Balanced Data, Imbalanced Spectra: Unveiling Class Disparities with Spectral Imbalance

Chiraag Kaushik [*1] **Ran Liu** [*1] **Chi-Heng Lin** [1 2] **Amrit Khera** [1] **Matthew Y Jin** [3] **Wenrui Ma** [1]
Vidya Muthukumar [1] **Eva L Dyer** [1]

Abstract

Classification models are expected to perform equally well for different classes, yet in practice, there are often large gaps in their performance. This issue of class bias is widely studied in cases of datasets with *sample imbalance*, but is relatively overlooked in balanced datasets. In this work, we introduce the concept of *spectral imbalance* in features as a potential source for class disparities and study the connections between spectral imbalance and class bias in both theory and practice. To build the connection between spectral imbalance and class gap, we develop a theoretical framework for studying class disparities and derive exact expressions for the per-class error in a high-dimensional mixture model setting. We then study this phenomenon in 11 different state-of-the-art pretrained encoders, and show how our proposed framework can be used to compare the quality of encoders, as well as evaluate and combine *data augmentation* strategies to mitigate the issue. Our work sheds light on the class-dependent effects of learning, and provides new insights into how state-of-the-art pretrained features may have unknown biases that can be diagnosed through their spectra. Code can be found at https://github.com/nerdslab/SpectraImbalance.

1. Introduction

A core objective in machine learning is to build fair classification models that provide good performance irrespective of the class to which the data belongs. In practice, however, models are often biased towards better performance on some classes of the data and may do a poor job on others.

Most studies and approaches for dealing with class bias have focused on *sample imbalance*, or the fact that different classes often have different number of samples (Zhang et al., 2023a). In this case, mechanisms to reweight the loss or rebalance the dataset can be used to correct the sample imbalance (Ren et al., 2020; Zhu et al., 2022). Somewhat surprisingly, *even without sample imbalance* there can still be significant performance gaps across classes, including on state-of-the-art encoders (Ma et al., 2022; 2023). These effects can become more pronounced depending on the types of regularization and/or data augmentation used during training (Balestriero et al., 2022; Kirichenko et al., 2023). Thus, we need a better understanding of the sources of class disparities and new approaches for identifying and mitigating underlying biases in models. This is particularly relevant due to the popularity of foundation and pretrained models, since underlying biases in the pretrained features may impact performance and robustness on downstream tasks.

In this work, we introduce a new framework for studying class-dependent generalization that relies on a concept that we call *spectral imbalance*. The central idea behind this perspective is that differences, or shifts, in the distribution of features across classes could be the source of class biases. We characterize these differences in feature geometry using the distribution of *eigenvalues*, i.e. the *spectrum*, for each class and show, both in theory and in practice, that deviations in spectra across classes will indeed produce interesting effects on the class gap. Our findings complement recent empirical work that connects certain geometric properties of learned features to per-class performance (Ma et al., 2022; 2023) while also providing a formal theoretical connection between spectral imbalance and class gaps in high-dimensional linear models.

First, we develop a theoretical framework for studying this phenomenon and dig deeper into the distinct ways in which spectral imbalance may impact class gap. In particular, we derive *exact asymptotic expressions* for the class-wise error in a Gaussian mixture model with class-specific covari-

[*]Equal contribution [1]Georgia Institute of Technology, Georgia, the USA. [2]Samsung Research. [3]Stanford University, California, the USA. Correspondence to: Ran Liu <rliu361@gatech.edu>, Chiraag Kaushik <ckaushik7@gatech.edu>, Eva L Dyer <evadyer@gatech.edu>, Vidya Muthukumar <vmuthukumar8@gatech.edu>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024.

ances. We use these expressions to uncover different types of spectral imbalance and characterize their consequences for the class gap. This theory provides a strong basis for the investigation of the spectra in per-class generalization.

Next, in extensive empirical investigations, we study the class-dependent spectra for 11 state-of-the-art pretrained encoders (with 6 distinct types of architecture) on ImageNet. Across the board, we observe that the class gap is strongly correlated with the spectral imbalance of the representation space and that through measuring different statistics of this imbalance, we can predict which classes will perform worse than others without actually testing the model. Furthermore, we define a classwise *spectral quantile measure* that we use to diagnose different models, predict which encoders will have smaller class gaps, and also determine smarter augmentation strategies. Our empirical results provide strong evidence that spectral imbalance plays an important role in the characteristics of representation space.

In summary, our major contributions are as follows:

- In Section 2, we introduce the concept of *spectral imbalance* and study how discrepancies in eigenvalues between classes are related to the problem of class-dependent generalization.
- In Section 3, we derive *exact* theoretical expressions for the per-class generalization in a high-dimensional linear mixture-model setting (Theorem 1). The resultant simulations demonstrate the effect of different spectral imbalances on the per-class performance gap.
- In Section 4, we empirically demonstrate that similar phenomena hold in the representation space of pretrained models. Thus, we design a *spectral quantile score*, that accurately measures the amount of 'imbalance' in representation space.
- Finally, in Section 4.4, we provide an initial exploration of how to improve augmentation design to mitigate spectral imbalance, and design a simple ensemble method to combine augmentations and improve performance across classes *without any re-training*.

2. Why spectral imbalance?

Effectively addressing class bias necessitates a deep understanding of the underlying structure of the representation space. One promising avenue is through analyzing the spectral properties of the features corresponding to each class. The spectrum, i.e., the set of eigenvalues of the covariance matrix of the representation, serves as a powerful indicator of the features' intrinsic geometry and dimensionality. It characterizes how variance is distributed across different principal components, offering a window into the structure of the representation space. We thus hypothesize that variations in these spectral properties will provide crucial insights into class-specific performance, potentially revealing biases that are not immediately discernible through traditional measures of imbalance (e.g. sample size per class).

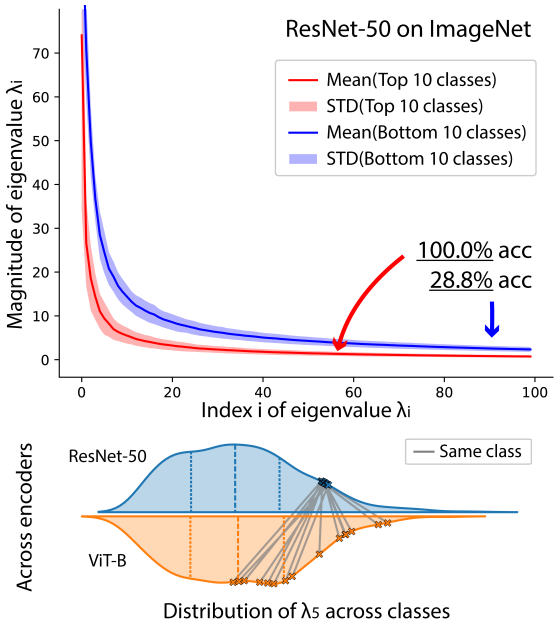

Figure 1. Spectral imbalance in different pretrained models. (A) We plot the spectrum of the top and bottom performing classes and find that classes that achieve lower accuracy have larger eigenvalues. (B) Histogram of the $k = 5$ eigenvalue across classes for two encoders (ResNet-50, ViT-B) with the quartiles indicated. If we look at classes with similarly ranked or "adjacent" eigenvalues (upper quartile) in ResNet-50, we find that they can be mapped to very different positions in the distribution of eigenvalues in another encoder (ViT-B).

We explore in particular the extent to which *spectral imbalance*, or variation in spectral characteristics across different classes, impacts class performance. As a motivating example, in Figure 1, we estimate and examine the spectrum for different classes in a pre-trained ResNet-50 (He et al., 2016). We make two important observations:

- First, we find that the eigenvalue distributions differ significantly between top-performing and worst-performing classes, with top-performing classes exhibiting a uniformly *smaller* spectrum (Figure 1(A)).
- Second, when comparing eigenvalues of different pretrained encoders, we observe that the distribution of the eigenvalues shifts considerably. For example, classes might have similarly ranked eigenvalues in one encoder, but vastly different eigenvalues in another (see Figure 1(B), more in Appendix Figure 4).

Both of these observations motivate the need to understand the differences in class-dependent spectra, and how different sources of spectral imbalance may impact downstream performance.

