sketch The proof uses a perturbation analysis of solutions to linearly constrained positive definite quadratic programs (Daniel, 1973), exploiting the sparsity of the optimal solutions to have dependence only on $\|S\|_{2,2s}$ and $\mu_{2s}(S)$ rather than $\|S\|_2$ and $\mu_k(S)$. ■

Proof (of Theorem 2.18). Define a Gaussian process Ω, indexed by U, by

$$\Omega_U := \sum_{i=1}^{m} \gamma_i \langle w, \varphi_{US}(x_i) \rangle.$$

Our goal is to apply Slepian's Lemma (Lemma 2.10) to bound the expectation of the supremum of Ω, which depends on φ_{US}, by the expectation of the supremum of a Gaussian process Γ which depends only on U.

$$\begin{aligned}
\mathsf{E}_{\boldsymbol{\gamma}} (\Omega_U - \Omega_{U'})^2 &= \mathsf{E}_{\boldsymbol{\gamma}} \left(\sum_{i=1}^{m} \gamma_i \langle w, \varphi_{US}(x_i) \rangle - \sum_{i=1}^{m} \gamma_i \langle w, \varphi_{U'S}(x_i) \rangle \right)^2 \\
&= \sum_{i=1}^{m} (\langle w, \varphi_{US}(x_i) - \varphi_{U'S}(x_i) \rangle)^2 \\
&\leq r^2 \sum_{i=1}^{m} \|\varphi_{US}(x_i) - \varphi_{U'S}(x_i)\|^2 \qquad (2.12)
\end{aligned}$$

Applying the result from Lemma 2.19, we have

$$\begin{aligned}
\mathsf{E}_{\boldsymbol{\gamma}} (\Omega_U - \Omega_{U'})^2 &\leq r^2 \sum_{i=1}^{m} \|\varphi_{US}(x_i) - \varphi_{U'S}(x_i)\|^2 \\
&\leq \left(\frac{2r\|S\|_{2,2s}}{\mu_{2s}(S)} \right)^2 \sum_{i=1}^{m} \left\| (U'^T - U^T) x_i \right\|_2^2 \\
&= \left(\frac{2r\|S\|_{2,2s}}{\mu_{2s}(S)} \right)^2 \sum_{i=1}^{m} \sum_{j=1}^{k} (\langle U' e_j, x_i \rangle - \langle U e_j, x_i \rangle)^2 \\
&= \left(\frac{2r\|S\|_{2,2s}}{\mu_{2s}(S)} \right)^2 \mathsf{E}_{\boldsymbol{\gamma}} \left(\left(\sum_{i=1}^{m} \sum_{j=1}^{k} \gamma_{ij} \langle U' e_j, x_i \rangle \right) - \left(\sum_{i=1}^{m} \sum_{j=1}^{k} \gamma_{ij} \langle U e_j, x_i \rangle \right) \right)^2 \\
&= \mathsf{E}_{\boldsymbol{\gamma}} (\Gamma_U - \Gamma_{U'})^2
\end{aligned}$$

for $$\Gamma_U := \frac{2r\|S\|_{2,2s}}{\mu_{2s}(S)} \sum_{i=1}^{m} \sum_{j=1}^{k} \gamma_{ij} \langle U e_j, x_i \rangle.$$

By Slepian's Lemma (Lemma 2.10), $\mathsf{E}_{\boldsymbol{\gamma}} \sup_U \Omega_U \leq \mathsf{E}_{\boldsymbol{\gamma}} \sup_U \Gamma_U$. It remains to bound

$E_\gamma \sup_U \Gamma_U$:

$$\begin{aligned}
\frac{\mu_{2s}(S)}{2r\|S\|_{2,2s}}\,\mathsf{E}_{\boldsymbol{\gamma}} \sup_U \Gamma_U &= \mathsf{E}_{\boldsymbol{\gamma}} \sup_U \sum_{i=1}^m \sum_{j=1}^k \gamma_{ij}\langle Ue_j, x_i\rangle \\
&= \mathsf{E}_{\boldsymbol{\gamma}} \sup_U \sum_{j=1}^k \langle Ue_j, \sum_{i=1}^m \gamma_{ij}x_i\rangle \\
&\le \mathsf{E}_{\boldsymbol{\gamma}} \sup_U \sum_{j=1}^k \|Ue_j\| \|\sum_{i=1}^m \gamma_{ij}x_i\| \\
&= k\,\mathsf{E}_{\boldsymbol{\gamma}}\, \|\sum_{i=1}^m \gamma_{i1}x_i\| \\
&\le k\sqrt{\mathsf{E}_{\boldsymbol{\gamma}}\, \|\sum_{i=1}^m \gamma_{i1}x_i\|^2} \\
&= k\sqrt{\mathsf{E}_{\boldsymbol{\gamma}} \left\langle \sum_{i=1}^m \gamma_{i1}x_i, \sum_{i=1}^m \gamma_{i1}x_i \right\rangle} = k\sqrt{\sum_{i=1}^m \|x_i\|^2} \le k\sqrt{m}.
\end{aligned}$$

Hence,

$$\mathsf{E}_{\boldsymbol{\gamma}} \sup_{U\in\mathcal{U}} \frac{2}{m}\sum_{i=1}^m \gamma_i\langle w, \varphi_{US}(x_i)\rangle \;\le\; \frac{4r\|S\|_{2,2s}k}{\mu_{2s}(S)\sqrt{m}} \;\le\; \frac{4rk\sqrt{2s}}{\mu_{2s}(S)\sqrt{m}},$$

where we used the fact that $\|S\|_{2,2s} \le \sqrt{2s}$ (see Lemma 2.25 in Section 2.8.4 for a proof).□

We present the main result of this section:

Theorem 2.20 (Rademacher Average of Mostly Good Random Subclasses).

$$\mathsf{Pr}_{\mathbf{z},\boldsymbol{\sigma}}\left\{\sup_{f\in\mathcal{F}_{\boldsymbol{\mu}^*,\eta}(\mathbf{x})} \frac{1}{m}\sum_{i=1}^m \sigma_i\ell(y_i, f(x_i)) \ge \frac{t}{4}\right\} \le \binom{m}{\eta}\left(\frac{8(r/2)^{1/(k+1)}}{\varepsilon}\right)^{(k+1)k} \exp(-mt_3^2/(2b^2)),$$

for
$$t_3 := \frac{t}{4} - \frac{L\varepsilon}{2\lambda}\left(\frac{3r\sqrt{s}}{\mu_s^*}+1\right) - \frac{2L\sqrt{\pi}rk\sqrt{s}}{\mu_{2s}^*\sqrt{m}} - \frac{2b\eta}{m}.$$

Proof (of Theorem 2.20). As before, each dictionary $D \in \mathcal{D}$ will be factorized as $D = US$ for U an isometry in $\mathcal{U}$ and $S \in \mathcal{S} = (B_{\mathbb{R}^k})^k$. Let $\mathcal{S}_\varepsilon$ be a minimum-cardinality proper ε-cover (in operator norm) of $\{S \in \mathcal{S} : \mu_s(S) \ge \mu_s^*, \mu_{2s}(S) \ge \mu_{2s}^*\}$, the set of suitably

incoherent elements of $\mathcal{S}$.

Recall that the goal is to control the Rademacher complexity of $\mathcal{F}_{\boldsymbol{\mu}^*,\eta}(\mathbf{x})$. Our strategy will be to control this complexity by controlling the complexity of each subclass from a partition of $\mathcal{F}_{\boldsymbol{\mu}^*,\eta}(\mathbf{x})$. For an arbitrary $f = (D, w) \in \mathcal{F}$, let an index i be *good* if and only if $\left[\mathsf{margin}_s(D, x_i) > \tau\right]$, and let an index be *bad* if and only if it is not good. Consider a fixed m-sample $\mathbf{z}$ and the occurrence of a set of $m - \eta$ good indices[7]. There are $N := \binom{m}{\eta}$ ways to choose this set of indices. We can partition $\mathcal{F}_{\boldsymbol{\mu}^*,\eta}(\mathbf{x})$ into N subclasses $\mathcal{F}^1_{\boldsymbol{\mu}^*,\eta}(\mathbf{x}), \ldots, \mathcal{F}^N_{\boldsymbol{\mu}^*,\eta}(\mathbf{x})$ such that for all functions in a given subclass, a particular set of $m - \eta$ indices is guaranteed to be good. To be precise, we can choose distinct good index sets $\Gamma_1, \ldots, \Gamma_N$, each of cardinality $m-\eta$, such that for each Γ_j, if $i \in \Gamma_j$ then all $f = (D, w)$ in $\mathcal{F}^j_{\boldsymbol{\mu}^*,\eta}$ satisfy $\left[\mathsf{margin}_s(D, x_i) > \tau\right]$.

Since the $\mathcal{F}^j_{\boldsymbol{\mu}^*,\eta}(\mathbf{x})$ form a partition, we can control the complexity of $\mathcal{F}_{\boldsymbol{\mu}^*,\eta}(\mathbf{x})$ via:

$$\sup_{f \in \mathcal{F}_{\boldsymbol{\mu}^*,\eta}(\mathbf{x})} \sum_{i=1}^{m} \sigma_i \ell(y_i, f(x_i)) = \max_{j \in [N]} \sup_{f \in \mathcal{F}^j_{\boldsymbol{\mu}^*,\eta}(\mathbf{x})} \sum_{i=1}^{m} \sigma_i \ell(y_i, f(x_i)).$$

To gain a handle on the complexity of each subclass $\mathcal{F}^j_{\boldsymbol{\mu}^*,\eta}(\mathbf{x})$, we will approximate the subclasses as follows. For each $j \in [N]$, define an ε-neighborhood of $\mathcal{F}^j_{\boldsymbol{\mu}^*,\eta}(\mathbf{x})$ as

$$\bar{\mathcal{F}}^j_{\mu^*,\eta}(\mathbf{x}) := \left\{ f = (US', w') : \begin{array}{llll} \|S - S'\| \le \varepsilon, & \|w - w'\| \le \varepsilon, & & \\ S \in \mathcal{S}, & w \in \mathcal{W}, & (US, w) \in \mathcal{F}^j_{\boldsymbol{\mu}^*,\eta}(\mathbf{x}) & \end{array} \right\};$$

note that the ε neighborhood is taken with respect to S and w but not U. Also, let $\mathcal{W}_\varepsilon$ be a minimum-cardinality ε-cover of $\mathcal{W}$ and define an infinite-cardinality epsilon-net of $\mathcal{F}$:

$$\mathcal{F}_\varepsilon := \left\{ f = (US', w') \in \mathcal{F} : U \in \mathcal{U}, S' \in \mathcal{S}_\varepsilon, w' \in \mathcal{W}_\varepsilon \right\}.$$

Finally, taking the intersection of $\bar{\mathcal{F}}^j_{\mu^*,\eta}(\mathbf{x})$ with $\mathcal{F}_\varepsilon$ yields the $\mathcal{F}^j_{\boldsymbol{\mu}^*,\eta}(\mathbf{x})$-approximating

[7] Each of the remaining indices can be either good or bad.

subclass, a disjoint union of subclasses equal to

$$\bigcup_{S' \in \mathcal{S}_\varepsilon, w' \in \mathcal{W}_\varepsilon} \mathcal{F}^{j,S',w'}_{\boldsymbol{\mu}^*,\eta}(\mathbf{x})$$

for

$$\mathcal{F}^{j,S',w'}_{\boldsymbol{\mu}^*,\eta}(\mathbf{x}) := \mathcal{F}^{j}_{\boldsymbol{\mu}^*,\eta}(\mathbf{x}) \cap \left\{ f \in \mathcal{F} : f = (US', w') : U \in \mathcal{U} \right\}.$$

To show that this disjoint union is a good approximator for $\mathcal{F}^{j}_{\boldsymbol{\mu}^*,\eta}(\mathbf{x})$, for each $j \in [N]$ and arbitrary $\boldsymbol{\sigma} \in \{-1,1\}^m$ we compare

$$\sup_{f \in \mathcal{F}^{j}_{\boldsymbol{\mu}^*,\eta}(\mathbf{x})} \frac{1}{m} \sum_{i=1}^{m} \sigma_i \ell(y_i, f(x_i)) \qquad \text{and} \qquad \max_{S' \in \mathcal{S}_\varepsilon, w' \in \mathcal{W}_\varepsilon} \sup_{f \in \mathcal{F}^{j,S',w'}_{\boldsymbol{\mu}^*,\eta}(\mathbf{x})} \sum_{i=1}^{m} \sigma_i \ell(y_i, f(x_i)).$$

Without loss of generality, choose $j = 1$ and take $\Gamma_1 = [m-\eta]$. If f is in $\mathcal{F}^{1}_{\boldsymbol{\mu}^*,\eta}(\mathbf{x})$, it follows that there exists an f' in the disjoint union $\bigcup_{S' \in \mathcal{S}_\varepsilon, w' \in \mathcal{W}_\varepsilon} \mathcal{F}^{1,S',w'}_{\boldsymbol{\mu}^*,\eta}(\mathbf{x})$ such that

$$\begin{aligned}
&\frac{1}{m} \sum_{i=1}^{m} \sigma_i \left| \ell(y_i, \langle w, \varphi_D(x_i) \rangle) - \ell(y_i, \langle w', \varphi_{D'}(x_i) \rangle) \right| \\
&\quad \leq \frac{L}{m} \left(\sum_{i=1}^{m-\eta} \sigma_i \left| \langle w, \varphi_D(x_i) \rangle - \langle w', \varphi_{D'}(x_i) \rangle \right| \right) + \frac{1}{m} \sum_{i=m-\eta+1}^{m} \sigma_i \left| \ell(y_i, \langle w, \varphi_D(x_i) \rangle) - \ell(y_i, \langle w', \varphi_{D'}(x_i) \rangle) \right| \\
&\quad \leq \frac{L\varepsilon}{2\lambda} \left(\frac{3r\sqrt{s}}{\mu_s^*} + 1 \right) + \frac{b\eta}{m},
\end{aligned}$$

where the last line is due to the Sparse Coding Stability Theorem (Theorem 2.4).

Therefore, for any $\boldsymbol{\sigma} \in \{-1,1\}^m$ it holds that

$$\begin{aligned}
&\sup_{f \in \mathcal{F}_{\boldsymbol{\mu}^*,\eta}(\mathbf{x})} \frac{1}{m} \sum_{i=1}^{m} \sigma_i \ell(y_i, f(x_i)) \\
&\leq \max_{j \in [N]} \max_{S' \in \mathcal{S}_\varepsilon, w' \in \mathcal{W}_\varepsilon} \sup_{f \in \mathcal{F}^{j,S',w'}_{\boldsymbol{\mu}^*,\eta}(\mathbf{x})} \frac{1}{m} \sum_{i=1}^{m} \sigma_i \ell(y_i, f(x_i)) + \frac{L\varepsilon}{2\lambda} \left(\frac{3r\sqrt{s}}{\mu_s^*} + 1 \right) + \frac{b\eta}{m}.
\end{aligned}$$

Thus, the approximation error from using the disjoint union is small (it is $O(\frac{1}{m})$ if $\varepsilon = \frac{1}{m}$).

It remains to control the complexity of the approximating subclass. From the above,

for fixed $\mathbf{z}$:

$$\begin{aligned}
&\Pr_{\boldsymbol{\sigma}}\left\{\sup_{f\in\mathcal{F}_{\boldsymbol{\mu}^*,\eta}(\mathbf{x})}\frac{1}{m}\sum_{i=1}^{m}\sigma_i\ell(y_i,f(x_i))\geq\frac{t}{4}\right\}\\
&\leq\Pr_{\boldsymbol{\sigma}}\left\{\max_{\substack{j\in[N]\\S'\in\mathcal{S}_\varepsilon,w'\in\mathcal{W}_\varepsilon}}\sup_{f\in\mathcal{F}^{j,S',w'}_{\boldsymbol{\mu}^*,\eta}(\mathbf{x})}\frac{1}{m}\sum_{i=1}^{m}\sigma_i\ell(y_i,f(x_i))\geq\frac{t}{4}-\frac{L\varepsilon}{2\lambda}\left(\frac{3r\sqrt{s}}{\mu_s^*}+1\right)-\frac{b\eta}{m}\right\}\\
&\leq N\left(\frac{8(r/2)^{1/(k+1)}}{\varepsilon}\right)^{(k+1)k}\cdot\\
&\qquad\max_{\substack{j\in[N]\\S'\in\mathcal{S}_\varepsilon,w'\in\mathcal{W}_\varepsilon}}\Pr_{\boldsymbol{\sigma}}\left\{\sup_{f\in\mathcal{F}^{j,S',w'}_{\boldsymbol{\mu}^*,\eta}(\mathbf{x})}\frac{1}{m}\sum_{i=1}^{m}\sigma_i\ell(y_i,f(x_i))\geq\frac{t}{4}-\frac{L\varepsilon}{2\lambda}\left(\frac{3r\sqrt{s}}{\mu_s^*}+1\right)-\frac{b\eta}{m}\right\}.
\end{aligned}$$

Now, from the Bounded Differences Inequality (Theorem 2.11), for any fixed $j\in[N]$, $S'\in\mathcal{S}_\varepsilon$ and $w'\in\mathcal{W}_\varepsilon$,

$$\Pr_{\boldsymbol{\sigma}}\left\{\sup_{f\in\mathcal{F}^{j,S',w'}_{\boldsymbol{\mu}^*,\eta}(\mathbf{x})}\frac{1}{m}\sum_{i=1}^{m}\sigma_i\ell(y_i,f(x_i))>\mathsf{E}_{\boldsymbol{\sigma}}\sup_{f\in\mathcal{F}^{j,S',w'}_{\boldsymbol{\mu}^*,\eta}(\mathbf{x})}\frac{1}{m}\sum_{i=1}^{m}\sigma_i\ell(y_i,f(x_i))+t_1\right\}$$

is at most $\exp(-mt_1^2/(2b^2))$.

To make the above useful, let us get a handle on the Rademacher complexity term

$$\mathsf{E}_{\boldsymbol{\sigma}}\sup_{f\in\mathcal{F}^{j,S',w'}_{\boldsymbol{\mu}^*,\eta}(\mathbf{x})}\frac{1}{m}\sum_{i=1}^{m}\sigma_i\ell(y_i,f(x_i)).$$

Without loss of generality, again take $j = 1$ and $\Gamma_1 = [m - \eta]$. Then

$$\begin{aligned}
&\mathsf{E}_{\boldsymbol{\sigma}} \sup_{f \in \mathcal{F}^{j,S',w'}_{\boldsymbol{\mu}^*,\eta}(\mathbf{x})} \frac{1}{m} \sum_{i=1}^{m} \sigma_i \ell(y_i, f(x_i)) \\
&\leq \mathsf{E}_{\sigma_1,\ldots,\sigma_{m-\eta}} \left\{ \sup_{f \in \mathcal{F}^{j,S',w'}_{\boldsymbol{\mu}^*,\eta}(\mathbf{x})} \frac{1}{m} \sum_{i=1}^{m-\eta} \sigma_i \ell(y_i, f(x_i)) \right\} \\
&\quad + \mathsf{E}_{\sigma_{m-\eta+1},\ldots,\sigma_m} \left\{ \sup_{f \in \mathcal{F}^{j,S',w'}_{\boldsymbol{\mu}^*,\eta}(\mathbf{x})} \frac{1}{m} \sum_{i=m-\eta+1}^{m} \sigma_i \ell(y_i, f(x_i)) \right\} \\
&\leq \mathsf{E}_{\sigma_1,\ldots,\sigma_{m-\eta}} \left\{ \sup_{f \in \mathcal{F}^{j,S',w'}_{\boldsymbol{\mu}^*,\eta}(\mathbf{x})} \frac{1}{m} \sum_{i=1}^{m-\eta} \sigma_i \ell(y_i, f(x_i)) \right\} + \frac{b\eta}{m}.
\end{aligned}$$

Now, Theorem 2.18, the Rademacher Loss Comparison Lemma (Lemma 2.8), and the Rademacher-Gaussian Average Comparison Lemma (Lemma 2.9) imply that

$$\begin{aligned}
\mathsf{E}_{\boldsymbol{\sigma}} \sup_{\{U \in \mathcal{U} : s\text{-sparse}(\varphi_{US}(\mathbf{x}))\}} \frac{1}{m} \sum_{i=1}^{m-\eta} \sigma_i \ell\big(y_i, \langle w, \varphi_{US}(x_i) \rangle\big) &\leq \frac{\sqrt{m-\eta}}{m} \frac{2L\sqrt{\pi} r k \sqrt{s}}{\mu_{2s}(S)} \\
&\leq \frac{2L\sqrt{\pi} r k \sqrt{s}}{\mu_{2s}(S)\sqrt{m}},
\end{aligned}$$

and hence

$$\mathsf{Pr}_{\boldsymbol{\sigma}} \left\{ \sup_{f \in \mathcal{F}^{j,S',w'}_{\boldsymbol{\mu}^*,\eta}(\mathbf{x})} \frac{1}{m} \sum_{i=1}^{m} \sigma_i \ell(f(x_i)) > \frac{2L\sqrt{\pi} r k \sqrt{s}}{\mu^*_{2s}\sqrt{m}} + \frac{b\eta}{m} + t_1 \right\} \leq \exp(-m t_1^2/(2b^2)).$$

Combining this bound with the fact that the bound is independent of the draw of $\mathbf{z}$ and applying Proposition 2.27 (with d set to k) to extend the bound over all choices of j, S', and w' yields the final result. □

For the case of $\eta = 0$, let

$$t_2 := \frac{t}{4} - \frac{L\varepsilon}{2\lambda} \left(\frac{3r\sqrt{s}}{\mu^*_s} + 1 \right) - \frac{2L\sqrt{\pi} r k \sqrt{s}}{\mu^*_{2s}\sqrt{m}}.$$

It is now possible to prove the generalization error bound for the infinite-dimensional setting.

Proof (of Theorem 2.6). Since $\mathcal{F}_{\mu_*}(\mathbf{x})$ is equivalent to $\mathcal{F}_{\boldsymbol{\mu}^*,0}(\mathbf{x})$, Lemma 2.17 and Theorem 2.20 imply that

$$\begin{aligned}
&\mathsf{Pr}_{\mathbf{z}\,\mathbf{z}'}\left\{\begin{array}{l}\exists f \in \mathcal{F}_{\boldsymbol{\mu}^*},\ \left[\mathsf{margin}_s(D,\mathbf{x}) > \tau\right] \textbf{ and } \left(\exists\, \tilde{\mathbf{x}} \subseteq_\eta \mathbf{x}'\ \left[\mathsf{margin}_s(D,\tilde{\mathbf{x}}) > \tau\right]\right) \\ \qquad \textbf{and } \left((\mathsf{P}_{\mathbf{z}'} - \mathsf{P}_{\mathbf{z}})\ell(\cdot, f) \geq \frac{t}{2}\right)\end{array}\right\} \\
&\leq \left(\frac{8(r/2)^{1/(k+1)}}{\varepsilon}\right)^{(k+1)k}\left(\exp(-mt_2^2/(2b^2)) + \binom{m}{\eta}\exp(-mt_3^2/(2b^2))\right) \\
&\leq 2\binom{m}{\eta}\left(\frac{8(r/2)^{1/(k+1)}}{\varepsilon}\right)^{(k+1)k}\exp(-mt_3^2/(2b^2)),
\end{aligned}$$

and consequently the full probability (2.10) in Proposition 2.15 can be upper bounded (using $\eta = \log\frac{1}{\delta}$) as:

$$\begin{aligned}
&\mathsf{Pr}_{\mathbf{z}\,\mathbf{x}''}\left\{\ \hat{f}_{\mathbf{z}} \in \mathcal{F}_{\boldsymbol{\mu}^*},\ \left[\mathsf{margin}_s(\hat{D}_{\mathbf{z}}, (\mathbf{x}\cup\mathbf{x}'')) > \tau\right] \textbf{ and } \left((\mathsf{P} - \mathsf{P}_{\mathbf{z}})\ell(\cdot, \hat{f}_{\mathbf{z}}) \geq t\right)\ \right\} \\
&\leq 4\binom{m}{\log\frac{1}{\delta}}\left(\frac{8(r/2)^{1/(k+1)}}{\varepsilon}\right)^{(k+1)k}\exp(-mt_3^2/(2b^2)) + 2\delta.
\end{aligned}$$

After some elementary manipulations and choosing $\varepsilon = \frac{1}{m}$, we nearly have the final learning bound. Let $\mu_s^*, \mu_{2s}^* > 0$, $s \in [k]$, and $m \geq \frac{27}{\tau^2\lambda}$ be fixed a priori. With probability at least $1-\delta$ over a labeled m-sample $\mathbf{z} \sim \mathsf{P}^m$ and a second, unlabeled m-sample $\mathbf{x}'' \sim \Pi^m$, if an algorithm learns hypothesis $\hat{f}_{\mathbf{z}} = (\hat{D}_{\mathbf{z}}, \hat{w}_{\mathbf{z}})$ from $\mathbf{z}$ such that $\mu_{2s}(\hat{D}_{\mathbf{z}}) \geq \mu_{2s}^*$, $\mu_s(\hat{D}_{\mathbf{z}}) \geq \mu_s^*$, s-$\mathsf{sparse}(\varphi_{\hat{D}_{\mathbf{z}}}(\mathbf{x}\cup\mathbf{x}''))$, and $\left[\mathsf{margin}_s(\hat{D}_{\mathbf{z}}, \mathbf{x}\cup\mathbf{x}'') > \tau\right]$ all hold, then the generalization error $(\mathsf{P} - \mathsf{P}_{\mathbf{z}})\ell(\cdot, \hat{f}_{\mathbf{z}})$ is bounded by:

$$\begin{aligned}
&\frac{8L\sqrt{\pi}rk\sqrt{s}}{\mu_{2s}^*\sqrt{m}} + b\sqrt{\frac{8\left((k+1)k\log(8m) + k\log\frac{r}{2} + (\log m + 1)\log\frac{4}{\delta} + \log 2\right)}{m}} \\
&\quad + \frac{1}{m}\left(\frac{2L}{\lambda}\left(\frac{3r\sqrt{s}}{\mu_s^*} + 1\right) + 8b\log\frac{4}{\delta}\right).
\end{aligned}$$

Making this bound adaptive to the incoherences, sparsity level, and margin on $\mathbf{z}$ and $\mathbf{x}''$ yields the following final result. With probability at least $1-\delta$ over a labeled m-sample $\mathbf{z} \sim \mathsf{P}^m$ and a second, unlabeled sample $\mathbf{x}'' \sim \Pi^m$, if an algorithm learns hypothesis $\hat{f}_{\mathbf{z}} =$

$(\hat{D}_{\mathbf{z}}, \hat{w}_{\mathbf{z}})$ such that $\varphi_{\hat{D}_{\mathbf{z}}}$ is s-sparse on $(\mathbf{x} \cup \mathbf{x}'')$, $\mu_{2s}(\hat{f}_{\mathbf{z}}) > 0$, and

$$m \geq \frac{27}{\text{margin}_s^2(\hat{D}_{\mathbf{z}}, \mathbf{x} \cup \mathbf{x}'') \cdot \lambda},$$

then the generalization error $(P - P_{\mathbf{z}})\ell(\cdot, \hat{f}_{\mathbf{z}})$ is bounded by:

$$\frac{16L\sqrt{\pi}rk\sqrt{s}}{\mu_{2s}(\hat{f}_{\mathbf{z}})\sqrt{m}} + b\sqrt{\frac{8\left((k^2+k)\log(8m) + k\log\frac{r}{2} + (\log m + 1)\log\frac{7\alpha k}{\delta} + \log 2\right)}{m}} + \frac{1}{m}\left(\frac{2L}{\lambda}\left(\frac{6r\sqrt{s}}{\mu_s(\hat{f}_{\mathbf{z}})} + 1\right) + 8b\log\frac{7\alpha k}{\delta}\right),$$

for $\alpha = \left(\log_2\left(\frac{4}{\mu_s(\hat{f}_{\mathbf{z}})}\right)\log_2\left(\frac{4}{\mu_{2s}(\hat{f}_{\mathbf{z}})}\right)\right)^2$. □

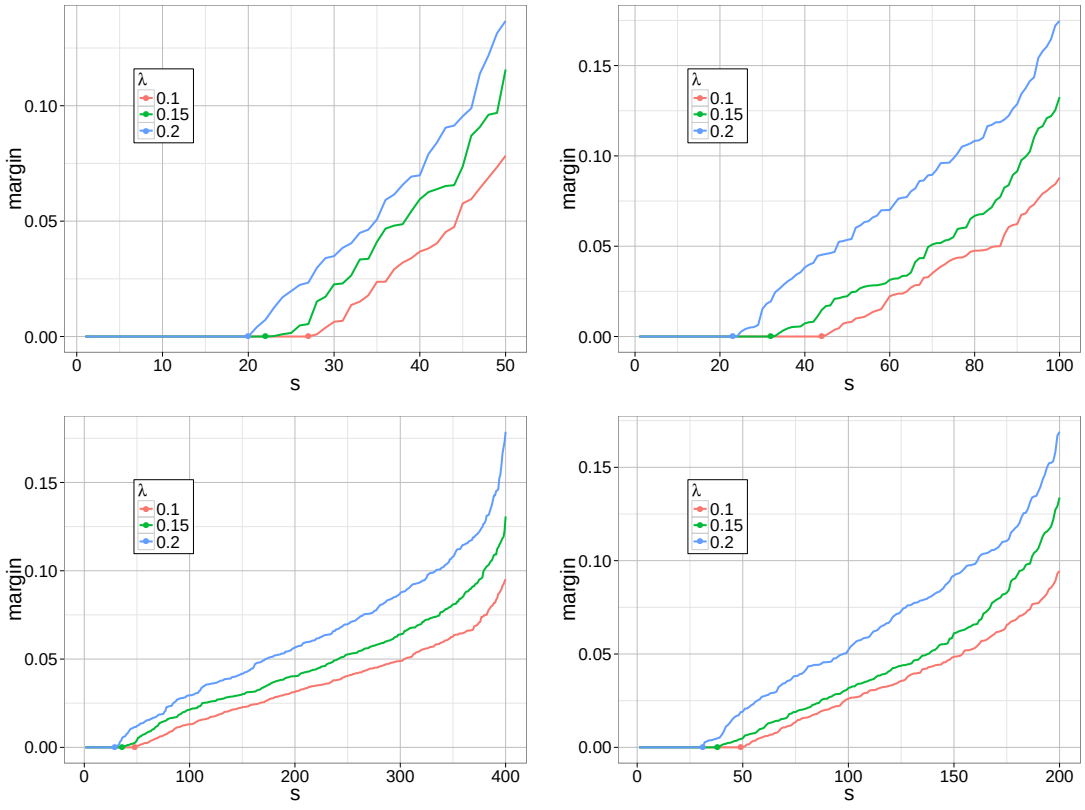


Figure 2.4: The s-margin for predictive sparse coding trained on the USPS training set, digit 4 versus all, for three settings of λ. Clockwise from top left: 50 atoms, 100 atoms, 200 atoms, and 400 atoms. The sparsity level (maximum number of non-zeros per code, taken across all codes of the training points) is indicated by the dots.

2.6 An empirical study of the s-margin

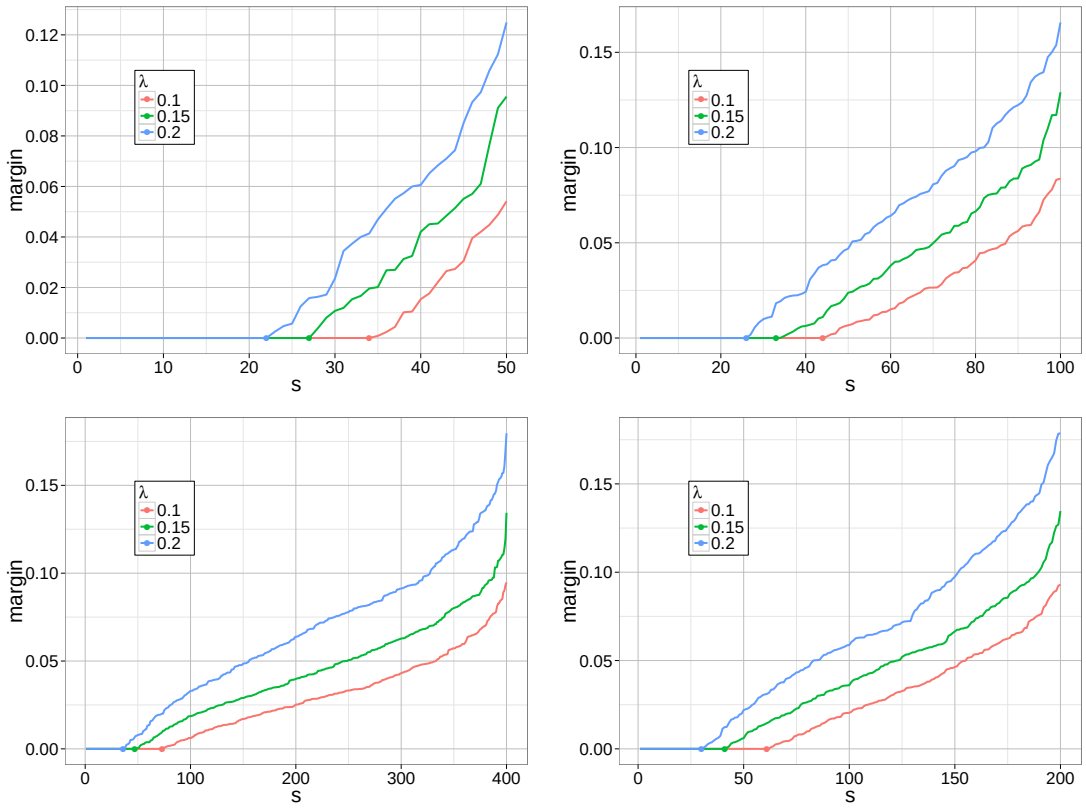


Figure 2.5: The s-margin for predictive sparse coding trained on the MNIST training set, digit 4 versus all, for three settings of λ. Clockwise from top left: 50 atoms, 100 atoms, 200 atoms, and 400 atoms. The sparsity level (maximum number of non-zeros per code, taken across all codes of the training points) is indicated by the dots.

Empirical evidence suggests that the s-margin is well above zero even when s is only slightly larger than the observed sparsity level. We performed experiments on two separate digit classification tasks, from the USPS dataset and the MNIST dataset LeCun et al. (1998). In both cases, we employed the single binary classification task of the digit 4 versus all the other digits, and for both datasets all the training data was used. The results for USPS and MNIST are shown in Figures 2.4 and 2.5 respectively. Each data point (an image) was normalized to unit norm. In all plots, it is apparent that when the minimum sparsity level is s (indicated by the colored dots on the x-axis of the plots), then using an $(s+\rho)$-margin for ρ a small positive integer yields a non-trivial margin. Using the $2s$-margin

when s-sparsity holds may ensure that there is a moderate margin for only a constant factor increase to s.