2.1. Examining the role of the spectrum in class-dependent generalization

To see more concretely how spectral balance across classes can play a role in *class-dependent generalization*, we first consider a simple linear classification setting, where the features follow a 2-class Gaussian mixture model (GMM):

$$\mathbf{x} \mid y \overset{\text{i.i.d.}}{\sim} \mathcal{N}(y\boldsymbol{\theta}^*, \boldsymbol{\Sigma}_y).$$

Here, $\boldsymbol{\theta}^* \in \mathbb{R}^p$ is an unknown signal controlling the class means, $y \in \{-1, 1\}$ is the label, and $\boldsymbol{\Sigma}_y$ are the covariance matrices for each class. For any fixed estimator $\hat{\boldsymbol{\theta}}$, inference is performed as $\hat{y} = \text{sign}(\mathbf{x}^\top \hat{\boldsymbol{\theta}})$. Then, the class-dependent probability-of-error (POE) can be written as:

$$\text{POE}(\hat{\boldsymbol{\theta}}|y) \coloneqq \mathbb{P}\{\text{sign}(\mathbf{x}^\top \hat{\boldsymbol{\theta}}) \neq y | \mathbf{x} \text{ in class } y\} \quad (1)$$

$$= Q\left(\frac{\langle \hat{\boldsymbol{\theta}}, \boldsymbol{\theta}^* \rangle}{\|\boldsymbol{\Sigma}_y^{1/2} \hat{\boldsymbol{\theta}}\|_2}\right) \quad (2)$$

where $Q(\cdot)$ is the Gaussian Q-function (see proof in Appendix A). Fixing the estimator $\hat{\boldsymbol{\theta}}$, we can see that the class dependency of the POE is captured by the term $\|\boldsymbol{\Sigma}_y^{1/2} \hat{\boldsymbol{\theta}}\|_2$. Hence, it is natural to consider differences in this quantity as indicators of how differently an estimator can be expected to perform on the two classes. In particular, an estimator $\hat{\boldsymbol{\theta}}$ will achieve small class gap when $\|\boldsymbol{\Sigma}_{y=1}^{1/2} \hat{\boldsymbol{\theta}}\|_2 \approx \|\boldsymbol{\Sigma}_{y=-1}^{1/2} \hat{\boldsymbol{\theta}}\|_2$, for which a sufficient (but by no means necessary) condition is that $\boldsymbol{\Sigma}_{y=1} \approx \boldsymbol{\Sigma}_{y=-1}$.

As a concrete example, suppose for simplicity that the eigendecompositions of the class covariances are $\boldsymbol{\Sigma}_y = \sum_j \lambda_j^{(y)} \mathbf{v}_j \mathbf{v}_j^T$, i.e., both share the same eigenvectors. Then,

$$\textit{ClassGap}(\hat{\boldsymbol{\theta}}) \coloneqq \left|\text{POE}(\hat{\boldsymbol{\theta}}|y=-1) - \text{POE}(\hat{\boldsymbol{\theta}}|y=1)\right|$$

is positively correlated with the quantity

$$\left|\sum_j (\lambda_j^{(y=-1)} - \lambda_j^{(y=1)}) \langle \hat{\boldsymbol{\theta}}, \mathbf{v}_j \rangle^2\right|$$

From these expressions, we can see that if the eigenvalues of $\boldsymbol{\Sigma}_1$ are uniformly smaller than those of $\boldsymbol{\Sigma}_{-1}$, Class 1 will have smaller generalization error. This conforms with the familiar intuition that classes with smaller variance (i.e., less noise) should be "easier" to identify, as shown in Figure 1. Moreover, the effect of an eigenvalue gap between the two classes is unevenly distributed between different coordinates, depending on the structure of the estimator $\hat{\boldsymbol{\theta}}$.

Of course, in general, the connection between the spectra of features corresponding to a specific class and the per-class generalization is considerably more nuanced, since $\hat{\boldsymbol{\theta}}$ is itself learned from training examples and depends crucially on the properties of the spectra of both classes. Indeed, many recent works on benign overfitting in linear regression and classification (Bartlett et al., 2020; Muthukumar et al., 2021; Cao et al., 2021; Wang & Thrampoulidis, 2021) have shown that the properties of the (overall) data spectrum play a vital role in achieving good overall performance, particularly in the overparameterized regime. In the forthcoming sections, we extend these insights to a fundamental understanding of how the *class-wise spectrum* affects *per-class* performance, and build the framework of spectral imbalance.

3. An exact characterization of the class gap

In this section, we explore the factors that govern class imbalances by studying the case of high-dimensional linear classification in the GMM setting introduced in Section 2, where the mixtures correspond to the classes. The GMM is a natural way to study class covariance differences and provides insights into the types of representations that might result in larger class biases in practice. Here, we provide an exact asymptotic characterization of the per-class generalization error and class gap. Our framework allows us to model nuanced relationships between the eigenvalues of each class covariance, providing a rigorous testbed for studying their effect on per-class generalization.

3.1. Problem formulation

We consider a training scheme where we are given n i.i.d. samples $(\mathbf{x}, y)$ from the GMM setting in Section 2.1, where n_1 (resp., n_{-1}) samples come from Class 1 (resp., Class -1). We study *empirical risk minimization* (ERM) estimators of the form

$$\hat{\boldsymbol{\theta}} \coloneqq \arg\min_{\boldsymbol{\theta} \in \mathbb{R}^p} \frac{1}{n} \sum_{i=1}^{n} \mathcal{L}(y_i \langle \mathbf{x}_i, \boldsymbol{\theta} \rangle) + r\|\boldsymbol{\theta}\|_2^2,$$

where $\mathcal{L}$ is a convex loss function and $r > 0$ is the ridge regularization parameter. Our goal is to characterize the per-class test error of the resultant $\hat{\boldsymbol{\theta}}$, given by Equation (1).

Our theory considers the modern *high-dimensional* setting where both the model complexity and the number of training samples are large. Specifically, we study the per-class error in the asymptotic limit where $n, p \to \infty$ jointly, with the ratios $\frac{n}{p} = \delta$, $\frac{n_y}{n} = \pi_y$ fixed (and clearly, $\pi_1 + \pi_{-1} = 1$).[1] Our theory allows for overparameterization ($\delta < 1$)

[1] We note here that the quantities $\hat{\boldsymbol{\theta}}, \boldsymbol{\Sigma}_y, \boldsymbol{\theta}^*$, and $\text{POE}(\hat{\boldsymbol{\theta}} \mid y)$

or underparameterization ($\delta > 1$), and for any constant regularization parameter $r > 0$.

Notation. The Mahalanobis norm of a vector, defined for a positive semi-definite matrix $\mathbf{H}$ is given by $\|\mathbf{x}\|_{\mathbf{H}} = \sqrt{\mathbf{x}^\top \mathbf{H}\mathbf{x}}$. We also define the following function, which we denote the *Lth order Moreau envelope function* of a given convex function $f : \mathbb{R} \to \mathbb{R}$:

$$\mathcal{M}_f^{(L)}(x_1, ..., x_L; \tau_1, ..., \tau_L) := \min_{v \in \mathbb{R}} f(v) + \sum_{\ell=1}^{L} \frac{1}{2\tau_\ell}(x_\ell - v)^2$$

Note that for $L = 1$ this is the classical Moreau envelope function of f. We will often use the shorthand notation $\mathcal{M}^{(2)}(x_y; \tau_y) = \mathcal{M}^{(2)}(x_1, x_{-1}; \tau_1, \tau_{-1})$. The notation $\xrightarrow{P}$ denotes convergence in probability, $\xrightarrow{\mathcal{W}_2}$ denotes convergence in Wasserstein-2 distance, and the empirical distribution of a random variable a_j is denoted $\frac{1}{p}\sum_{j=1}^{p} \delta(a_j)$.

3.2. Main results

We now state our general asymptotic result for the GMM with spectral imbalance. Our analysis relies on the Convex Gaussian Min-Max Theorem (Thrampoulidis et al., 2018) and is influenced (particularly in the treatment of anisotropic covariances) by (Montanari et al., 2019) and (Taheri et al., 2020), who consider, respectively, max-margin classification and adversarial training in the GMM *without* spectral imbalance. After stating the assumptions and theorem, we briefly outline major differences in the proof technique while deferring the full proof to Appendix A.

Assumption 1. The target signal $\boldsymbol{\theta}^*$ and class covariances $\boldsymbol{\Sigma}_y$ satisfy the following:

1. The $\boldsymbol{\Sigma}_y$ are diagonal.
2. $\|\boldsymbol{\theta}^*\|_{\boldsymbol{\Sigma}_y^{-1}} \to \zeta_y$ and $\|\boldsymbol{\theta}^*\|_2 \to C$ for some constants $\zeta_y, C > 0$.
3. The empirical joint distribution of $(\sqrt{p}\boldsymbol{\theta}_j^*, \lambda_j^{(1)}, \lambda_j^{(-1)})$ converges to some distribution, denoted Π: $\frac{1}{p}\sum_{j=1}^{p} \delta(\sqrt{p}\boldsymbol{\theta}_j^*, \lambda_j^{(1)}, \lambda_j^{(-1)}) \xrightarrow{\mathcal{W}_2} \Pi$.

Due to rotational invariance, the first assumption is for simplicity and can be relaxed to the setting where the covariances are simultaneously diagonalizable. We also note that the last assumption is quite flexible; as we will see in Section 3.3, various choices of Π can reflect very different types of spectral imbalance between the two classes.

Theorem 1. *Let* $G, H_1, H_{-1} \overset{i.i.d.}{\sim} \mathcal{N}(0,1)$ *and* $(T, L_1, L_{-1}) \sim \Pi$. *Under Assumption 1, the per-class POE can be written as a function of scalars* μ_y, α_y, *and* ζ_y*:*

$$POE(\hat{\boldsymbol{\theta}} \mid y) = Q\left(\frac{\langle \hat{\boldsymbol{\theta}}, \boldsymbol{\theta}^* \rangle}{\|\boldsymbol{\Sigma}_y^{1/2}\hat{\boldsymbol{\theta}}\|_2}\right) \xrightarrow{P} Q\left(\frac{\mu_y \zeta_y^2}{\sqrt{\alpha_y^2 + \mu_y^2 \zeta_y^2}}\right),$$

where (μ_y, α_y) *are found in the optimal solution (if it is unique) of the following min-max problem over 12 scalar variables:*

$$\min_{\substack{\alpha_y, \tau_y \geq 0 \\ \mu_y \in \mathbb{R}}} \max_{\substack{\beta_y, \gamma_y \geq 0 \\ \eta_y \in \mathbb{R}}} r\, \mathbb{E}\, \mathcal{M}_{(\cdot)^2}^{(2)}\left(\frac{\alpha_y \beta_y H_y}{\gamma_y \sqrt{\delta L_y}} + \frac{\eta_y \alpha_y T}{\gamma_y \zeta_y^2 L_y}; \frac{\alpha_y r}{\gamma_y L_y}\right)$$
$$+ \sum_{y \in \{-1,1\}} \left[\pi_y\, \mathbb{E}_G\, \mathcal{M}_{\mathcal{L}}^{(1)}\left(\mu_y \zeta_y^2 + \sqrt{\mu_y^2 \zeta_y^2 + \alpha_y^2} G; \frac{\tau_y}{\beta_y}\right)\right.$$
$$\left. - \frac{\eta_y^2 \alpha_y}{2\gamma_y \zeta_y^2} - \frac{\mu_y^2 \gamma_y \zeta_y^2}{2\alpha_y} - \frac{\alpha_y \beta_y^2}{2\delta\gamma_y} + \frac{\beta_y \tau_y}{2} - \frac{\alpha_y \gamma_y}{2} + \eta_y \mu_y\right]. \tag{3}$$

First, we note that the min-max optimization specified in Theorem 1 is straightforward to solve by gradient descent/ascent. Hence, this theorem provides a simple analytical framework to assess the *exact* asymptotic impact of various high dimensional and spectrally-imbalanced GMM settings. Specifically, we can apply the above theorem for different choices of the distribution of features and ground truth (i.e., Π) to characterize the precise impact of various types of spectral imbalance. While these will be the main factors which we study below, the theorem also provides the flexibility to analyze the effect of other parameters like the choice of loss function $\mathcal{L}$ or regularization strength r. While we do not explore this here, the result also allows for comparison between recently proposed alternative performance metrics in class-imbalanced settings, which are functions of the per-class accuracies (Mullick et al., 2020).

A note on the proof technique: The objective function in Equation (3) is quite similar to the min-max program derived for the GMM with equal class covariances in (Taheri et al., 2020); however, there are a few key differences in our proof technique which allow us to generalize that result and obtain a more refined *per-class* characterization of the error. Firstly, we use a *decoupling trick* that introduces two variables (one for each class) for each variable in the objective function of (Taheri et al., 2020). This allows us to isolate the quantities of interest for each class and apply the CGMT separately for each class. Secondly, in contrast to (Taheri et al., 2020), our result features the second order Moreau envelope term, which functions as a way to link the decision variables corresponding to each class.

3.3. Insights on spectral imbalance

We can use the predictions of Theorem 1 to precisely study the impact of various types of spectral imbalance on per-class performance. In Figure 2, we plot the asymptotic predictions of the per-class error and the class gap for three

should be interpreted as sequences indexed by n; however, we will suppress the dependence on n in the notation for clarity.

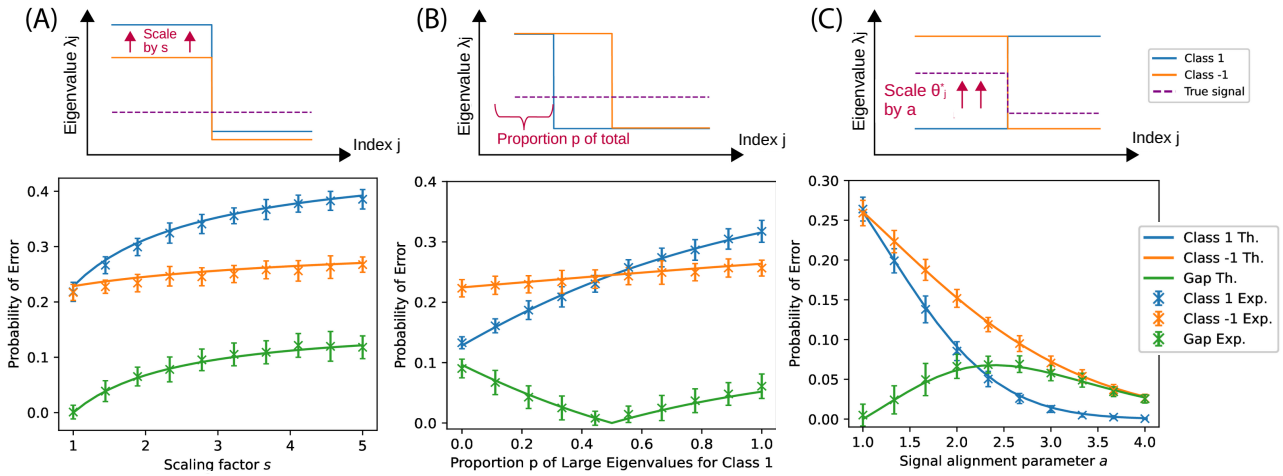

Figure 2. Spectral imbalance in the GMM for the settings in Section 3.3. Top Row: Visualization of the spectra of both classes. Bottom Row: Theoretical predictions of Theorem 1 (solid lines) and numerical simulations (average over 50 trials ± 1 std.) for the per-class error.

different types of spectral imbalance which can arise in GMMs. To isolate the effect of spectral imbalance, the predictions in this section fix the amount of data from each class as identical (i.e. $\pi_y = 0.5$), the parameterization ratio $\delta = 2$, regularization parameter $r = 0.5$ and the loss as the squared hinge loss, $\mathcal{L}(t) = \max(0, 1-t)^2$. In each case, we verify the theoretical predictions via numerical simulations averaged over 50 independent draws of $n = 1000$ training examples. The results indicate that the asymptotic predictions accurately describe the per-class behavior even for moderate values of n and p. We provide further details of the setup for these simulations in Appendix A.2. We also explore the scenario where sample imbalance and spectral imbalance are *both* present in Appendix A.3.

(A) Impact of eigenvalue scaling: In this example, we vary the relative scale of some of the eigenvalues of Class 1 while fixing the target signal as $\sqrt{p}\boldsymbol{\theta}^*_j = 1$ for all j. Specifically, we fix the eigenvalues of Class -1 to be either 0.5 or 2, in equal proportion. Then, we set all the eigenvalues for Class 1 to be a multiplicative scaling $s \geq 1$ of the eigenvalues of Class -1. We can see in Figure 2(A) that as s increases, the performance of Class 1 degrades severely while that of Class -1 also degrades, but to a lesser extent. In line with the general intuitions in Section 2.1, this results in a net increase in the class gap.

(B) Impact of eigenvalue decay rate: Fixing $\sqrt{p}\boldsymbol{\theta}^*_j = 1$ for all i, we again consider a bi-level covariance model, where the eigenvalues of each class take one of two values: 2 or 0.5. We fix the proportion of larger eigenvalues in Class -1 to be 0.5 while varying the proportion $p \in [0, 1]$ of larger eigenvalues for Class 1. This example indicates that there can be an important trade-off between overall performance and the class gap: the choice $p = 0$ (all small eigenvalues) yields the best per-class performance for both classes while $p = 0.5$ minimizes the class gap.

(C) Impact of alignment with the target signal: The previous notions of spectral imbalance ignored the relative shape of each marginal distribution with respect to the target signal $\boldsymbol{\theta}^*$. We now consider a situation where the two class covariances are misaligned, so that either $(\lambda_j^{(1)}, \lambda_j^{(-1)}) = (0.5, 2)$ or $(\lambda_j^{(1)}, \lambda_j^{(-1)}) = (2, 0.5)$. We then increase the magnitude of $\boldsymbol{\theta}^*_j$ by a factor a, *only* in the directions $j \in [p]$ where $\lambda_j^{(-1)} = 2$. Here, as we increase the parameter $a \geq 1$, the overall signal strength increases, so the global performance improves. However, due to the differences in the covariance alignment, the improvement is not uniform between the two classes, and Class 1, which has smaller variance in the directions of increased signal strength, improves more quickly. Interestingly, for large a the class gap decreases as the effect of large signal strength outweighs the differences between the class spectra.