2.7 Discussion and open problems

We have shown the first generalization error bounds for predictive sparse coding. The learning bounds in Theorems 2.5 and 2.6 are intimately related to the stability of the sparse encoder, and consequently the bounds depend on both the learned dictionary and the training sample. Using the techniques of this chapter, in the infinite-dimensional setting it is unclear whether one can achieve the encoder stability guarantees without measuring properties of the encoder on an independent, unlabeled sample. It is an important open problem whether there is a generalization error bound for the infinite-dimensional setting which does not rely on the second sample. Additionally, the PRP condition in the Sparse Coding Stability Theorem (Theorem 2.4) appears to be much stronger than what should be required. We conjecture that the PRP should actually be $O(\varepsilon)$ rather than $O(\sqrt{\varepsilon})$. If this conjecture turns out to be true, then the number of samples required before Theorems 2.5 and 2.6 kick in would be greatly reduced, and many constants in these results would likewise be massively reduced.

While this chapter establishes upper bounds on the generalization error for predictive sparse coding, two things remain unclear. How close are these bounds to the optimal ones? Also, what lower bounds can be established in each of the settings? If the conditions on which these bounds rely are of fundamental importance, then the presented data-dependent bounds provide motivation for an algorithm to prefer dictionaries for which small subdictionaries are well-conditioned and to additionally encourage large coding margin on the training sample.

2.8 Additional proofs

2.8.1 Proof of Sparse Coding Stability Theorem

The flow of this section is as follows. We first establish some preliminary notation and summarize important conditions. Several lemmas are then presented to support a key sparsity lemma. This sparsity lemma establishes that the solution to the perturbed problem is sparse provided the perturbation is not too large. Finally, the sparsity of this new solution is exploited to bound the difference of the new solution from the old solution. This flow is embodied by the proof flowchart in Figure 2.6.

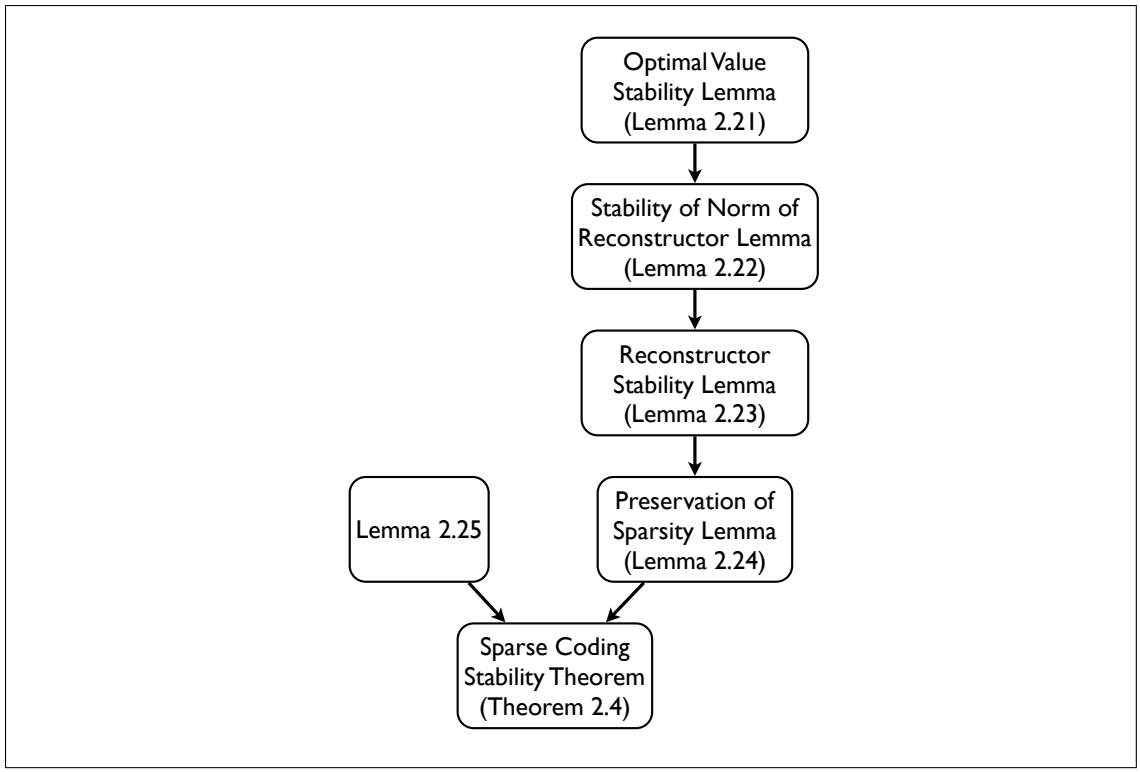


Figure 2.6: Proof flowchart for the Sparse Coding Stability Theorem (Theorem 2.4).

Notation and assumptions

Let α and $\tilde{\alpha}$ respectively denote the solutions to the Lasso problems:

$$\alpha = \arg\min_z \frac{1}{2}\|x - Dz\|_2^2 + \lambda\|z\|_1 \qquad \tilde{\alpha} = \arg\min_z \frac{1}{2}\|x - \tilde{D}z\|_2^2 + \lambda\|z\|_1.$$

First, let's review the optimality conditions for the Lasso (Asif and Romberg, 2010, conditions L1 and L2):

$$\langle D_j, x - D\alpha \rangle = \mathsf{sign}(\alpha_j)\lambda \quad \text{if } \alpha_j \neq 0,$$
$$|\langle D_j, x - D\alpha \rangle| < \lambda \quad \text{otherwise.}$$

Note that the above optimality conditions imply that if $\alpha_j \neq 0$ then

$$|\langle D_j, x - D\alpha \rangle| = \lambda.$$

Assumptions

The statement of the Sparse Coding Stability Theorem (Theorem 2.4) makes the following assumptions:

(A1) - Closeness D and $\tilde{D}$ are close, as measured by operator norm:

$$\|\tilde{D} - D\|_2 \leq \varepsilon.$$

(A2) - Incoherence There is a $\mu > 0$ such that, for all $J \subseteq [k]$ satisfying $|J| = s$:

$$\sigma_{\mathsf{min}}(D_J) \geq \mu.$$

(A3) - Sparsity with Margin For some fixed $\tau > 0$, there is a $\mathcal{I} \subseteq [k]$ with $|\mathcal{I}| = k - s$ such that for all $i \in \mathcal{I}$:

$$|\langle D_i, x - D\alpha \rangle| < \lambda - \tau.$$

Consequently, all $i \in \mathcal{I}$ satisfy $\alpha_i = 0$.

Useful observations

Let v_D^* be the optimal value of the Lasso for dictionary D:

$$\begin{aligned} v_D^* &= \min_z \frac{1}{2}\|x - Dz\|_2^2 + \lambda\|z\|_1 \\ &= \frac{1}{2}\|x - D\alpha\|_2^2 + \lambda\|\alpha\|_1 \end{aligned}$$

Likewise, let

$$v_{\tilde{D}}^* = \frac{1}{2}\|x - \tilde{D}\tilde{\alpha}\|_2^2 + \lambda\|\tilde{\alpha}\|_1$$

The first observation is that the values of the optimal solutions are close:

Lemma 2.21 (Optimal Value Stability). *If $\|D - \tilde{D}\|_2 \leq \varepsilon$, then*

$$\left|v_D^* - v_{\tilde{D}}^*\right| \leq \frac{5}{8}\frac{\varepsilon}{\lambda}.$$

Proof. The proof is simple:

$$\begin{aligned} v_{\tilde{D}}^* &\leq \frac{1}{2}\|x - \tilde{D}\alpha\|_2^2 + \lambda\|\alpha\|_1 \\ &= \frac{1}{2}\|x - D\alpha + (D - \tilde{D})\alpha\|_2^2 + \lambda\|\alpha\|_1 \\ &\leq \frac{1}{2}\left(\|x - D\alpha\|_2^2 + 2\|x - D\alpha\|_2\|(D - \tilde{D})\alpha\|_2 + \|(D - \tilde{D})\alpha\|_2^2\right) + \lambda\|\alpha\|_1 \\ &\leq \frac{1}{2}\|x - D\alpha\|_2^2 + \lambda\|\alpha\|_1 + \frac{1}{2}\left(\frac{\varepsilon}{\lambda} + \frac{1}{4}\left(\frac{\varepsilon}{\lambda}\right)^2\right) \\ &\leq v_D^* + \frac{5}{8}\frac{\varepsilon}{\lambda} \end{aligned}$$

for $\frac{\varepsilon}{\lambda} \leq 1$. A symmetric argument shows that $v_D^* \leq v_{\tilde{D}}^* + \frac{5}{8}\frac{\varepsilon}{\lambda}$. □

The second observation shows that the norms of the optimal reconstructors are close.

Lemma 2.22 (Stability of Norm of Reconstructor). *If $\|D - \tilde{D}\|_2 \leq \varepsilon$, then*

$$\left|\|D\alpha\|_2^2 - \|\tilde{D}\tilde{\alpha}\|_2^2\right| \leq \frac{5}{4}\frac{\varepsilon}{\lambda}.$$

Showing this is more involved than the previous observation.

Proof. First, we claim (and show) that

$$(x - D\alpha)^T D\alpha = \lambda \|\alpha\|_1. \tag{2.13}$$

The proof of the claim comes directly from Osborne et al. (2000, circa (2.8)) To see (2.13), recall that the Lasso objective is

$$\underset{z}{\text{minimize}} \quad \frac{1}{2}\|x - Dz\|_2^2 + \lambda \|z\|_1.$$

The subgradient of this objective with respect to z is

$$-D^T(x - Dz) + \lambda v,$$

where $v_j = 1$ if $z_j > 0$, $v_j = -1$ if $z_j < 0$, and $v_j \in [-1, 1]$ if $z_j = 0$. From the definition of v, it follows that

$$v^T z = \|z\|_1.$$

At an optimal point α, $\partial_z \mathcal{L}(\alpha, \lambda) = 0$, and hence

$$\begin{gathered}
D^T(x - D\alpha) = \lambda v \\
\Updownarrow \\
(x - D\alpha)^T D = \lambda v^T \\
\Downarrow \\
(x - D\alpha)^T D\alpha = \lambda v^T \alpha \\
\Updownarrow \\
(x - D\alpha)^T D\alpha = \lambda \|\alpha\|_1,
\end{gathered}$$

as claimed.

Now, we use the fact that the values of the optimal solutions are close (Lemma 2.21):

$$\left|v_D^* - v_{\tilde{D}}^*\right| \leq \frac{5}{8}\frac{\varepsilon}{\lambda}.$$

But v_D^* is just

$$\begin{aligned}
\frac{1}{2}\langle x - D\alpha, x - D\alpha\rangle + \lambda\|\alpha\|_1 &= \frac{1}{2}\langle x - D\alpha, x - D\alpha\rangle + \langle x - D\alpha, D\alpha\rangle \\
&= \frac{1}{2}\langle x, x - D\alpha\rangle - \frac{1}{2}\langle x - D\alpha, D\alpha\rangle + \langle x - D\alpha, D\alpha\rangle \\
&= \frac{1}{2}\left(\langle x, x - D\alpha\rangle + \langle x - D\alpha, D\alpha\rangle\right) \\
&= \frac{1}{2}\langle x + D\alpha, x - D\alpha\rangle \\
&= \frac{1}{2}\left(\|x\|_2^2 - \|D\alpha\|_2^2\right).
\end{aligned}$$

Consequently,

$$\left|\frac{1}{2}\left(\|x\|_2^2 - \|D\alpha\|_2^2\right) - \frac{1}{2}\left(\|x\|_2^2 - \|\tilde{D}\tilde{\alpha}\|_2^2\right)\right| \leq \frac{5}{8}\frac{\varepsilon}{\lambda}$$

and hence

$$\left|\|D\alpha\|_2^2 - \|\tilde{D}\tilde{\alpha}\|_2^2\right| \leq \frac{5}{4}\frac{\varepsilon}{\lambda}. \qquad \square$$

Finally, we prove stability of the optimal reconstructor. Rather than showing that $\|D\alpha - \tilde{D}\tilde{\alpha}\|_2^2$ is $O(\varepsilon)$, it will be more convenient for later purposes to prove the following roughly equivalent result.

Lemma 2.23 (Reconstructor Stability). *If $\|D - \tilde{D}\|_2 \leq \varepsilon$, then*

$$\|D\alpha - D\tilde{\alpha}\|_2^2 \leq \frac{13\varepsilon}{\lambda}.$$

Proof. Let $\alpha' := \frac{1}{2}(\alpha + \tilde{\alpha})$. From the optimality of α, it follows that $v_D(\alpha) \leq v_D(\alpha')$, or

more explicitly:

$$\frac{1}{2}\|x - D\alpha\|_2^2 + \lambda\|\alpha\|_1 \leq \frac{1}{2}\|x - D\alpha'\|_2^2 + \lambda\|\alpha'\|_1. \tag{2.14}$$

First, note that $\left|\|D\tilde{\alpha}\|_2^2 - \|\tilde{D}\tilde{\alpha}\|_2^2\right| \leq \frac{7}{4}\frac{\varepsilon}{\lambda}$, because

$$\begin{aligned}
\left|\|D\tilde{\alpha}\|_2^2 - \|\tilde{D}\tilde{\alpha}\|_2^2\right| &\leq 2\left|\langle D\tilde{\alpha}, (\tilde{D} - D)\alpha\rangle\right| + \|(\tilde{D} - D)\tilde{\alpha}\|_2^2 \\
&\leq 2\|D\tilde{\alpha}\|_2\|\tilde{D} - D\|_2\|\tilde{\alpha}\|_2 + \left(\|\tilde{D} - D\|_2\|\tilde{\alpha}\|_2\right)^2 \\
&\leq 2\left(1 + \frac{\varepsilon}{2\lambda}\right)\frac{\varepsilon}{2\lambda} + \frac{1}{4}\left(\frac{\varepsilon}{\lambda}\right)^2 \\
&\leq \frac{7}{4}\frac{\varepsilon}{\lambda},
\end{aligned}$$

assuming $\varepsilon \leq \lambda$. Combining this fact with Lemma 2.22, $\left|\|D\alpha\|_2^2 - \|\tilde{D}\tilde{\alpha}\|_2^2\right| \leq \frac{5}{4}\frac{\varepsilon}{\lambda}$, yields

$$\left|\|D\alpha\|_2^2 - \|D\tilde{\alpha}\|_2^2\right| \leq \frac{3\varepsilon}{\lambda}.$$

By the convexity of the 1-norm, the RHS of (2.14) obeys:

$$\begin{aligned}
&\frac{1}{2}\left\|x - D\left(\frac{\alpha + \tilde{\alpha}}{2}\right)\right\|_2^2 + \lambda\left\|\frac{\alpha + \tilde{\alpha}}{2}\right\|_1 \\
&\leq \frac{1}{2}\left\|x - \frac{1}{2}(D\alpha + D\tilde{\alpha})\right\|_2^2 + \frac{\lambda}{2}\|\alpha\|_1 + \frac{\lambda}{2}\|\tilde{\alpha}\|_1 \\
&= \frac{1}{2}\left(\|x\|_2^2 - 2\langle x, \frac{1}{2}(D\alpha + D\tilde{\alpha})\rangle + \frac{1}{4}\|D\alpha + D\tilde{\alpha}\|_2^2\right) + \frac{\lambda}{2}\|\alpha\|_1 + \frac{\lambda}{2}\|\tilde{\alpha}\|_1 \\
&= \frac{1}{2}\|x\|_2^2 - \frac{1}{2}\langle x, D\alpha\rangle - \frac{1}{2}\langle x, D\tilde{\alpha}\rangle + \frac{1}{8}\left(\|D\alpha\|_2^2 + \|D\tilde{\alpha}\|_2^2 + 2\langle D\alpha, D\tilde{\alpha}\rangle\right) + \frac{\lambda}{2}\|\alpha\|_1 + \frac{\lambda}{2}\|\tilde{\alpha}\|_1 \\
&\leq \frac{1}{2}\|x\|_2^2 - \frac{1}{2}\langle x, D\alpha\rangle - \frac{1}{2}\langle x, D\tilde{\alpha}\rangle + \frac{1}{4}\|D\alpha\|_2^2 + \frac{1}{4}\langle D\alpha, D\tilde{\alpha}\rangle + \frac{\lambda}{2}\|\alpha\|_1 + \frac{\lambda}{2}\|\tilde{\alpha}\|_1 + \frac{3}{8}\frac{\varepsilon}{\lambda} \\
&\leq \frac{1}{2}\|x\|_2^2 - \frac{1}{2}\langle x, D\alpha\rangle - \frac{1}{2}\langle x, D\tilde{\alpha}\rangle + \frac{1}{4}\|D\alpha\|_2^2 + \frac{1}{4}\langle D\alpha, D\tilde{\alpha}\rangle + \frac{1}{2}\langle x - D\alpha, D\alpha\rangle \\
&\quad + \frac{1}{2}\langle x - \tilde{D}\tilde{\alpha}, \tilde{D}\tilde{\alpha}\rangle + \frac{3}{8}\frac{\varepsilon}{\lambda} \\
&\leq \frac{1}{2}\|x\|_2^2 - \frac{1}{2}\langle x, D\alpha\rangle - \frac{1}{2}\langle x, D\tilde{\alpha}\rangle + \frac{1}{4}\|D\alpha\|_2^2 + \frac{1}{4}\langle D\alpha, D\tilde{\alpha}\rangle + \frac{1}{2}\langle x, D\alpha\rangle - \frac{1}{2}\|D\alpha\|_2^2 \\
&\quad + \frac{1}{2}\langle x, D\tilde{\alpha}\rangle - \frac{1}{2}\|D\alpha\|_2^2 + \left(\frac{3}{8} + \frac{1}{4} + \frac{5}{8}\right)\frac{\varepsilon}{\lambda}
\end{aligned}$$

which simplifies to

$$\frac{1}{2}\|x\|_2^2 - \frac{3}{4}\|D\alpha\|_2^2 - \frac{1}{2}\langle x, D\alpha\rangle - \frac{1}{2}\langle x, D\tilde{\alpha}\rangle + \frac{1}{4}\langle D\alpha, D\tilde{\alpha}\rangle + \frac{1}{2}\langle x, D\alpha\rangle + \frac{1}{2}\langle x, D\tilde{\alpha}\rangle + \frac{5}{4}\frac{\varepsilon}{\lambda}$$
$$= \frac{1}{2}\|x\|_2^2 - \frac{3}{4}\|D\alpha\|_2^2 + \frac{1}{4}\langle D\alpha, D\tilde{\alpha}\rangle + \frac{5}{4}\frac{\varepsilon}{\lambda}.$$

Now, taking the (expanded) LHS of (2.14) and the newly derived upper bound of the RHS of (2.14) yields the inequality:

$$\frac{1}{2}\|x\|_2^2 - \langle x, D\alpha\rangle + \frac{1}{2}\|D\alpha\|_2^2 + \lambda\|\alpha\|_1$$
$$\leq \frac{1}{2}\|x\|_2^2 - \frac{3}{4}\|D\alpha\|_2^2 + \frac{1}{4}\langle D\alpha, D\tilde{\alpha}\rangle + \frac{5}{4}\frac{\varepsilon}{\lambda}.$$

which implies that

$$-\langle x, D\alpha\rangle + \frac{1}{2}\|D\alpha\|_2^2 + \lambda\|\alpha\|_1$$
$$\leq -\frac{3}{4}\|D\alpha\|_2^2 + \frac{1}{4}\langle D\alpha, D\tilde{\alpha}\rangle + \frac{5}{4}\frac{\varepsilon}{\lambda}.$$

Replacing $\lambda\|\alpha\|_1$ with $\langle x - D\alpha, D\alpha\rangle$ yields:

$$-\langle x, D\alpha\rangle + \frac{1}{2}\|D\alpha\|_2^2 + \langle x, D\alpha\rangle - \|D\alpha\|_2^2$$
$$\leq -\frac{3}{4}\|D\alpha\|_2^2 + \frac{1}{4}\langle D\alpha, D\tilde{\alpha}\rangle + \frac{5}{4}\frac{\varepsilon}{\lambda},$$

implying that

$$\frac{1}{4}\|D\alpha\|_2^2 \leq \frac{1}{4}\langle D\alpha, D\tilde{\alpha}\rangle + \frac{5}{4}\frac{\varepsilon}{\lambda}.$$

Hence,

$$\|D\alpha\|_2^2 \leq \langle D\alpha, D\tilde{\alpha}\rangle + \frac{5\varepsilon}{\lambda}.$$

Now, note that

$$\begin{aligned}\|D\alpha - D\tilde{\alpha}\|_2^2 &= \|D\alpha\|_2^2 + \|D\tilde{\alpha}\|_2^2 - 2\langle D\alpha, D\tilde{\alpha}\rangle \\ &\leq \|D\alpha\|_2^2 + \|D\tilde{\alpha}\|_2^2 - 2\|D\alpha\|_2^2 + 10\frac{\varepsilon}{\lambda} \\ &\leq \|D\alpha\|_2^2 + \|D\alpha\|_2^2 - 2\|D\alpha\|_2^2 + 13\frac{\varepsilon}{\lambda} \\ &= 13\frac{\varepsilon}{\lambda}.\end{aligned}$$

□

The sparsity lemma

We now prove that the solution to the perturbed problem is sparse for sufficiently small ε.

Lemma 2.24 (Preservation of Sparsity). *Under Assumptions (A1)-(A3), if*

$$\tau \geq \varepsilon\left(1 + \frac{1}{2\lambda}\right) + \sqrt{\frac{13\varepsilon}{\lambda}},$$

then $\tilde{\alpha}_i = 0$ *for all* $i \in \mathcal{I}$.

Proof. Let $i \in \mathcal{I}$ be arbitrary. To prove that $\tilde{\alpha}_i = 0$, it is sufficient to show that

$$\left|\langle \tilde{D}_i, x - \tilde{D}\tilde{\alpha}\rangle\right| < \lambda,$$

since $\tilde{\alpha}_i$ is hence zero.

First, note that

$$\begin{aligned}\left|\langle \tilde{D}_i, x - \tilde{D}\tilde{\alpha}\rangle\right| &= \left|\langle D_i + \tilde{D}_i - D_i, x - \tilde{D}\tilde{\alpha}\rangle\right| \\ &\leq \left|\langle D_i, x - \tilde{D}\tilde{\alpha}\rangle\right| + \|\tilde{D}_i - D_i\|_2\|x - \tilde{D}\tilde{\alpha}\|_2 \\ &\leq \left|\langle D_i, x - \tilde{D}\tilde{\alpha}\rangle\right| + \varepsilon \qquad (\text{since } \|x\|_2 \leq 1)\end{aligned}$$

and

$$\begin{aligned}\left|\langle D_i, x - \tilde{D}\tilde{\alpha}\rangle\right| &= \left|\langle D_i, x - (D + \tilde{D} - D)\tilde{\alpha}\rangle\right| \\ &\leq |\langle D_i, x - D\tilde{\alpha}\rangle| + \left|\langle D_i, (\tilde{D} - D)\tilde{\alpha}\rangle\right| \\ &\leq |\langle D_i, x - D\tilde{\alpha}\rangle| + \|D_i\|_2\|\tilde{D} - D\|_2\|\tilde{\alpha}\|_2 \\ &\leq |\langle D_i, x - D\tilde{\alpha}\rangle| + \frac{\varepsilon}{2\lambda}.\end{aligned}$$

Hence,

$$\left|\langle \tilde{D}_i, x - \tilde{D}\alpha\rangle\right| \leq |\langle D_i, x - D\tilde{\alpha}\rangle| + \varepsilon\left(1 + \frac{1}{2\lambda}\right),$$

and so it is sufficient to show that

$$|\langle D_i, x - D\tilde{\alpha}\rangle| < \lambda - \varepsilon\left(1 + \frac{1}{2\lambda}\right).$$

Now,

$$\begin{aligned}|\langle D_i, x - D\tilde{\alpha}\rangle| &= |\langle D_i, x - D\tilde{\alpha} + D\alpha - D\alpha\rangle| \\ &\leq |\langle D_i, x - D\alpha\rangle| + |\langle D_i, D\alpha - D\tilde{\alpha}\rangle| \\ &< \lambda - \tau + \|D_i\|_2\|D\alpha - D\tilde{\alpha}\|_2 \\ &< \lambda - \tau + \sqrt{\frac{13\varepsilon}{\lambda}}, \qquad (2.15)\end{aligned}$$

where (2.15) is due to Lemma 2.23. Consequently, it is sufficient if τ is chosen to satisfy

$$\lambda - \tau + \sqrt{\frac{13\varepsilon}{\lambda}} \leq \lambda - \varepsilon\left(1 + \frac{1}{2\lambda}\right),$$

yielding:

$$\tau \geq \varepsilon\left(1 + \frac{1}{2\lambda}\right) + \sqrt{\frac{13\varepsilon}{\lambda}}. \qquad \square$$

Proof of the Sparse Coding Stability Theorem

Proof (of Theorem 2.4). Recall that $\varphi_D(x)$ is the unique optimal solution to the problem

$$\min_{z\in\mathbb{R}^k} \frac{1}{2}\|x - Dz\|_2^2 + \lambda\|z\|_1.$$

If not for the ℓ_1 penalty, in standard form the quadratic program is

$$\min_{z\in\mathbb{R}^k} \frac{1}{2} z^T D^T D z - z^T (D^T x) + \lambda\|z\|_1.$$

Denoting $\bar{z} := \begin{pmatrix} z \\ z^+ \\ z^- \end{pmatrix}$ with $z^+, z^- \in \mathbb{R}^k$, an equivalent formulation is

$$\begin{aligned} \underset{\bar{z}\in\mathbb{R}^{3k}}{\text{minimize}} \quad & Q(\bar{z}) := \frac{1}{2}\bar{z}^T \begin{pmatrix} D^T D & 0_{k\times 2k} \\ 0_{2k\times k} & 0_{2k\times 2k} \end{pmatrix} \bar{z} - \bar{z}^T \left(\begin{pmatrix} D^T \\ 0_{2k\times d} \end{pmatrix} x \right) + \lambda(0_k^T 1_{2k}^T)\bar{z} \\ \text{subject to} \quad & z^+ \geq 0_k \qquad z^- \geq 0_k \qquad z - z^+ + z^- = 0_k. \end{aligned}$$

Similarly, let $\tilde{Q}(\cdot)$ be the objective using $\tilde{D}$ instead of D.

For optimal solutions $\bar{z}_* := \begin{pmatrix} z_* \\ z_*^+ \\ z_*^- \end{pmatrix}$ and $\bar{t}_* := \begin{pmatrix} t_* \\ t_*^+ \\ t_*^- \end{pmatrix}$ of Q and $\tilde{Q}$ respectively, from Daniel (1973), we have

$$(\bar{u} - \bar{z}_*)^T \nabla Q(\bar{z}_*) \geq 0 \tag{2.16}$$

$$(\bar{u} - \bar{t}_*)^T \nabla \tilde{Q}(\bar{t}_*) \geq 0 \tag{2.17}$$

for all feasible $\bar{u} \in \mathbb{R}^{3k}$. Setting $\bar{u}$ to $\bar{t}_*$ in (2.16) and $\bar{u}$ to $\bar{z}_*$ in (2.17) and adding (2.17) and (2.16) yields

$$(\bar{t}_* - \bar{z}_*)^T (\nabla Q(\bar{z}_*) - \nabla \tilde{Q}(\bar{t}_*)) \geq 0,$$

which is equivalent to

$$(\bar{t}_* - \bar{z}_*)^T(\nabla\tilde{Q}(\bar{t}_*) - \nabla\tilde{Q}(\bar{z}_*)) \leq (\bar{t}_* - \bar{z}_*)^T(\nabla Q(\bar{z}_*) - \nabla\tilde{Q}(\bar{z}_*)) \tag{2.18}$$

Here,

$$\nabla Q(z) = \frac{1}{2}\begin{pmatrix} D^TD & 0_{k\times 2k} \\ 0_{2k\times k} & 0_{2k\times 2k} \end{pmatrix} z - \begin{pmatrix} D^T \\ 0_{2k\times d} \end{pmatrix} x + \lambda \begin{pmatrix} 0_k \\ 1_{2k} \end{pmatrix}.$$

After plugging in the expansions of ∇Q and $\nabla\tilde{Q}$ and incurring cancellations from the zeros, (2.18) becomes

$$\begin{aligned}(t_* - z_*)^T\tilde{D}^T\tilde{D}(t_* - z_*) &\leq (t_* - z_*)^T\left((D^TD - \tilde{D}^T\tilde{D})z_* + 2(\tilde{D} - D)^Tx\right) \qquad (2.19)\\ &\leq (t_* - z_*)^T(D^TD - \tilde{D}^T\tilde{D})z_* + 2\|t_* - z_*\|_2\|(\tilde{D} - D)^Tx\|_2 \\ &\leq (t_* - z_*)^T(D^TD - \tilde{D}^T\tilde{D})z_* + \|t_* - z_*\|_2(2\varepsilon) \qquad (2.20)\end{aligned}$$

Let us gain a handle on the first term.

Below, we will use an operator which we dub the *s-restricted 2-norm* (which previously was defined before Lemma 2.19): for a dictionary $A \in (B_{\mathbb{R}^d})^k$, the s-restricted 2-norm of A is defined as $\|A\|_{2,s} := \sup_{\{t\in\mathbb{R}^k:\|t\|=1,|\,\mathrm{supp}(t)|\leq s\}} \|At\|_2$. Now, note that $\tilde{D} = D + E$ for some

E satisfying $\|E\|_2 \leq \varepsilon$. Hence,

$$
\begin{aligned}
&(t_* - z_*)^T (D^T D - \tilde{D}^T \tilde{D}) z_* \\
&= \left| (t_* - z_*)^T (E^T D + D^T E + E^T E) z_* \right| \\
&\leq \left| (t_* - z_*)^T E^T D z_* \right| + \left| (t_* - z_*)^T D^T E z_* \right| + \left| (t_* - z_*)^T E^T E z_* \right| \\
&\leq \|E(t_* - z_*)\|_2 \|D z_*\|_2 + \|D(t_* - z_*)\|_2 \|E z_*\|_2 + \|E(t_* - z_*)\|_2 \|E z_*\|_2 \\
&\leq \|t_* - z_*\|_2 \Big(\|E\|_2 \|D\|_{2,s} \|z_*\|_2 + \|D\|_{2,s} \|E\|_2 \|z_*\|_2 + \|E\|_2^2 \|z_*\|_2 \Big) \\
&\leq \|t_* - z_*\|_2 \left(\frac{\varepsilon\sqrt{s}}{2\lambda} + \frac{\varepsilon\sqrt{s}}{2\lambda} + \frac{\varepsilon^2}{2\lambda} \right) \\
&\leq \|t_* - z_*\|_2 \frac{3}{2} \frac{\varepsilon\sqrt{s}}{\lambda},
\end{aligned}
$$

where the penultimate step follows because

1. if $\|z_*\|_0 \leq s$, then Lemma 2.25 in Section 2.8.4 implies that $\|D z_*\|_2 \leq \sqrt{s}\|z_*\|_2$ (and $\|z_*\|_2 \leq \|z_*\|_1 \leq \frac{1}{2\lambda}$); and
2. Lemma 2.24 implies that $\|t_* - z_*\|_0 \leq s$.