4. Spectral imbalance in pretrained features

In this section, we build on our theoretical findings to explore how the framework of spectral imbalance may be used to understand and mitigate class bias in the image representation space of pre-trained models.

4.1. Experimental setup

Settings. We consider a common scenario in representation learning, where encoders are used to extract features, and linear classifiers are trained on these features for downstream evaluation. Formally, a dataset $D = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$ of n images is used to train a network consisting of a pre-

trained encoder $f_\omega(\cdot)$ and a linear classifier $g_\theta(\cdot) = \langle \theta, \cdot \rangle$. The encoder produces a latent representation for each image following $\mathbf{z}_i = f_\omega(\mathbf{x}_i)$. In this work, we aim to understand the properties of the resulting representation space $\mathcal{Z} = \{\mathbf{z}_i\}_{i=1}^n$, and how the (estimated) spectral properties of the distribution on $\mathcal{Z}$ can be correlated to the classification results of the classifier $g_\theta(\cdot)$ for each class. Note that the role of the latent features $\{\mathbf{z}_i\}_{i=1}^n$ in representation learning is directly comparable to the role of the original features $\{\mathbf{x}_i\}_{i=1}^n$ in the preceding GMM; this is because in both cases the downstream model is linear.

Empirical estimation of the class-dependent spectra. To obtain the estimated eigenspectrum, we first estimate the empirical class-dependent covariance matrix as:

$$\mathbf{\Sigma}_C\left(f_\omega\right) = \frac{1}{|\Omega_C|} \sum_{i \in \Omega_C} (\mathbf{z}_i - \overline{\mathbf{z}}_c)(\mathbf{z}_i - \overline{\mathbf{z}}_c)^T,$$

where C is a class, Ω_C denotes the set of examples with label $y_i = C$, and $\overline{\mathbf{z}}_c$ denotes the empirical mean of features with class C.

The eigenvalue decomposition of the covariance matrix is then computed as $\mathbf{\Sigma}_C = \mathbf{V}_C \mathbf{\Lambda}_C \mathbf{V}_C^{-1}$, where $\mathbf{\Lambda}_C$ is a diagonal matrix with nonnegative entries $\lambda_i^{(C)}$, and the columns of $\mathbf{V}_C$ correspond to the eigenvectors of $\mathbf{\Sigma}_C$. Without loss of generality, we assume that $\lambda_1^{(C)} \geq \lambda_2^{(C)} \ldots \geq \lambda_m^{(C)}$, where m is the rank of $\mathbf{\Sigma}_C$. The resulting set of eigenvalues $\lambda_i^{(C)}$ is the (empirical) eigenspectrum of class C.

Pretrained networks. In what follows, we evaluate the representation space of a diverse set of classification models to understand their class bias. Specifically:

- **Residual networks:** Using residual connections to bridge convolutional blocks, ResNets are a class of deep convolutional neural networks (CNNs) that is instrumental in many visual recognition tasks (He et al., 2016).
- **Improved CNNs:** DenseNet (Huang et al., 2017) and EfficientNet (Tan & Le, 2019) are two representative convolutional architectures that aim to improve ResNets. They design different strategies to build connections across layers to improve the information flow inside the networks.
- **Vision transformers:** The vision transformer splits an image into patches and uses self-attention to study their interactions (Dosovitskiy et al., 2020). ViT aggregates global information at earlier layers, creating significantly different representations than CNNs (Raghu et al., 2021).
- **Transformers without self-attention:** Recent studies show that self-attention is not required for obtaining good representations in vision. Along this line, we study MLP-Mixer (Tolstikhin et al., 2021) and PoolFormer (Yu et al., 2022), which uses multi-layer perceptrons and average pooling to mix information across patches, respectively.

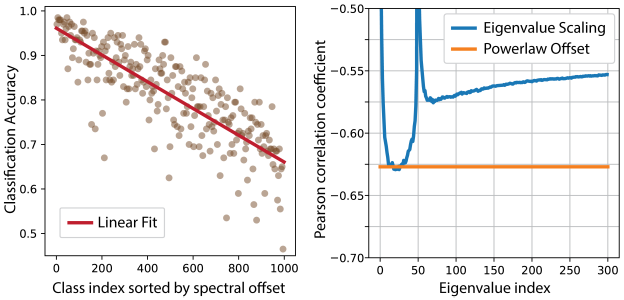

Figure 3. Examining the relationship between class-dependent spectra and performance. (left) The spectral offset of each class vs. their classification accuracy, computed for ResNet-50 on the validation set of ImageNet. (right) Pearson correlation coefficient between class accuracy and individual eigenvalues (blue) and the power law offset (red).

We found all checkpoints of the pretrained models in Torchvision (Marcel & Rodriguez, 2010) and timm (Wightman, 2019) (see details in Appendix B). For all experiments, we use the standard ImageNet ILSVRC 2012 dataset (Deng et al., 2009), which contains $C = 1000$ object classes with an average of 1281/50 training/validation images per class. We also provide results on smaller datasets in Appendix B.3.

4.2. Spectra correlate with class-wise performance

Individual eigenvalues. We first analyze the correlation between the class-wise performance and the magnitude of individual eigenvalues of each class. To do this, we compute the Pearson correlation coefficient (PCC) between the set of eigenvalues $\{\lambda_j^{(c)}\}$ (validation set) and the set of across-class accuracies, $\{\text{Accuracy}(c)\}$, for every eigenvalue index j. In Figure 3, we observe that the class-wise accuracy shows a strong negative correlation with most individual eigenvalues (right), in line with the theoretical results on eigenvalue scaling in Figure 2(A). Across the 11 models examined, we obtained an average maximum negative correlation of -0.69. Interestingly, unlike the examples in Section 3.3, the first few eigenvalues do not show a strong correlation for most models, potentially illustrating unique properties of the features learned by real-world encoders.

Power law decay offsets. Natural images are known to exhibit a power law decay in their eigenvalues (Ruderman, 1994). Recent work has shown that this also holds in visual representations in pretrained models (Ghosh et al., 2022) and can be used to examine the quality of learned representations. Thus, we fit the class conditional spectrum to

$\lambda_i = ai^{-b}$, and observe how the parameters (a, b) indicate the class-wise accuracy. Interestingly, we observed that the scale a (the *offset* on a log-scale) correlates most strongly with class accuracy (an average of -0.68 across 11 models), while the power (i.e. rate of decay) b does not strongly correlate with class-wise performance. This phenomenon demonstrates that there is a nuanced relationship between the specific distribution and decay rate of eigenvalues (cf. Figure 2(B)) and the class performance. We provide further details and also measure the correlation between other decay rate metrics and class-wise performance in Appendix B.2.

4.3. Spectral imbalance in your pretrained model

Next, we conduct a systematic investigation of the per-class accuracy discrepancies across encoders and show that spectral imbalance is often a measurable property of the encoder. Surprisingly, out of the 55 unique pairs of encoders that we consider, an average of 31% classes had a ranking (with duplicates) difference over 100 out of $C = 1000$ total classes, with the input data held constant. To better understand the large variability of per-class accuracy across encoders, we propose the *Spectral Quantile Score* (SQS), which measures the spectral imbalance between classes in a given encoder.

Spectral Quantile Score. Writing the set of per-class accuracies as a sorted list of numbers $\{Acc_1, \ldots, Acc_C\}$ in descending order, we define the *empirical* class bias score of an encoder as:

$$\textit{Class bias} = \left(\sum_{k=1}^{L} Acc_k - \sum_{k=C-L+1}^{C} Acc_k \right) / \left(L \times \overline{Acc} \right),$$

where C is the total number of classes, $\overline{Acc}$ is the average accuracy over all classes, and L is a selected cutoff. In our experiments on ImageNet, we set $L = 100$ throughout.

To compute the imbalance across classes in their spectra, we can similarly sort our per-class spectral metric of interest (i.e., spectral offset, specific eigenvalues) as $\{s_1, \ldots, s_C\}$ and compute the the Spectral Quantile Score (SQS):

$$\text{SQS} = s_L / s_1.$$

Again, we set $L = 100$ for our experiments. As shown in Figure 4(A), we found the SQS of different encoders is indicative of their empirical class bias (0.90 PCC), where latent spaces with a higher SQS would likely give higher empirical class bias. The strong correlation between SQS and empirical class bias emphasizes the importance of investigating the spectra imbalance in encoders, which accurately reflects and evaluates the encoders' empirical class bias. Moreover, as the worst class performance is often strongly associated with the final average performance of an encoder (Arjovsky et al., 2022), understanding spectral imbalance provides an additional lens into the representation space without training actual classifiers.