Combining this result with the fact that $\tilde{D}$ has s-incoherence lower bounded by μ implies the desired result:

$$
\|t_* - z_*\|_2 \leq \frac{3}{2} \frac{\varepsilon\sqrt{s}}{\lambda\mu}. \qquad \square
$$

2.8.2 Proof of Symmetrization by Ghost Sample Lemma

Proof (of Lemma 2.7). Replace $\mathcal{F}(\sigma_n)$ from the notation of Mendelson and Philips (2004) with $\mathcal{F}(\mathbf{z}, \mathbf{x}'')$. A modified one-sided version of (Mendelson and Philips, 2004, Lemma 2.2) that uses the more favorable Chebyshev-Cantelli inequality implies that, for every $t > 0$:

$$
\begin{aligned}
\left(1 - \frac{4 \sup_{f \in \mathcal{F}} \mathsf{Var}(\ell(\cdot, f))}{4 \sup_{f \in \mathcal{F}} \mathsf{Var}(\ell(\cdot, f)) + m t^2}\right) &\mathsf{Pr}_{\mathbf{z}\,\mathbf{x}''} \left\{ \exists f \in \mathcal{F}(\mathbf{z}, \mathbf{x}''),\ (\mathsf{P} - \mathsf{P}_{\mathbf{z}}) \ell(\cdot, f) \geq t \right\} \\
&\leq \mathsf{Pr}_{\mathbf{z}\,\mathbf{z}'\mathbf{x}''} \left\{ \exists f \in \mathcal{F}(\mathbf{z}, \mathbf{x}''),\ (\mathsf{P}_{\mathbf{z}'} - \mathsf{P}_{\mathbf{z}}) \ell(\cdot, f) \geq \frac{t}{2} \right\}.
\end{aligned}
$$

As the losses lie in $[0, b]$ by assumption, it follows that $\sup_{f \in \mathcal{F}} \mathsf{Var}(\ell(\cdot, f)) \leq \frac{b^2}{4}$. The lemma follows since the left hand factor of the LHS of the above inequality is at least $\frac{1}{2}$ whenever $m \geq \left(\frac{b}{t}\right)^2$. □

2.8.3 Proofs for overcomplete setting

Proof (of Theorem 2.5). Proposition 2.12 and Lemmas 2.13 and 2.14 imply that

$$\mathsf{Pr}_{\mathbf{z}} \left\{ \begin{array}{l} \exists f \in \mathcal{F}_\mu, \ \left[\mathsf{margin}_s(D, \mathbf{x}) > \iota\right] \\ \qquad \textbf{and} \left((\mathsf{P} - \mathsf{P}_{\mathbf{z}})\ell(\cdot, f) > t\right) \end{array} \right\} \\ \leq 2 \left(\left(\frac{8(r/2)^{1/(d+1)}}{\varepsilon} \right)^{(d+1)k} \exp(-m\varpi^2/(2b^2)) + \delta \right).$$

Equivalently,

$$\mathsf{Pr}_{\mathbf{z}} \left\{ \begin{array}{l} \exists f \in \mathcal{F}_\mu, \ \left[\mathsf{margin}_s(D, \mathbf{x}) > \iota\right] \\ \qquad \textbf{and} \left((\mathsf{P} - \mathsf{P}_{\mathbf{z}})\ell(\cdot, f) > 2\left(\varpi + 2L\beta + \frac{b\eta(m,d,k,\varepsilon,\delta)}{m}\right)\right) \end{array} \right\} \\ \leq 2 \left(\left(\frac{8(r/2)^{1/(d+1)}}{\varepsilon} \right)^{(d+1)k} \exp(-m\varpi^2/(2b^2)) + \delta \right).$$

Now, expand β and η and replace δ with $\delta/4$:

$$\mathsf{Pr}_{\mathbf{z}} \left\{ \begin{array}{l} \exists f \in \mathcal{F}_\mu, \ \left[\mathsf{margin}_s(D, \mathbf{x}) > \iota\right] \textbf{ and} \\ \qquad (\mathsf{P} - \mathsf{P}_{\mathbf{z}})\ell(\cdot, f) > 2\left(\varpi + 2L\varepsilon\frac{1}{2\lambda}\left(1 + \frac{3r\sqrt{s}}{\mu}\right) + \frac{b(dk\log\frac{1944}{\mathsf{margin}_s^2(D,\mathbf{x})\cdot\lambda} + \log(2m+1) + \log\frac{4}{\delta})}{m}\right) \end{array} \right\} \\ \leq 2 \left(\frac{8(r/2)^{1/(d+1)}}{\varepsilon} \right)^{(d+1)k} \exp(-m\varpi^2/(2b^2)) + \frac{\delta}{2}.$$

Choosing $\frac{\delta}{4} = \left(\frac{8(r/2)^{1/(d+1)}}{\varepsilon}\right)^{(d+1)k} \exp(-m\varpi^2/(2b^2))$ yields

$$\mathsf{Pr}_{\mathbf{z}}\left\{\begin{array}{l}\exists f \in \mathcal{F}_\mu,\ \left[\mathsf{margin}_s(D,\mathbf{x}) > \iota\right] \textbf{ and} \\ \qquad (\mathsf{P}-\mathsf{P}_{\mathbf{z}})\ell(\cdot,f) > 2\left(\begin{array}{l}\varpi + L\varepsilon\frac{1}{\lambda}\left(1+\frac{3r\sqrt{s}}{\mu}\right) + \\ \frac{b(dk\log\frac{1944}{\mathsf{margin}_s^2(D,\mathbf{x})\cdot\lambda} + \log(2m+1) + (d+1)k\log\frac{\varepsilon}{8(r/2)^{1/(d+1)}} + \frac{m\varpi^2}{b^2})}{m}\end{array}\right)\end{array}\right\}$$
$$\leq 4\cdot\left(\frac{8(r/2)^{1/(d+1)}}{\varepsilon}\right)^{(d+1)k}\exp(-m\varpi^2/(2b^2)),$$

which is equivalent to

$$\mathsf{Pr}_{\mathbf{z}}\left\{\begin{array}{l}\exists f \in \mathcal{F}_\mu,\ \left[\mathsf{margin}_s(D,\mathbf{x}) > \iota\right] \textbf{ and} \\ \qquad (\mathsf{P}-\mathsf{P}_{\mathbf{z}})\ell(\cdot,f) > 2\left(\begin{array}{l}\varpi + L\varepsilon\frac{1}{\lambda}\left(1+\frac{3r\sqrt{s}}{\mu}\right) + \\ \frac{b(dk\log\frac{1944}{\mathsf{margin}_s^2(D,\mathbf{x})\cdot\lambda} - (d+1)k\log\frac{8}{\varepsilon} + k\log\frac{2}{r} + \log(2m+1) + \frac{m\varpi^2}{b^2})}{m}\end{array}\right)\end{array}\right\}$$
$$\leq 4\cdot\left(\frac{8(r/2)^{1/(d+1)}}{\varepsilon}\right)^{(d+1)k}\exp(-m\varpi^2/(2b^2)),$$

Let δ (a new variable, not related to the previous incarnation of δ) be equal to the upper bound, and solve for ϖ, yielding:

$$\varpi = b\sqrt{\frac{2((d+1)k\log\frac{8}{\varepsilon} + k\log\frac{r}{2} + \log\frac{4}{\delta})}{m}}$$

and hence

$$\mathsf{Pr}_{\mathbf{z}}\left\{\begin{array}{l}\exists f = (D,w) \in \mathcal{F}_\mu,\ \left[\mathsf{margin}_s(D,\mathbf{x}) > \iota\right] \textbf{ and} \\ \qquad (\mathsf{P}-\mathsf{P}_{\mathbf{z}})\ell(\cdot,f) > 2\left(\begin{array}{l}b\sqrt{\frac{2((d+1)k\log\frac{8}{\varepsilon} + k\log\frac{r}{2} + \log\frac{4}{\delta})}{m}} + L\varepsilon\frac{1}{\lambda}\left(1+\frac{3r\sqrt{s}}{\mu}\right) + \\ \frac{b(dk\log\frac{1944}{\mathsf{margin}_s^2(D,\mathbf{x})\cdot\lambda} + \log(2m+1) + \log\frac{4}{\delta})}{m}\end{array}\right)\end{array}\right\}$$
$$\leq \delta,$$

If we set $\varepsilon = \frac{1}{m}$, then provided that $m > \frac{243}{\mathsf{margin}_s^2(D,\mathbf{x})\cdot\lambda}$:

$$\mathsf{Pr}_{\mathbf{z}}\left\{\begin{array}{l}\exists f \in \mathcal{F}_\mu,\ \left[\mathsf{margin}_s(D,\mathbf{x}) > \iota\right] \textbf{ and} \\ \qquad (\mathsf{P}-\mathsf{P}_{\mathbf{z}})\ell(\cdot,f) > 2\left(\begin{array}{l} b\sqrt{\frac{2((d+1)k\log(8m)+k\log\frac{r}{2}+\log\frac{4}{\delta})}{m}} + \frac{L}{m}\left(\frac{1}{\lambda}(1+\frac{3r\sqrt{s}}{\mu})\right) + \\ \frac{b}{m}\left(dk\log\frac{1944}{\mathsf{margin}_s^2(D,\mathbf{x})\cdot\lambda} + \log(2m+1) + \log\frac{4}{\delta}\right)\end{array}\right)\end{array}\right\}$$

$\leq \delta$.

It remains to distribute a prior across the bounds for each choice of s and μ. To each choice of $s \in [k]$ assign prior probability $\frac{1}{k}$. To each choice of $i \in \mathbb{N}\cup\{0\}$ for $2^{-i} \leq \mu$ assign prior probability $(i+1)^{-2}$. For a given choice of $s \in [k]$ and $2^{-i} \leq \mu$ we use $\delta(s,i) := \frac{6}{\pi^2}\frac{1}{(i+1)^2}\frac{1}{k}\delta$ (since $\sum_{i=1}^{\infty}\frac{1}{i^2} = \frac{\pi^2}{6}$). Then, provided that

$$m > \frac{243}{\mathsf{margin}_s(D,\mathbf{x})^2\lambda},$$

the generalization error $(\mathsf{P}-\mathsf{P}_{\mathbf{z}})\ell(\cdot,f)$ is bounded by:

$$2b\sqrt{\frac{2\left((d+1)k\log(8m)+k\log\frac{r}{2}+\log\frac{2\pi^2\left(\log_2\frac{4}{\mu_s(D)}\right)^2 k}{3\delta}\right)}{m}}$$
$$+\frac{2b}{m}\left(dk\log\frac{1944}{\mathsf{margin}_s^2(D,\mathbf{x})\cdot\lambda}+\log(2m+1)+\log\frac{2\pi^2\left(\log_2\frac{4}{\mu_s(D)}\right)^2 k}{3\delta}\right)$$
$$+\frac{2L}{m}\left(\frac{1}{\lambda}(1+\frac{6r\sqrt{s}}{\mu_s(D)})\right). \qquad \square$$

2.8.4 Infinite-dimensional setting

Proof (of Lemma 2.16). Recall that $\eta = \log\frac{1}{\delta}$. Suppose, as in the event being measured, that there is no subset of the ghost sample $\mathbf{x}'$ of size at least η such that the τ-level s-margin condition holds for the entire subset. Equivalently, there is a subset of at least η points in the ghost sample $\mathbf{x}'$ that violate the τ-level s-coding margin condition. From the permutation argument, if no point of $\mathbf{x}''$ violates $\left[\mathsf{margin}_s(D,\cdot) > \tau\right]$, then the probability

that over $\eta = \log \frac{1}{\delta}$ points of $\mathbf{x}'$ will violate $\left[\mathsf{margin}_s(D, \cdot) > \tau\right]$ is at most δ. $\square$

Proof (of Lemma 2.19). By definition, $\varphi_{US}(x) = \arg\min_{z \in \mathbb{R}^k} \frac{1}{2}\|x - USv\|_2 + \lambda\|z\|_1$. Note that $\arg\min_{z \in \mathbb{R}^k} \|x - USz\|_2 = \arg\min_{z \in \mathbb{R}^k} \|U^T x - U^T USz\|_2 = \arg\min_{z \in \mathbb{R}^k} \|U^T x - Sz\|_2$, where the first equality follows because any x in the complement of the image of U will be orthogonal to USz, for any z; hence, it is sufficient to approximate the projection of x onto the range of U. Thus, $\varphi_{US}(x) = \arg\min_{z \in \mathbb{R}^k} \frac{1}{2}\|U^T x - Sz\|_2^2 + \lambda\|z\|_1$. It will be useful to apply a well-known reformulation of this minimization problem as a quadratic program with linear constraints. Denoting $\bar{z} := \bar{z} := (z^T {z^+}^T {z^-}^T)^T$, an equivalent formulation is

$$\underset{\bar{z} \in \mathbb{R}^{3k}}{\text{minimize}} \quad Q_U(\bar{z}) := \frac{1}{2}\bar{z}^T \begin{pmatrix} S^T S & 0_{k \times 2k} \\ 0_{2k \times k} & 0_{2k \times 2k} \end{pmatrix} \bar{z} - \frac{1}{2}\bar{z}^T \left(\begin{pmatrix} 2S^T U^T \\ 0_{2k \times d} \end{pmatrix} x \right) + \lambda(0_k^T \, 1_{2k}^T)\bar{z}$$

$$\text{subject to} \quad z^+ \geq 0_k \qquad z^- \geq 0_k \qquad z - z^+ + z^- = 0_k,$$

For optimal solutions $\bar{z}_* := \begin{pmatrix} z_* \\ z_*^+ \\ z_*^- \end{pmatrix}$ and $\bar{t}_* := \begin{pmatrix} t_* \\ t_*^+ \\ t_*^- \end{pmatrix}$ of Q_U and $Q_{U'}$ respectively, from Daniel (1973), we have

$$(\bar{u} - \bar{z}_*)^T \nabla Q_U(\bar{z}_*) \geq 0 \tag{2.21}$$

$$(\bar{u} - \bar{t}_*)^T \nabla Q_{U'}(\bar{t}_*) \geq 0 \tag{2.22}$$

for all $\bar{u} \in \mathbb{R}^{3k}$. Setting $\bar{u}$ to $\bar{t}_*$ in (2.21) and $\bar{u}$ to $\bar{z}_*$ in (2.22) and adding (2.21) and (2.22) yields

$$(\bar{t}_* - \bar{z}_*)^T (\nabla Q_U(\bar{z}_*) - \nabla Q_{U'}(\bar{t}_*)) \geq 0,$$

which is equivalent to

$$(\bar{t}_* - \bar{z}_*)^T (\nabla Q_{U'}(\bar{t}_*) - \nabla Q_{U'}(\bar{z}_*)) \leq (\bar{t}_* - \bar{z}_*)^T (\nabla Q_U(\bar{z}_*) - \nabla Q_{U'}(\bar{z}_*)). \tag{2.23}$$

Here, $\nabla Q_U(z) = \begin{pmatrix} S^T S & 0_{k\times 2k} \\ 0_{2k\times k} & 0_{2k\times 2k} \end{pmatrix} z - \begin{pmatrix} 2S^T U^T \\ 0_{2k\times d} \end{pmatrix} x + \lambda \begin{pmatrix} 0_k \\ 1_{2k} \end{pmatrix}$. After plugging in the expansions of ∇Q_U and $\nabla Q_{U'}$ and incurring cancellations from the zeros, (2.23) becomes

$$(t_* - z_*)^T (S^T S t_* - 2S^T {U'}^T x - S^T S z_* + 2S^T {U'}^T x) \\ \leq (t_* - z_*)^T (S^T S z_* - 2S^T U^T x - S^T S z_* + 2S^T {U'}^T x),$$

which reduces to

$$(t_* - z_*)^T S^T S (t_* - z_*) \leq 2(t_* - z_*)^T S^T ({U'}^T - U^T) x.$$

Since both t_* and z_* are s-sparse, wherever we typically would consider the operator norm $\|S\|_2 := \sup_{\|t\|=1} \|St\|_2$, we instead need only consider the $2s$-restricted operator norm $\|S\|_{2,2s}$.

Note that $(t_* - z_*)^T S^T S (t_* - z_*) \geq \mu_{2s}(S) \|t_* - z_*\|_2^2$, which implies that

$$\|t_* - z_*\|_2^2 \leq \frac{2}{\mu_{2s}(S)} \|t_* - z_*\| \|S\|_{2,2s} \|({U'}^T - U^T) x\|$$

and hence

$$\|t_* - z_*\|_2 \leq \frac{2\|S\|_{2,2s}}{\mu_{2s}(S)} \|({U'}^T - U^T) x\|_2. \qquad \square$$

Lemma 2.25. *If $S \in (B_{\mathbb{R}^k})^k$, then $\|S\|_{2,s} \leq \sqrt{s}$.*

Proof. Define S_Λ as the submatrix of S that selects the columns indexed by Λ. Similarly,

for $t \in \mathbb{R}^k$ define the coordinate projection t_Λ of t.

$$
\begin{aligned}
&\sup_{\{t:\|t\|=1,|\operatorname{supp}(t)|\le s\}} \|St\|_2 \\
&= \max_{\{\Lambda\subseteq[k]:|\Lambda|\le s\}} \sup_{\{t:\|t\|=1,\operatorname{supp}(t)\subseteq\Lambda\}} \|S_\Lambda t_\Lambda\|_2 \\
&= \max_{\{\Lambda\subseteq[k]:|\Lambda|\le s\}} \sup_{\{t:\|t\|=1,\operatorname{supp}(t)\subseteq\Lambda\}} \left\| \sum_{\omega\in\Lambda} t_\omega S_\omega \right\|_2 \\
&\le \max_{\{\Lambda\subseteq[k]:|\Lambda|\le s\}} \sup_{\{t:\|t\|=1,\operatorname{supp}(t)\subseteq\Lambda\}} \sum_{\omega\in\Lambda} |t_\omega| \|S_\omega\|_2 \\
&\le \max_{\{\Lambda\subseteq[k]:|\Lambda|\le s\}} \sup_{\{t:\|t\|=1,\operatorname{supp}(t)\subseteq\Lambda\}} \sum_{\omega\in\Lambda} |t_\omega| \\
&\le \max_{\{\Lambda\subseteq[k]:|\Lambda|\le s\}} \sup_{\{t:\|t\|=1,\operatorname{supp}(t)\subseteq\Lambda\}} \|t_\Lambda\|_1 \\
&\le \max_{\{\Lambda\subseteq[k]:|\Lambda|\le s\}} \sup_{\{t:\|t\|=1,\operatorname{supp}(t)\subseteq\Lambda\}} \sqrt{s}\|t_\Lambda\|_2 \\
&= \sqrt{s}. \qquad \square
\end{aligned}
$$

2.8.5 Covering numbers

For a Banach space E of dimension d, the ε-covering numbers of the radius r ball of E are bounded as $\mathcal{N}(rB_E, \varepsilon) \le (4r/\varepsilon)^d$ (Cucker and Smale, 2002, Chapter I, Proposition 5).

For spaces of dictionaries obeying some deterministic property, such as

$$\mathcal{D}_\mu = \{D \in \mathcal{D} : \mu_s(D) \ge \mu\},$$

one must be careful to use a *proper* ε-cover so that the representative elements of the cover also obey the desired property; a proper cover is more restricted than a cover in that a proper cover must be a subset of the set being covered, rather than simply being a subset of the ambient Banach space. That is, if A is a proper cover of a subset T of a Banach space E, then $A \subseteq T$. For a cover, we need only $A \subseteq E$. The following bound relates proper covering numbers to covering numbers (a simple proof can be found in Vidyasagar 2002,

Lemma 2.1): If E is a Banach space and $T \subseteq E$ is a bounded subset, then

$$\mathcal{N}(E, \varepsilon, T) \leq \mathcal{N}_{\mathsf{proper}}(E, \varepsilon/2, T).$$

Let $d, k \in \mathbb{N}$. Define $E_\mu := \{E \in (B_{\mathbb{R}^d})^k : \mu_s(D) \geq \mu\}$ and $\mathcal{W} := rB_{\mathbb{R}^d}$. The following bounds derive directly from the above.

Proposition 2.26. *The proper ε-covering number of E_μ is bounded by $(8/\varepsilon)^{dk}$.*

Proposition 2.27. *The product of the proper ε-covering number of E_μ and the ε-covering number of $\mathcal{W}$ is bounded by*

$$\left(\frac{8(r/2)^{1/(d+1)}}{\varepsilon}\right)^{(d+1)k}.$$

CHAPTER 3

MULTI-TASK PREDICTIVE SPARSE CODING

3.1 Introduction

The previous chapter looked at the sample complexity of dictionary learning for sparse coding. The upper bounds presented there are the first and (at this time) only upper bounds on the generalization error of predictive sparse coding. In the overcomplete setting, we saw that the estimation error decays roughly at the rate $O(\sqrt{dk/m})$, where d is the dimension of the input space, k is the dimension of the learned feature space, and m is the size of the training sample. However, in many real-world learning settings, a large amount of data might not available for individual tasks, and consequently, it may be impossible to learn a good predictive sparse coding representation from a single task. In some of these scenarios, it can be beneficial to pool together multiple related tasks, each with limited data, in order to learn a shared representation for all of them.

Predictive sparse coding may be an ideal candidate for multi-task learning for two reasons: the sample complexity of predictive sparse coding may be quite high, and furthermore, predictive sparse coding involves learning a representation that can be shared across tasks. This chapter introduces a multi-task learning formulation of predictive sparse coding, as well as a new model for multi-task sparse coding as an intermediate step.

The form of the new dictionary model distills to each task having its own two-part dictionary, consisting of a shared part that is common to all the dictionaries and a task-specific part that is exclusive to that task. In the extreme where the size of the task-specific subdictionaries goes to zero, all tasks use the same global sparse coding representation.

In the other extreme where the size of the shared subdictionary goes to zero, each task independently learns its own representation, and the multi-task learning problem collapses into independent single-task problems.

Since this multi-task extension takes place at the dictionary level, the resulting multi-task sparse coding model can operate in both the unsupervised and supervised settings and gives rise to multi-task sparse coding and multi-task predictive sparse coding respectively. While the ideas for the unsupervised multi-task sparse coding model existed in some sense in a multi-class classification work of Ramirez et al. (2010), their model never explicitly shares atoms between the different dictionaries, and their multi-class formulation is different from a typical multi-task formulation. There are some previous works that considered the extreme settings of full sharing (no task-specific part) and no sharing (no shared part). Mairal et al. (2009) considered supervised learning of a single shared dictionary in a multi-class setting, using a multi-class loss function. Since they did not reduce the multi-class task to multiple tasks, this model was not a multi-task model. In a later work, Mairal et al. (2012) used predictive sparse coding in a multi-task setting via a reduction of multi-class classification with c classes to c one-vs-all binary classification problems. The difference is that, after the reduction, they did not consider learning a single, shared dictionary nor using multiple dictionaries with partial sharing; instead, they trained a separate dictionary for each class. Notably, Mairal et al. (2012) did consider a semi-supervised learning model, and although we do not analyze the effect of unlabeled data here, this would be a very good direction for future work. Finally, Yu et al. (2009b) considered learning a dictionary that is shared across a set of 10 one-vs-all tasks from the MNIST digits classification problem. Although they did not pursue supervised dictionary learning, the bounds presented in this chapter still apply to their setting in the case of the sparse coding model.

Contributions The first contribution of this chapter is a new multi-task dictionary model that allows for atoms to be shared across the tasks' dictionaries. In addition, we have developed two new models, theoretical results supporting these models, and algorithms for learning. Specific developments include:

1. Multi-task sparse coding and multi-task predictive sparse coding, each of which use

the multi-task dictionary model.

2. For the unsupervised setting of multi-task sparse coding, Theorem 3.1 provides a bound on the average estimation error for dictionary learning that decays with the number of tasks and the number of points per task.

3. For multi-task predictive sparse coding, a suitably adapted multi-task version of the analysis for the single-task case gives rise to the generalization error bound of Theorem 3.2.

4. A stochastic subgradient descent learning algorithm for multi-task predictive sparse coding (see Section 3.5) and empirical support for this algorithm in Section 3.6.

In the next section, we formally introduce the new multi-task dictionary model, multi-task sparse coding, and multi-task predictive sparse coding. Section 3.3 presents the main results, proofs of which can be found in Section 3.4. A learning algorithm for multi-task predictive sparse coding is developed in Section 3.5. Finally, Section 3.6 presents the results of experiments that used this algorithm to compare dictionary models with varying levels of sharing, and we close with a discussion.

3.2 Multi-task predictive sparse coding

3.2.1 Representation

In the new multi-task dictionary model, each task maintains its own dictionary, but certain constraints are placed to enforce similarity between the dictionaries. The first k_s atoms of each task's subdictionary are shared, while the remaining $k_e = k - k_s$ atoms of each task's dictionary are exclusive to that task. More precisely, we denote the dictionary for the t^{th} task via $\bar{D}^{(t)} = \left(D^{(0)}\ D^{(t)}\right)$, where $D^{(0)} \in \mathcal{D}^{(s)}$ and $D^{(t)} \in \mathcal{D}^{(e)}$. Here, $\mathcal{D}^{(s)}$ and $\mathcal{D}^{(e)}$ are the spaces of dictionaries of k_s atoms and k_e atoms respectively all with ℓ_2-norm bounded by 1. From the partial sharing property of dictionaries $\bar{D}^{(1)}, \ldots, \bar{D}^{(T)}$, the constraint $(\bar{D}^{(s)})_j = (\bar{D}^{(t)})_j$ holds for any $s, t \in [T]$ and $1 \leq j \leq k_s$, where T is the number of tasks. Since this model allows atoms between dictionaries to be shared, we sometimes

refer to the model as the *atomic sharing model*.

Some remarks are in order. Intuitively, with regards to the shared subdictionary $D^{(0)}$ the average estimation error should decay with the total number of points Tm (for T tasks and m points per task), and hence the size of the shared subdictionary can grow with the number of tasks. Conversely, with respect to the task-specific subdictionaries $D^{(1)}, \ldots, D^{(T)}$ the average estimation error should decay only with m, the number of points *per task*. Additionally, when $k_e = 0$, the dictionary $\bar{D}^{(t)}$ for each task is equal to $D^{(0)}$; consequently, the average estimation error depends only on the estimation of $D^{(0)}$ and hence should decay with the total number of points Tm.

Using this atomic sharing model with standard (unsupervised) sparse coding gives rise to a flexible multi-task sparse coding model. This model can be further extended to a flexible multi-task predictive sparse coding model. We will proceed by starting with single-task sparse coding, shifting to multi-task sparse coding, and finally deriving multi-task predictive sparse coding.

Sparse coding Recall from (2.1) the sparse auto-encoder $\varphi_D(x) = \arg\min_z \frac{1}{2}\|x - Dz\|_2^2 + \lambda\|z\|_1$, for some dictionary D. Let P be a probability measure on $B_{\mathbb{R}^d}$, recalling that the input space $B_{\mathbb{R}^d}$ is the unit ℓ_2-ball in $\mathbb{R}^d$. In the unsupervised, single-task setting, the goal is to minimize the (regularized) reconstruction error objective

$$\min_{D \in \mathcal{D}} \mathsf{E}_{x \sim \mathsf{P}} \|x - D\varphi_D(x)\|_2^2 + \lambda\|\varphi_D(x)\|_1.$$

It will be useful to define the dictionary-indexed function class $\mathcal{F} := \{f_D : D \in \mathcal{D}\}$, where

$$f_D(x) = \min_{z \in \mathbb{R}^k} \frac{1}{2}\|x - Dz\|_2^2 + \lambda\|z\|_1.$$

With this definition in place, the ideal objective can be rewritten as $\min_{f \in \mathcal{F}} \mathsf{P} f$. In practice, we only have access to a sample $\mathbf{x}$ composed of m points $x_1, \ldots, x_m$ drawn iid from P. The central theoretical question can then be framed as finding a good high probability upper

bound on

$$\sup_{f \in \mathcal{F}} (\mathsf{P} - \mathsf{P}_{\mathbf{x}}) f,$$

where $\mathsf{P}_{\mathbf{x}}$ is the empirical measure with respect to $\mathbf{x}$, acting on a function g of $\mathbb{R}^d$ as $\mathsf{P}_{\mathbf{x}} = \frac{1}{m} \sum_{j=1}^{m} g(x_j)$. Such bounds were obtained by Maurer and Pontil (2008), and in particular they apply to empirical risk minimization (or local minima obtained when attempting to solve the non-convex empirical risk minimization problem).

Multi-task sparse coding We now move to the multi-task sparse coding model. Let $\mathsf{P}_1, \ldots, \mathsf{P}_T$ be probability measures on $B_{\mathbb{R}^d}$. In the multi-task sparse coding model, the ideal objective is

$$\min_{\substack{D^{(0)} \in \mathcal{D}^{(s)} \\ D^{(1)}, \ldots, D^{(T)} \in \mathcal{D}^{(e)}}} \frac{1}{T} \sum_{t=1}^{T} \left(\mathsf{E}_{x \sim \mathsf{P}_t} \|x - D \varphi_{\bar{D}^{(t)}}(x)\|_2^2 + \lambda \|\varphi_{\bar{D}^{(t)}}\|_1 \right). \tag{3.1}$$

By redefining $\mathcal{F}$ as

$$\mathcal{F} := \left\{ (f_{\bar{D}^{(1)}}, \ldots, f_{\bar{D}^{(T)}}) : D^{(0)} \in \mathcal{D}^{(s)}, D^{(1)}, \ldots, D^{(T)} \in \mathcal{D}^{(e)} \right\},$$

with (as before)

$$f_{\bar{D}^{(t)}}(x) := \min_{z \in \mathbb{R}^k} \frac{1}{2} \left\| x - \bar{D}^{(t)} z \right\|_2^2 + \lambda \|z\|_1,$$

the objective (3.1) can be rewritten as

$$\min_{\mathbf{f} \in \mathcal{F}} \frac{1}{T} \sum_{t=1}^{T} \mathsf{P}_t f_t.$$

In the above, the notation $\mathbf{f}$ encapsulates the T component functions as $\mathbf{f} = (f_1, \ldots, f_T)$. This notation will be used from here on out. As before, in practice rather than observing the probability measures $\mathsf{P}_1, \ldots, \mathsf{P}_T$ we instead only observe an empirical sample. In this case, the sample actually is a meta-sample, consisting of T m-samples $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(T)}$, where

$\mathbf{x}^{(t)}$ is the t^{th} task's m-sample containing points $x_1^{(t)}, \ldots, x_m^{(t)}$. To this end, consider the empirical risk minimization surrogate objective

$$\min_{\mathbf{f} \in \mathcal{F}} \frac{1}{T} \sum_{t=1}^{T} \mathsf{P}_{\mathbf{x}^{(t)}} f_t.$$

To characterize our performance when using the surrogate empirical objective, it is of theoretical importance to obtain high probability upper bounds on the uniform (over $\mathcal{F}$) average estimation error

$$\sup_{\mathbf{f} \in \mathcal{F}} \left\{ \frac{1}{T} \sum_{t=1}^{T} \mathsf{P}_t f_t - \frac{1}{T} \sum_{t=1}^{T} \mathsf{P}_{\mathbf{x}^{(t)}} f_t \right\}.$$

We will further analyze this quantity in the next section.

Multi-task predictive sparse coding In multi-task predictive sparse coding, the probability measures $\mathsf{P}_1, \ldots, \mathsf{P}_T$ are instead over $\mathcal{B}_{\mathbb{R}^d} \times \mathcal{Y}$, where $\mathcal{Y}$ is a space of univariate labels. The learner additionally maintains a linear hypothesis in $\mathcal{W} := r\mathcal{B}_{\mathbb{R}^k}$ for each task, with linear hypothesis W_t corresponding to the t^{th} task. The ideal objective is

$$\min_{\substack{D^{(0)} \in \mathcal{D}^{(s)} \\ D^{(1)}, \ldots, D^{(T)} \in \mathcal{D}^{(e)} \\ W_1, \ldots, W_T \in \mathcal{W}}} \frac{1}{T} \sum_{t=1}^{T} \mathsf{E}_{(x,y) \sim \mathsf{P}_t} \ell\left(y, \langle W_t, \varphi_{\bar{D}^{(t)}}(x) \rangle\right). \tag{3.2}$$

Empirically, we observe a meta-sample, consisting of T labeled m-samples $\mathbf{z}^{(1)}, \ldots, \mathbf{z}^{(T)}$, where $\mathbf{z}^{(t)}$ is the t^{th} task's m-sample containing labeled points $z_1^{(t)}, \ldots, z_m^{(t)}$, and any labeled point $z_j^{(t)}$ is equal to $(x_j^{(t)}, y_j^{(t)})$. The (regularized) empirical objective which is optimized in practice, is

$$\min_{\substack{D^{(0)} \in \mathcal{D}^{(s)} \\ D^{(1)}, \ldots, D^{(T)} \in \mathcal{D}^{(e)} \\ W_1, \ldots, W_T \in \mathcal{W}}} \frac{1}{T} \sum_{t=1}^{T} \left(\frac{1}{m} \sum_{j=1}^{m} \ell\left(y_j^{(t)}, \left\langle W_t, \varphi_{\bar{D}^{(t)}}(x_j^{(t)}) \right\rangle\right) + \frac{1}{r} \|W_t\|_2^2 \right). \tag{3.3}$$

It is useful to rewrite (3.3) as a global optimization over the shared representation parameter $D^{(0)}$ and T independent optimizations (conditional on $D^{(0)}$) over the task-specific

parameters $(D^{(1)}, w^{(1)}), \ldots, (D^{(T)}, w^{(T)})$:

$$\min_{D^{(0)} \in \mathcal{D}^{(s)}} \frac{1}{T} \sum_{t=1}^{T} \min_{\substack{D^{(t)} \in \mathcal{D}^{(e)} \\ W_t \in \mathcal{W}}} \frac{1}{m} \sum_{j=1}^{m} \ell\left(y_j^{(t)}, \left\langle W_t, \varphi_{\bar{D}^{(t)}}(x_j^{(t)}) \right\rangle\right) + \frac{1}{r}\|W_t\|_2^2.$$

As before, we frame the ideal objective and (unregularized) empirical objective in terms of a function class $\mathcal{F}$, now redefined as

$$\mathcal{F} := \left\{ (f_{\bar{D}^{(1)}, W_1}, \ldots, f_{\bar{D}^{(T)}, W_T}) : D^{(0)} \in \mathcal{D}^{(s)}, D^{(1)}, \ldots, D^{(T)} \in \mathcal{D}^{(e)}, W_1, \ldots, W_T \in \mathcal{W} \right\}$$

with

$$f_{D,w}(x, y) := \ell\left(y, \langle w, \varphi_D(x) \rangle\right).$$

The main theoretical question then is to obtain a high probability upper bound on

$$\frac{1}{T} \sum_{t=1}^{T} \mathsf{P}_t \, f_t - \frac{1}{T} \sum_{t=1}^{T} \mathsf{P}_{\mathbf{z}^{(t)}} \, f_t, \tag{3.4}$$

that holds for all $\mathbf{f} = (f_1, \ldots, f_T) \in \mathcal{F}$ simultaneously. As compared to the unsupervised setting, we will entertain bounds that can depend on the particular hypothesis $\mathbf{f}$, rather than obtaining uniform upper confidence bounds on (3.4). This difference is owed to our reuse of the data-and-hypothesis dependent analysis for single-task predictive sparse coding, which appears to be much more difficult to analyze than (unsupervised) single-task sparse coding. We analyze (3.4) in the next section.

3.3 Generalization error bounds

This section contains the statement of the two main learning bounds for the unsupervised and supervised settings respectively. We first present the bound for (unsupervised) multi-task sparse coding.

Theorem 3.1 (Unsupervised Learning Bound). *Let $\mathcal{F}$ be the function class*

$$\mathcal{F} := \left\{ (f_{\bar{D}^{(1)}}, \ldots, f_{\bar{D}^{(T)}}) : D^{(0)} \in \mathcal{D}^{(s)}, D^{(1)}, \ldots, D^{(T)} \in \mathcal{D}^{(e)} \right\}$$

with $f_{\bar{D}^{(t)}}(x) := \min_{z \in \mathbb{R}^k} \frac{1}{2}\|x - \bar{D}^{(t)} z\|_2^2 + \lambda \|z\|_1$. *Suppose* $\mathsf{P}_1, \ldots, \mathsf{P}_T$ *are probability measures on* $\mathcal{B}_{\mathbb{R}^d}$ *and for each* $t \in [T]$, $\mathbf{x}^{(t)}$ *is an* m*-sample composed of* $x_1^{(t)}, \ldots, x_m^{(t)}$ *drawn iid from* P_t. *Then with probability at least* $1 - \delta$, *all* $f \in \mathcal{F}$ *satisfy*

$$\frac{1}{T}\sum_{t=1}^{T} \mathsf{P}_t f_t - \frac{1}{T}\sum_{t=1}^{T} \mathsf{P}_{\mathbf{x}^{(t)}} f_t \leq b\sqrt{\left(\frac{dk_s}{Tm} + \frac{dk_e}{m}\right)\log(8Tm) + \frac{\log\frac{1}{\delta}}{Tm}} + \frac{4}{\lambda}\frac{1}{Tm}.$$

In the case when all the tasks use the same dictionary ($k = k_s$, $k_e = 0$), the estimation error decays at the rate $\sqrt{\frac{dk}{Tm}}$ (ignoring $\log$ terms). Hence, the number of points per task m can be much less than dk provided that the number of tasks is sufficiently large. Of course, it only makes sense to set $k_e = 0$ if there exists a single dictionary which provides a good sparse code representation for all of the tasks; if this is not the case, then the average (regularized) empirical risk $\frac{1}{T}\mathsf{P}_{\mathbf{x}^{(t)}} f_t$ will not be small.

When might it be reasonable to select $k_e > 0$? Suppose that m is much larger than d. Then for k_e small, $\frac{dk_e}{m}$ should be very small. In addition, our bound on the estimation error is dominated by $\frac{dk_e}{m}$ until the number of shared atoms k_s exceeds roughly Tk_e. As the number of tasks in multi-task learning often is very large, it therefore is reasonable to select k_s to be very large whenever k_e is non-zero.

For multi-task predictive sparse coding, we have the following learning bound.