Testing for class disparity in a new encoder. Next, we investigate how class-dependent spectra can be used to estimate the performance of a new encoder. Following (Balestriero et al., 2022), we used a one-sided Welch's t-test (Welch, 1947) to validate our hypothesis that the per-class accuracy is significantly lower when their training set spectra have a higher offset parameter a. We define the random variable as the accuracy difference between two encoders $\Delta_{f_1,f_2}(c) := \text{Acc}_{f_1}(c) - \text{Acc}_{f_2}(c)$, and Δ_- represents the set of classes with spectral offset difference $a_{f_1}(c) - a_{f_2}(c) < 0$ being negative. We define the null hypothesis as $H_0 = \mathbb{E}[\Delta] > \mathbb{E}[\Delta_-]$.Intuitively, rejecting the hypothesis means the classes with a higher spectral property would give lower class-wise accuracies. We obtain that there is enough evidence to reject this hypothesis with 95% confidence for 98 out of 110 pairs of encoders; and 99% confidence for 80 out of 110 pairs of encoders. This demonstrates that spectral properties might reliably predict class-wise accuracies for new encoders, despite the noise in spectrum estimation.

4.4. Combine augmentations using spectral information

Recent work identified that commonly used data augmentations can in fact worsen class gap (Balestriero et al., 2022). Thus, we sought to study the effect of *data augmentation* on class disparity and ask whether different augmentations can be used to alleviate gaps in class accuracies.

Do different augmentations impact spectral imbalance? To begin, we supplement the results of Section 4.3 by showing that spectral imbalance correlates strongly with class gap not only across encoders, but also across data transformations. Following (Hendrycks & Dietterich, 2019), we generated 16 versions of ImageNet where the images are transformed differently. Specifically, given a sample $\mathbf{x}$ from our train/test dataset $\mathcal{D}$, we start by transforming every image with the 16 candidate augmentations as $\mathbf{x}^\ell = t_\ell(\mathbf{x})$, where $t_\ell \in \mathcal{T}_\ell$ is a sampled augmentation operation from the ℓ^{th} augmentation plan. After feeding these transformed images through our pretrained encoder f_ω, we obtain 16 different sets of latents $\{\mathbf{z}_\ell = f_\omega(\mathbf{x}^\ell)\}_{\ell=1}^{16}$. Then, our prediction corresponding to the ℓ^{th} augmented test sample is given by the C-dimensional logit output $\mathbf{L}_\ell = g_\theta(\mathbf{z}_\ell)$.

As in Section 4.2, we fit a power law to all classes and use the offset parameter a as the studied spectral property. Between the training set spectral properties and the testing set class bias, we compute the SQS score across all 16 types of augmentations and obtain a PCC of 0.90, demonstrating strong positive correlation. Within each class, the spectral property is also statistically significant for 78.9% out of 1000 classes. See Appendix B.5 for more details.

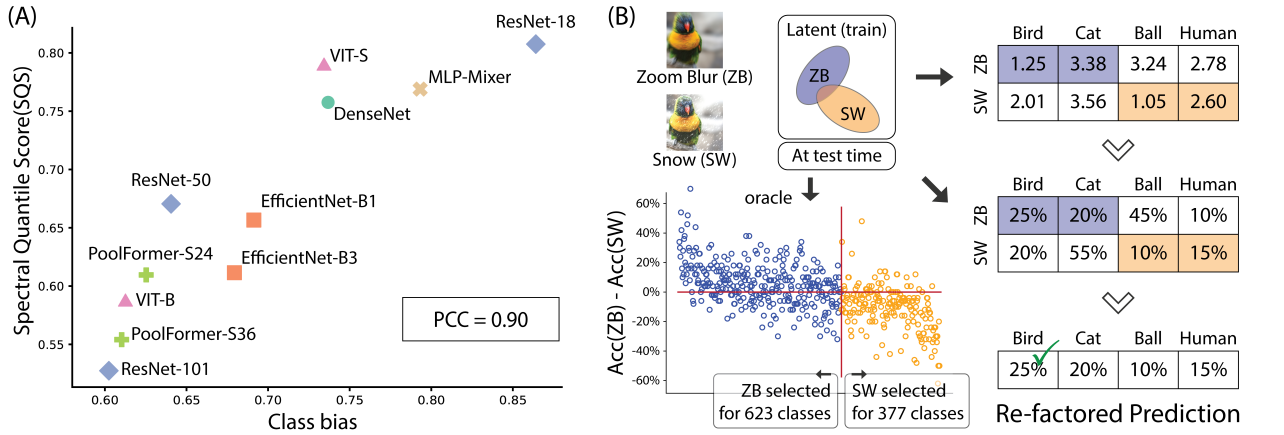

Figure 4. Selecting model and data with spectra. (A) Given different encoders, the Spectral Quantile Score (SQS) is predictive of the empirical class bias of the representation spaces. Note that SQS is estimated on training set without deploying the classifier(s). (B) Given different data augmentation plans, we can use the class-level spectra to assign data to their optimal augmentation plan in a class-dependent way. This assignment can be used at test time to re-factor the prediction and boost performance across all classes without any re-training or modification to the encoder.

An ensembling method to combine augmentations using spectral information. Based on the observed correlation between imbalance of the spectral offset and performance for each class, we propose a *spectral re-factoring* method that adaptively combines the predictions across different augmented views at *test time* to improve performance. To do this, we extract the spectral offset parameter a for each class ($C = 1000$) across all candidate augmentations ($K = 16$) on the training dataset. This results in a matrix $\mathbf{M} \in \mathbb{R}^{K \times C}$, where each entry $M_{\ell,c} = a_{\ell,c}$ denotes the offset parameter for all latents $\{\mathbf{z}_\ell\}$ in class c.

Consider a test sample $\mathbf{x}$. Suppose that an 'oracle' could give us its true class c^*. Then, we would use the prediction corresponding to the augmented view that achieves minimum 'spectral score' of class c^*, i.e. use the augmentation ℓ^* that minimizes M_{ℓ,c^*}. We depict this idea in Figure 4(B) with two candidates Zoom Blur (ZB) and Snow (SW). In this case, ZB achieves a smaller spectral score for 623/1000 classes, while SW acheives a smaller spectral score for the other 377/1000 classes. Thus, our oracle would select the predictions of ZB for the 623 classes with smaller spectra and SW for the next 377 classes. This approach would yield significant improvement across all classes: 6.0% on 623 classes (blue) and 10.2% on 377 classes (yellow).

Of course, in reality we do not have access to the class information c^* to apply the 'correct augmentation'. Instead, we propose an ensembling method for assigning final class probabilities based upon the spectral property matrix $\mathbf{M}$:

(Step 1) Pass K candidate augmentations of each sample through f and g to produce a multi-view logits matrix $\mathbf{L} \in \mathbb{R}^{K \times C}$, where $\mathbf{L}_\ell$ corresponds to the logits generated by prediction on augmented data $\mathbf{x}^\ell$.

(Step 2) Create a new logit vector $\hat{\mathbf{l}}_c = \mathbf{L}_{\ell^*(c),c}$ where $\ell^*(c)$ *minimizes* $M_{\ell,c}$. Intuitively, we select the logit corresponding to the 'optimal' augmentation for class c, based upon the spectral scores in $\mathbf{M}$.

(Step 3) Classify test data $\mathbf{x}$ as $c(\mathbf{x}) = \arg\max_{c \in [C]} \hat{L}_c$.

This re-factored prediction is depicted in Figure 4(B). Essentially, the procedure first *estimates* the correct augmentation from the training set for each class to use in steps 1 and 2, selects the corresponding logit for that class, and uses these adaptively selected logits in a standard classification procedure in step 3. This ensembling procedure improves Zoom Blur (65.01%) and Snow (65.12%) to a combined accuracy of 67.616%, giving an average performance improvement of 2.55% and an average class gap improvement of 3.52% *without any re-training required.*

Using the same approach, we repeat the experiments on all combinations of augmentations provided by (Hendrycks & Dietterich, 2019) in Appendix B.5. We show that, across 15x16 = 240 runs, our proposed ensemble method can stably improve the average performance across two augmentations and the worse augmentation by 2.6% and 5.9% respectively.

,

5. Related work

Class-dependent generalization and class bias. Recently, (Balestriero et al., 2022; Kirichenko et al., 2023) put forward the interesting observation that data augmentations can create class bias, and that the issue worsens the

stronger the augmentation(s) is. In contrast to our work, these works focus exclusively on the impact of data augmentation and do not identify or characterize the role of spectra in class bias. They propose a different hypothesis that is less quantifiable; in particular, that classes that are inherently more 'fine-grained', or harder to distinguish, are most affected. Comparatively, our proposed measures of spectral imbalance are precise, quantitative, and demonstrated to correlate with class bias through formal statistical analysis.