Theorem 3.2. *With probability at least* $1 - \delta$ *over* $\mathbf{z}^{(1)} \sim (\mathsf{P}_1)^m, \ldots, \mathbf{z}^{(T)} \sim (\mathsf{P}_T)^m$, *for any* $s \in [k]$ *and any* $f = (D^{(0)}, D^{(1)}, \ldots, D^{(T)}, w^{(1)}, \ldots, w^{(T)}) \in \mathcal{F}$ *satisfying (for all* $t \in [T]$*)* $s\text{-sparse}(\varphi_{\bar{D}^{(t)}}(\mathbf{x}^{(t)}))$ *and* $Tm > \frac{243}{\min_{t\in[T]} \text{margin}_s(\bar{D}^{(t)}, \mathbf{x}^{(t)})^2 \lambda}$, *the generalization error*

$\sum_{t=1}^{T} \mathsf{P}_t \, \ell(\cdot, f_t) - \sum_{t=1}^{T} \mathsf{P}_{\mathbf{z}^{(t)}} \, \ell(\cdot, f_t)$ *is bounded by:*

$$2\sqrt{2}b\sqrt{\frac{k_s d \log(16Tm)}{Tm} + \frac{(k_s + (d+1)k_e)\log(16Tm) + (k_s + k_e)\log\frac{r}{4} + \log\frac{2\pi^2\left(\log_2 \frac{4}{\mu_s(\bar{D})}\right)^2 k}{3\delta}}{m}}$$
$$+ 2b\left(\frac{d(\frac{k_s}{T} + k_e)\log\frac{3888}{\min_{t\in[T]} \mathrm{margin}_s^2(\bar{D}^{(t)}, \mathbf{x}^{(t)})\cdot\lambda}}{m} + \frac{\log(2Tm+1) + \log\frac{2\pi^2\left(\log_2 \frac{4}{\mu_s(\bar{D})}\right)^2 k}{3\delta}}{Tm}\right)$$
$$+ \frac{2L}{Tm}\left(\frac{1}{\lambda}(1 + \frac{6r\sqrt{s}}{\mu_s(D)})\right).$$

Although daunting, the important thing to note is that the estimation error due to dictionary learning can be controlled similar to the unsupervised multi-task sparse coding setting, with qualifications on the incoherence and s-margin properties of the learned dictionaries that were anticipated from the single-task predictive sparse coding learning bound Theorem 2.5. Unlike the bound for unsupervised multi-task sparse coding, even when $k_e = 0$ it no longer is sufficient for $\frac{dk_s}{Tm}$ to be small in order to obtain a useful bound; since multi-task predictive sparse coding involves learning a linear hypothesis W_t for each task t, we roughly need $\frac{k}{m}$ to be small as well. However, this is still a substantial improvement upon single-task predictive sparse coding, wherein we roughly needed $\frac{dk}{m}$ to be small to obtain a useful bound.

3.4 Proofs for generalization error bounds

This section contains the narrative for proving the main results, including all technical proofs. We first handle Theorem 3.1, the result for the unsupervised setting, and then prove the predictive setting result of Theorem 3.2.

3.4.1 Unsupervised setting: proof of Theorem 3.1

For the generalization error bound in the unsupervised setting, the goal is to control

$$\sup_{f\in\mathcal{F}} \frac{1}{T}\sum_{t=1}^{T} \mathsf{P}_t \, f_t - \frac{1}{T}\sum_{t=1}^{T} \mathsf{P}_{\mathbf{x}^{(t)}} \, f_t.$$

Recall that in the unsupervised sparse coding setting,

$$\mathcal{F} := \left\{ (f_{\bar{D}^{(1)}}, \ldots, f_{\bar{D}^{(T)}}) : D^{(0)} \in \mathcal{D}^{(s)}, D^{(1)}, \ldots, D^{(T)} \in \mathcal{D}^{(e)} \right\}$$

with

$$f_{\bar{D}^{(t)}}(x) := \min_{z \in \mathbb{R}^k} \frac{1}{2}\|x - \bar{D}^{(t)} z\|_2^2 + \lambda \|z\|_1.$$

Our high-level strategy will involve constructing a finite ε-net for $\mathcal{F}$ and then using large deviation bounds for each element of this ε-net. To advance toward this goal, we first study the behavior of some function element f_D, evaluated at an arbitrary point x, under perturbations to the dictionary D.

Let $D, D' \in \mathcal{D}$ be dictionaries satisfying $\|D - D'\|_2 \leq \varepsilon$. One way to understand how some element f_D behaves under perturbations is to bound the size of $\sup_{x \in B_{\mathbb{R}^d}} |f_D(x) - f_{D'}(x)|$. For an arbitrary $x \in B_{\mathbb{R}^d}$, let z^* be a minimizer of $f_D(x)$ and let z'^* be a minimizer of $f_{D'}(x)$. Observe that

$$\begin{aligned}
&f_D(x) - f_{D'}(x) \\
&= \left(\frac{1}{2}\|x - Dz^*\|_2^2 + \lambda \|z^*\|_1 \right) - \left(\frac{1}{2}\|x - D'z'^*\|_2^2 + \lambda \|z'^*\|_1 \right) \\
&\leq \left(\frac{1}{2}\|x - Dz'^*\|_2^2 + \lambda \|z'^*\|_1 \right) - \left(\frac{1}{2}\|x - D'z'^*\|_2^2 + \lambda \|z'^*\|_1 \right) \\
&= \frac{1}{2}\left(\|x - Dz'^*\|_2^2 - \|x - D'z'^*\|_2^2 \right).
\end{aligned}$$

Let $D' = D + E$. Some simple linear algebra yields

$$\begin{aligned}
&\|x - Dz'^*\|_2^2 - \|x - D'z'^*\|_2^2 \\
&= \|x\|_2^2 - 2\langle x, Dz'^*\rangle + \|Dz'^*\|_2^2 - \|x\|_2^2 + 2\langle x, D'z'^*\rangle - \|D'z'^*\|_2^2 \\
&= 2\langle x, (D' - D)z'^*\rangle + \|Dz'^*\|_2^2 - \|D'z'^*\|_2^2 \\
&= 2\langle x, Ez'^*\rangle + \|(D' - E)z'^*\|_2^2 - \|D'z'^*\|_2^2 \\
&\leq 2\langle x, Ez'^*\rangle + \|D'z'^*\|_2^2 + 2\|D'z'^*\|_2\|Ez'^*\|_2 + \|Ez'^*\|_2^2 - \|D'z'^*\|_2^2 \\
&= 2\langle x, Ez'^*\rangle + 2\|D'z'^*\|_2\|Ez'^*\|_2 + \|Ez'^*\|_2^2 \\
&\leq \frac{\varepsilon}{\lambda} + \frac{2\varepsilon}{\lambda} + \left(\frac{\varepsilon}{2\lambda}\right)^2 \\
&\leq \frac{13}{4}\frac{\varepsilon}{\lambda},
\end{aligned}$$

where the penultimate inequality follows since $\|D'z'^*\|_2 \leq 2$ by the triangle inequality (using $\|x\|_2 \leq 1$ and $\|x - D'z'^*\|_2 \leq 1$, the latter of which follows from the optimality of z'^* for $f_{D'}(x)$). Consequently,

$$f_D(x) - f_{D'}(x) \leq \frac{13}{8}\frac{\varepsilon}{\lambda} \leq \frac{2\varepsilon}{\lambda}.$$

A symmetric argument works for bounding $f_{D'}(x) - f_D(x)$.

We have just proved the following lemma.

Lemma 3.3. *Let $D, D' \in \mathcal{D}$ be dictionaries satisfying $\|D - D'\|_2 \leq \varepsilon$. Then*

$$\sup_{x \in B_{\mathbb{R}^d}} |f_D(x) - f_{D'}(x)| \leq \frac{2\varepsilon}{\lambda}.$$

This lemma establishes that if two dictionaries are close, then their sparse coding optimal objective values are uniformly close (for all x in the ℓ_2-ball of $\mathbb{R}^d$). Recall that we are seeking an upper confidence bound on the deviation $\frac{1}{T}\sum_{t=1}^{T} \mathsf{P}_t\, f_t - \frac{1}{T}\sum_{t=1}^{T} \mathsf{P}_{\mathbf{x}^{(t)}}$ that holds uniformly over $\mathcal{F}$. If we can approximate $\mathcal{F}$ with a finite approximating set (an ε-net), then such bounds readily follow from a simple application of the union bound and Hoeffding's inequality. Our strategy therefore will be to construct such an ε-net.

To this end, observe that every $f \in \mathcal{F}$ is fully specified by a choice $(D^{(0)}, D^{(1)}, \ldots, D^{(T)})$ in $\mathcal{D}^{(s)} \times \left(\mathcal{D}^{(e)}\right)^T$. If we can find a finite approximating set for $\mathcal{D}^{(s)} \times \left(\mathcal{D}^{(e)}\right)^T$ such that the dictionary for each task (e.g. $\bar{D}^{(t)}$ for the t^{th} task) has a close representative in the approximating set, then Lemma 3.3 implies that this finite approximating set induces a finite approximating set for $\mathcal{F}$ as well. It will be convenient to endow the space $\mathcal{D}^{(s)} \times \left(\mathcal{D}^{(e)}\right)^T$ with the metric

$$\Delta\left((D^{(0)}, D^{(1)}, \ldots, D^{(T)}), (D'^{(0)}, D'^{(1)}, \ldots, D'^{(T)})\right) := \max_{t \in [t]} \|\bar{D}^{(t)} - \bar{D}'^{(t)}\|_2, \tag{3.5}$$

where $\bar{D}'^{(t)}$ is the dictionary $(D'^{(0)} D'^{(t)})$, for $t \in [T]$. The advantage of this metric is that if a finite set is a good approximation for $\mathcal{D}^{(s)} \times \left(\mathcal{D}^{(e)}\right)^T$ in this metric, then the finite set gives rise to a good approximation for every task's dictionary.

Constructing a good finite approximating set turns out to be easy, as shown by the following lemma.

Lemma 3.4. *If $\mathcal{D}_\varepsilon^{(s)}$ is an ε_1-net for $\mathcal{D}^{(s)}$, and if $\mathcal{D}_\varepsilon^{(e)}$ is an ε_2-net for $\mathcal{D}^{(e)}$, then $\mathcal{D}_\varepsilon^{(s)} \times \left(\mathcal{D}_\varepsilon^{(e)}\right)^T$ is an $(\varepsilon_1 + \varepsilon_2)$-net for $\mathcal{D}^{(s)} \times \left(\mathcal{D}^{(e)}\right)^T$ in the metric (3.5).*

Proof. Below, we denote the coordinate projection of z onto the first k_s coordinates as $z^{(0)}$ and the coordinate projection onto the remaining k_e coordinates as $z^{(1)}$.
Let $(D^{(0)}, D^{(1)}, \ldots, D^{(T)})$ and $(D'^{(0)}, D'^{(1)}, \ldots, D'^{(T)})$ be arbitrary elements of $\mathcal{D}^{(s)} \times \left(\mathcal{D}^{(e)}\right)^T$. Now, observe that for any $t \in [T]$:

$$\begin{aligned}
&\sup_{\|z\|_2=1} \left\| \left((D^{(0)}\ D^{(t)}) - (D'^{(0)}\ D'^{(t)})\right) z \right\|_2 \\
&= \sup_{\|z\|_2=1} \left\| (D^{(0)} - D'^{(0)}) z^{(0)} + (D^{(t)} - D'^{(t)}) z^{(1)} \right\|_2 \\
&\leq \sup_{\|z\|_2=1} \left\| (D^{(0)} - D'^{(0)}) z^{(0)} \right\|_2 + \left\| (D^{(t)} - D'^{(t)}) z^{(1)} \right\|_2 \\
&\leq \sup_{\|z^{(0)}\|_2=1} \left\| (D^{(0)} - D'^{(0)}) z^{(0)} \right\|_2 + \sup_{\|z^{(1)}\|_2=1} \left\| (D^{(t)} - D'^{(t)}) z^{(1)} \right\|_2 \\
&= \|D^{(0)} - D'^{(0)})\|_2 + \|D^{(t)} - D'^{(t)})\|_2.
\end{aligned}$$

Consequently,

$$\max_{t\in[T]} \|\bar{D}^{(t)} - \bar{D}'^{(t)}\|_2 \leq \|D^{(0)} - D'^{(0)}\|_2 + \max_{t\in[T]} \|D^{(t)} - D'^{(t)}\|_2.$$

Now, let $(D^{(0)}, D^{(1)}, \ldots, D^{(T)})$ be an arbitrary element of the space $\mathcal{D}^{(s)} \times \left(\mathcal{D}^{(e)}\right)^T$. By definition, there exists a choice $(D'^{(0)}, D'^{(1)}, \ldots, D'^{(T)})$ in the approximating set $\mathcal{D}_\varepsilon^{(s)} \times \left(\mathcal{D}_\varepsilon^{(e)}\right)^T$ satisfying $\|D^{(0)} - D'^{(0)}\| \leq \varepsilon_1$. and $\|D^{(t)} - D'^{(t)}\| \leq \varepsilon_2$ for $t \in [T]$. This implies that $\max_{t\in[T]} \|\bar{D}^{(t)} - \bar{D}'^{(t)}\|_2 \leq \varepsilon_1 + \varepsilon_2$. □

We now bound the cardinality of the approximating set $\mathcal{D}_\varepsilon^{(s)} \times \left(\mathcal{D}_\varepsilon^{(e)}\right)^T$, after which nearly everything will be in place for the final result. We can use Proposition 2.26 (with a factor 4 rather than 8, since proper ε-covering numbers are not required here) to bound the cardinality of this set by

$$\inf_{\substack{\varepsilon_1,\varepsilon_2>0 \\ \varepsilon_1+\varepsilon_2=\varepsilon}} \left(\frac{4}{\varepsilon_1}\right)^{dk_s} \left(\frac{4}{\varepsilon_2}\right)^{Tdk_e}.$$

Selecting $\varepsilon_1 = \varepsilon_2 = \frac{\varepsilon}{2}$ yields an ε-net $\mathcal{D}_\varepsilon^{(s)} \times \left(\mathcal{D}_\varepsilon^{(e)}\right)^T$ of cardinality at most

$$\left(\frac{8}{\varepsilon}\right)^{d(k_s+Tk_e)}.$$

Recall that the goal is to obtain a high probability bound on

$$\frac{1}{T}\sum_{t=1}^{T} \mathsf{P}_t f_t - \frac{1}{T}\sum_{t=1}^{T} \mathsf{P}_{\mathbf{x}^{(t)}} f_t.$$

We now have sufficient results to do this.

Proof (of Theorem 3.1). Let $\varepsilon' = \frac{2\varepsilon}{\lambda}$. From the ε'-net $\mathcal{F}_\varepsilon$ induced from $\mathcal{D}_\varepsilon^{(s)} \times \left(\mathcal{D}_\varepsilon^{(e)}\right)^T$, we

have:

$$\Pr\left\{\exists f \in \mathcal{F} : \frac{1}{T}\sum_{t=1}^{T} \mathsf{P}_t\, f_t - \frac{1}{T}\sum_{t=1}^{T} \mathsf{P}_{\mathbf{x}^{(t)}}\, f_t \geq \alpha\right\} \\ \leq \Pr\left\{\exists f \in \mathcal{F}_\varepsilon : \frac{1}{T}\sum_{t=1}^{T} \mathsf{P}_t\, f_t - \frac{1}{T}\sum_{t=1}^{T} \mathsf{P}_{\mathbf{x}^{(t)}}\, f_t \geq \alpha - 2\varepsilon'\right\},$$

which for arbitrary $f \in \mathcal{F}_\varepsilon$, is not greater than

$$|\mathcal{F}_\varepsilon| \Pr\left\{\frac{1}{T}\sum_{t=1}^{T} \mathsf{P}_t\, f_t - \frac{1}{T}\sum_{t=1}^{T} \mathsf{P}_{\mathbf{x}^{(t)}}\, f_t \geq \alpha - 2\varepsilon'\right\}.$$

Expanding this last expression, we see that it is just

$$\Pr\left\{\frac{1}{T}\sum_{t=1}^{T} \mathsf{P}_t\, f_t - \frac{1}{T}\sum_{t=1}^{T} \mathsf{P}_{\mathbf{x}^{(t)}}\, f_t \geq \alpha - 2\varepsilon'\right\} \\ = \Pr\left\{\frac{1}{T}\sum_{t=1}^{T} \mathsf{P}_t\, f_t - \frac{1}{T}\sum_{t=1}^{T}\frac{1}{m}\sum_{j=1}^{m} f_t(x_j^{(t)}) \geq \alpha - 2\varepsilon'\right\}.$$

Now, observe that

$$\begin{aligned}\mathsf{E}\left(\frac{1}{T}\sum_{t=1}^{T}\frac{1}{m}\sum_{j=1}^{m} f_t(x_j^{(t)})\right) &= \mathsf{P}_1 \ldots \mathsf{P}_T\, \frac{1}{T}\sum_{t=1}^{T}\frac{1}{m}\sum_{j=1}^{m} f_t(x_j^{(t)}) \\ &= \frac{1}{T}\sum_{t=1}^{T} \mathsf{P}_t\, \frac{1}{m}\sum_{j=1}^{m} f_t(x_j^{(t)}) \\ &= \frac{1}{T}\sum_{t=1}^{T} \mathsf{P}_t\, f_t.\end{aligned}$$

This observation, combined with the fact that for any f, t, j we have $f(x_j^{(t)}) \in [0, b]$ surely (by assumption), unlocks an application of Hoeffding's inequality:

$$\Pr\left\{\frac{1}{T}\sum_{t=1}^{T} \mathsf{P}_t\, f_t - \frac{1}{T}\sum_{t=1}^{T}\frac{1}{m}\sum_{j=1}^{m} f_t(x_j^{(t)}) \geq \beta\right\} \leq \exp\left(-\frac{2Tm\beta^2}{b^2}\right),$$

for $\beta := \alpha - 2\varepsilon'$.

Gathering the above results, it follows that

$$\begin{aligned}\Pr\left\{\exists f \in \mathcal{F} : \frac{1}{T}\sum_{t=1}^{T} \mathrm{P}_t f_t - \frac{1}{T}\sum_{t=1}^{T} \mathrm{P}_{\mathbf{x}^{(t)}} f_t \geq \alpha\right\} &\leq |\mathcal{F}_\varepsilon| \exp\left(-\frac{2Tm\beta^2}{b^2}\right) \\ &\leq \left(\frac{8}{\varepsilon}\right)^{d(k_s+Tk_e)} \exp\left(-\frac{2Tm\beta^2}{b^2}\right).\end{aligned}$$

Setting δ equal to the last line above and solving for β yields:

$$\beta = b\sqrt{\left(\frac{dk_s}{Tm} + \frac{dk_e}{m}\right)\log\frac{8}{\varepsilon} + \frac{\log\frac{1}{\delta}}{Tm}}.$$

The result follows after substituting $\alpha = \beta + \frac{4\varepsilon}{\lambda}$ and selecting $\varepsilon = \frac{1}{Tm}$. □

3.4.2 Predictive setting: proof of Theorem 3.2

As in the unsupervised setting, we again seek to control

$$\frac{1}{T}\sum_{t=1}^{T} \mathrm{P}_t f_t - \frac{1}{T}\sum_{t=1}^{T} \mathrm{P}_{\mathbf{z}^{(t)}} f_t.$$

However, in the predictive sparse coding setting, the function class $\mathcal{F}$ is defined as

$$\mathcal{F} := \left\{(f_{\bar{D}^{(1)},W_1}, \ldots, f_{\bar{D}^{(T)},W_T} : D^{(0)} \in \mathcal{D}^{(s)}, D^{(1)}, \ldots, D^{(T)} \in \mathcal{D}^{(e)}, W_1, \ldots, W_T \in \mathcal{W}\right\}$$

with

$$f_{D,w}(x,y) := \ell\left(y, \langle w, \varphi_D(x)\rangle\right).$$

At a high level, the proof strategy for the multi-task setting of predictive sparse coding mimics the strategy for single-task predictive sparse coding (which was presented in Chapter 2). In the single-task setting (where $T = 1$), the key steps were:

1. For hypotheses where the dictionary has large s-margin, showing that the probability of the true risk exceeding the empirical risk by a large amount is not much greater than the probability of a large deviation between the empirical risks on two independent

m-samples. This was handled by Proposition 2.12.

2. Applying the Good Ghost Lemma (Lemma 2.13) in order to guarantee that the learned auto-encoder has large s-margin on a second iid sample.

3. Applying Lemma 2.14 to bound the probability of a large deviation between the risks on two independent m-samples.

The steps from the single-task setting straightforwardly carry forward with significant but relatively easy modifications throughout. We will present multi-task versions of each of the above three results before concluding with a proof of Theorem 3.2

Notation The exposition of our proof will assisted by some notation. It will be convenient to collect the random variables $\mathbf{z}^{(1)}, \ldots, \mathbf{z}^{(T)}$ into a meta-sample $\underline{\mathbf{z}}$. Similarly, in the below ghost samples $\mathbf{z}'^{(1)}, \ldots, \mathbf{z}'^{(T)}$ are collected into a ghost meta-sample $\underline{\mathbf{z}}'$.

Let $\underline{\tilde{\mathbf{x}}} \subseteq_\eta \underline{\mathbf{x}}$ indicate that $\tilde{\mathbf{x}}$ is a meta-sample collecting samples $\tilde{\mathbf{x}}^{(1)}, \ldots, \tilde{\mathbf{x}}^{(T)}$ obeying the following constraints:

1. For $t \in [T]$, the sample $\tilde{\mathbf{x}}^{(t)}$ is a subset of sample $\mathbf{x}^{(t)}$.

2. For $t \in [T]$, let c_t be the number of elements of $\mathbf{x}^{(t)}$ that are not in $\tilde{\mathbf{x}}^{(t)}$. Then $\sum_{t=1}^{T} c_t \leq \eta$. That is, cumulatively over the points in $\tilde{\mathbf{x}}^{(1)}, \ldots, \tilde{\mathbf{x}}^{(T)}$, at most η elements of the points in $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(T)}$ have been removed.

Finally, define $\mathcal{E}_\mu := \{(D^{(0)}, D^{(1)}, \ldots, D^{(T)}) \in \mathcal{D}^{(s)} \times \left(\mathcal{D}^{(e)}\right)^T : \mu_s(\bar{D}^{(t)}) \geq \mu, t \in [T]\}$, which is the subset of $\mathcal{D}^{(s)} \times \left(\mathcal{D}^{(e)}\right)^T$ for which all task dictionaries are μ-incoherent (for some fixed s).

Proof exposition The first result is a straightforward multi-task extension of Lemma 2.2 of Mendelson and Philips (2004), which applies symmetrization by a ghost sample for random subclasses.

Lemma 3.5. *If* $\alpha^2 \geq \frac{2b^2}{Tm}$, *then*

$$\mathsf{Pr}\left\{\exists \mathbf{f} \in \mathcal{F}(\underline{\mathbf{z}}), \frac{1}{T}\sum_{t=1}^{T} \mathsf{P}_t\, f_t - \frac{1}{T}\sum_{t=1}^{T} \mathsf{P}_{\mathbf{z}^{(t)}}\, f_t \geq \alpha\right\} \leq 2\,\mathsf{Pr}\left\{\exists \mathbf{f} \in \mathcal{F}(\underline{\mathbf{z}}), \frac{1}{T}\sum_{t=1}^{T} \mathsf{P}_{\mathbf{z}^{(t)}}\, f_t - \frac{1}{T}\sum_{t=1}^{T} \mathsf{P}_{\mathbf{z}'^{(t)}}\, f_t \geq \alpha/2\right\}.$$

Proof. Define two stochastic processes U_i and W_i via:

$$U_j(\mathbf{f}) = \frac{1}{T}\sum_{t=1}^{T} f_t(z_j^{(t)}) - \mathsf{P}_t\, f_t(z_j^{(t)}) \qquad W_j(\mathbf{f}) = \frac{1}{T}\sum_{t=1}^{T} f_t(z_j'^{(t)}) - \mathsf{P}_t\, f_t(z_j'^{(t)}).$$

Observe from the triangle inequality that $\frac{1}{m}\sum_{j=1}^{m} U_j(\mathbf{f}) \geq \alpha$ and $\frac{1}{m}\sum_{j=1}^{m} W_j(\mathbf{f}) \leq \alpha/2$ imply that $\frac{1}{m}\sum_{j=1}^{m}(U_j(\mathbf{f}) - W_j(\mathbf{f})) \geq \alpha/2$.

Now, let A be the set

$$A = \{\underline{\mathbf{z}} : \exists \mathbf{f} \in \mathcal{F}(\underline{\mathbf{z}}), \sum_{j=1}^{m} U_j(\mathbf{f}) \geq t\}.$$

Observe that for every element of A, there exists a $\mathbf{f} \in \mathcal{F}(\underline{\mathbf{z}})$ as well as a realization of U such that $\frac{1}{m}\sum_{j=1}^{m} U_j(f) \geq \alpha$. Hence, from the triangle inequality, for this $\mathbf{f}$ and U it follows that

$$\mathsf{Pr}\left\{\underline{\mathbf{z}}' : \frac{1}{m}\sum_{j=1}^{m} W_j(\mathbf{f}) \leq \alpha/2\right\} \leq \mathsf{Pr}\left\{\underline{\mathbf{z}}' : \frac{1}{m}\sum_{j=1}^{m}(U_j(\mathbf{f}) - W_j(\mathbf{f})) \geq \alpha/2\right\}.$$

The LHS is lower bounded by taking the infimum of the probability, with respect to $\mathbf{f}$, and likewise the RHS is upper bounded by taking the supremum (with respect to $\mathbf{f} \in \mathcal{F}(\underline{\mathbf{z}})$):

$$\inf_{\mathbf{f}\in\mathcal{F}} \mathsf{Pr}\left\{\frac{1}{m}\sum_{j=1}^{m} W_j(\mathbf{f}) \leq \alpha/2\right\} \leq \mathsf{Pr}\left\{\underline{\mathbf{z}}' : \exists \mathbf{f} \in \mathcal{F}(\underline{\mathbf{z}}), \frac{1}{m}\sum_{j=1}^{m}(U_j(\mathbf{f}) - W_j(\mathbf{f})) \geq \alpha/2\right\}.$$

The inequality does not depend on the particular $\mathbf{f}$ from before. This inequality holds not only with respect to the particular realization of U we selected but in fact holds for any element of A.

Taking the probability measure of $\underline{\mathbf{z}}$ on the set A yields:

$$\begin{aligned}
&\Pr\left\{\exists \mathbf{f} \in \mathcal{F}(\underline{\mathbf{z}}), \sum_{j=1}^{m} U_j(\mathbf{f}) \geq t\}\right\} \inf_{\mathbf{f}\in\mathcal{F}} \Pr\left\{\frac{1}{m}\sum_{j=1}^{m} U_j(\mathbf{f}) \leq \alpha/2\right\} \\
&\leq \Pr\left\{\exists \mathbf{f} \in \mathcal{F}(\underline{\mathbf{z}}), \frac{1}{m}\sum_{j=1}^{m}(U_j(\mathbf{f}) - W_j(\mathbf{f})) \geq \alpha/2\Big) \textbf{ and} \Big(\sum_{j=1}^{m} U_j(\mathbf{f}) \geq t\Big)\right\} \\
&\leq \Pr\left\{\exists \mathbf{f} \in \mathcal{F}(\underline{\mathbf{z}}), \frac{1}{m}\sum_{j=1}^{m}(U_j(\mathbf{f}) - W_j(\mathbf{f})) \geq \alpha/2\right\}.
\end{aligned}$$

Now, observe that for any $\mathbf{f} \in \mathcal{F}$, Chebyshev's inequality implies that

$$\begin{aligned}
&\Pr\left\{\frac{1}{m}\sum_{j=1}^{m} U_j(\mathbf{f}) > \alpha/2\right\} \\
&= \Pr\left\{\frac{1}{m}\sum_{j=1}^{m}\frac{1}{T}\sum_{t=1}^{T}\left(f_t(z_j^{(t)}) - \mathsf{P}_t\, f_t(z_j^{(t)})\right) > \alpha/2\right\} \\
&\leq \frac{4\mathrm{Var}\left(\frac{1}{T}\sum_{t=1}^{T} f_t(z_1^{(t)})\right)}{m\alpha^2} \\
&\leq \frac{b^2}{Tm\alpha^2},
\end{aligned}$$

where the last line follows because we assume the range of each f_t is in $[0, b]$ and hence its variance is at most $\frac{b^2}{4}$. Since the above works for any $\mathbf{f}$, we have

$$\begin{aligned}
\left(1 - \frac{b^2}{Tm\alpha^2}\right) \Pr\left\{\exists \mathbf{f} \in \mathcal{F}(\underline{\mathbf{z}}), \sum_{j=1}^{m} U_j(\mathbf{f}) \geq t\}\right\} \\
\leq \Pr\left\{\exists \mathbf{f} \in \mathcal{F}(\underline{\mathbf{z}}), \frac{1}{m}\sum_{j=1}^{m}(U_j(\mathbf{f}) - W_j(\mathbf{f})) \geq \alpha/2\right\}.
\end{aligned}$$

Finally, selecting $\alpha^2 \geq \frac{2b^2}{Tm}$ finishes the proof. □

The following corollary specializes this result to a mirror image of Proposition 2.12. Similar to the single task analysis of predictive sparse coding, $\mathcal{F}_\mu$ will denote the subclass of $\mathcal{F}$ induced by restricting to dictionaries $\bar{D}^{(1)}, \ldots, \bar{D}^{(T)}$ with s-incoherence of at least μ.

Corollary 3.6. *If* $Tm \geq 2\left(\frac{b}{\alpha}\right)^2$*, then*

$$\begin{aligned}&\mathsf{Pr}_{\underline{\mathbf{z}}}\left\{\exists f \in \mathcal{F}_\mu,\ \left(\forall t \in [T] : \left[\mathsf{margin}_s(\bar{D}^{(t)}, \mathbf{x}^{(t)})) > \iota\right]\right) \textbf{ and } ((\mathsf{P} - \mathsf{P}_{\mathbf{z}})\ell(\cdot, f) > \alpha)\right\} \\ &\leq 2\mathsf{Pr}_{\underline{\mathbf{z}},\underline{\mathbf{z}}'}\left\{\exists f \in \mathcal{F}_\mu,\ \left(\forall t \in [T] : \left[\mathsf{margin}_s(\bar{D}^{(t)}, \mathbf{x}^{(t)}) > \iota\right]\right) \textbf{ and } ((\mathsf{P}_{\mathbf{z}'} - \mathsf{P}_{\mathbf{z}})\ell(\cdot, f) > \alpha/2)\right\}.\end{aligned}$$

Similar to the single-task setting, in the RHS above, let the event whose probability is being measured be

$$J := \left\{\underline{\mathbf{z}}\,\underline{\mathbf{z}}' : \exists \mathbf{f} \in \mathcal{F}_\mu,\ \left(\forall t \in [T] : \left[\mathsf{margin}_s(\bar{D}^{(t)}, \mathbf{x}^{(t)}) > \iota\right]\right) \textbf{ and } ((\mathsf{P}_{\mathbf{z}'} - \mathsf{P}_{\mathbf{z}})\ell(\cdot, f) > \alpha/2)\right\}.$$

Again, similar to the single-task setting, define Z as the event that there exists a hypothesis with stable codes on each task's original sample (using that task's dictionary in the hypothesis), in the sense of the Sparse Coding Stability Theorem (Theorem 2.4), but more than $\eta(T, m, d, k, D, \underline{\mathbf{x}}, \delta)$ points of the ghost samples (in aggregate) whose codes are not guaranteed stable by the Sparse Coding Stability Theorem:

$$Z := \left\{\underline{\mathbf{z}}\,\underline{\mathbf{z}}' : \begin{array}{r}\exists \mathbf{f} \in \mathcal{F}_\mu,\ \left(\forall t \in [T] : \left[\mathsf{margin}_s(\bar{D}^{(t)}, \mathbf{x}^{(t)}) > \iota\right]\right) \\ \textbf{and } \left(\nexists\, \tilde{\underline{\mathbf{x}}} \subseteq_\eta \underline{\mathbf{x}}'\ \left(\forall t \in [T] : \left[\mathsf{margin}_s(\bar{D}^{(t)}, \tilde{\mathbf{x}}^{(t)}) > \tfrac{1}{3}\mathsf{margin}_s(\bar{D}^{(t)}, \mathbf{x}^{(t)})\right]\right)\right)\end{array}\right\}.$$

We will bound $\mathsf{Pr}(J) = \mathsf{Pr}(J \cap \bar{Z}) + \mathsf{Pr}(J \cap Z)$ by bounding $\mathsf{Pr}(J \cap \bar{Z}) + \mathsf{Pr}(Z)$. The next lemma establishes an upper bound on $\mathsf{Pr}(Z)$. It is a multi-task adaptation of the Good Ghost Lemma (Lemma 2.13).

Lemma 3.7 (MTL Good Ghost). *Fix* $\mu, \lambda > 0$*, and* $s \in [k]$*. With probability at least* $1 - \delta$ *over* T *m-samples* $\mathbf{x}^{(1)} \sim (P_1)^m, \ldots, \mathbf{x}^{(T)} \sim (P_T)^m$ *and* T *second m-samples* $\mathbf{x}'^{(1)} \sim (P_1)^m, \ldots, \mathbf{x}'^{(T)} \sim (P_T)^m$*, for any* $(D^{(0)}, D^{(1)}, \ldots, D^{(T)})$ *such that for all* $t \in [T]$*:*

1. $\bar{D}^{(t)} \in \mathcal{D}_\mu$
2. $\varphi_{\bar{D}^{(t)}}$ *is* s*-sparse on* $\mathbf{x}^{(t)}$*,*

it holds for all $t \in [T]$ *that at least* $m - \eta(T, m, d, k, \bar{D}, x, \delta)$ *points* $\mathbf{x}'^{(t)} \subseteq_\eta \mathbf{x}'^{(t)}$ *satisfy*

$\left[\mathsf{margin}_s(\bar{D}^{(t)}, \mathbf{x}'^{(t)}) > \frac{1}{3}\min_{t\in[T]}\mathsf{margin}_s(\bar{D}^{(t)}, \mathbf{x}^{(t)})\right]$, *with*

$$\eta := d(k_s + Tk_e)\log\frac{3888}{\min_{t\in[T]}\mathsf{margin}_s^2(\bar{D}^{(t)}, \mathbf{x}^{(t)})\cdot\lambda} + \log(2Tm+1) + \log\frac{1}{\delta}.$$

Proof. We begin with a fixed $(D^{(0)}, D^{(1)}, \ldots, D^{(T)}) \in \mathcal{E}_\mu$. Also, for now let the minimal margin $\tau = \min_{t\in[T]}\mathsf{margin}_s(\bar{D}^{(t)}, \mathbf{x}^{(t)})$ be fixed a priori. We later extend to the general case of arbitrary $(D^{(0)}, D^{(1)}, \ldots, D^{(T)}) \in \mathcal{E}_\mu$ and τ.

Let $\varepsilon = \frac{(\frac{1}{3}\tau)^2\cdot\lambda}{27}$, and take $\mathcal{E}_\varepsilon$ to be a minimum-cardinality proper ε-cover of $\mathcal{E}_\mu$. Let $(D'^{(0)}, D'^{(1)}, \ldots, D'^{(T)})$ be a candidate element of $\mathcal{E}_\varepsilon$ satisfying $\|\bar{D}^{(t)} - \bar{D}'^{(t)}\|_2 \leq \varepsilon$, for $t \in [T]$. Then the Sparse Coding Stability Theorem (Theorem 2.4) implies that for each $t \in [T]$ the coding margin of $\bar{D}'^{(t)}$ on $\mathbf{x}^{(t)}$ retains over two-thirds the coding margin of $\bar{D}^{(t)}$ on $\mathbf{x}^{(t)}$; that is, $\left[\mathsf{margin}_s(\bar{D}'^{(t)}, \mathbf{x}^{(t)}) > \frac{2}{3}\tau\right]$.