The role of spectra in generalization. Many recent works have highlighted the importance of the data spectrum in generalization (Bartlett et al., 2020; Muthukumar et al., 2021; Lin et al., 2022), especially in the overparameterized regime. Specifically, the works (Chatterji & Long, 2021; Wang & Thrampoulidis, 2021; Cao et al., 2021; Wang et al., 2021) characterize the behavior of overparameterized linear classifiers in GMMs through non-asymptotic, finite-sample bounds and show good generalization under favorable spectral properties of the data. However, they all assume equal class covariances and do not explicitly consider the effects of spectral imbalance. Another line of work provides exact asymptotic bounds on the classification error achieved by linear classifiers in the GMM (Deng et al., 2022; Mai et al., 2019; Loureiro et al., 2021; Kini et al., 2021; Taheri et al., 2020; Javanmard & Soltanolkotabi, 2022). Of these, only (Loureiro et al., 2021) allow for different covariances between classes. Their work studies a more general multiclass setting, but does not systematically study per-class generalization. By contrast, we focus on a simpler binary classification setting and use an alternative proof technique based on Gaussian comparison inequalities to obtain a much simpler expression that is easier to numerically evaluate, and thus allows us to model various spectral imbalances. Finally, our empirical results estimate the eigenspectra either on validation/test data, on augmented training data, or on training data but non-zero training error. This circumvents "neural collapse" on training data (Papyan et al., 2020) which would yield misleading estimates of eigenspectra.

Evaluating representations. Assessing the quality of representations is a critical aspect of representation learning. The importance stems from its ability to bypass the need for training classifiers, especially when the quality and quantity of downstream data are lacking (Martin et al., 2021; Liu et al., 2022; 2023). Representations can be studied from the perspective of manifold analysis (Hauser & Ray, 2017); Previous works show that estimated geometric properties like intrinsic dimension correlate with performance (Ansuini et al., 2019; Cohen et al., 2020; Doimo et al., 2020; Valeriani et al., 2023). Recently, (Ghosh et al., 2022; Agrawal et al., 2022) uses the power law decay of eigenspectrum to measure the quality of representations, and follow-up works show that spectral properties can also be used to improve self-supervised learning (He & Ozay, 2022; Zhang et al., 2023b; Weng et al., 2023). In contrast to these works, we especially evaluate the spectral properties of the representation space from the perspective of class bias and class-dependent generalization error.

The recent works (Ma et al., 2022; 2023) also provide a geometric interpretation of the class covariances of learned representations and connect these differences in geometry to differences in performance, both for long-tailed and balanced datasets. Our work provides an alternate and complementary perspective, studying the decay rate and relative magnitude of eigenvalues, rather than the "volume", which is related to the product of eigenvalues (Ma et al., 2022) or the Gauss curvature (Ma et al., 2023) of the learned features. Compared to these works, we provide an explicit theoretical framework for studying spectral imbalance in high-dimensional GMMs, and we focus our attention on pre-trained representations and sample-balanced datasets. We compare our empirical findings to (Ma et al., 2022) in more detail in Appendix B.6.

6. Discussion

In this work, we introduced the concept of spectral imbalance as a way to characterize class-dependent bias. We studied spectral imbalance in theory and in practice, and provided a new framework for studying the relationship between class and spectral imbalance. While we identify spectral imbalance as *one of* the factors influencing class disparity, there could be others which we have as yet not identified. Moreover, our theory assumes linear models and our experiments assume pre-trained encoders. In the future, we would like to understand how feature learning in neural networks could lead to spectral imbalance in the first place. We hope such understanding will help us effectively mitigate the class bias issue, either through re-training or post-hoc manipulation of pre-trained features.

Acknowledgements

We are grateful to the anonymous ICML reviewers for valuable feedback. This project is supported by an NSF Graduate Research Fellowship (DGE-2039655), NSF award (IIS-2039741), NSF award (CIF:RI:2212182), NSF CAREER awards (IIS-2146072, CCF-223915), as well as generous gifts from the Alfred Sloan Foundation, the McKnight Foundation, the CIFAR Azrieli Global Scholars Program, Amazon Research and Adobe Research.

Impact Statement

Our work provides new insights into how to mitigate class biases by measuring the spectral properties of representation space. The presented results suggest that large-scale pre-

trained models may possess inherent biases in their decision-making processes. Our method provides researchers with tools to better understand and address these issues. While this work focuses on evaluating and mitigating the biases in discriminative tasks, extending this approach to generative models and tasks can have major societal benefits regarding fairness and auditing of pre-trained models.

References

Agrawal, K. K., Mondal, A. K., Ghosh, A., and Richards, B. Assessing representation quality in self-supervised learning by measuring eigenspectrum decay. *Advances in Neural Information Processing Systems*, 35:17626–17638, 2022.

Andersen, P. K. and Gill, R. D. Cox's regression model for counting processes: a large sample study. *The Annals of Statistics*, pp. 1100–1120, 1982.

Ansuini, A., Laio, A., Macke, J. H., and Zoccolan, D. Intrinsic dimension of data representations in deep neural networks. *Advances in Neural Information Processing Systems*, 32, 2019.

Arjovsky, M., Chaudhuri, K., and Lopez-Paz, D. Throwing away data improves worst-class error in imbalanced classification. *arXiv preprint arXiv:2205.11672*, 2022.

Balestriero, R., Bottou, L., and LeCun, Y. The effects of regularization and data augmentation are class dependent. *arXiv preprint arXiv:2204.03632*, 2022.

Bartlett, P. L., Long, P. M., Lugosi, G., and Tsigler, A. Benign overfitting in linear regression. *Proceedings of the National Academy of Sciences*, 117(48):30063–30070, 2020.

Cao, Y., Gu, Q., and Belkin, M. Risk bounds for over-parameterized maximum margin classification on sub-gaussian mixtures. *Advances in Neural Information Processing Systems*, 34:8407–8418, 2021.

Chatterji, N. S. and Long, P. M. Finite-sample analysis of interpolating linear classifiers in the overparameterized regime. *J. Mach. Learn. Res.*, 22:129–1, 2021.

Cohen, U., Chung, S., Lee, D. D., and Sompolinsky, H. Separability and geometry of object manifolds in deep neural networks. *Nature Communications*, 11(1):746, 2020.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255. Ieee, 2009.

Deng, Z., Kammoun, A., and Thrampoulidis, C. A model of double descent for high-dimensional binary linear classification. *Information and Inference: A Journal of the IMA*, 11(2):435–495, 2022.

Dhifallah, O. and Lu, Y. On the inherent regularization effects of noise injection during training. In *International Conference on Machine Learning*, pp. 2665–2675. PMLR, 2021.

Doimo, D., Glielmo, A., Ansuini, A., and Laio, A. Hierarchical nucleation in deep neural networks. *Advances in Neural Information Processing Systems*, 33:7526–7536, 2020.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Ghosh, A., Mondal, A. K., Agrawal, K. K., and Richards, B. Investigating power laws in deep representation learning. *arXiv preprint arXiv:2202.05808*, 2022.

Hauser, M. and Ray, A. Principles of riemannian geometry in neural networks. *Advances in Neural Information Processing Systems*, 30, 2017.

He, B. and Ozay, M. Exploring the gap between collapsed & whitened features in self-supervised learning. In *International Conference on Machine Learning*, pp. 8613–8634. PMLR, 2022.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016.

Hendrycks, D. and Dietterich, T. Benchmarking neural network robustness to common corruptions and perturbations. *arXiv preprint arXiv:1903.12261*, 2019.

Huang, G., Liu, Z., Van Der Maaten, L., and Weinberger, K. Q. Densely connected convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4700–4708, 2017.

Javanmard, A. and Soltanolkotabi, M. Precise statistical analysis of classification accuracies for adversarial training. *The Annals of Statistics*, 50(4):2127–2156, 2022.

Kini, G. R., Paraskevas, O., Oymak, S., and Thrampoulidis, C. Label-imbalanced and group-sensitive classification under overparameterization. *Advances in Neural Information Processing Systems*, 34:18970–18983, 2021.

Kirichenko, P., Balestriero, R., Ibrahim, M., Vedantam, S. R., Firooz, H., and Wilson, A. G. Understanding the class-specific effects of data augmentations. In *ICLR 2023 Workshop on Pitfalls of limited data and computation for Trustworthy ML*, 2023.

Lin, C.-H., Kaushik, C., Dyer, E. L., and Muthukumar, V. The good, the bad and the ugly sides of data augmentation: An implicit spectral regularization perspective. *arXiv preprint arXiv:2210.05021*, 2022.

Liu, R., Azabou, M., Dabagia, M., Xiao, J., and Dyer, E. Seeing the forest and the tree: Building representations of both individual and collective dynamics with transformers. *Advances in Neural Information Processing Systems*, 35:2377–2391, 2022.

Liu, R., Zippi, E. L., Pouransari, H., Sandino, C., Nie, J., Goh, H., Azemi, E., and Moin, A. Frequency-aware masked autoencoders for multimodal pretraining on biosignals. *arXiv preprint arXiv:2309.05927*, 2023.