Next, we consider how many points from any of the ghost samples, in aggregate for all $t \in [T]$, fail to satisfy $\left[\mathsf{margin}_s(\bar{D}'^{(t)}, \cdot) > \frac{2}{3}\tau\right]$. We tentatively take $(D^{(0)}, D^{(1)}, \ldots, D^{(T)})$ and hence $(D'^{(0)}, D'^{(1)}, \ldots, D'^{(T)})$ as fixed. Suppose there are at least η violations in aggregate over the ghost samples. In particular, consider the case that there are at least c_t violations in ghost sample $\mathbf{x}'^{(t)}$ for $t \in [T]$; that is, we fix a particular choice $(c_1, \ldots, c_T)$ satisfying $c_t \geq 0$ and $\sum_{t\in[T]} c_t = \eta$. Without loss of generality, assume that the violations in $\mathbf{x}'^{(t)}$ occur in the first c_t slots.

It is readily apparent that the probability that we see 0 violations in sample $\mathbf{x}^{(t)}$ but at least c_t violations in the ghost sample $\mathbf{x}'^{(t)}$ is at most 2^{-c_t}. Consequently, the probability of seeing 0 violations in samples $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(T)}$ but at least $c_1, \ldots, c_T$ violations in $\mathbf{x}'^{(1)}, \ldots, \mathbf{x}'^{(T)}$ respectively is at most $2^{-\eta}$. This bound holds regardless of our particular choice of $(c_1, \ldots, c_T)$, and hence marginalizing over the $(c_1, \ldots, c_T)$, we conclude that the probability of seeing 0 violations (in aggregate) on the original samples but at least η violations (in aggregate) on the ghost samples is at most $2^{-\eta}$.

This result easily extends to the case where τ is not fixed a priori. Since the VC-dimension of a threshold functions is 1, for a fixed hypothesis there are only $2Tm+1$ ways to label each of the $2Tm$ points in terms of whether or not they violate the margin threshold. Hence, only with probability at most $(2Tm+1)2^{-\eta}$ do at least η points of the

ghost samples (in aggregate over $t \in [T]$) violate their respective conditions

$$\left[\mathsf{margin}_s(\bar{D}'^{(t)}, \cdot) > \frac{2}{3} \min_{t\in[T]} \mathsf{margin}_s(\bar{D}^{(t)}, \mathbf{x}^{(t)})\right], \text{ for } t \in [T]. \tag{3.6}$$

This result also can be extended for arbitrary $(D'^{(0)}, D'^{(1)}, \ldots, D'^{(T)}) \in \mathcal{E}_\varepsilon$ using a bound on the proper covering numbers of $\mathcal{D}^{(s)} \times \left(\mathcal{D}^{(e)}\right)^T$ in the metric (3.5), yielding: for all $(D'^{(0)}, D'^{(1)}, \ldots, D'^{(T)}) \in \mathcal{E}_\varepsilon$, only with probability at most

$$\left(\log \frac{16}{\varepsilon}\right)^{d(k_s+Tk_e)} (2Tm+1)2^{-\eta}$$

do at least η points (in aggregate over $t \in [T]$) violate their respective conditions (3.6).

Finally, consider the at least $Tm-\eta$ points in the ghost samples (in aggregate) satisfying their respective conditions (3.6). Since $\|\bar{D}'^{(t)} - \bar{D}^{(t)}\| \leq \varepsilon$ for $t \in [T]$, the Sparse Coding Stability Theorem (Theorem 2.4) implies that these points satisfy their respective conditions

$$\left[\mathsf{margin}_s(\bar{D}^{(t)}, \cdot) > \frac{1}{3} \min_{t\in[T]} \mathsf{margin}_s(\bar{D}^{(t)}, \mathbf{x}^{(t)})\right], \text{ for } t \in [T]. \qquad \square$$

The last piece of the puzzle is to present a multi-task version of Lemma 2.14. We will need a particular ε-net composed of the product of a proper ε-net for the dictionaries and a (not necessarily proper) ε-net for the set of linear hypotheses:

Proposition 3.8. *The product of proper ε-covering numbers for $\mathcal{E}_\mu$ and the ε-covering numbers for $\mathcal{W}^T$ is bounded by*

$$\left(\frac{16}{\varepsilon}\right)^{k_s(d+T)+k_eT(d+1)} \left(\frac{r}{4}\right)^{T(k_s+k_e)}.$$

Lemma 3.9 (MTL Large Deviation on Good Ghost). *Let $\varpi := \alpha/2 - \left(2L\beta + \frac{b\eta}{Tm}\right)$, $\beta := \frac{\varepsilon}{2\lambda}\left(1 + \frac{3r\sqrt{s}}{\mu}\right)$. Then*

$$\mathsf{Pr}(J \cap \bar{Z}) \leq \left(\frac{16}{\varepsilon}\right)^{k_s(d+T)+k_eT(d+1)} \left(\frac{r}{4}\right)^{T(k_s+k_e)} \exp(-Tm\varpi^2/(2b^2)).$$

The proof of Lemma 3.9 is withheld as it is nearly identical to the proof of Lemma 2.14, with small modifications for the multi-task setting. The only crucial difference is the size of the ε-net used to cover the hypothesis space and the better concentration in the application of Hoeffding's inequality (as we are averaging over Tm bounded independent random variables in the sum rather than only m bounded independent random variables).

Finally, by following the derivation of the final learning bound for the single-task setting (Theorem 2.5), we arrive at the multi-task predictive sparse coding generalization error bound (Theorem 3.2).

3.5 Learning

This section presents a simple learning algorithm for multi-task predictive sparse coding using stochastic subgradient descent. The main difficulty is to work out the subgradient computations; fortunately, in the single-task setting Mairal et al. (2012) already worked out a subgradient of the expected loss with respect to the dictionary. We first summarize the updates in the single-task setting due to Mairal et al. (2012), after which the updates for the multi-task setting with partially shared dictionaries will follow naturally.

Single-task setting In single-task predictive sparse coding, the model parameters consist of a dictionary $D \in \mathbb{R}^{d \times k}$ of k atoms in $\mathbb{R}^d$ and a linear estimator $w \in \mathbb{R}^k$. The goal is to minimize the (regularized) stochastic objective

$$\mathsf{E}_{(x,y)\sim \mathsf{P}}\, \ell\big(y, \langle w, \varphi_D(x)\big) + \lambda_w \|w\|_2^2$$

using a labeled m-sample $\mathbf{z}$ consisting of labeled points $(x_1, y_1), \ldots, (x_m, y_m)$.

This objective can be minimized via stochastic subgradient descent, provided that we can compute a subgradient of the objective

$$\ell\big(y, \langle w, \varphi_D(x)\big) + \lambda_w \|w\|_2^2,$$

where (x, y) is a labeled point randomly drawn from distribution P.

For completeness, the subgradient computations shown by Mairal et al. (2012, Proposition 1) will be reproduced here. To simplify the exposition, define the sparse code variable $\alpha := \varphi_D(x)$ and the support variable $J := \mathsf{supp}(\varphi_D(x))$. One choice of subgradient of the objective with respect to D can be described in terms of a subgradient with respect to the active subdictionary D_J and the gradient with respect to inactive subdictionary D_{J^c} (where J^c is the complement, $[k] \setminus J$). The choice of subgradient with respect to D_J will be

$$-D_J \beta \alpha^T + (x - D\alpha)\beta^T,$$

with

$$\beta := (D_J^T D_J)^{-1} \frac{\partial \ell}{\partial \alpha_J}\left(y, \langle w, \alpha \rangle\right).$$

The gradient with respect to D_{J^c} is simply 0. The subdifferential of the objective with respect to w follows easily from the chain rule and is

$$\frac{\partial}{\partial w}\left[\ell\big(y, \langle w, \alpha \rangle\big) + \lambda_w \|w\|_2^2\right]\Big|_{(w,\alpha)} = \alpha \left(\frac{\partial \ell(y, \hat{y})}{\partial \hat{y}}\Big|_{\langle w, \alpha \rangle}\right) + 2\lambda_w w.$$

The experiments in the next section will employ the hinge loss $\ell(y, \hat{y}) = \mathsf{max}\{0, 1 - y\hat{y}\}$, and so we provide the updates for this case. First, consider the case when $1 - y\langle w, \alpha \rangle \geq 0$. In this case, $-\alpha y + 2\lambda_w w$ is an element of the subdifferential

$$\frac{\partial}{\partial w}\left[\ell\big(y, \langle w, \alpha \rangle\big) + \lambda_w \|w\|_2^2\right]\Big|_{(w,\alpha)},$$

and subdifferential with respect to D_J contains, among others, the subgradient element

$$y\left(D_J (D_J^T D_J)^{-1} w_J \alpha^T - (x - D\alpha) w_J^T (D_J^T D_J)^{-1}\right).$$

In the case when $1 - y\langle w, \alpha \rangle) < 0$, the gradients for both w and D are zero, and hence no update is made. This outcome is business as usual with the hinge loss.

Multi-task setting In multi-task predictive sparse coding, recall that the dictionary $\bar{D}^{(t)}$ for each task consists of a k_s-atom shared subdictionary $D^{(0)} \in (B_{\mathbb{R}^d})^{k_s}$ and a task-specific k_e-atom subdictionary $D^{(t)} \in (B_{\mathbb{R}^d})^{k_e}$, and recall that these subdictionaries are combined as $\bar{D}^{(t)} = (D^{(0)}\ D^{(t)})$.

Before writing the ideal objective, we first define Q as the uniform probability measure over the probability measures $\mathsf{P}_1, \ldots, \mathsf{P}_T$. The goal is then to minimize the (regularized) stochastic objective

$$\mathsf{E}_{\mathsf{P}\sim\mathsf{Q}}\left[\mathsf{E}_{(x,y)\sim\mathsf{P}} \sum_{t=1}^{T} \delta_{\mathsf{P}_t}(\mathsf{P})\ell\left(y, \left\langle W_t, \varphi_{\bar{D}^{(t)}}(x)\right\rangle\right)\right] + \lambda_w \sum_{t=1}^{T} \|W_t\|_2^2. \tag{3.7}$$

Performing stochastic subgradient descent with the multi-task objective (3.7) is similar to stochastic subgradient descent in the single-task setting. Define the objective for a single point-label pair $z = (x, y)$ from the t^{th} task as

$$f_{t,z}(D^{(0)}, D^{(t)}, W_t) := \ell\left(y, \left\langle W_t, \varphi_{\bar{D}^{(t)}}(x)\right\rangle\right) + \lambda_w \|W_t\|_2^2. \tag{3.8}$$

Algorithm 1 shows a mini-batch stochastic subgradient descent algorithm for approximately optimizing the regularized stochastic objective using a meta-sample of T labeled m-samples $\mathbf{z}^{(1)}, \ldots, \mathbf{z}^{(T)}$.

To do an update, this algorithm draws a task uniformly at random by drawing a probability measure P_t from Q, approximately draws a mini-batch of q examples $(x_1, y_1), \ldots, (x_q, y_q)$ from P_t by drawing some $(x_{j_1}^{(t)}, y_{j_1}^{(t)}), \ldots, (x_{j_q}^{(t)}, y_{j_q}^{(t)})$ from the empirical measure $\mathsf{P}_{\mathbf{z}^{(t)}}$), and finally uses the subgradient updates shown in Figure 3.1. The choice of step size, $\eta_i := \frac{1}{q}\min\left\{1, \frac{n}{10i}\right\}$, was adopted from Mairal et al. (2012); observe that the step size is normalized by the mini-batch size so that for each of the n rounds the step size is effectively $\min\left\{1, \frac{n}{10i}\right\}$.

Considering the full dictionary $\bar{D}^{(t)} = (D^{(0)}\ D^{(t)})$, the updates come readily from the single-task setting updates shown above. An update to $D^{(0)}$ in the multi-task model translates to an update to the first k_s columns of D in the single-task model; similarly, an update to $D^{(t)}$ in the multi-task model corresponds to an update to the remaining k_e columns of

Algorithm 1: A mini-batch stochastic subgradient descent learning algorithm for multi-task predictive sparse coding. The subgradient computations can be found in Figure 3.1.

Input: T datasets $\left\{\mathbf{z}^{(t)}\right\}_{r\in[T]}$, where $\mathbf{z}^{(t)} = \left\{(\mathbf{x}_j^{(t)}, \mathbf{y}_j^{(t)})\right\}_{j\in[m]}$ for $t \in [T]$

begin

  for $i = 1 \to n$ **do**

    $\eta_i := \frac{1}{q} \min\left\{1, \frac{n}{10i}\right\}$

    Draw t uniformly at random over $[T]$

    $\mathbf{z}' :=$ [q points drawn from $\mathbf{z}^{(t)}$ uniformly at random]

    $(D^{(0)}_{\mathsf{new}}, D^{(t)}_{\mathsf{new}}, w_{\mathsf{new}}) := (D^{(0)}, D^{(t)}, W_t)$

    for $j = 1 \to q$ **do**

      Let $f_{z'_j}$ be objective induced by z'_j, $(D^{(0)}, D^{(t)})$, and W_t.

      $D^{(0)}_{\mathsf{new}} := D^{(0)}_{\mathsf{new}} - \eta_i \frac{\partial f_{t,z'_j}}{\partial D^{(0)}}(D^{(0)}, D^{(t)}, W_t)$

      $D^{(t)}_{\mathsf{new}} := D^{(t)}_{\mathsf{new}} - \eta_i \frac{\partial f_{t,z'_j}}{\partial D^{(t)}}(D^{(0)}, D^{(t)}, W_t)$

      $w_{\mathsf{new}} := w_{\mathsf{new}} - \eta_i \frac{\partial f_{t,z'_j}}{\partial W_t}(D^{(0)}, D^{(t)}, W_t)$

    end

    $(D^{(0)}, D^{(t)}, W_t) := (D^{(0)}_{\mathsf{new}}, D^{(t)}_{\mathsf{new}}, w_{\mathsf{new}})$

  end

end

D in the single-task model.

3.6 Experiments

In this section, we explore the extent to which multi-task learning helps (or hurts) the performance of predictive sparse coding. The approach taken here will be to focus on several controlled experiments in which a single parameter, such as the number of exclusive atoms k_e, is varied while all other parameters are fixed. We are not aware of any previous study in this setting comparing the relative differences between models with varying levels of sharing/exclusivity between tasks.

All the experiments in this section use the MNIST digits dataset (LeCun et al., 1998) with the official training / test splits. The 10-class digit classification task was reduced with a one-vs-all decomposition into 10 binary classification tasks suitable for the multi-

$$\frac{\partial f_{t,z}}{\partial D^{(0)}}(D^{(0)}, D^{(t)}, W_t) = \frac{\partial \ell\left(y, \langle W_t, \varphi_{\bar{D}^{(t)}}(x)\rangle\right)}{\partial D^{(0)}}\Big|_{(D^{(0)}, D^{(t)}, W_t)}$$

$$\frac{\partial f_{t,z}}{\partial D^{(l)}}(D^{(0)}, D^{(t)}, W_t) = \begin{cases} \dfrac{\partial \ell\left(y, \langle W_t, \varphi_{\bar{D}^{(t)}}(x)\rangle\right)}{\partial D^{(t)}}\Big|_{(D^{(0)}, D^{(t)}, W_t)} & \text{if } l = t \\ 0 & \text{if } l \neq t \end{cases}$$

$$\frac{\partial f_{t,z}}{\partial W^{(l)}}(D^{(0)}, D^{(t)}, W_t) = \begin{cases} \dfrac{\partial \ell\left(y, \langle W_t, \varphi_{\bar{D}^{(t)}}(x)\rangle\right)}{\partial W_t}\Big|_{(D^{(0)}, D^{(t)}, W_t)} + 2\lambda_w W_t & \text{if } l = t \\ 0 & \text{if } l \neq t \end{cases}$$

Figure 3.1: Subgradient updates for multi-task predictive sparse coding.

task setting. For $c \in \{0, 1, \ldots, 9\}$, the c^{th} task involves discriminating between the digit c versus a class that is a union of all the rest of the digits $\{0, 1, \ldots, 9\} \setminus \{c\}$.

Experimental intricacies In all the experiments, the hinge loss $\ell\big(y, y'\big) = \max\{0, 1 - yy'\}$ was used, the linear hypothesis regularization parameter λ_w was set to 10^{-4}, the ℓ_1-norm regularization parameter was set to 0.1, and the mini-batch size n was set to 10. Prior to dictionary learning, dictionaries were initialized via the following method. First, each dictionary atom was set to the average of 3 random selected data points in the training set, scaled to unit ℓ_2 norm. The resulting initial random dictionary was then trained via 50 iterations of the standard bi-convex alternating algorithm for sparse coding (using LARS for the sparse coding step and the dual method of Lee et al. (2007) for the dictionary update step). For multi-task models that use several dictionaries, such as when $k_e > 0$ (and in particular when $k_s = 0$ and $k_e = k$), all of the dictionaries were initialized to the same dictionary using the described procedure. For each task, the linear hypothesis W_t was initialized by running 10^5 iterations of Pegasos (Shalev-Shwartz et al., 2011) on the labeled points for that task; here, the input space representation arose from the sparse-coding-initialized dictionary.

After this initialization procedure, learning proceeded as per the multi-task predictive sparse coding algorithm described in Algorithm 1, with a few modifications. First, rather than drawing random points for the mini-batch $\mathbf{z}'$, we instead cycled over a permuted version

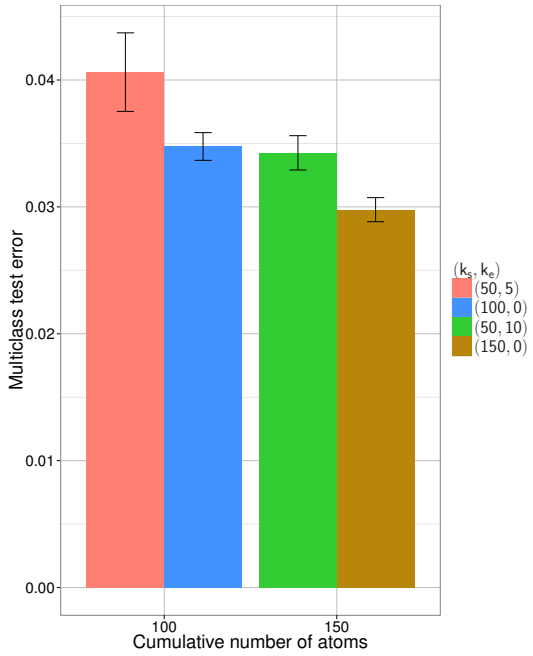
of the data, grabbing q points in each mini-batch. Once all the points had been traversed, the last mini-batch was closed (possibly being smaller in size than the other mini-batches), and the data again was placed into a random permutation. Additionally, after running this version of Algorithm 1, for each task Pegasos again was run for 10^5 iterations to learn W_t.

We have not yet described the number of training iterations n for which multi-task predictive sparse coding ran. For each model configuration (k_s, k_e), the number of iterations of training was determined as follows. For each value of n in some set, the first 80% of the unpermuted training data was used for training and the remaining 20% for validation. The selected value of n had the lowest average multi-class test error across the five repetitions. We tried values of n ranging from 10,000 to 180,000 iterations, in increments of 10,000. Hence, for different settings of k_s and k_e, it is possible that a different number of training iterations was used.

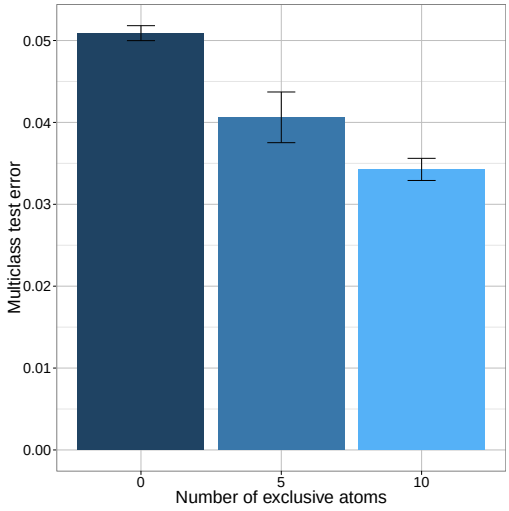
Experimental investigations The first investigation fixes the cumulative number of atoms $k_s + Tk_e$ in the multi-task model while varying their allocation across the shared dictionary and the task-specific dictionaries. Since there are ten tasks, a one atom increase to k_e must be compensated by decreasing k_s by ten atoms. Figure 3.2a shows the results of this experiment. From the plot, it appears more beneficial to use full-sharing ($k_e = 0$), at least compared to small departures from the full sharing model in which k_e is small.

The second experiment fixes the size of the shared dictionary at $k_s = 50$ and compares the performances of the cases of 0, 5, or 10 exclusive atoms per task-specific dictionary. The results are shown in Figure 3.2b. This experiment is more of a sanity check to test whether Algorithm 1 is able to learn from complicated partial-sharing models. As the dataset size is large relative to the dictionary sizes, overlearning does not appear to take place as the model size increases, and the additional task-exclusive atoms appear to decrease the test error.

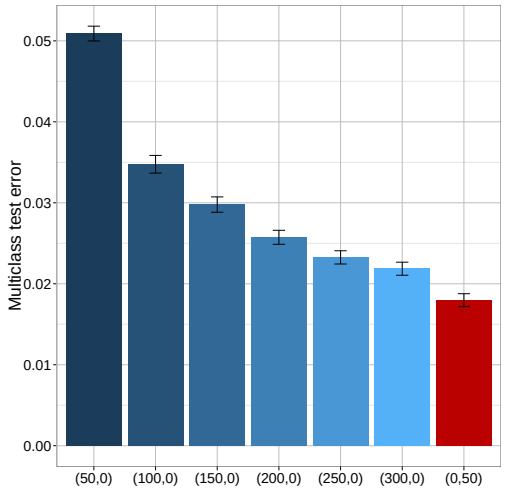
The third and final experiment compares three full sharing models, in which k_e is set to zero and k_s varies between 50 and 300, and a fully exclusive (single-task) model in which each task uses its own 50-atom dictionary. The fully exclusive model translates to 500 atoms cumulatively, and so if it is possible to achieve similar performance using fewer cumulative



(a) Full versus partial sharing



(b) Increasing exclusive atoms



(c) Increasing size of fully-shared dictionary

Figure 3.2: Results of three experiments investigating performance of the sharing model of multi-task predictive sparse coding. The plot (a) compares two models each with 100 atoms cumulatively, and it also compares two models each with 150 atoms cumulatively. The plot (b) compares three models, each with 50 shared atoms, and a varying number of exclusive atoms per task-specific dictionary. In plot (c), the effect of increasing the size of the dictionary in a full-sharing model is compared to 50-atom per dictionary fully exclusive model.

atoms in a shared model, then stronger generalization error guarantees can be made using the results from Section 3.3. As shown by the results in Figure 3.2c, the full sharing model

makes progress toward the test error achieved by the fully exclusive model; however, further experiments with larger full sharing models (up to $k_s = 500$) are necessary to see if the full sharing model can match or even outperform the fully exclusive model.

3.7 Discussion

This chapter introduced a new multi-task dictionary model for sparse coding. This model can be applied in the unsupervised and supervised settings, yielding multi-task sparse coding and multi-task predictive sparse coding respectively. In the unsupervised setting, it was shown with high probability that the task-wise average of the expected reconstruction error for any hypothesis exceeds the task-wise average of that hypothesis's empirically observed reconstruction errors by at most $O(\sqrt{dk_s/(Tm)} + \sqrt{dk_e/m})$. Hence, when the number of tasks is very large, the size of the shared dictionary (k_s) also can be very large. A similar result holds in the predictive setting, with additional modifications due to sparse auto-encoder stability properties and the statistical cost of estimating a separate linear estimator for each task.

The empirical results suggest that when the number of cumulative atoms in the multi-task dictionary model is fixed, it is beneficial to use a non-trivial amount of sharing ($k_s > 0$). Additionally, the simple stochastic subgradient descent algorithm for the sharing model of multi-task predictive sparse coding appears to work well. An interesting avenue for future exploration would be to experiment with much larger dictionary sizes, to the point where using a fully exclusive model severely overlearns; in this setting, the learning bounds developed in this work suggest that incorporating sharing into the multi-task dictionary model can stave off overlearning in common low-sample regimes.

Part II

NEW REPRESENTATION LEARNING PARADIGMS

CHAPTER 4

MINIMAX MULTI-TASK LEARNING

4.1 Introduction

The essence of machine learning is to exploit what we observe in order to form accurate predictors of what we cannot. A multi-task learning (MTL) algorithm learns an inductive bias to learn several tasks together. MTL is incredibly pervasive in machine learning: it has natural connections to random effects models (Yu et al., 2009a); user preference prediction (including collaborative filtering) can be framed as MTL (Zhang et al., 2011); multi-class classification admits the popular *one-vs-all* and *all-pairs* MTL reductions; and MTL admits provably good learning in settings where single-task learning is hopeless (Baxter, 2000; Maurer, 2009). But if we see examples from a random set of tasks today, which of these tasks will matter tomorrow? Not knowing in the present what challenges nature has in store for the future, a sensible strategy is to mitigate the worst case by ensuring some minimum proficiency on each task.

Consider a simple learning scenario: A music preference prediction company is in the business of predicting what 5-star ratings different users would assign to songs. At training time, the company learns a shared representation for predicting the users' song ratings by pooling together the company's limited data on each user's preferences. Given this learned representation, a separate predictor for each user can be trained very quickly. At test time, the environment draws a user according to some (possibly randomized) rule and solicits from the company a prediction of that user's preference for a particular song. The environment may also ask for predictions about new users, described by a few ratings each, and so

the company must leverage its existing representation to rapidly learn new predictors and produce ratings for these new users.

Classically, multi-task learning has sought to minimize the (regularized) sum of the empirical risks over a set of tasks. In this way, classical MTL implicitly assumes that once the learner has been trained, it will be tested on test tasks drawn uniformly at random from the empirical task distribution of the training tasks. Notably, there are several reasons why classical MTL may not be ideal:

- While at training time the usual flavor of MTL commits to a fixed distribution over users (typically either uniform or proportional to the number of ratings available for each user), at test time there is no guarantee what user distribution we will encounter. In fact, there may not exist any fixed user distribution: the sequence of users for which ratings are elicited could be adversarial.

- Even in the case when the distribution over tasks is not adversarial, it may be in the interest of the music preference prediction company to guarantee some minimum level of accuracy per user in order to minimize negative feedback and a potential loss of business, rather than maximizing the mean level of accuracy over all users.

- Whereas minimizing the average prediction error is very much a teleological endeavor, typically at the expense of some locally egregious outcomes, minimizing the worst-case prediction error respects a notion of fairness to all tasks (or people).

This chapter introduces *minimax multi-task learning* as a response to the above scenario.[1] In addition, we cast a spectrum of multi-task learning. At one end of the spectrum lies minimax MTL, and departing from this point progressively relaxes the "hardness" of the maximum until full relaxation reaches the second endpoint and recovers classical MTL. We further sculpt a generalized loss-compositional paradigm for MTL which includes this spectrum and several other new MTL formulations. This paradigm equally applies to the problem of *learning to learn* (LTL), in which the goal is to learn a hypothesis space from a set of training tasks such that this representation admits good hypotheses on future tasks.

[1] Note that minimax MTL does not refer to the *minimax estimators* of statistical decision theory.

In truth, MTL and LTL typically are handled equivalently at training time — this work will be no exception — and they diverge only in their test settings and hence the learning theoretic inquiries they inspire.

Contributions. The first contribution of this chapter is to introduce minimax MTL and a continuum of relaxations. Second, we introduce a generalized loss-compositional paradigm for MTL which admits a number of new MTL formulations and also includes classical MTL as a special case. Third, we empirically evaluate the performance of several MTL formulations from this paradigm in the multi-task learning and learning to learn settings, under the task-wise maximum test risk and task-wise mean test risk criteria, on four datasets (one synthetic, three real). Finally, Theorem 4.1 is the core theoretical contribution of this chapter and shows the following: If it is possible to obtain maximum empirical risk across a set of training tasks below some level γ, then it is likely that the maximum true risk obtained by the learner on a new task is bounded by roughly γ. Hence, if the goal is to minimize the worst case outcome over new tasks, the theory suggests minimizing the maximum of the empirical risks across the training tasks rather than their mean.

In the next section, we recall the settings of multi-task learning and learning to learn, formally introduce minimax MTL, and motivate it theoretically. In Section 4.3, we introduce a continuously parameterized family of minimax MTL relaxations and the new generalized loss-compositional paradigm. Section 4.4 presents an empirical evaluation of various MTL/LTL formulations with different models on four datasets. Finally, we close with a discussion.

4.2 Minimax multi-task learning

We begin with a promenade through the basic MTL and LTL setups, with an effort to abide by the notation introduced by Baxter (2000). Throughout the rest of the chapter, each labeled example (x, y) will live in $\mathcal{X} \times \mathcal{Y}$ for input instance x and label y. Typical choices of $\mathcal{X}$ include $\mathbb{R}^n$ or a compact subset thereof, while $\mathcal{Y}$ typically is a compact subset of $\mathbb{R}$ or the binary $\{-1, 1\}$. In addition, define a loss function $\ell : \mathbb{R} \times \mathcal{Y} \to \mathbb{R}_+$. For simplicity,

this chapter considers ℓ_2 loss (squared loss) $\ell(y, y') = (y - y')^2$ for regression and hinge loss $\ell(y, y') = \max\{0, 1 - yy'\}$ for classification.

MTL and LTL often are framed as applying an inductive bias to learn a common hypothesis space, selected from a fixed family of hypothesis spaces, and thereafter learning from this hypothesis space a hypothesis for each task observed at training time. It will be useful to formalize the various sets and elements present in the preceding statement. Let $\mathbb{H}$ be a family of hypothesis spaces. Any hypothesis space $\mathcal{H} \in \mathbb{H}$ itself is a set of hypotheses; each hypothesis $h \in \mathcal{H}$ is a map $h : \mathcal{X} \to \mathbb{R}$.

Learning to learn. In learning to learn, the goal is to achieve inductive transfer to learn the best $\mathcal{H}$ from $\mathbb{H}$. Unlike in MTL, there is a notion of an *environment* of tasks: an unknown probability measure Q over a space of task probability measures $\mathscr{P}$. The goal is to find the optimal representation via the objective

$$\inf_{\mathcal{H} \in \mathbb{H}} \mathsf{E}_{\mathsf{P} \sim \mathsf{Q}} \inf_{h \in \mathcal{H}} \mathsf{E}_{(x,y) \sim \mathsf{P}} \ell(y, h(x)). \tag{4.1}$$

In practice, T (unobservable) training task probability measures $\mathsf{P}_1, \ldots, \mathsf{P}_T \in \mathscr{P}$ are drawn iid from Q, and from each task t a set of m examples are drawn iid from P_t.

Multi-task learning. Whereas in learning to learn there is a distribution over tasks, in multi-task learning there is a fixed, finite set of tasks indexed by $[T] := \{1, \ldots, T\}$. Each task $t \in [T]$ is coupled with a fixed but unknown probability measure P_t. Classically, the goal of MTL is to minimize the expected loss at test time under the uniform distribution on $[T]$:

$$\inf_{\mathcal{H} \in \mathbb{H}} \frac{1}{T} \sum_{t \in [T]} \inf_{h \in \mathcal{H}} \mathsf{E}_{(x,y) \sim \mathsf{P}_t} \ell(y, h(x)). \tag{4.2}$$

Notably, this objective is equivalent to (4.1) when Q is the uniform distribution on the set of probability measures $\{\mathsf{P}_1, \ldots, \mathsf{P}_T\}$. In terms of the data generation model, MTL differs from LTL since the tasks are fixed; however, just as in LTL, from each task t a set of m examples are drawn iid from P_t .

4.2.1 Minimax MTL

A natural generalization of classical MTL results by introducing a prior distribution π over the index set of tasks $[T]$. Given π, the (idealized) objective of this generalized MTL is

$$\inf_{\mathcal{H}\in\mathbb{H}} \mathsf{E}_{t\sim\pi} \inf_{h\in\mathcal{H}} \mathsf{E}_{(x,y)\sim \mathsf{P}_t} \ell(y, h(x)), \tag{4.3}$$

given only the training data $\{(x_1^{(t)}, y_1^{(t)}), \ldots, (x_m^{(t)}, y_m^{(t)})\}_{t\in[T]}$. The classical MTL objective (4.2) equals (4.3) when π is taken to be the uniform prior over $[T]$. We argue that in many instances, that which is most relevant to minimize is not the expected error under a uniform distribution over tasks, or even any pre-specified π, but rather the expected error for the worst π. We propose to minimize the maximum error over tasks under an adversarial choice of π, yielding the objective:

$$\inf_{\mathcal{H}\in\mathbb{H}} \sup_{\pi} \mathsf{E}_{t\sim\pi} \inf_{h\in\mathcal{H}} \mathsf{E}_{(x,y)\sim \mathsf{P}_t} \ell(y, h(x)),$$

where the supremum is taken over the T-dimensional simplex. As the supremum (assuming it is attained) is attained at an extreme point of the simplex, this objective is equivalent to

$$\inf_{\mathcal{H}\in\mathbb{H}} \max_{t\in[T]} \inf_{h\in\mathcal{H}} \mathsf{E}_{(x,y)\sim \mathsf{P}_t} \ell(y, h(x)).$$

In practice, we approximate the true objective via a regularized form of the empirical objective

$$\inf_{\mathcal{H}\in\mathbb{H}} \max_{t\in[T]} \inf_{h\in\mathcal{H}} \sum_{i=1}^{m} \ell(y_i^{(t)}, h(x_i^{(t)})).$$

In the next section, we motivate minimax MTL theoretically by showing that the worst-case performance on future tasks likely will not be much higher than the maximum of the empirical risks for the training tasks. We restrict attention to the case of finite $\mathbb{H}$.

4.2.2 A learning to learn bound for the maximum risk

In this subsection, we use the following notation. Let $\mathsf{P}_1, \ldots, \mathsf{P}_T$ be probability measures drawn iid from Q, and for $t \in [T]$ let $\mathbf{z}^{(t)}$ be an m-sample (a sample of m points) from P_t with corresponding empirical measure $\mathsf{P}_{\mathbf{z}^{(t)}}$. Also, if P is a probability measure then $\mathsf{P}\,\ell(\cdot, h) := \mathsf{E}\,\ell(y, h(x))$; similarly, if $\mathsf{P}_{\mathbf{z}}$ is an empirical measure with respect to m-sample $\mathbf{z}$, then $\mathsf{P}_{\mathbf{z}}\,\ell(\cdot, h) := \frac{1}{m}\sum_{i=1}^{m} \ell(y_i, h(x_i))$.

Our focus is the learning to learn setting with a minimax lens: when one learns a representation $\mathcal{H} \in \mathbb{H}$ from multiple training tasks and observes maximum empirical risk γ, we would like to guarantee that $\mathcal{H}$'s true risk on a newly drawn test task will be bounded by roughly γ. Such a goal is in striking contrast to the classical emphasis of learning to learn, where the goal is to obtain bounds on $\mathcal{H}$'s expected true risk. Using $\mathcal{H}$'s expected true risk and Markov's inequality, Baxter (2000, the display prior to (25)) showed that the probability that $\mathcal{H}$'s true risk on a newly drawn test task is above some level γ decays as the expected true risk over γ:

$$\mathsf{Pr}\left\{\inf_{h\in\mathcal{H}} \mathsf{P}\,\ell(\cdot, h) \geq \gamma\right\} \leq \frac{\frac{1}{T}\sum_{t\in[T]} \mathsf{P}_{\mathbf{z}^{(t)}}\,\ell(\cdot, h_t) + \varepsilon}{\gamma} \tag{4.4}$$

where the size of ε is controlled by T, m, and the complexities of certain spaces.

The expected true risk is not of primary interest for controlling the tail of the (random) true risk, and a more direct approach yields a much better bound. We restrict the space of representations $\mathbb{H}$ to be finite with cardinality $\mathcal{C}$; in this case, the analysis is particularly simple and illuminates the idea for proving the general case. The next theorem is the main result of this section:

Theorem 4.1. *Let $|\mathbb{H}| = \mathcal{C}$, and let the loss ℓ be L-Lipschitz in its second argument and bounded by B. Suppose T tasks $\mathsf{P}_1, \ldots, \mathsf{P}_T$ are drawn iid from Q and from each task P_t an iid m-sample $\mathbf{z}^{(t)}$ is drawn. Suppose there exists $\mathcal{H} \in \mathbb{H}$ such that all $t \in [T]$ satisfy $\min_{h\in\mathcal{H}} \mathsf{P}_{\mathbf{z}^{(t)}}\,\ell(\cdot, h) \leq \gamma$. Let P be newly drawn probability measure from Q. Let $\hat{h}$ be the empirical risk minimizer over the test m-sample. With probability at least $1-\delta$ with respect*

to the random draw of the T tasks and their T corresponding m-samples:

$$\Pr\left\{ \mathrm{P}\,\ell\big(\cdot,\hat{h}\big) > \gamma + \frac{1}{T} + L\max_{\mathcal{H}\in\mathbb{H}} \mathcal{R}_m(\mathcal{H}) + B\sqrt{\frac{\log\frac{2}{\delta}}{2m}} \right\} \leq \frac{\log\frac{2\mathcal{C}}{\delta} + \log\lceil B\rceil + \log(T+1)}{T}. \tag{4.5}$$

In the above, $\mathcal{R}_m(\mathcal{H})$ is the Rademacher complexity of $\mathcal{H}$ (cf. (Bartlett and Mendelson, 2002)), defined as

$$\mathcal{R}_m(\mathcal{H}) := \mathrm{E}_{\mathbf{x}}\,\mathrm{E}_{\sigma} \sup_{h\in\mathcal{H}} \frac{2}{m}\sum_{i=1}^{m} \sigma_i h(x_i),$$

where $\sigma_1,\ldots,\sigma_m$ are iid Rademacher random variables (uniform over $\{-1,1\}$. Critically, in (4.5) the probability of observing a task with high true risk decays with T, whereas in (4.4) the decay is independent of T. Hence, when the goal is to minimize the probability of bad performance on future tasks uniformly, this theorem motivates minimizing the *maximum* of the empirical risks as opposed to their mean.

For the proof of Theorem 4.1, first consider the singleton case $\mathbb{H} = \{\mathcal{H}_1\}$. Suppose that for γ fixed a priori, the maximum of the empirical risks is bounded by γ, i.e. $\max_{t\in[T]} \min_{h\in\mathcal{H}_1} \mathrm{P}_{\mathbf{z}^{(t)}}\,\ell(\cdot,h) \leq \gamma$.

Let a new probability measure P drawn from Q correspond to a new test task, with accompanying m-sample $\mathbf{z}$. Suppose the probability of the event $[\min_{h\in\mathcal{H}_1} \mathrm{P}_{\mathbf{z}}\,\ell(\cdot,h) > \gamma]$ is at least ε. Then the probability that γ bounds all T empirical risks is at most $(1-\varepsilon)^T \leq e^{-T\varepsilon}$. Hence, with probability at least $1-e^{-T\varepsilon}$:

$$\Pr\left\{ \min_{h\in\mathcal{H}_1} \mathrm{P}_{\mathbf{z}}\,\ell(\cdot,h) > \gamma \right\} \leq \varepsilon.$$

A simple application of the union bound extends this result for finite $\mathbb{H}$:

Lemma 4.2. *Under the same conditions as Theorem 4.1, with probability at least $1-\delta/2$ with respect to the random draw of the T tasks and their T corresponding m-samples:*

$$\Pr\left\{ \min_{h\in\mathcal{H}} \mathrm{P}_{\mathbf{z}}\,\ell(\cdot,h) > \gamma \right\} \leq \frac{\log\frac{2\mathcal{C}}{\delta}}{T}.$$

The bound in the lemma states a $1/T$ rate of decay for the probability that the empirical risk obtained by $\mathcal{H}$ on a new task exceeds γ. Next, we relate this empirical risk to the true risk obtained by the empirical risk minimizer. Note that at test time $\mathcal{H}$ is fixed and hence independent of any test m-sample. Then, from standard learning theory results shown below:

Lemma 4.3. *Take loss ℓ as in Theorem 4.1. With probability at least $1-\delta/2$, for all $h \in \mathcal{H}$ uniformly:*

$$\mathsf{P}\,\ell(\cdot, h) \leq \mathsf{P}_{\mathbf{z}}\,\ell(\cdot, h) + L\mathcal{R}_m(\mathcal{H}) + B\sqrt{\frac{\log(2/\delta)}{2m}}.$$

Proof sketch From the bounded differences inequality (Theorem 2.11), with probability at least $1-\delta/2$ the random quantity $\sup_{h\in\mathcal{H}} \mathsf{P}\,\ell(\cdot, h) - \mathsf{P}_{\mathbf{z}}$ is upper bounded by the its expectation plus $B\sqrt{\frac{\log(2/\delta)}{2m}}$. From symmetrization, $\mathsf{E}\sup_{h\in\mathcal{H}} \mathsf{P}\,\ell(\cdot, h) - \mathsf{P}_{\mathbf{z}}$ is upper bounded by $\mathcal{R}_m(\{\ell(\cdot, h) : h \in \mathcal{H}\})$, which from Meir and Zhang (2003, Theorem 7) is upper bounded by $L\mathcal{R}_m(\mathcal{H})$. ■

In particular, with high probability the true risk of the empirical risk minimizer is not much larger than its empirical risk. Theorem 4.1 now follows from Lemmas 4.2 and 4.3 and a union bound over $\gamma \in \Gamma := \{0, 1/T, 2/T, \ldots, \lceil B \rceil\}$; note that mapping the observed maximum empirical risk γ to $\min\{\gamma' \in \Gamma \mid \gamma \leq \gamma'\}$ picks up the additional $\frac{1}{T}$ term in (4.5).

In the next section, we introduce a loss-compositional paradigm for multi-task learning which includes as special cases minimax MTL and classical MTL.

4.3 A generalized loss-compositional paradigm for MTL

The paradigm can benefit from a bit of notation. Given a set of T tasks, we represent the empirical risk for hypothesis $h_t \in \mathcal{H}$ ($\in \mathbb{H}$) on task $t \in [T]$ as $\hat{\ell}_t(h_t) := \sum_{i=1}^m \ell(y_i^{(t)}, h_t(x_i^{(t)}))$. Additionally define a set of hypotheses for multiple tasks $\mathbf{h} := (h_1, \ldots, h_T) \in \mathcal{H}^T$ and the vector of empirical risks $\hat{\boldsymbol{\ell}}(\mathbf{h}) := (\hat{\ell}_1(h_1), \ldots, \hat{\ell}_T(h_T))$.

With this notation set, the proposed loss-compositional paradigm encompasses any regularized minimization of a (typically convex) function $\phi : \mathbb{R}_+^T \to \mathbb{R}_+$ of the empirical risks:

$$\inf_{\mathcal{H}\in\mathbb{H}} \inf_{\mathbf{h}\in\mathcal{H}^T} \phi\big(\hat{\boldsymbol{\ell}}(\mathbf{h})\big) + \underline{\Omega}\big((\mathcal{H},\mathbf{h})\big), \tag{4.6}$$

where $\underline{\Omega}(\cdot) : \mathbb{H} \times \cup_{\mathcal{H}\in\mathbb{H}}\mathcal{H}^T \to \mathbb{R}_+$ is a regularizer.

$\boldsymbol{\ell_p}$ MTL. One notable specialization that is still quite general is the case when ϕ is an ℓ_p-norm, yielding ℓ_p *MTL*. This subfamily encompasses classical MTL and many new MTL formulations:

- Classical MTL as ℓ_1 *MTL*:

$$\inf_{\mathcal{H}\in\mathbb{H}} \inf_{\mathbf{h}\in\mathcal{H}^T} \frac{1}{T}\sum_{t\in[T]} \hat{\ell}(h_t) + \underline{\Omega}\big((\mathcal{H},\mathbf{h})\big) \quad \equiv \quad \inf_{\mathcal{H}\in\mathbb{H}} \inf_{\mathbf{h}\in\mathcal{H}^T} \frac{1}{T}\|\hat{\boldsymbol{\ell}}(\mathbf{h})\|_1 + \underline{\Omega}\big((\mathcal{H},\mathbf{h})\big).$$

- Minimax MTL as ℓ_∞ *MTL*:

$$\inf_{\mathcal{H}\in\mathbb{H}} \inf_{\mathbf{h}\in\mathcal{H}^T} \max_{t\in[T]} \hat{\ell}(h_t) + \underline{\Omega}\big((\mathcal{H},\mathbf{h})\big) \quad \equiv \quad \inf_{\mathcal{H}\in\mathbb{H}} \inf_{\mathbf{h}\in\mathcal{H}^T} \|\hat{\boldsymbol{\ell}}(\mathbf{h})\|_\infty + \underline{\Omega}\big((\mathcal{H},\mathbf{h})\big).$$

- A new formulation, ℓ_2 *MTL*:

$$\inf_{\mathcal{H}\in\mathbb{H}} \inf_{\mathbf{h}\in\mathcal{H}^T} \Big(\frac{1}{T}\sum_{t\in[T]} \big(\hat{\ell}(h_t)\big)^2\Big)^{1/2} + \underline{\Omega}\big((\mathcal{H},\mathbf{h})\big) \quad \equiv \quad \inf_{\mathcal{H}\in\mathbb{H}} \inf_{\mathbf{h}\in\mathcal{H}^T} \frac{1}{\sqrt{T}}\|\hat{\boldsymbol{\ell}}(\mathbf{h})\|_2 + \underline{\Omega}\big((\mathcal{H},\mathbf{h})\big).$$

A natural question is why one might consider minimizing ℓ_p-norms of the empirical risks vector for $1 < p < \infty$, as in ℓ_2 MTL. The contour of the ℓ_1-norm of the empirical risks evenly trades off empirical risks between different tasks; however, it has been observed that overfitting often happens near the end of learning, rather than the beginning (Roux et al., 2008). More precisely, when the empirical risk is high, the gradient of the empirical risk (taken with respect to the parameter $(\mathcal{H},\mathbf{h})$) is likely to have positive inner product with the gradient of the true risk. Therefore, given a candidate solution with a corresponding vector of empirical risks, a sensible strategy is to take a step in solution space which places more emphasis on tasks with higher empirical risk. This strategy is particularly appropriate when the class of learners has high capacity relative to the amount of available data. This observation sets the foundation for an approach that minimizes norms of the empirical risks.

In this work, we also discuss an interesting subset of the loss-compositional paradigm which does not fit into ℓ_p MTL; this subfamily embodies a continuum of relaxations of minimax MTL.

α-minimax MTL. In some cases, minimizing the maximum loss can exhibit certain disadvantages because the maximum loss is not robust to situations when a small fraction of the tasks are fundamentally harder than the remaining tasks. Consider the case when the empirical risk for each task in this small fraction can not be reduced below a level u. Rather than rigidly minimizing the maximum loss, a more robust alternative is to minimize the maximize loss in a soft way. Intuitively, the idea is to ensure that most tasks have low empirical risk, but a small fraction of tasks are permitted to have higher loss. We formalize this as α-*minimax MTL*, via the relaxed objective:

$$\underset{\mathcal{H}\in\mathbb{H},\mathbf{h}\in\mathcal{H}^T}{\text{minimize}} \quad \min_{b\geq 0}\Big\{ b + \frac{1}{\alpha}\sum_{t\in[T]} \max\{0, \hat{\ell}_t(h_t) - b\}\Big\} + \underline{\Omega}\big((\mathcal{H},\mathbf{h})\big).$$

In the above, ϕ from the loss-compositional paradigm (4.6) is a variational function of the empirical risks vector. The above optimization problem is equivalent to the perhaps more intuitive problem:

$$\begin{aligned} \underset{\mathcal{H}\in\mathbb{H},\mathbf{h}\in\mathcal{H}^T, b\geq 0, \boldsymbol{\xi}\geq 0}{\text{minimize}} \quad & b + \frac{1}{\alpha}\sum_{t\in[T]} \xi_t + \underline{\Omega}\big((\mathcal{H},\mathbf{h})\big) \\ \text{subject to} \quad & \hat{\ell}_t(h_t) \leq b + \xi_t,\ \ t\in[T]. \end{aligned}$$

Here, b plays the role of the relaxed maximum, and each ξ_t's deviation from zero indicates the deviation from the (loosely enforced) maximum. We expect $\boldsymbol{\xi}$ to be sparse.

To help understand how α affects the learning problem, let us consider a few cases:

(1) When $\alpha > T$, the optimal value of b is zero, and the problem is equivalent to classical MTL. To see this, note that for a given candidate solution with $b > 0$ the objective always can be reduced by reducing b by some ε and increasing each ξ_t by the same ε.

(2) Suppose one task is much harder than all the other tasks (e.g. an outlier task), and its empirical risk is separated from the maximum empirical risk of the other tasks by

ρ. Let $1 < \alpha < 2$; now, at the optimal hard maximum solution (where $\boldsymbol{\xi} = 0$), the objective can be reduced by increasing one of the ξ_t's by ρ and decreasing b by ρ. Thus, the objective can focus on minimizing the maximum risk of the set of $T - 1$ easier tasks. In this special setting, this argument can be extended to the more general case $k < \alpha < k + 1$ and k outlier tasks, for $k \in [T]$.

(3) As α approaches 0, we recover the hard maximum case of minimax MTL.

This work focuses on α-minimax MTL with $\alpha = 2/(\lceil 0.1T + 0.5 \rceil^{-1} + \lceil 0.1T + 1.5 \rceil^{-1})$ i.e. the harmonic mean of $\lceil 0.1T + 0.5 \rceil$ and $\lceil 0.1T + 1.5 \rceil$. The reason for this choice is that in the idealized case (2) above, for large T this setting of α makes the relaxed maximum consider all but the hardest 10% of the tasks. We also try the 20% level (i.e. $0.2T$ replacing $0.1T$ in the above).

After the work in this chapter was published (see (Mehta et al., 2012)), we discovered a work by Dekel et al. (2007) that performs multi-task learning in an online setting, using a framework similar to ours. A key difference however is that their analysis focuses on certain online learning guarantees for multi-task learning, whereas our analysis focuses on learning to learn guarantees for the offline learning setting. Additionally, it is unclear whether the α-minimax MTL relaxation fits into the framework of Dekel et al. (2007). This point appears to be an important one, because the experimental results in Section 4.4 indicate that α-minimax MTL often performs better than the other MTL formulations that fall out of our paradigm.

Models. We now provide examples of how specific models fit into this framework. We consider two convex multi-task learning formulations: Evgeniou and Pontil's regularized multi-task learning (the *EP model*) (Evgeniou and Pontil, 2004) and Argyriou, Evgeniou, and Pontil's convex multi-task feature learning (the *AEP model*) (Argyriou et al., 2008). The EP model is a linear model with a shared parameter $v_0 \in \mathbb{R}^d$ and task-specific parameters $v_t \in \mathbb{R}^d$ (for $t \in [T]$). Evgeniou and Pontil presented this model as

$$\min_{v_0, \{v_t\}_{t \in [T]}} \sum_{t \in [T]} \sum_{i=1}^{m} \ell(y_i^{(t)}, \langle v_0 + v_t, x_i^{(t)} \rangle) + \lambda_0 \|v_0\|^2 + \frac{\lambda_1}{T} \sum_{t \in [T]} \|v_t\|^2,$$

for ℓ the hinge loss or squared loss. This can be set in the new paradigm via $\mathbb{H} = \{\mathcal{H}_{v_0} \mid v_0 \in \mathbb{R}^d\}$, $\mathcal{H}_{v_0} = \{h : x \mapsto \langle v_0 + v_t, x\rangle \mid v_t \in \mathbb{R}^d\}$, and $\hat{\ell}_t(h_t) = \frac{1}{m}\sum_{i=1}^m \ell\big(y_i^{(t)}, \langle v_0 + v_t, x_i^{(t)}\rangle\big)$.

The AEP model minimizes the task-wise average loss with the trace norm (nuclear norm) penalty:

$$\min_W \sum_t \sum_{i=1}^m \ell(y_i^{(t)}, \langle W_t, x_i^{(t)}\rangle) + \lambda \|W\|_{\mathsf{tr}},$$

where $\|\cdot\|_{\mathsf{tr}} : W \mapsto \sum_i \sigma_i(W)$ is the trace norm. In the new paradigm, $\mathbb{H}$ is a set where each element is a k-dimensional subspace of linear estimators (for $k \ll d$). Each $h_t = W_t$ in some $\mathcal{H} \in \mathbb{H}$ lives in $\mathcal{H}$'s corresponding low-dimensional subspace. Also, $\hat{\ell}_t(h_t) = \frac{1}{m}\sum_{i=1}^m \ell\big(y_i^{(t)}, \langle h_t, x_i^{(t)}\rangle\big)$.

For easy empirical comparison between the various MTL formulations from the paradigm, at times it will be convenient to use constrained formulations of the EP and AEP model. If the regularized forms are used, a fair comparison of the methods warrants plotting results according to the size of the optimal parameter found (i.e. $\|W\|_{\mathsf{tr}}$ for AEP). For EP, the constrained form is:

$$\begin{aligned} \underset{v_0, \{v_t\}_{t\in[T]}}{\text{minimize}} \quad & \sum_{t\in[T]} \sum_{i=1}^m \ell(y_i^{(t)}, \langle v_0 + v_t, x_i^{(t)}\rangle) \\ \text{subject to} \quad & \|v_0\| \le \tau_0, \\ & \|v_t\| \le \tau_1, \ \ t \in [T]. \end{aligned}$$

For AEP, the constrained form is:

$$\begin{aligned} \underset{W}{\text{minimize}} \quad & \sum_t \sum_{i=1}^m \ell(y_i^{(t)}, \langle W_t, x_i^{(t)}\rangle) \\ \text{subject to} \quad & \|W\|_{\mathsf{tr}} \le r. \end{aligned}$$

4.4 Empirical evaluation

We consider four learning problems; the first three involve regression (MTL model in parentheses):

- A synthetic dataset composed from *two modes* of tasks (EP model),
- The *school* dataset from the Inner London Education Authority (EP model),
- The conjoint analysis *personal computer* ratings dataset [2] (Lenk et al., 1996) (AEP model).

The fourth problem is multi-class classification from the *MNIST* digits dataset (LeCun et al., 1998) with a reduction to multi-task learning using a tournament of pairwise (binary) classifiers. We use the AEP model. Given data, each problem involved a choice of MTL formulation (e.g. minimax MTL), model (EP or AEP), and choice of regularized versus constrained. All the problems were solved with just a few lines of code using `CVX` (Grant and Boyd, 2011, 2008). In this work, we considered convex multi-task learning formulations in order to make clear statements about the optimal solutions attained for various learning problems.

Two modes. The two modes regression problem consists of 50 linear prediction tasks for the first type of task and 5 linear prediction tasks for the second task type. The true parameter for the first task type is a vector μ drawn uniformly from the sphere of radius 5; the true parameter for the second task type is -2μ. Each task is drawn from an isotropic Gaussian with mean taken from the task type and the standard deviation of all dimensions set to σ_{task}. Each data point for each task is drawn from a product of 10 standard normals (so $x_i^{(t)} \in \mathbb{R}^{10}$). The targets are generated according to $\langle W_t, x_i^{(t)} \rangle + \varepsilon_t$, where the ε_t's are iid univariate centered normals with standard deviation σ_{noise}. We fixed τ_0 to a large value (in this case, $\tau_0 = 10$ is sufficient since the mean for the largest task fits into a ball of radius 10) and τ_1 to a small value ($\tau_1 = 2$). We compute the average mean and maximum test error over 100 instances of the 55-task multi-task problem. Each task's training set and test set are 5 and 15 points respectively. The average maximum (mean) test error is the 100-experiment-average of the task-wise maximum (mean) of the ℓ_2 risks. For each LTL experiment, 55 new test tasks were drawn using the same μ as from the training tasks.

[2] This data, collected at the University of Michigan MBA program, generously was provided by Peter Lenk.

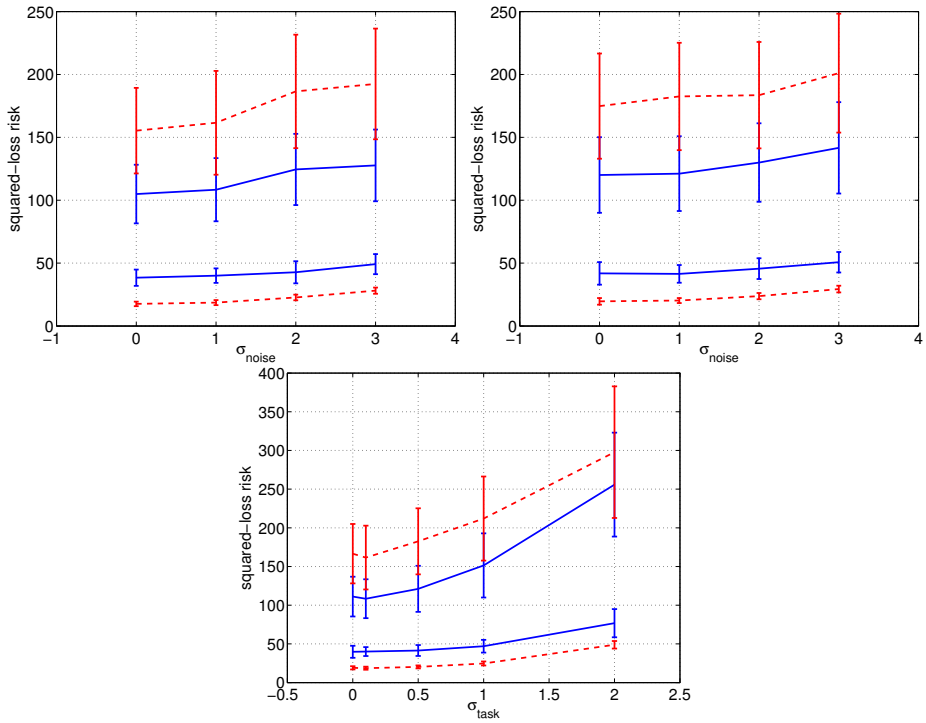


Figure 4.1: Max ℓ_2-risk (Top two lines) and mean ℓ_2-risk (Bottom two lines). At Top Left and Top Right: ℓ_2-risk vs noise level, for $\sigma_{\mathsf{task}} = 0.1$ and $\sigma_{\mathsf{task}} = 0.5$ respectively. At Bottom: ℓ_2-risk vs task variation, for $\sigma_{\mathsf{noise}} = 0.1$. Dashed red is ℓ_1, dashed blue is minimax. Error bars indicate one standard deviation. MTL results (not shown) were similar to LTL results (shown), with MTL-LTL relative difference below 6.8% for all points plotted.

Figure 4.1 shows a tradeoff: when each task group is fairly homogeneous (left and center plots), minimax is better at minimizing the maximum of the test risks while ℓ_1 is better at minimizing the mean of the test risks. As task homogeneity decreases (right plot), the gap in performance closes with respect to the maximum of the test risks and remains roughly the same with respect to the mean.

School. The school dataset has appeared in many previous works (Goldstein, 1991; Bakker and Heskes, 2003; Evgeniou et al., 2007). For brevity we just say the goal is to predict student test scores using certain student-level features. Each school is treated as a separate task. We report both the task-wise maximum of the root mean square error (RMSE) and the task-wise mean of the RMSE (normalized by number of points per task, as in previous works).

The results (see Figure 4.2) demonstrate that when the learner has moderate shared

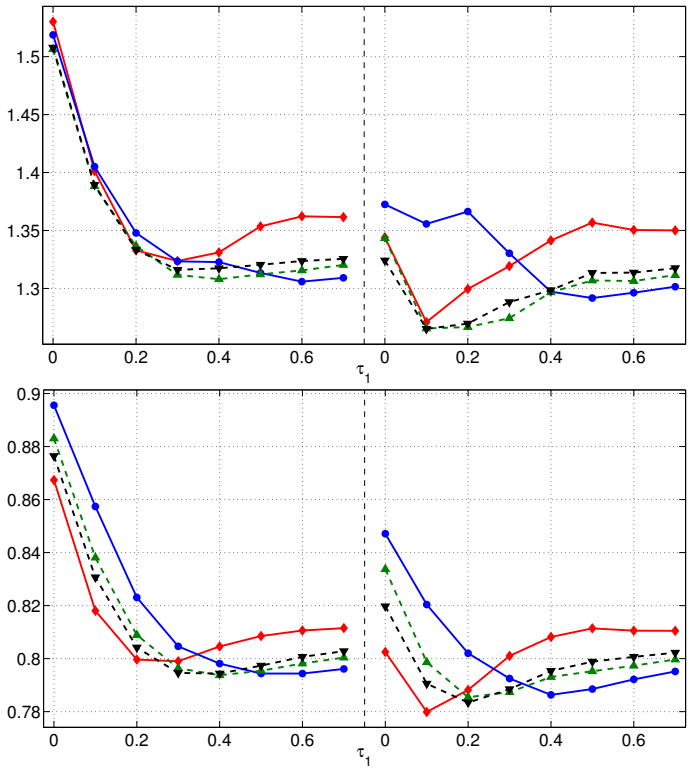


Figure 4.2: Maximum RMSE (Top) and normalized mean RMSE (Bottom) versus task-specific parameter bound τ_1, for shared parameter bound τ_0 fixed. In each figure, Left section is τ_0 is 0.2 and Right section is $\tau_0 = 0.6$. Solid red ♦ is ℓ_1, solid blue ● is minimax, dashed green ▲ is $(0.1T)$-minimax, dashed black ▼ is $(0.2T)$-minimax. The results for ℓ_2 MTL were visually identical to ℓ_1 MTL and hence were not plotted.

capacity τ_0 and high task-specific capacity τ_1, minimax MTL outperforms ℓ_1 MTL for the max objective; additionally, for the max objective in almost all parameter settings $(0.1T)$-minimax and $(0.2T)$-minimax MTL outperform ℓ_1 MTL, and they also outperform minimax MTL when the task-specific capacity τ_1 is not too large. We hypothesize that minimax MTL performs the best in the high$-\tau_1$ regime because stopping learning once the maximum of the empirical risks cannot be improved invokes early stopping and its built-in regularization properties (see e.g. (Murata and Amari, 1999)). Interestingly, for the normalized mean RMSE objective, both minimax relaxations are competitive with ℓ_1 MTL; however, when the shared capacity τ_0 is high (right section, right plot), ℓ_1 MTL performs the best. For high task-specific capacity τ_1, minimax MTL and its relaxations again seem to resist overfitting compared to ℓ_1 MTL.

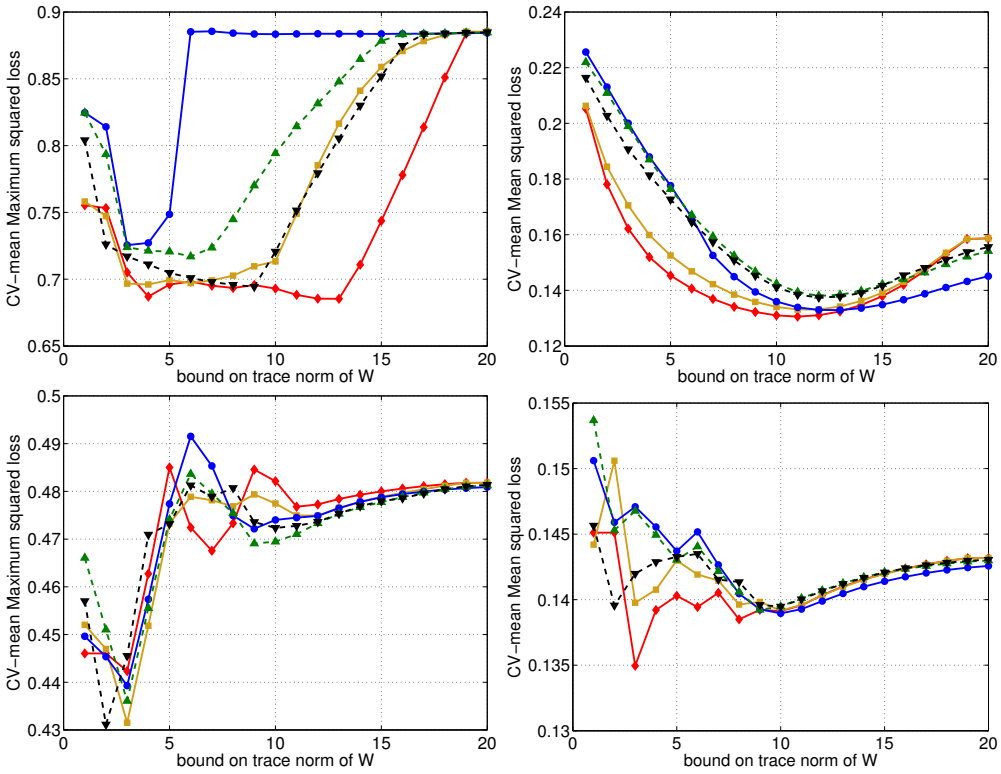


Figure 4.3: MTL (Top) and LTL (Bottom). Maximum ℓ_2 risk (Left) and Mean ℓ_2 risk (Right) vs bound on $\|W\|_{\text{tr}}$. LTL used 10-fold cross-validation (10% of tasks left out in each fold). Solid red ♦ is ℓ_1, solid blue ● is minimax, dashed green ▲ is $(0.1T)$-minimax, dashed black ▼ is $(0.2T)$-minimax, solid gold ■ is ℓ_2.

Personal computer. The personal computer dataset is composed of 189 human subjects each of which rated on a 0-10 scale the same 20 computers (16 training, 4 test). Each computer has 13 binary features (amount of memory, screen size, price, etc.).

The results are shown in Figure 4.3. In the MTL setting, for both the maximum RMSE objective and the mean RMSE objective, ℓ_1 MTL appears to perform the best. When the trace norm of W is high, minimax MTL displays resistance to overfitting and obtains the lowest mean RMSE. In the LTL setting for the maximum RMSE objective, ℓ_2, minimax, and $(0.1T)$-minimax MTL all outperform ℓ_1 MTL. For the mean RMSE, ℓ_1 MTL obtains the lowest risk for almost all parameter settings.

MNIST. The MNIST task is a 10-class problem; we approach it via a reduction to a tournament of 45 binary classifiers trained via the AEP model. The dimensionality was

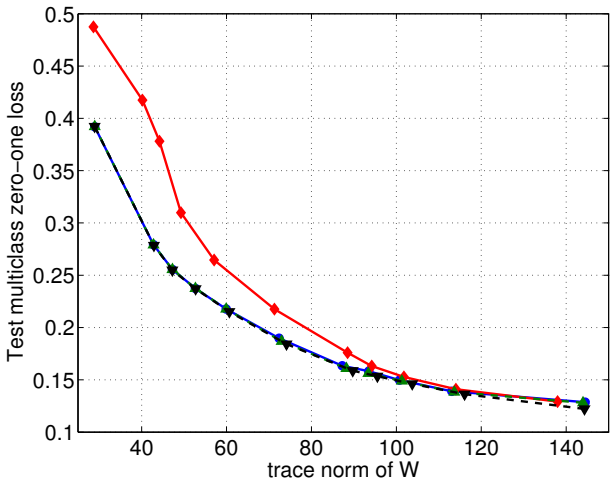


Figure 4.4: Test multiclass 0-1 loss vs $\|W\|_{\mathsf{tr}}$. Solid red is ℓ_1 MTL, solid blue is minimax, dashed green is $(0.1T)$-minimax, dashed black is $(0.2T)$-minimax. Regularized AEP used for speed and trace norm of W's computed, so samples differ per curve.

reduced to 50 using principal component analysis (computed on the full training set), and only the first 2% of each class's training points were used for training.

Intuitively, the performance of the tournament tree of binary classifiers can only be as accurate as its paths, and the accuracy of each path depends on the accuracy of the nodes. Hence, our hypothesis is that minimax MTL should outperform ℓ_1 MTL. The results in Figure 4.4 confirm our hypothesis. Minimax MTL outperforms ℓ_1 MTL when the capacity $\|W\|_{\mathsf{tr}}$ is somewhat limited, with the gap widening as the capacity decreases. Furthermore, at every capacity minimax MTL is competitive with ℓ_1 MTL.

4.5 Discussion

We have established a continuum of formulations for MTL which recovers as special cases classical MTL and the newly formulated minimax MTL. In between these extreme points lies a continuum of relaxed minimax MTL formulations. More generally, we introduced a loss-compositional paradigm that operates on the vector of empirical risks, inducing the additional ℓ_p MTL paradigms. The empirical evaluations indicate that α-minimax MTL at either the 10% or 20% level often outperforms ℓ_1 MTL in terms of the maximum test risk objective and sometimes even in the mean test risk objective. All the minimax or

α-minimax MTL formulations exhibit a built-in safeguard against overfitting in the case of learning with a model that is very complex relative to the available data.

Although efficient algorithms may make the various new MTL learning formulations practical for large problems, a proper effort to develop fast algorithms in this setting would have detracted from the main point of this first study. A good direction for the future is to obtain efficient algorithms for minimax and α-minimax MTL. In fact, such algorithms might have applications beyond MTL and even machine learning. Another area ripe for exploration is to establish more general learning bounds for minimax MTL and to extend these bounds to α-minimax MTL.

CHAPTER 5

SAMPLE VARIANCE PENALIZED META-LEARNING

5.1 Introduction

The choice of representation is a fundamental problem in machine learning. In machine learning pop culture, there is a common sentiment that given the "right" features, learning problems become easy. A natural way by which to judge the performance of a representation is by how it performs on tasks drawn from some environment. For instance, we might conclude that we found good visual features if those features make it easy to solve a variety of visual tasks such as object recognition and scene classification; similarly, if a set of features that describe music can be used linearly to accurately predict individual people's musical preferences, the representation embodying those features probably is a good one.

Given samples from training tasks drawn from the environment, a meta-learner seeks to learn a representation, or hypothesis space, that affords good hypotheses for the training tasks. Once a representation is learned, certain questions arise about the performance of this representation on future tasks, such as:

Q1. What is the true risk of the learned hypothesis space's empirical risk minimizer on a new task, *in expectation* over the draw of the new task?

Q2. What is the *probability* that the true risk of the learned hypothesis space's empirical risk minimizer on a new task exceeds some level ε?

This chapter introduces *sample variance penalized meta-learning*, a new mode of meta-learning designed to perform well on average across all tasks while also obtaining *stable*

performance across tasks. By stable, we mean that the meta-learner's performance on new tasks should have low variance. Since the new tasks are drawn randomly within this framework, the performance is a random variable even after conditioning on the training samples of the training tasks.