Loureiro, B., Sicuro, G., Gerbelot, C., Pacco, A., Krzakala, F., and Zdeborová, L. Learning gaussian mixtures with generalized linear models: Precise asymptotics in high-dimensions. *Advances in Neural Information Processing Systems*, 34:10144–10157, 2021.

Ma, Y., Jiao, L., Liu, F., Li, Y., Yang, S., and Liu, X. Delving into semantic scale imbalance. *arXiv preprint arXiv:2212.14613*, 2022.

Ma, Y., Jiao, L., Liu, F., Yang, S., Liu, X., and Li, L. Curvature-balanced feature manifold learning for long-tailed classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15824–15835, 2023.

Mai, X., Liao, Z., and Couillet, R. A large scale analysis of logistic regression: Asymptotic performance and new insights. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3357–3361. IEEE, 2019.

Marcel, S. and Rodriguez, Y. Torchvision the machine-vision package of torch. In *Proceedings of the 18th ACM International Conference on Multimedia*, pp. 1485–1488, 2010.

Martin, C. H., Peng, T., and Mahoney, M. W. Predicting trends in the quality of state-of-the-art neural networks without access to training or testing data. *Nature Communications*, 12(1):4122, 2021.

Montanari, A., Ruan, F., Sohn, Y., and Yan, J. The generalization error of max-margin linear classifiers: High-dimensional asymptotics in the overparametrized regime. *arXiv preprint arXiv:1911.01544*, 2019.

Mullick, S. S., Datta, S., Dhekane, S. G., and Das, S. Appropriateness of performance indices for imbalanced data classification: An analysis. *Pattern Recognition*, 102: 107197, 2020.

Muthukumar, V., Narang, A., Subramanian, V., Belkin, M., Hsu, D., and Sahai, A. Classification vs regression in overparameterized regimes: Does the loss function matter? *Journal of Machine Learning Research*, 22(222):1–69, 2021. URL http://jmlr.org/papers/v22/20-603.html.

Papyan, V., Han, X., and Donoho, D. L. Prevalence of neural collapse during the terminal phase of deep learning training. *Proceedings of the National Academy of Sciences*, 117(40):24652–24663, 2020.

Raghu, M., Unterthiner, T., Kornblith, S., Zhang, C., and Dosovitskiy, A. Do vision transformers see like convolutional neural networks? *Advances in Neural Information Processing Systems*, 34:12116–12128, 2021.

Ren, J., Yu, C., Ma, X., Zhao, H., Yi, S., et al. Balanced meta-softmax for long-tailed visual recognition. *Advances in Neural Information Processing Systems*, 33: 4175–4186, 2020.

Ruderman, D. L. The statistics of natural images. *Network: Computation in Neural Systems*, 5(4):517, 1994.

Taheri, H., Pedarsani, R., and Thrampoulidis, C. Asymptotic behavior of adversarial training in binary classification. *arXiv preprint arXiv:2010.13275*, 2020.

Tan, M. and Le, Q. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International Conference on Machine Learning*, pp. 6105–6114. PMLR, 2019.

Thrampoulidis, C., Abbasi, E., and Hassibi, B. Precise error analysis of regularized m-estimators in high dimensions. *IEEE Transactions on Information Theory*, 64(8):5592–5628, 2018.

Tolstikhin, I. O., Houlsby, N., Kolesnikov, A., Beyer, L., Zhai, X., Unterthiner, T., Yung, J., Steiner, A., Keysers, D., Uszkoreit, J., et al. Mlp-mixer: An all-mlp architecture for vision. *Advances in Neural Information Processing Systems*, 34:24261–24272, 2021.

Tsigler, A. and Bartlett, P. L. Benign overfitting in ridge regression. *arXiv preprint arXiv:2009.14286*, 2020.

Valeriani, L., Doimo, D., Cuturello, F., Laio, A., Ansuini, A., and Cazzaniga, A. The geometry of hidden representations of large transformer models. *arXiv preprint arXiv:2302.00294*, 2023.

Wang, K. and Thrampoulidis, C. Binary classification of gaussian mixtures: abundance of support vectors, benign overfitting and regularization. *arXiv preprint arXiv:2011.09148*, 2021.

Wang, K., Muthukumar, V., and Thrampoulidis, C. Benign overfitting in multiclass classification: All roads lead to interpolation. *Advances in Neural Information Processing Systems*, 34:24164–24179, 2021.

Welch, B. L. The generalization of 'student's'problem when several different population varlances are involved. *Biometrika*, 34(1-2):28–35, 1947.

Weng, X., Ni, Y., Song, T., Luo, J., Anwer, R. M., Khan, S., Khan, F. S., and Huang, L. Modulate your spectrum in self-supervised learning. *arXiv preprint arXiv:2305.16789*, 2023.

Wightman, R. Pytorch image models. https://github.com/rwightman/pytorch-image-models, 2019.

Yu, W., Luo, M., Zhou, P., Si, C., Zhou, Y., Wang, X., Feng, J., and Yan, S. Metaformer is actually what you need for vision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10819–10829, 2022.

Zhang, Y., Kang, B., Hooi, B., Yan, S., and Feng, J. Deep long-tailed learning: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023a.

Zhang, Y., Zhu, H., Song, Z., Koniusz, P., and King, I. Spectral feature augmentation for graph contrastive learning and beyond. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 11289–11297, 2023b.

Zhu, J., Wang, Z., Chen, J., Chen, Y.-P. P., and Jiang, Y.-G. Balanced contrastive learning for long-tailed visual recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6908–6917, 2022.

Appendix

A. Theoretical Results for the Gaussian Mixture Model

A.1. Proofs

This section contains the proofs for spectrally-imbalanced GMM setting in Section 2.1.

Lemma 2 (Expression for the POE in the imbalanced GMM)**.** *In the GMM setting of Section 2.1, given an estimator $\hat{\boldsymbol{\theta}} \in \mathbb{R}^p$, the per-class error and class gap satisfy*

$$\begin{aligned} \textit{POE}(\hat{\boldsymbol{\theta}}|y) &:= \mathbb{P}\{\textit{sign}(\mathbf{x}^\top \hat{\boldsymbol{\theta}}) \neq y | \mathbf{x} \textit{ in class } y\} = Q\left(\frac{\langle \hat{\boldsymbol{\theta}}, \boldsymbol{\theta}^* \rangle}{\|\boldsymbol{\Sigma}_y^{1/2}\hat{\boldsymbol{\theta}}\|_2}\right) \\ \textit{ClassGap}(\hat{\boldsymbol{\theta}}) &= \left| Q\left(\frac{\langle \hat{\boldsymbol{\theta}}, \boldsymbol{\theta}^* \rangle}{\|\boldsymbol{\Sigma}_1^{1/2}\hat{\boldsymbol{\theta}}\|_2}\right) - Q\left(\frac{\langle \hat{\boldsymbol{\theta}}, \boldsymbol{\theta}^* \rangle}{\|\boldsymbol{\Sigma}_{-1}^{1/2}\hat{\boldsymbol{\theta}}\|_2}\right) \right| \\ &\leq \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}\min_y \frac{\langle \tilde{\boldsymbol{\theta}}, \boldsymbol{\theta}^* \rangle^2}{\|\boldsymbol{\Sigma}_y^{1/2}\hat{\boldsymbol{\theta}}\|_2^2}\right) |\langle \hat{\boldsymbol{\theta}}, \boldsymbol{\theta}^* \rangle| \left| \|\boldsymbol{\Sigma}_y^{1/2}\hat{\boldsymbol{\theta}}\|_2^{-1} - \|\boldsymbol{\Sigma}_y^{1/2}\hat{\boldsymbol{\theta}}\|_2^{-1} \right|, \end{aligned}$$

where $Q(\cdot)$ is the Gaussian Q-function.