In addition to introducing sample variance penalized meta-learning, the core contributions of this chapter are:

- High probability learning theoretic guarantees answering Q1 and Q2.
- A forward stepwise learner for sample variance penalized meta-learning when selecting a representation corresponds to feature selection.
- A simple convex relaxation of sample variance penalized meta-learning which arises from a similar, new convex relaxation of sample variance penalized empirical risk minimization in the single-task setting.

5.2 Meta-learning & sample variance penalization

We begin with a standard single-task learning setup. Consider an input space $\mathcal{X}$, an output space $\mathcal{Y}$, and a joint space $\mathcal{Z} := \mathcal{X} \times \mathcal{Y}$. Let P be a probability measure on $\mathcal{Z}$. In this setup, m labeled points $z_1, \ldots, z_m$ are drawn iid from P, with $z_j = (x_j, y_j)$ for $j \in [m]$. This m-sample is collected into $\mathbf{z}$. Throughout this chapter, $\ell : \mathcal{Y} \times \mathbb{R} \to [0, b]$ will be a loss function that is L-Lipschitz in its second argument, for some constant $b > 0$. To simplify the theoretical results in the next section, we will take $b = 1$ without loss of generality. The case of finite $b > 1$ can be recovered by rescaling. We frequently use the loss-composed function $\ell(\cdot, h) : \mathcal{Y} \times \mathcal{X} \to [0, b]$, defined as the map $(x, y) \mapsto \ell\big(y, h(x)\big)$.

A popular and principled approach in single-task learning is to use regularized empirical risk minimization. For a hypothesis space $\mathcal{H}$ and a regularizer $\Omega : \mathcal{H} \times \mathcal{Z}^m \to \mathbb{R}_+$, define the Ω-regularized empirical risk minimization algorithm as

$$\mathcal{A}_{\mathcal{H}}(\mathbf{z}) = \operatorname*{arg\,min}_{h \in \mathcal{H}} \frac{1}{m} \sum_{j=1}^{m} \ell\big(y_j, h(x_j)\big) + \Omega(h, \mathbf{z}).$$

From this definition, $\mathcal{A}_{\mathcal{H}}$ is a map from a space of labeled m-samples $\mathcal{Z}^m$ to the hypothesis space $\mathcal{H}$. The algorithm $\mathcal{A}_{\mathcal{H}}(\mathbf{z})$ minimizes an empirical objective, but of course its true purpose, lurking in the shadows of optimization, is to find a hypothesis $h \in \mathcal{H}$ that minimizes the true risk $\mathsf{E}_{z\sim \mathsf{P}}\, \ell\big(y, h(x)\big)$.

Meta-learning Suppose that $\mathcal{H}$ is no longer fixed but instead can be selected by a higher-level learning algorithm, the *meta-learner*. The meta-learner will select the hypothesis space from $\mathbb{H}$, a family of hypothesis spaces, or *meta-hypothesis space*. The quality of the true risk minimizer depends intimately on the choice of hypothesis space.

In order to describe the input of the meta-learner, we proceed with a generative model for how these inputs are drawn. Adopting the setting of Baxter (2000), T tasks $\mathsf{P}_1, \ldots, \mathsf{P}_T$ are drawn iid from an *environment*, a distribution over tasks. Here, tasks are identified with probability measures over $\mathcal{Z}$. Next, for each task P_t, an m-*sample* $\mathbf{z}^{(t)} = \big(z_1^{(t)}, \ldots, z_m^{(t)}\big)$ is generated by drawing the examples $z_1^{(t)}, \ldots, z_m^{(t)}$ iid from P_t. Let us collect the m-samples $\mathbf{z}^{(1)}, \ldots, \mathbf{z}^{(T)}$ into a *meta-sample* $\underline{\mathbf{z}}$.

We now can introduce the meta-learner. The meta-learner $\mathbf{A}$ is a map from meta-samples to algorithms, or more formally, $\mathbf{A} : \big(\mathcal{Z}^m\big)^T \to \mathscr{A}$, where $\mathscr{A}$ is a set of learning algorithms. In this chapter, $\mathscr{A}$ will be the set of regularized empirical risk minimizers $\mathscr{A} := \{\mathcal{A}_{\mathcal{H}} : \mathcal{H} \in \mathbb{H}\}$ induced by the family $\mathbb{H}$ of hypothesis spaces.

Learning representations In this chapter, we consider a special but powerful class of meta-hypothesis spaces. For meta-hypothesis spaces in this class, choosing a hypothesis space is equivalent to choosing a *representation*: a set of features arrived at by processing the original features of $\mathcal{X}$ in some way. Each hypothesis will be fully specified by selecting a representation and a linear function acting on this representation. Hence, meta-learning a hypothesis space $\widehat{\mathcal{H}}$ in the meta-hypothesis space $\mathbb{H}$ simply corresponds to learning a set of features. Then, for an individual task, empirical risk minimization over $\widehat{\mathcal{H}}$ corresponds to finding the best hypothesis that is linear in the learned representation. We further constrain the linear hypothesis to have an ℓ_2-norm of at most r.

A natural measure of value for a representation is how well it can be leveraged to

solve predictions tasks from a class of interest. Given the "right" features, learning in these tasks becomes easy. Examples of this are character recognition using sparse codes with a learned dictionary, discriminating faces using hand-crafted geometric features, and predicting people's music preferences via musician-crafted features. A common element of these applications is the presence of many tasks, perhaps even infinitely many drawn from a pool, and the need to learn a representation that performs well for most of them. Hence, meta-learning is a natural lens through which we can study representation learning.

Sample variance penalization The sample variance of a sample $a_1, \ldots, a_m$ is the symmetric function

$$\mathsf{V}_m(a_1, \ldots, a_m) = \frac{1}{m(m-1)} \sum_{1 \le i < j \le m} \frac{(a_i - a_j)^2}{2}.$$

We then can express the sample variance of the loss of hypothesis h with respect to sample $\mathbf{z}$ as

$$\mathsf{V}_{\mathbf{z}}(h) := \mathsf{V}_m\Big(\ell\big(y_1, h(x_1)\big), \ldots, \ell\big(y_m, h(x_m)\big)\Big).$$

Similarly, the sample variance of $\mathcal{A}_{\mathcal{H}}$ on $\underline{\mathbf{z}}$, i.e. the sample variance of empirical risk minimization over $\mathcal{H}$ on meta-sample $\underline{\mathbf{z}}$, is

$$\mathsf{V}_{\underline{\mathbf{z}}}(\mathcal{A}_{\mathcal{H}}) := \mathsf{V}_T\Big(\mathsf{P}_{\mathbf{z}^{(1)}}\, \ell\big(\cdot, \mathcal{A}_{\mathcal{H}}(\mathbf{z}^{(1)})\big), \ldots, \mathsf{P}_{\mathbf{z}^{(T)}}\, \ell\big(\cdot, \mathcal{A}_{\mathcal{H}}(\mathbf{z}^{(T)})\big)\Big).$$

5.3 Learning guarantees

Notation Before embarking on a tour of a theoretical landscape, it will pay off to equip ourselves with some notation. A probability measure P on $\mathcal{Z}$ operates on functions with domain $\mathcal{Z}$ as $\mathsf{P}\, f = \mathsf{E}_{z \sim \mathsf{P}}\, f(z)$. We denote the empirical measure associated with some sample $\mathbf{z}$ as $\mathsf{P}_{\mathbf{z}}$, which is defined as $\mathsf{P}_{\mathbf{z}} := \frac{1}{m} \sum_{j=1}^{m} \delta_{z_j}$ for δ_z the Dirac measure concentrated at $z \in \mathcal{Z}$. The empirical measure operates as $\mathsf{P}_{\mathbf{z}}\, f = \frac{1}{m} \sum_{j=1}^{m} f(z_j)$. In the meta-learning setting, the environment is specified by a probability measure Q over tasks, the tasks themselves being

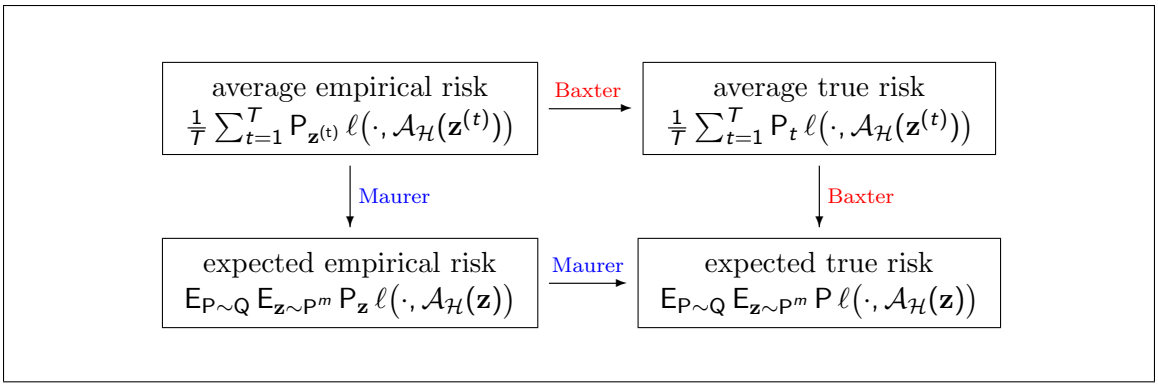


Figure 5.1: Commutative diagram showing different strategies for bounding transfer risk in terms of empirical risk.

identified with probability measures. Formally, Q will be a probability measure on a space $\mathscr{P}$ of probability measures on $\mathcal{Z}$. As mentioned previously, the theoretical results in this section assume that the loss function ℓ is a map from $\mathcal{Y} \times \mathbb{R}$ into $[0, 1]$, and it is L-Lipschitz in its second argument.

Risk bounds The *transfer risk* of an algorithm $\mathcal{A}_{\mathcal{H}}$ (Maurer, 2005) is

$$\mathsf{E}_{\mathsf{P}\sim\mathsf{Q}}\, \mathsf{E}_{\mathbf{z}\sim\mathsf{P}^m}\, \mathsf{P}\, \ell\big(\cdot, \mathcal{A}_{\mathcal{H}}\big). \tag{5.1}$$

The objective of meta-learning is to select a hypothesis space that minimizes the transfer risk. In practice the environment Q is unknown and a meta-learner only has access to a meta-sample $\underline{\mathbf{z}}$, consisting of $\mathbf{z}^{(1)}, \ldots, \mathbf{z}^{(T)}$. The most direct way to attempt to minimize the transfer risk given the meta-sample is to use a meta-learner that performs regularized empirical risk minimization via the objective

$$\min_{\mathcal{H}\in\mathbb{H}} \frac{1}{T}\sum_{t=1}^{T} \mathsf{P}_{\mathbf{z}^{(t)}}\, \ell\big(\cdot, \mathcal{A}_{\mathcal{H}}(\mathbf{z}^{(t)})\big) + \underline{\Omega}(\mathcal{H}, \underline{\mathbf{z}}),$$

where $\underline{\Omega}(\cdot) : \mathbb{H} \times (\mathcal{Z}^m)^T \to \mathbb{R}_+$ is a regularizer that typically does not depend on the meta-sample $\underline{\mathbf{z}}$. However, since our main purpose is to minimize the mean of the empirical risks with a penalty on the sample variance of the empirical risks, in this work $\underline{\Omega}(\cdot)$ will depend intimately on $\underline{\mathbf{z}}$.

In particular, the regularizer in play will be a function of the empirical risks of the empirical risk minimizers $\mathcal{A}_{\mathcal{H}}(\mathbf{z}^{(1)}), \ldots, \mathcal{A}_{\mathcal{H}}(\mathbf{z}^{(T)})$ on their respective m-samples. A critical point, and a seemingly devastating one, is that the regularizer is algorithmic in that it depends on the empirical risk minimizers of $\mathcal{H}$ on the m-samples in the meta-sample. We will expand on this point in Section 5.5.

We are now equipped enough to state the theoretical goals of this chapter. First, we seek upper confidence bounds on the transfer risk by bounding

$$\mathsf{E}_{\mathsf{P}\sim\mathsf{Q}}\, \mathsf{E}_{\mathbf{z}\sim\mathsf{P}^m}\, \mathsf{P}\, \ell\big(y, \mathcal{A}_{\mathcal{H}}(\mathbf{z})\big) - \frac{1}{T}\sum_{t=1}^{T} \mathsf{P}_{\mathbf{z}^{(t)}}\, \ell\big(\cdot, \mathcal{A}_{\mathcal{H}}(\mathbf{z}^{(t)})\big) \tag{5.2}$$

simultaneously for all $\mathcal{H} \in \mathbb{H}$; however, rather than seeking uniform bounds, we seek algorithm-and-data-dependent bounds so as to attain higher resolution when the learned representation is stable across the observed tasks.

For the second goal, consider a second sample of $\widetilde{T}$ test tasks, $\tilde{\mathsf{P}}_1, \ldots, \tilde{\mathsf{P}}_{\widetilde{T}}$, drawn iid from Q, with respective m-samples $\tilde{\mathbf{z}}^{(1)}, \ldots, \tilde{\mathbf{z}}^{(\widetilde{T})}$. For the representation learned from $\underline{\mathbf{z}}$, we wish to obtain tail bounds on the average true risk of empirical risk minimization (induced by the learned representation) on the test tasks. This will be accomplished by obtaining tail bounds on the quantity

$$\frac{1}{\widetilde{T}}\sum_{s=1}^{\widetilde{T}} \tilde{\mathsf{P}}_s \ell\big(y, \mathcal{A}_{\mathcal{H}}(\tilde{\mathbf{z}}^{(s)})\big) - \frac{1}{T}\sum_{t=1}^{T} \mathsf{P}_{\mathbf{z}^{(t)}}\, \ell\big(\cdot, \mathcal{A}_{\mathcal{H}}(\mathbf{z}^{(t)})\big). \tag{5.3}$$

We will show Chebyshev-type tail bounds by exploiting the sample variance of the empirical risks to arrive at an empirical Chebyshev bound.

We first discuss bounding the transfer risk and then address the relatively easier Chebyshev-type bound. Two strategies previously have been applied to bounding the transfer risk in meta-learning. The first strategy decomposes (5.2) into two large deviation bounds:

- a bound on the deviation between the average empirical risk (the right-most term of (5.2)) and the average true risk $\frac{1}{T}\sum_{t=1}^{T} \mathsf{P}_t\, \ell\big(\cdot, \mathcal{A}_{\mathcal{H}}(\mathbf{z}^{(t)})\big)$, and
- a bound on the deviation between the average true risk and the transfer risk (5.1).

Baxter (2000) previously used a similar approach, except rather than using the transfer risk of empirical risk minimization in the left term of (5.2), he considered the expected Bayes risk $\mathsf{E}_{P\sim Q}\inf_{h\in\mathcal{H}} P\,\ell(\cdot, h)$. The second strategy also decomposes (5.2) into two large deviation bounds:

- a bound on the deviation between the average empirical risk and the expected empirical risk $\mathsf{E}_{P\sim Q}\,\mathsf{E}_{\mathbf{z}\sim P^m}\,P_{\mathbf{z}}\,\ell\big(\cdot, \mathcal{A}_{\mathcal{H}}\big)$, and
- a bound on the deviation between the expected empirical risk and the transfer risk.

Maurer (2009) and Maurer et al. (2012, see proof of Theorem 2)[1] used this strategy in a setting where the representation is induced by a choice of linear preprocessor for the original features.

These two strategies are shown in the commutative diagram in Figure 5.1. The second strategy seems more attractive for our goal because we can incorporate the sample variance of the tasks' empirical risks when going from the average empirical risk to the expected empirical risk. We therefore will employ this strategy.

In order to state the main theoretical results, we need to introduce the conditional Rademacher complexity and uniform Rademacher complexity of a function class $\mathcal{F}$. Let $\sigma_1, \ldots, \sigma_m$ be iid Rademacher random variables (uniformly distributed on $\{-1, 1\}$). The conditional Rademacher complexity of a function class $\mathcal{H}$ with respect to an m-sample $\mathbf{x}$ is defined as

$$\mathcal{R}_{m|\mathbf{z}}(\mathcal{H}) := \mathsf{E}_\sigma \sup_{h\in\mathcal{H}} \frac{2}{m}\sum_{j=1}^{m} \sigma_j h(x_j),$$

and the uniform Rademacher complexity of $\mathcal{H}$ is defined as

$$\overline{\mathcal{R}}_m(\mathcal{H}) := \sup_{\mathbf{x}\in\mathcal{X}^m} \mathcal{R}_{m|\mathbf{x}}(\mathcal{H}).$$

We now present bounds for two settings, the case of finite $\mathbb{H}$ and an extension for an important subclass of infinite $\mathbb{H}$'s.

[1] Maurer et al. (2012) actually bound the deviation between the transfer risk of the empirically optimal $\mathcal{H}$ and the Bayes transfer risk $\inf_{\mathcal{H}\in\mathbb{H}} \mathsf{E}_{P\sim Q}\inf_{h\in\mathcal{H}} P\,\ell\big(\cdot, h\big)$, but in the process they essentially follow the second strategy mentioned above.

Theorem 5.1. *Let $\mathbb{H}$ be finite. With probability at least $1-\delta$ over the meta-sample $\underline{\mathbf{z}} = (\mathbf{z}^{(1)}, \ldots, \mathbf{z}^{(T)})$, all $\mathcal{H} \in \mathbb{H}$ satisfy*

$$\mathsf{E}_{\mathsf{P}\sim\mathsf{Q}}\, \mathsf{E}_{\mathbf{z}\sim\mathsf{P}^m}\, \mathsf{P}\, \ell\big(\cdot, \mathcal{A}_{\mathcal{H}}(\mathbf{z})\big) - \frac{1}{T}\sum_{t=1}^{T} \mathsf{P}_{\mathbf{z}^{(t)}}\, \ell\big(\cdot, \mathcal{A}_{\mathcal{H}}(\mathbf{z}^{(t)})\big) \\ \leq L \max_{\mathcal{H}\in\mathbb{H}} \overline{\mathcal{R}}_m(\mathcal{H}) + \sqrt{\frac{2\,\mathsf{V}_{\underline{\mathbf{z}}}(\mathcal{A}_{\mathcal{H}})\ln\frac{2|\mathbb{H}|}{\delta}}{T}} + \frac{7\ln\frac{2|\mathbb{H}|}{\delta}}{3(T-1)}.$$

We now consider a powerful and important class of infinite families of representations. In this class, $\mathbb{H}$ is indexed by a parameter θ living in some metric space $(\Theta, \|\cdot\|)$; furthermore, a choice $\theta \in \Theta$ gives rise to hypothesis space $\mathcal{H}_\theta$ using a representation induced by a preprocessor $\varphi_\theta : \mathcal{X} \to \mathbb{R}^k$. Finally, we assume that the class of feature maps $\{\varphi_\theta : \theta \in \Theta\}$ satisfies

$$\|\theta - \theta'\| \leq \varepsilon \Rightarrow \sup_{x\in\mathcal{X}} \|\varphi_\theta(x) - \varphi_{\theta'}(x)\| \leq C\varepsilon, \tag{5.4}$$

for some constant C. The utility of (5.4) is that it implies that the ε-net for Θ is a $(C\varepsilon)$-net for the space of feature maps $\{\varphi_\theta : \theta \in \Theta\}$, as measured in $\sup$-norm over $\mathcal{X}$. We will use $\mathcal{N}(\Theta, \varepsilon)$ to indicate the minimum cardinality of an optimal ε-net for Θ (in the norm $\|\cdot\|$).

Theorem 5.2. *Let $\mathbb{H}$ be metrizable via a metric space Θ, let a choice of θ correspond to a choice of feature map $\varphi_\theta : \mathcal{X} \to \mathbb{R}^k$, and let φ_θ be C-Lipschitz in θ as in (5.4). With probability at least $1-\delta$ over the meta-sample $\underline{\mathbf{z}} = (\mathbf{z}^{(1)}, \ldots, \mathbf{z}^{(T)})$, all $\mathcal{H} \in \mathbb{H}$ satisfy*

$$\mathsf{E}_{\mathsf{P}\sim\mathsf{Q}}\, \mathsf{E}_{\mathbf{z}\sim\mathsf{P}^m}\, \mathsf{P}\, \ell\big(\cdot, \mathcal{A}_{\mathcal{H}}(\mathbf{z})\big) - \frac{1}{T}\sum_{t=1}^{T} \mathsf{P}_{\mathbf{z}^{(t)}}\, \ell\big(\cdot, \mathcal{A}_{\mathcal{H}}(\mathbf{z}^{(t)})\big) \\ \leq L \sup_{\mathcal{H}\in\mathbb{H}} \overline{\mathcal{R}}_m(\mathcal{H}) + \sqrt{\frac{2\,\mathsf{V}_{\underline{\mathbf{z}}}(\mathcal{A}_{\mathcal{H}})\ln\frac{2\mathcal{N}(\Theta,\frac{1}{T})}{\delta}}{T}} + \frac{7\ln\frac{2\mathcal{N}(\Theta,\frac{1}{T})}{\delta}}{3(T-1)} + \frac{2LCr}{T}.$$

Finally, the following result provides a tail bound for the average risk on a collection of $\widetilde{T}$ tasks drawn in the future:

Theorem 5.3. *Let $\mathbb{H}$ be as in Theorem 5.2. Let $\tilde{\mathsf{P}}_1, \ldots, \tilde{\mathsf{P}}_{\widetilde{T}}$ be drawn iid from Q. The*

probability that

$$\left| \frac{1}{\widetilde{T}} \sum_{s=1}^{\widetilde{T}} \tilde{\mathsf{P}}_s \ell\big(\cdot, \mathcal{A}_{\mathcal{H}}(\tilde{\mathbf{z}}^{(s)})\big) - \frac{1}{T} \sum_{t=1}^{T} \mathsf{P}_{\mathbf{z}^{(t)}}\, \ell\big(\cdot, \mathcal{A}_{\mathcal{H}}(\mathbf{z}^{(t)})\big) \right| > \alpha + L \sup_{\mathcal{H} \in \mathbb{H}} \overline{\mathcal{R}}_m(\mathcal{H}) + \sqrt{\frac{2\, \mathsf{V}_{\underline{\mathbf{z}}}(\mathcal{A}_{\mathcal{H}}) \ln \frac{2\mathcal{N}(\Theta, \frac{1}{T})}{\delta}}{T}} + \frac{7 \ln \frac{2\mathcal{N}(\Theta, \frac{1}{T})}{\delta}}{3(T-1)} + \frac{2LCr}{T}$$

is at most

$$\frac{\mathsf{V}_{\underline{\mathbf{z}}}(\mathcal{A}_{\mathcal{H}}) + \sqrt{\frac{2\, \mathsf{V}_{\underline{\mathbf{z}}}(\mathcal{A}_{\mathcal{H}}) \log \frac{\mathcal{N}(\Theta, \frac{1}{T})}{\delta}}{T-1}}}{\widetilde{T} \alpha^2} + 2\delta.$$

Since δ can be chosen to be very small, Theorem 5.3 yields linear decay in $\widetilde{T}$ while also enjoying higher confidence when the sample variance is low. The proof of this empirical Chebyshev bound is straightforward given a bound on the expected empirical risk and the variance of the empirical risk, where expectations are over a random task and a random m-sample drawn from that task.

All three theorems are proved in the next section.

5.4 Proof sketches

We now discuss our strategy for obtaining upper confidence bounds on (5.2) (adopted from Maurer (2009)). First, observe that for any $\mathcal{H} \in \mathbb{H}$:

$$\mathsf{E}_{\mathsf{P} \sim \mathsf{Q}}\, \mathsf{E}_{\mathbf{z} \sim \mathsf{P}^m}\, \mathsf{P}\, \ell\big(\cdot, \mathcal{A}_{\mathcal{H}}(\mathbf{z})\big) - \frac{1}{T} \sum_{t=1}^{T} \mathsf{P}_{\mathbf{z}^{(t)}}\, \ell\big(\cdot, \mathcal{A}_{\mathcal{H}}(\mathbf{z}^{(t)})\big) \leq$$

$$\mathsf{E}_{\mathsf{P} \sim \mathsf{Q}}\, \mathsf{E}_{\mathbf{z} \sim \mathsf{P}^m}\, \mathsf{P}\, \ell\big(\cdot, \mathcal{A}_{\mathcal{H}}(\mathbf{z})\big) - \mathsf{E}_{\mathsf{P} \sim \mathsf{Q}}\, \mathsf{E}_{\mathbf{z} \sim \mathsf{P}^m}\, \mathsf{P}_{\mathbf{z}}\, \ell\big(\cdot, \mathcal{A}_{\mathcal{H}}(\mathbf{z})\big) \tag{5.5}$$

$$+ \mathsf{E}_{\mathsf{P} \sim \mathsf{Q}}\, \mathsf{E}_{\mathbf{z} \sim \mathsf{P}^m}\, \mathsf{P}_{\mathbf{z}}\, \ell\big(\cdot, \mathcal{A}_{\mathcal{H}}(\mathbf{z})\big) - \frac{1}{T} \sum_{t=1}^{T} \mathsf{P}_{\mathbf{z}^{(t)}}\, \ell\big(\cdot, \mathcal{A}_{\mathcal{H}}(\mathbf{z}^{(t)})\big). \tag{5.6}$$

Hence, it is sufficient to bound (5.5) and (5.6) for all $\mathcal{H} \in \mathbb{H}$ with high probability.

Bounding (5.5) Let us first bound (5.5). We first let $\mathcal{H} \in \mathbb{H}$ and P be arbitrary and consider

$$\mathsf{E}_{\mathbf{z}\sim\mathsf{P}^m}\,\mathsf{P}\,\ell\big(\cdot, \mathcal{A}_{\mathcal{H}}(\mathbf{z})\big) - \mathsf{E}_{\mathbf{z}\sim\mathsf{P}^m}\,\mathsf{P}_{\mathbf{z}}\,\ell\big(\cdot, \mathcal{A}_{\mathcal{H}}(\mathbf{z})\big). \tag{5.7}$$

Let $\mathbf{z}'$ be an independent copy of $\mathbf{z}$. This expression is bounded above by

$$\begin{aligned}
&\mathsf{E}_{\mathbf{z}\sim\mathsf{P}^m} \sup_{h\in\mathcal{H}} \big(\mathsf{P} - \mathsf{P}_{\mathbf{z}})\ell\big(\cdot, \mathcal{A}_{\mathcal{H}}(\mathbf{z})\big)\big) \\
&= \mathsf{E}_{\mathbf{z}\sim\mathsf{P}^m} \sup_{h\in\mathcal{H}} \mathsf{E}_{\mathbf{z}'\sim\mathsf{P}^m} \frac{1}{m}\sum_{j=1}^{m}\Big(\ell\big(y_j', h(x_j')\big) - \ell\big(y_j, h(x_j)\big)\Big) \\
&\le \mathsf{E}_{\mathbf{z}\sim\mathsf{P}^m}\,\mathsf{E}_{\mathbf{z}'\sim\mathsf{P}^m} \sup_{h\in\mathcal{H}} \frac{1}{m}\sum_{j=1}^{m}\Big(\ell\big(y_j', h(x_j')\big) - \ell\big(y_j, h(x_j)\big)\Big),
\end{aligned}$$

where the last step is due to Jensen's inequality. For iid Rademacher random variables $\sigma_1, \ldots, \sigma_m$ (uniformly distributed on $\{-1, 1\}$), the above is equal to

$$\begin{aligned}
&\mathsf{E}_{\boldsymbol{\sigma}}\,\mathsf{E}_{\substack{\mathbf{z}\sim\mathsf{P}^m \\ \mathbf{z}'\sim\mathsf{P}^m}} \sup_{h\in\mathcal{H}} \frac{1}{m}\sum_{j=1}^{m}\sigma_j\Big(\ell\big(y_j', h(x_j')\big) - \ell\big(y_j, h(x_j)\big)\Big) \\
&\le \mathsf{E}_{\sigma}\,\mathsf{E}_{\mathbf{z}\sim\mathsf{P}^m} \sup_{h\in\mathcal{H}} \frac{2}{m}\sum_{j=1}^{m}\sigma_j\ell\big(h(x_j), y_j\big),
\end{aligned}$$

where the last step follows from the triangle inequality.

Rewriting the last expression in terms of the conditional Rademacher complexity, we have that (5.7) is upper bounded by

$$\mathsf{E}_{\mathbf{z}\sim\mathsf{P}^m}\,\mathcal{R}_{m|\mathbf{z}}(\{\ell(\cdot, h) : h \in \mathcal{H}\}) \le L\,\mathsf{E}_{\mathbf{z}\sim\mathsf{P}^m}\,\mathcal{R}_{m|\mathbf{z}}(\mathcal{H}),$$

where the inequality follows from the fact that ℓ is L-Lipschitz in its second argument, after which it is possible to apply Theorem 7 of Meir and Zhang (2003) to bound the conditional Rademacher complexity of the loss-composed $\mathcal{H}$ in terms of the conditional Rademacher complexity of $\mathcal{H}$.

Using this result, it follows that (5.5) is bounded by

$$L \sup_{\mathcal{H} \in \mathbb{H}} \mathsf{E}_{\mathsf{P} \sim \mathsf{Q}} \mathsf{E}_{\mathbf{z} \sim \mathsf{P}^m} \mathcal{R}_{m|\mathbf{z}}(\mathcal{H}).$$

Since the uniform Rademacher complexity $\overline{\mathcal{R}}_m(\mathcal{H})$ always upper bounds the conditional Rademacher complexity, it follows that

$$\mathsf{E}_{\mathbf{z} \sim \mathsf{P}^m} \mathsf{P} \, \ell\big(\cdot, \mathcal{A}_{\mathcal{H}}(\mathbf{z})\big) - \mathsf{E}_{\mathbf{z} \sim \mathsf{P}^m} \mathsf{P}_{\mathbf{z}} \, \ell\big(\cdot, \mathcal{A}_{\mathcal{H}}(\mathbf{z})\big) \leq L \sup_{\mathcal{H} \in \mathbb{H}} \overline{\mathcal{R}}_m(\mathcal{H}). \tag{5.8}$$

Bounding (5.6) For now, take a fixed $\mathcal{H} \in \mathbb{H}$. It will be convenient to define the probability measure ρ on task m-samples, defined as

$$\rho(\mathbf{z}) := \mathsf{E}_{\mathsf{P} \sim \mathsf{Q}} \mathsf{P}^m(\mathbf{z}).$$

Rewriting (5.6) as

$$\mathsf{E}_{\mathbf{z} \sim \rho} \mathsf{P}_{\mathbf{z}} \, \ell\big(\cdot, \mathcal{A}_{\mathcal{H}}(\mathbf{z})\big) - \frac{1}{T} \sum_{t=1}^{T} \mathsf{P}_{\mathbf{z}^{(t)}} \, \ell\big(\cdot, \mathcal{A}_{\mathcal{H}}(\mathbf{z}^{(t)})\big),$$

it is now easy to see that we can apply the empirical Bernstein bound of Maurer and Pontil (2009, Theorem 11), yielding that with probability at least $1 - \delta$ over $\underline{\mathbf{z}}$:

$$\mathsf{E}_{\mathbf{z} \sim \rho} \mathsf{P}_{\mathbf{z}} \, \ell\big(\cdot, \mathcal{A}_{\mathcal{H}}(\mathbf{z})\big) - \frac{1}{T} \sum_{t=1}^{T} \mathsf{P}_{\mathbf{z}^{(t)}} \, \ell\big(\cdot, \mathcal{A}_{\mathcal{H}}(\mathbf{z}^{(t)})\big) \leq \sqrt{\frac{2 \, \mathsf{V}_{\underline{\mathbf{z}}}(\mathcal{A}_{\mathcal{H}}) \ln \frac{2}{\delta}}{T}} + \frac{7 \ln \frac{2}{\delta}}{3(T-1)}. \tag{5.9}$$

Proofs of the theorems In the case of finite meta-hypothesis spaces, the bounds on (5.5) and (5.6), provided by (5.8) and (5.9) respectively, along with a union bound over $\mathbb{H}$, yields Theorem 5.1.

For the case of meta-hypothesis spaces as in Theorem 5.2, consider an optimal $\frac{1}{T}$-covering $\mathbb{H}_\varepsilon$ of $\mathbb{H}$ of cardinality $\mathcal{N}(\Theta, \varepsilon)$. Using a union bound over $\mathbb{H}_\varepsilon$ and picking up approximation error at most $\frac{2C_r}{T}$ (recall that we restrict linear hypotheses to norm at most r), Theorem 11 of Maurer and Pontil (2009) can be extended over all of $\mathbb{H}$, yielding Theorem 5.2.

The proof of Theorem 5.3 follows by an application of an empirical version of Chebyshev's inequality followed by single-task concentration. To apply Chebyshev's inequality, we first control the expected empirical risk, $\mathsf{E}_{\mathbf{z}\sim\rho}\,\mathsf{P}_{\mathbf{z}}\,\ell\big(\cdot,\mathcal{A}_{\mathcal{H}}(\mathbf{z})\big)$, and the (true) variance of the empirical risk, which can be written as $\mathsf{E}\,\mathsf{V}_{\underline{\mathbf{z}}}(\mathcal{A}_{\mathcal{H}})$. For the expected empirical risk bound, applying (5.9) with a union bound over $\mathbb{H}_{\varepsilon}$ yields that with probability at least $1-\delta$, all $\mathcal{H}\in\mathbb{H}$ satisfy

$$\mathsf{E}_{\mathbf{z}\sim\rho}\,\mathsf{P}_{\mathbf{z}}\,\ell\big(\cdot,\mathcal{A}_{\mathcal{H}}(\mathbf{z})\big)-\frac{1}{T}\sum_{t=1}^{T}\mathsf{P}_{\mathbf{z}^{(t)}}\,\ell\big(\cdot,\mathcal{A}_{\mathcal{H}}(\mathbf{z}^{(t)})\big)\\ \leq\sqrt{\frac{2\,\mathsf{V}_{\underline{\mathbf{z}}}(\mathcal{A}_{\mathcal{H}})\ln\frac{2\mathcal{N}(\Theta,\frac{1}{T})}{\delta}}{T}}+\frac{7\ln\frac{2\mathcal{N}(\Theta,\frac{1}{T})}{\delta}}{3(T-1)}+\frac{2LCr}{T}. \quad (5.10)$$

We now turn to the variance bound. Maurer and Pontil (2009, Equation 5) showed that if $\mathbf{G}=(G_1,\ldots,G_n)$ is a vector of independent random variables in $[0,1]$, then

$$\mathsf{Pr}\{\mathsf{E}\,\mathsf{V}_{\mathbf{G}}-\mathsf{V}_{\mathbf{G}}\geq s\}\leq\exp\left(\frac{-(n-1)s^2}{2\,\mathsf{V}_{\mathbf{G}}}\right).$$

By inversion, it follows that with probability at least $1-\delta$,

$$\mathsf{E}\,\mathsf{V}_{\mathbf{G}}\leq\mathsf{V}_{\mathbf{G}}+\sqrt{\frac{2\,\mathsf{V}_{\mathbf{G}}\log\frac{1}{\delta}}{n-1}}.$$

Specializing to our setting, we have with probability at least $1-\delta$,

$$\mathsf{E}\,\mathsf{V}_{\underline{\mathbf{z}}}(\mathcal{A}_{\mathcal{H}})\leq\mathsf{V}_{\underline{\mathbf{z}}}(\mathcal{A}_{\mathcal{H}})+\sqrt{\frac{2\,\mathsf{V}_{\underline{\mathbf{z}}}(\mathcal{A}_{\mathcal{H}})\log\frac{1}{\delta}}{T-1}}. \quad (5.11)$$

The proof of Theorem 5.3 is completed by applying Chebyshev's inequality (using the bound (5.10) on the expected empirical risk and the bound (5.11) on the variance) and finally applying the single-task concentration result (5.8).

5.5 Convexity & algorithms

This section focuses on schemes for sample variance penalization. In general, direct sample variance penalization of the empirical risk is non-convex due to the sample variance term; however, when the penalty is sufficiently light, it turns out that the objective admits a convenient convex relaxation.

We first consider the linear single-task setting, both due to its simplicity and because we are not aware of previous results characterizing when sample variance penalized empirical risk minimization is convex in the single-task setting. Let $\mathbf{z}$ be an m-sample and h a hypothesis. Recall that $\mathsf{V}_{\mathbf{z}}(h)$ is the sample variance of the empirical losses. We assume the loss $\ell : \mathcal{Y} \times \mathbb{R} \to [0, b]$ is convex. The optimization problem for sample variance penalized empirical risk minimization is

$$\min_{h \in \mathcal{H}} \frac{1}{m} \sum_{j=1}^{m} \ell\big(y_j, h(x_j)\big) + \lambda \, \mathsf{V}_{\mathbf{z}}(h), \tag{5.12}$$

where $\lambda \geq 0$. This objective is non-convex.

Rather than optimizing this objective, consider an alternative convex relaxation, for $\rho \geq 0$:

$$\min_{h \in \mathcal{H}} \frac{1}{m} \sum_{j=1}^{m} \ell\big(y_j, h(x_j)\big) + \rho \frac{1}{m} \sum_{j=1}^{m} \ell^2\big(y_j, h(x_j)\big). \tag{5.13}$$

A simple exercise in linear algebra yields the identity

$$\frac{1}{m} \sum_{j=1}^{m} \ell^2\big(y_j, h(x_j)\big) = \left(\frac{1}{m} \sum_{j=1}^{m} \ell\big(y_j, h(x_j)\big) \right)^2 + \frac{m}{m-1} \mathsf{V}_{\mathbf{z}}(h),$$

implying that (5.13) is equivalent to

$$\min_{h \in \mathcal{H}} \frac{1}{m} \sum_{j=1}^{m} \ell\big(y_j, h(x_j)\big) + \rho \left(\frac{1}{m} \sum_{j=1}^{m} \ell\big(y_j, h(x_j)\big) \right)^2 + \rho \frac{m}{m-1} \mathsf{V}_{\mathbf{z}}(h). \tag{5.14}$$

For losses in $[0, b]$, it follows that

$$\left(\frac{1}{m}\sum_{j=1}^{m}\ell\big(y_j, h(x_j)\big)\right)^2 \leq b\frac{1}{m}\sum_{j=1}^{m}\ell\big(y_j, h(x_j)\big).$$

Hence, the objective (5.14) places a relatively *heavier* penalty on the sample variance than

$$\min_{h\in\mathcal{H}}\frac{1}{m}\sum_{j=1}^{m}\ell\big(y_j, h(x_j)\big) + \frac{1}{\frac{1}{\rho}+b}\frac{m}{m-1}\mathsf{V}_{\mathbf{z}}(h). \tag{5.15}$$

Taking $\rho \in [0, \infty)$, it follows that for $\lambda \in \left[0, \left(\frac{m}{m-1}\right)\frac{1}{b}\right]$, (5.12) admits the convex lower bound (5.13), and this convex problem places a relatively heavier penalty on the variance than does (5.12).

Extension to meta-learning The ideal objective for sample variance penalized meta-learning is

$$\min_{\mathcal{H}\in\mathbb{H}}\frac{1}{T}\sum_{t=1}^{T}\mathsf{P}_{\mathbf{z}^{(t)}}\,\ell\big(\cdot, \mathcal{A}_{\mathcal{H}}(\mathbf{z}^{(t)})\big) + \lambda\,\mathsf{V}_{\underline{\mathbf{z}}}(\mathcal{A}_{\mathcal{H}}). \tag{5.16}$$

Not only is this problem non-convex, but the sample variance penalty for each hypothesis space $\mathcal{H}$ depends on $\mathcal{H}$'s empirical risk minimizer $\mathcal{A}_{\mathcal{H}}$. Consequently, a naïve multi-task formulation such as

$$\min_{\substack{\mathcal{H}\in\mathbb{H}\\ h_1,\ldots,h_T\in\mathcal{H}}}\frac{1}{T}\sum_{t=1}^{T}\mathsf{P}_{\mathbf{z}^{(t)}}\,\ell\big(\cdot, h_t\big) + \lambda\,\mathsf{V}_T\big(\mathsf{P}_{\mathbf{z}^{(1)}}\,\ell\big(\cdot, h_1\big), \ldots, \mathsf{P}_{\mathbf{z}^{(T)}}\,\ell\big(\cdot, h_T\big)\big) \tag{5.17}$$

does not lead to the desired variance penalization of empirical risk minimization. The issue is that for any fixed $\mathcal{H}$ the $h_1, \ldots, h_T$ that are optimal for (5.17) might not correspond to the empirical risk minimizers $\{\mathcal{A}_{\mathcal{H}}(\mathbf{z}^{(t)})\}_{t\in[T]}$. Therefore, even if the variance term is small for the learned representation $\mathcal{H}$, empirical risk minimization using $\mathcal{H}$ might still have high variance.

The ideal objective (5.16) in fact belongs to a class of optimization problems known as

bilevel programs. When rewritten canonically as a bilevel program, (5.16) takes the form

$$\begin{aligned} \min_{\mathcal{H}\in\mathbb{H}} \quad & \frac{1}{T}\sum_{t=1}^{T} \mathsf{P}_{\mathbf{z}^{(t)}}\,\ell\big(\cdot, h_t\big) + \lambda\, \mathsf{V}_T\big(\mathsf{P}_{\mathbf{z}^{(1)}}\,\ell\big(\cdot, h_1\big), \ldots, \mathsf{P}_{\mathbf{z}^{(T)}}\,\ell\big(\cdot, h_T\big)\big) \\ \text{s.t.} \quad & h_t \in \operatorname*{arg\,min}_{h\in\mathcal{H}} \mathsf{P}_{\mathbf{z}^{(t)}}\,\ell\big(\cdot, h\big),\, t \in [T]. \end{aligned} \tag{5.18}$$

Although there is a good amount of literature on seeking local optima for bilevel programs, at this stage we prefer selecting the regularization parameter to be light enough where the overall problem is convex.

Problem (5.13) can be extended to the meta-learning setting, yielding the objective:

$$\min_{\mathcal{H}\in\mathbb{H}} \frac{1}{T}\sum_{t=1}^{T} \min_{h\in\mathcal{H}} \left\{ \mathsf{P}_{\mathbf{z}^{(t)}}\,\ell\big(\cdot, h\big) + \rho\,\big(\mathsf{P}_{\mathbf{z}^{(t)}}\,\ell\big(\cdot, h\big)\big)^2 \right\}. \tag{5.19}$$

For a fixed $\mathcal{H}$, the learning problem separates into T single-task learning problems of the form $\min_{h\in\mathcal{H}} \mathsf{P}_{\mathbf{z}^{(t)}}\,\ell\big(\cdot, h\big) + \rho\,(\mathsf{P}_{\mathbf{z}^{(t)}})^2$. Furthermore, since each problem's objective monotonically increases with its respective empirical risk, by taking $\mathcal{A}_{\mathcal{H}}$ as empirical risk minimization over $\mathcal{H}$, the objective can be rewritten as

$$\min_{\mathcal{H}\in\mathbb{H}} \frac{1}{T}\sum_{t=1}^{T} \mathsf{P}_{\mathbf{z}^{(t)}}\,\ell\big(\cdot, \mathcal{A}_{\mathcal{H}}(\mathbf{z}^{(t)})\big) + \rho\left(\mathsf{P}_{\mathbf{z}^{(t)}}\,\ell\big(\cdot, \mathcal{A}_{\mathcal{H}}(\mathbf{z}^{(t)})\big)\right)^2. \tag{5.20}$$

Finally, similar to (5.15), this last problem places a relatively heavier penalty on the sample variance than

$$\min_{h\in\mathcal{H}} \frac{1}{T}\sum_{t=1}^{T} \mathsf{P}_{\mathbf{z}^{(t)}}\,\ell\big(\cdot, \mathcal{A}_{\mathcal{H}}(\mathbf{z}^{(t)})\big) + \frac{1}{\frac{1}{\rho}+b}\frac{T}{T-1}\,\mathsf{V}_{\underline{\mathbf{z}}}(\mathcal{A}_{\mathcal{H}}).$$

Algorithm for feature selection representations In some situations, the meta-learning objective (5.16) might not be convex: this can happen either because λ is not sufficiently small or because even standard meta-learning (minimization of the average empirical risk with respect to $\mathcal{H}\in\mathbb{H}$ and $h_1, \ldots, h_T \in \mathcal{H}$) is non-convex. Recalling the comments above on problem (5.17), we cannot simply locally (or even globally, were it possible) optimize (5.17) and hope for a representation with low variance.

In the case where a choice of meta-hypothesis $\mathcal{H} \in \mathbb{H}$ corresponds to feature selection, a simple forward stepwise approach to meta-learning can be used to locally optimize the bilevel program (5.18). Let d be the number of features. If $J \subset [d]$, then $\mathcal{H}_J \in \mathbb{H}$ will be the space of linear hypotheses restricted to use the features indexed by J.

Algorithm 2: Forward Stepwise Meta-Learner.

Input: A meta-sample $\mathbf{z}^{(1)}, \ldots, \mathbf{z}^{(T)}$
begin
  $J = \{\}$
  $v^*_{\text{last}} = \infty$
  while $|J| < d$ **do**
    for $j \in \{1, \ldots, d\} \setminus J$ **do**
      $J' = J \cup \{j\}$
      for $t = 1$ **to** T **do**
        $h_t = \mathcal{A}_{\mathcal{H}_{J'}}(\mathbf{z}^{(t)})$
        $r_t = \mathsf{P}_{\mathbf{z}^{(t)}}\, \ell\big(\cdot, h_t\big)$
      end
      $v_j = \dfrac{1}{T}\displaystyle\sum_{t=1}^{T} r_t + \lambda\sqrt{\mathsf{V}_T(r_1, \ldots, r_T)}$
    end
    $j^* = \underset{1 \le j \le d}{\arg\min}\, v_j$
    $v^* = \underset{1 \le j \le d}{\min}\, v_j$
    if $v^* < v^*_{\text{last}}$ **then**
      $J = J \cup \{j^*\}$
      $v^*_{\text{last}} = v^*$
    else
      return $\mathcal{H}_J$
    end
  end
  return $\mathcal{H}_J$
end

Algorithm 2 contains the forward stepwise algorithm. In the algorithm, the meta-learner iteratively adds one feature at a time to the representation. In each round, the meta-learner computes the value of the sample variance penalized objective for each potential feature addition, and it selects the feature that minimizes the objective. The algorithm terminates once no feature addition improves upon the objective.

One alternatively could frame a forward-and-backward stepwise version, allowing for both feature addition and deletion up until some stopping criterion is met. Such a variant

will not be further developed here.

5.6 Experiments

We employed a very simple two-dimensional model to highlight situations when sample variance penalized meta-learning can perform better than a meta-learner that minimizes the task-average of the empirical risks.

The data for each task is generated as follows. For all tasks, each input point $x_j^{(t)}$ is drawn iid from the uniform distribution on the two-dimensional square $[-1, 1]^2 \subset \mathbb{R}^2$. For each task t, a random vector $w^{(t)}$ in $\mathbb{R}^2$ is drawn by selecting the first component $w_1^{(t)}$ from a univariate normal distribution with mean μ_1 and standard deviation $\sigma^{(w)}$. The second component is set to a constant value μ_2. For the t^{th} task and the j^{th} input point $x_j^{(t)}$, the corresponding output/target was drawn from a univariate normal distribution with mean $\langle w^{(t)}, x_j^{(t)} \rangle$ and standard deviation $\sigma^{(y)}$.

In the experiments, we set the distributional parameters as $(\mu_1, \sigma^{(w)}) = (1.01, 0.2)$, $\mu_2 = 1$, and $\sigma^{(y)} = 0.01$. The task of the meta-learner was to select a single feature from a training meta-sample, after which this feature would be used to learn a single linear regression coefficient for each test task's training sample. The choice of distributional parameters ensured that the meta-learner's optimal strategy for transfer risk minimization was to select the first feature, as it had slightly larger correlation with the output.

We used 100 training tasks, 10 training points per task, and 1000 test tasks, again with 10 training points per task. In addition, 100 test points were drawn from each test task to form an unbiased empirical estimate of the meta-learner's transfer risk.

This random experiment was repeated 10000 times, and the results are plotted in Figure 5.2. The meta-learner that uses empirical risk minimization (henceforth referred to as the ERM meta-learner) expresses much more variance in its test transfer risk than does the the sample variance penalized meta-learner (henceforth the SVP meta-learner). The reason for its higher variance is simple: in many experiments, the ERM meta-learner selects the second feature, whereas selecting the first feature leads to a lower transfer risk. Also, when the first feature is selected, the transfer risk is more heavily concentrated than when the

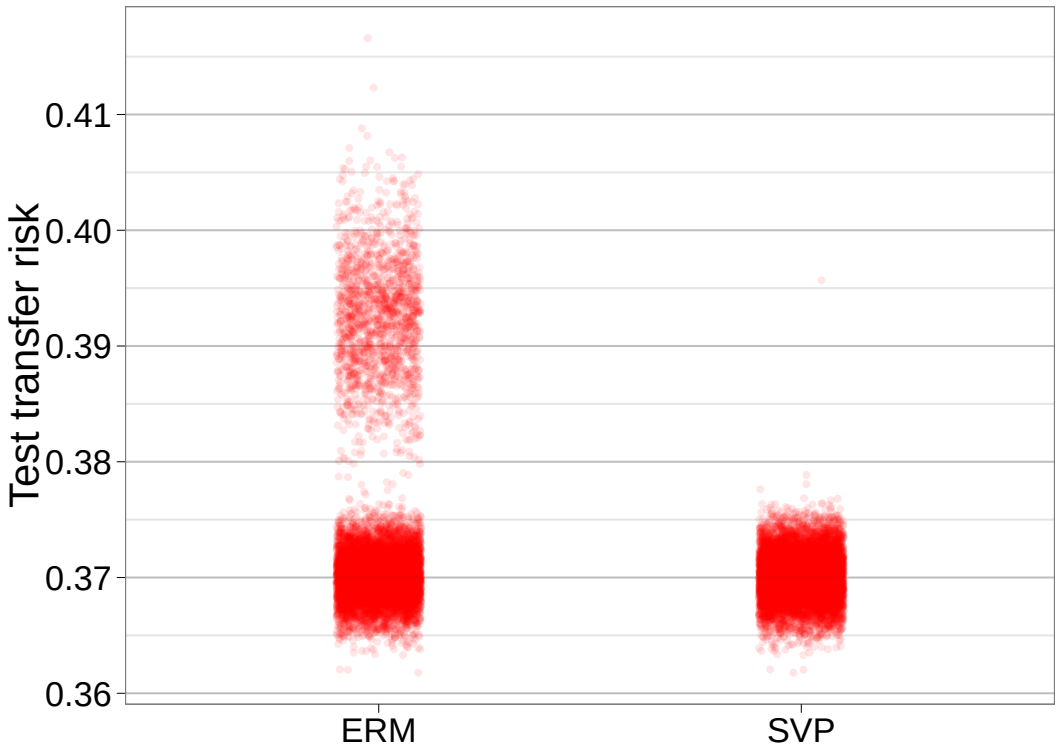


Figure 5.2: The empirically observed test transfer risk on the simulated data, for both empirical risk minimization (ERM) and sample variance penalized meta learning (SVP).

second feature is selected.

The first experiment paints an impressionistic picture for how the ERM and SVP meta-learners differ for a particular number of training tasks. The next experiment compares the empirical rates of decrease of the ERM and SVP meta-learners' transfer risk as the number of training tasks increases. In this experiment, we varied the number of training tasks from 10 to 1000, in increments of 10. For each setting of the number of training tasks, 10000 random experiments were performed. The results of this simulation, with standard errors (clipped below zero), are shown in Figure 5.3a. Rather than plotting the test transfer risk, the excess test transfer risk is instead plotted. The excess test transfer risk is an average like the test transfer risk, except the optimal test transfer risk (exhibited by a meta-learner that always picks up the first feature) is subtracted from the test transfer risk.

For this simulated data, it is evident that sample variance penalization enables a meta-learner to select the correct hypothesis space (i.e. feature) with high probability after having

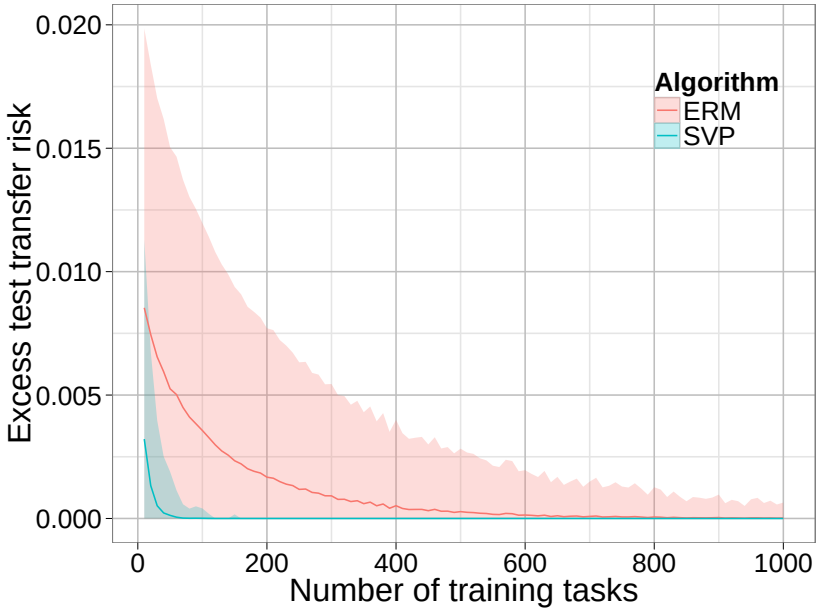
seen only a small number of training tasks. In contrast, the ERM meta-learner requires a much larger number of training tasks before it converges onto the right hypothesis space. Although the differences in the experiment-wise averages of the excess test transfer risks may appear small (see the solid lines), the sample variance of the ERM meta-learner's excess test transfer risk is much higher than that of the SVP meta-learner; consequently, the tail of the ERM meta-learner's excess test transfer risk has a much larger extent.

We conducted a similar version of the second experiment, this time zooming in on the regime of low training task size. In this experiment, we varied the number of training tasks from every integer from 2 to 100. The results of this simulation are shown in Figure 5.3b. This figure shows more clearly that in the small training task regime, the SVP meta-learner already has an excess test transfer risk that rapidly decays to zero. In contrast, the ERM meta-learner exhibits much slower decay.

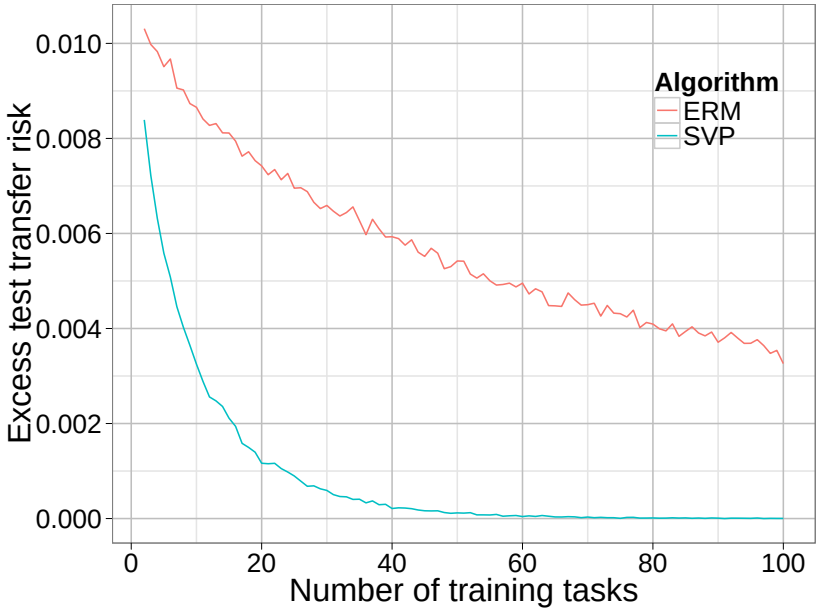
5.7 Discussion

We have presented potentially tighter bounds on the transfer risk of a meta-learner, as well as new high probability tail bounds on the average true risk suffered on new test tasks. These bounds incorporate the observable sample variance of the empirical risks suffered by a meta-learner on training tasks. Moreover, the bounds suggest incorporating sample variance penalization into the meta-learner's objective, yielding sample variance penalized meta-learning. Although this objective is non-convex, it was shown that in a certain regime of sample variance penalization a very natural convex relaxation is possible.

For the special case of meta-learners whose task is feature selection, we presented a forward stepwise meta-learning algorithm. Empirical simulations comparing a meta-learner that minimizes the average empirical risk with a meta-learner that uses sample variance penalization suggest that sample variance penalized meta-learning potentially can offer a faster decrease in the transfer risk as well as a more sharply decreasing tail. For future work, it would be interesting to study the performance of sample variance penalized meta learning, or its convex relaxation, on real-world datasets.



(a) Excess test transfer risk with standard errors.



(b) Excess test transfer risk for low number of training tasks.

Figure 5.3: The top and bottom plots show the excess test transfer risk, computed by comparing against the optimal meta-hypothesis which selects the first feature, for the empirical risk minimization (ERM) meta-learner and the sample variance penalization (SVP) meta-learner. Each point on each line represents the excess test risk averaged over 10,000 experiments. In the top plot, the shaded regions (clipped at zero) represent one standard deviation above and below the mean. The bottom plot zooms in on the regime of a small number of training tasks.

CHAPTER 6

CONCLUSION

This thesis set out with the dual purposes of obtaining a theoretical understanding of learning sparse representations and establishing new representation learning paradigms for multi-task and meta-learning. The main contributions of this thesis were:

- the s-margin, a new, sample-dependent way of measuring the coding stability of the LASSO-based sparse auto-encoder
- the Sparse Coding Stability Theorem, a result on the stability of the Lasso with respect to dictionary perturbations
- the first generalization error bounds for predictive sparse coding, with versions specialized to the overcomplete setting and the high/infinite-dimensional setting
- a new multi-task dictionary model for sparse coding, giving rise to a new unsupervised multi-task sparse coding model and multi-task predictive sparse coding model
- generalization error bounds for these two models showing how multi-task learning helps to reduce the estimation error due to dictionary learning
- a new framework for multi-task and meta-learning, including minimax multi-task learning and the α-relaxed minimax MTL relaxations
- tail bounds on the future test risk, which are directly optimized by minimax MTL
- potentially tighter upper confidence bounds on the transfer risk, using the sample variance of the empirical risks

- tail bounds for the future test risk of a meta-learner, using the sample variance of the empirical risks
- a new framework for meta-learning, sample variance penalized meta-learning

A common theme in this work was to further develop the area of representation learning, both in terms of the kinds of learning frameworks that are used and the kinds of learning guarantees that can be made. In the minimax MTL and sample variance penalized meta-learning works, new theoretical questions on future performance have been asked and first attempts have been made to answer them. Although epistemological reasons prevent any definitive assessments, the field of machine learning appears to be moving quickly and the state of single-task learning (ignoring very interesting develops in frameworks like active learning) seems stable. On the other hand, representation learning naturally lends itself to pooling together information from multiple related tasks, and much less focus has been given to theoretical guarantees about the future performance of meta-learners on new tasks. Such kinds of guarantees may bring us closer to developing more versatile learning agents that can automatically learn the inductive biases necessary for learning new tasks. The hope is that this thesis helps set the stage for future theoretical work on representation learning, including a deeper understanding of the difficulties in learning sparse representations.

Future work With respect to learning sparse representations, interesting directions for future work include a learning theoretic analysis of convex relaxations of sparse coding, such as the convex coding work of Bradley and Bagnell (2009a). Additionally, since many of the multi-task applications for predictive sparse coding arise from the multi-class setting, a proper multi-class treatment of the theory would be very useful for the machine learning and computer vision communities. On the meta-learning of representations front, one weakness is that much of the analysis of meta-learning or learning to learn relies on quite rigid assumptions that all tasks be drawn iid from an environment. The theory for meta-learning would be considerably more useful if it could apply to much less idealized models, including adversarial environments. Minimax MTL was a first step in addressing this problem, but even our theoretical results for minimax MTL adhere to the classical iid version of the

environment. Perhaps the best way to take on adaptive environments is by studying meta-learning in the online setting.

REFERENCES

Ando, R. K. and Zhang, T. (2005). A framework for learning predictive structures from multiple tasks and unlabeled data. *Journal of Machine Learning Research*, 6:1817–1853.

Argyriou, A., Evgeniou, T., and Pontil, M. (2008). Convex multi-task feature learning. *Machine Learning*, 73(3):243–272.

Asif, M. S. and Romberg, J. (2010). On the LASSO and Dantzig selector equivalence. In *Information Sciences and Systems (CISS), 2010 44th Annual Conference on*, pages 1–6. IEEE.

Bakker, B. and Heskes, T. (2003). Task clustering and gating for bayesian multitask learning. *Journal of Machine Learning Research*, 4:83–99.

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

Baxter, J. (2000). A model of inductive bias learning. *Journal of Artificial Intelligence Research*, 12(1):149–198.

Ben-David, S. and Schuller, R. (2003). Exploiting task relatedness for multiple task learning. *Learning Theory and Kernel Machines*, pages 567–580.

Bradley, D. M. and Bagnell, J. A. (2009a). Convex coding. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, pages 83–90. AUAI Press.

Bradley, D. M. and Bagnell, J. A. (2009b). Differentiable sparse coding. *Advances in Neural Information Processing Systems*, 21:113–120.

Caruana, R. (1997). Multitask learning. *Machine learning*, 28(1):41–75.

Cucker, F. and Smale, S. (2002). On the mathematical foundations of learning. *Bulletin of the American Mathematical Society*, 39:1–49.

Daniel, J. W. (1973). Stability of the solution of definite quadratic programs. *Mathematical Programming*, 5(1):41–53.

Dekel, O., Long, P. M., and Singer, Y. (2007). Online learning of multiple tasks with a shared loss. *Journal of Machine Learning Research*, 8:2233–2264.

Donoho, D. L. and Elad, M. (2003). Optimally sparse representation in general (nonorthogonal) dictionaries via 1 minimization. *Proceedings of the National Academy of Sciences*, 100(5):2197–2202.

Evgeniou, T. and Pontil, M. (2004). Regularized multi-task learning. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 109–117. ACM.

Evgeniou, T., Pontil, M., and Toubia, O. (2007). A convex optimization approach to modeling consumer heterogeneity in conjoint estimation. *Marketing Science*, 26(6):805–818.

Fazel, M. (2002). Matrix rank minimization with applications. *Elec Eng Dept Stanford University*, 54:1–130.

Goldstein, H. (1991). Multilevel modelling of survey data. *Journal of the Royal Statistical Society. Series D (The Statistician)*, 40(2):235–244.

Grant, M. C. and Boyd, S. P. (2008). Graph implementations for nonsmooth convex programs. In Blondel, V., Boyd, S., and Kimura, H., editors, *Recent Advances in Learning and Control*, Lecture Notes in Control and Information Sciences, pages 95–110. Springer-Verlag Limited.

Grant, M. C. and Boyd, S. P. (2011). CVX: Matlab software for disciplined convex programming, version 1.21.

Herman, M. A. and Strohmer, T. (2010). General deviants: An analysis of perturbations in compressed sensing. *Selected Topics in Signal Processing, IEEE Journal of*, 4(2):342–349.

Kakade, S. M., Sridharan, K., and Tewari, A. (2009). On the complexity of linear prediction: Risk bounds, margin bounds, and regularization. In Koller, D., Schuurmans, D., Bengio, Y., and Bottou, L., editors, *Advances in Neural Information Processing Systems 21*, pages 793–800. MIT Press.

Lanckriet, G. R. G., Cristianini, N., Bartlett, P., Ghaoui, L. E., and Jordan, M. I. (2004). Learning the kernel matrix with semidefinite programming. *Journal of Machine Learning Research*, 5:27–72.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.

Ledoux, M. and Talagrand, M. (1991). *Probability in Banach Spaces: Isoperimetry and Processes*. Springer.

Lee, H., Battle, A., Raina, R., and Ng, A. Y. (2007). Efficient sparse coding algorithms. *Advances in neural information processing systems*, 19:801.

Lenk, P. J., DeSarbo, W. S., Green, P. E., and Young, M. R. (1996). Hierarchical bayes conjoint analysis: Recovery of partworth heterogeneity from reduced experimental designs. *Marketing Science*, pages 173–191.

Mairal, J., Bach, F., and Ponce, J. (2012). Task-driven dictionary learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(4):791–804.

Mairal, J., Bach, F., Ponce, J., Sapiro, G., and Zisserman, A. (2009). Supervised dictionary learning. In Koller, D., Schuurmans, D., Bengio, Y., and Bottou, L., editors, *Advances in Neural Information Processing Systems 21*, pages 1033–1040. MIT Press.

Maurer, A. (2005). Algorithmic stability and meta-learning. *Journal of Machine Learning Research*, 6:967–994.

Maurer, A. (2006). Bounds for linear multi-task learning. *Journal of Machine Learning Research*, 7:117–139.

Maurer, A. (2009). Transfer bounds for linear feature learning. *Machine learning*, 75(3):327–350.

Maurer, A. and Pontil, M. (2008). Generalization bounds for K-dimensional coding schemes in Hilbert spaces. In *Algorithmic Learning Theory*, pages 79–91. Springer.

Maurer, A. and Pontil, M. (2009). Empirical Bernstein bounds and sample variance penalization. In *Conference on Learning Theory*.

Maurer, A. and Pontil, M. (2010). K-dimensional coding schemes in hilbert spaces. *IEEE Transactions on Information Theory*, 56(11):5839–5846.

Maurer, A., Pontil, M., and Romera-Paredes, B. (2012). Sparse coding for multitask and transfer learning. *arXiv preprint arXiv:1209.0738*.

McDiarmid, C. (1989). On the method of bounded differences. *Surveys in combinatorics*, 141(1):148–188.

Mehta, N. A. and Gray, A. G. (2013). Sparsity-based generalization bounds for predictive sparse coding. In Dasgupta, S. and McAllester, D., editors, *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, ICML '13. JMLR.

Mehta, N. A., Lee, D., and Gray, A. G. (2012). Minimax multi-task learning and a generalized loss-compositional paradigm for MTL. In Bartlett, P., Pereira, F., Burges, C., Bottou, L., and Weinberger, K., editors, *Advances in Neural Information Processing Systems 25*, pages 2159–2167.

Meir, R. and Zhang, T. (2003). Generalization error bounds for bayesian mixture algorithms. *Journal of Machine Learning Research*, 4:839–860.

Mendelson, S. and Philips, P. (2004). On the importance of small coordinate projections. *Journal of Machine Learning Research*, 5:219–238.

Murata, N. and Amari, S. (1999). Statistical analysis of learning dynamics. *Signal Processing*, 74(1):3–28.

Olshausen, B. A. and Field, D. J. (1997). Sparse coding with an overcomplete basis set: A strategy employed by V1? *Vision research*, 37(23):3311–3325.

Osborne, M. R., Presnell, B., and Turlach, B. A. (2000). On the LASSO and its dual. *Journal of Computational and Graphical Statistics*, pages 319–337.

Raina, R., Battle, A., Lee, H., Packer, B., and Ng, A. Y. (2007). Self-taught learning: transfer learning from unlabeled data. In *Proceedings of the 24th international conference on Machine learning*, pages 759–766. ACM.

Ramirez, I., Sprechmann, P., and Sapiro, G. (2010). Classification and clustering via dictionary learning with structured incoherence and shared features. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 3501–3508. IEEE.

Roux, N. L., Manzagol, P.-A., and Bengio, Y. (2008). Topmoumoute online natural gradient algorithm. In Platt, J., Koller, D., Singer, Y., and Roweis, S., editors, *Advances in Neural Information Processing Systems 20*, pages 849–856. MIT Press, Cambridge, MA.

Shalev-Shwartz, S., Singer, Y., Srebro, N., and Cotter, A. (2011). Pegasos: primal estimated sub-gradient solver for svm. *Mathematical Programming*, 127(1):3–30.

Shawe-Taylor, J., L., P., Williamson, R. C., and Anthony, M. (1998). Structural risk minimization over data-dependent hierarchies. *Information Theory, IEEE Transactions on*, 44(5):1926–1940.

Slepian, D. (1962). The one-sided barrier problem for Gaussian noise. *Bell System Technical Journal*, 41(2):463–501.

Steinwart, I. and Christmann, A. (2008). *Support vector machines*. Springer.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288.

Tropp, J. A. (2006). Just relax: Convex programming methods for identifying sparse signals in noise. *Information Theory, IEEE Transactions on*, 52(3):1030–1051.

Vainsencher, D., Mannor, S., and Bruckstein, A. M. (2011). The sample complexity of dictionary learning. *Journal of Machine Learning Research*, 12:3259–3281.

Vapnik, V. N. and Chervonenkis, A. Y. (1968). Uniform convergence of frequencies of occurence of events to their probabilities. In *Dokl. Akad. Nauk SSSR*, volume 181, pages 915–918.

Vidyasagar, M. (2002). *Learning and Generalization with Applications to Neural Networks*. Springer.

Yu, K., Lafferty, J., Zhu, S., and Gong, Y. (2009a). Large-scale collaborative prediction using a nonparametric random effects model. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 1185–1192. ACM.

Yu, K., Zhang, T., and Gong, Y. (2009b). Nonlinear learning using local coordinate coding. In Bengio, Y., Schuurmans, D., Lafferty, J., Williams, C. K. I., and Culotta, A., editors, *Advances in Neural Information Processing Systems 22*, pages 2223–2231. MIT Press.

Zhang, L., Agarwal, D., and Chen, B.-C. (2011). Generalizing matrix factorization through flexible regression priors. In *Proceedings of the fifth ACM conference on Recommender systems*, pages 13–20. ACM.

Zou, H. and Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(2):301–320.

VITA

Nishant Mehta is a human from the 20^{th} century, born in the sunny city of Greenville, South Carolina. He has been interested in artificial intelligence and machine learning since the 11th grade of high school, and around this time he picked up the nickname "Niche." After enjoying studies for a BS in Computer Science at Georgia Tech, Nishant took a brief one-year interlude on the beaches of Melbourne, Florida before returning to complete his PhD at Georgia Tech (see page 1 of this document). Following his PhD studies, an imminent departure to Canberra, Australia awaits for a much-anticipated postdoc with Bob Williamson; Nishant hopes to visit the other Melbourne too.