Proof. Note we can write $\mathbf{x}$ as $y\boldsymbol{\theta}^* + \boldsymbol{\Sigma}_y^{1/2}\mathbf{z}$, for $\mathbf{z} \sim \mathcal{N}(0, \mathbf{I})$. Then,

$$\begin{aligned} \mathbb{P}\{\text{sign}(\mathbf{x}^\top \hat{\boldsymbol{\theta}}) \neq y | \mathbf{x} \text{ in class } y\} &= \mathbb{P}\{y(\mathbf{x}^\top \hat{\boldsymbol{\theta}}) < 0 | \mathbf{x} \text{ in class } y\} \\ &= \mathbb{P}\{y((y\boldsymbol{\theta}^* + \boldsymbol{\Sigma}_y^{1/2}\mathbf{z})^\top \hat{\boldsymbol{\theta}}) < 0\} \\ &= \mathbb{P}\{\langle \hat{\boldsymbol{\theta}}, \boldsymbol{\theta}^* \rangle + y\hat{\boldsymbol{\theta}}^\top \boldsymbol{\Sigma}_y^{1/2}\mathbf{z} < 0\} \\ &= \mathbb{P}\{\langle \hat{\boldsymbol{\theta}}, \boldsymbol{\theta}^* \rangle + \|\boldsymbol{\Sigma}_y^{1/2}\hat{\boldsymbol{\theta}}\|_2 z < 0\} \\ &= \mathbb{P}\left\{z < -\frac{\langle \hat{\boldsymbol{\theta}}, \boldsymbol{\theta}^* \rangle}{\|\boldsymbol{\Sigma}_y^{1/2}\hat{\boldsymbol{\theta}}\|_2}\right\} \\ &= Q\left(\frac{\langle \hat{\boldsymbol{\theta}}, \boldsymbol{\theta}^* \rangle}{\|\boldsymbol{\Sigma}_y^{1/2}\hat{\boldsymbol{\theta}}\|_2}\right), \end{aligned}$$

where, in the fourth line, we use the fact that $y\hat{\boldsymbol{\theta}}^\top \boldsymbol{\Sigma}_y^{1/2}\mathbf{z} \overset{d}{=} \|\boldsymbol{\Sigma}_y^{1/2}\hat{\boldsymbol{\theta}}\|_2 z$, for a scalar $z \sim \mathcal{N}(0, 1)$. The upper bound on the class gap follows from the mean value theorem, after noting that the derivative of the Q-function is the negative of the Gaussian pdf. □

Before proceeding with the full proof of our main technical result, we first provide a brief sketch of the proof technique and the primary differences from previous works which study the balanced GMM setting.

Proof outline and remarks Our proof generalizes the result of (Taheri et al., 2020) to the case where each class has a different covariance and allows us to characterize the error in a per-class way. To make the connections to this analysis (and the key differences) clear, we make an attempt to follow the same notation when possible. Due to the covariate imbalance between classes, our analysis of the ERM objective function differs in a few important respects. Typically, application of the CGMT to anisotropic covariates requires a "whitening" step to obtain standard normal variables. The key step in our proof is to *separately* whiten the data from each class, which "decouples" the objective function into a portion corresponding to each class with its own decision variables. These two portions are tied together via the introduction of a new constraint.

We then invoke the CGMT once for each class (or more precisely, via a multivariate extension of (Dhifallah & Lu, 2021)) to write the problem as an equivalent "Auxiliary Optimization" (AO) objective which has a similarly decoupled form. Each of these portions of the AO can be simplified separately using the same "scalarization" techniques as in (Taheri et al., 2020). The additional constraint we introduced during the whitening step then plays a role in the final step of the proof, yielding the second-order Moreau envelope function which depends on the decision variables corresponding to both classes. Finally,

we note that, unlike (Taheri et al., 2020), we do not consider the effect of adversarial training; however, the same proof technique can be extended in a natural way to deal with this case.

We now proceed to the full proof of Theorem 1.

Proof of Theorem 1. We first analyze the ERM optimization through a series of simplifications.

$$\begin{aligned}\min_{\boldsymbol{\theta}\in\mathbb{R}^p} \frac{1}{n}\sum_{i=1}^n \mathcal{L}(y_i\langle \mathbf{x}_i,\boldsymbol{\theta}\rangle) + r\|\boldsymbol{\theta}\|_2^2 &= \min_{\boldsymbol{\theta}\in\mathbb{R}^p} \frac{1}{n}\sum_{i=1}^n \mathcal{L}(y_i\langle y_i\boldsymbol{\theta}^* + \boldsymbol{\Sigma}_{y_i}^{1/2}\mathbf{z}_i,\boldsymbol{\theta}\rangle) + r\|\boldsymbol{\theta}\|_2^2 \\ &\overset{d}{=} \min_{\boldsymbol{\theta}\in\mathbb{R}^p} \frac{1}{n}\sum_{i=1}^n \mathcal{L}(\langle \boldsymbol{\theta}^* + \boldsymbol{\Sigma}_{y_i}^{1/2}\mathbf{z}_i,\boldsymbol{\theta}\rangle) + r\|\boldsymbol{\theta}\|_2^2,\end{aligned}$$

where the last line follows from $y_i\mathbf{z}_i \overset{d}{=} \mathbf{z}_i$. Then, we re-index the data points as $\mathbf{z}_{y,i}$ for $y\in\{-1,1\}$ and $i\in[n_y]$ to get

$$\min_{\boldsymbol{\theta}\in\mathbb{R}^p} \frac{1}{n}\sum_{y\in\{-1,1\}}\sum_{i=1}^{n_y} \mathcal{L}(\langle \boldsymbol{\theta}^* + \boldsymbol{\Sigma}_y^{1/2}\mathbf{z}_{y,i},\boldsymbol{\theta}\rangle) + r\|\boldsymbol{\theta}\|_2^2$$

Next, let $\tilde{\boldsymbol{\theta}}_y \coloneqq \boldsymbol{\Sigma}_y^{1/2}\boldsymbol{\theta}$ and $\tilde{\boldsymbol{\theta}}_y^* \coloneqq \boldsymbol{\Sigma}_y^{-1/2}\boldsymbol{\theta}^*$. We now use a change of variable to write the problem as an optimization over the two variables $\tilde{\boldsymbol{\theta}}_1$ and $\tilde{\boldsymbol{\theta}}_{-1}$ while enforcing that both correspond to the same $\boldsymbol{\theta}$.

$$\min_{\substack{\tilde{\boldsymbol{\theta}}_y\in\mathbb{R}^p \\ \text{s.t.}\boldsymbol{\Sigma}_1^{-1/2}\tilde{\boldsymbol{\theta}}_1=\boldsymbol{\Sigma}_{-1}^{-1/2}\tilde{\boldsymbol{\theta}}_{-1}}} r\|\boldsymbol{\Sigma}_1^{-1/2}\tilde{\boldsymbol{\theta}}_1\|_2^2 + \frac{1}{n}\sum_{y\in\{-1,1\}}\sum_{i=1}^{n_y}\mathcal{L}(\langle\tilde{\boldsymbol{\theta}}_y,\tilde{\boldsymbol{\theta}}_y^*\rangle + \langle\mathbf{z}_{y,i},\tilde{\boldsymbol{\theta}}_y\rangle)$$

By duality, we can now write this as a min-max problem.

$$\begin{aligned}&= \min_{\tilde{\boldsymbol{\theta}}_y\in\mathbb{R}^p}\max_{\mathbf{w}\in\mathbb{R}^p} \quad r\|\boldsymbol{\Sigma}_1^{-1/2}\tilde{\boldsymbol{\theta}}_1\|_2^2 + \langle\mathbf{w},\boldsymbol{\Sigma}_1^{-1/2}\tilde{\boldsymbol{\theta}}_1 - \boldsymbol{\Sigma}_{-1}^{-1/2}\tilde{\boldsymbol{\theta}}_{-1}\rangle + \frac{1}{n}\sum_y\sum_{i=1}^{n_y}\mathcal{L}(\langle\tilde{\boldsymbol{\theta}}_y,\tilde{\boldsymbol{\theta}}_y^*\rangle + \langle\mathbf{z}_{y,i},\tilde{\boldsymbol{\theta}}_y\rangle) \\ &= \min_{\tilde{\boldsymbol{\theta}}_y\in\mathbb{R}^p}\max_{\mathbf{w}\in\mathbb{R}^p} \quad r\|\boldsymbol{\Sigma}_1^{-1/2}\tilde{\boldsymbol{\theta}}_1\|_2^2 + \langle\mathbf{w},\boldsymbol{\Sigma}_1^{-1/2}\tilde{\boldsymbol{\theta}}_1 - \boldsymbol{\Sigma}_{-1}^{-1/2}\tilde{\boldsymbol{\theta}}_{-1}\rangle \\ &\qquad\qquad + \frac{1}{n}\sum_y\left[\sum_{i=1}^{n_y}\mathcal{L}(\langle\tilde{\boldsymbol{\theta}}_y,\tilde{\boldsymbol{\theta}}_y^*\rangle + \langle\mathbf{z}_{y,i},\boldsymbol{\Theta}_y\tilde{\boldsymbol{\theta}}_y\rangle + \langle\mathbf{z}_{y,i},\boldsymbol{\Theta}_y^\perp\tilde{\boldsymbol{\theta}}_y\rangle)\right]\end{aligned}$$

where we define $\boldsymbol{\Theta}_y \coloneqq \frac{\tilde{\boldsymbol{\theta}}_y^*\tilde{\boldsymbol{\theta}}_y^{*\top}}{\|\tilde{\boldsymbol{\theta}}_y^*\|_2^2}$ as the orthogonal projection matrix onto $\tilde{\boldsymbol{\theta}}_y^*$. Now, introduce the constraints $v_{y,i} = \langle\tilde{\boldsymbol{\theta}}_y,\tilde{\boldsymbol{\theta}}_y^*\rangle + \langle\mathbf{z}_{y,i},\boldsymbol{\Theta}_y\tilde{\boldsymbol{\theta}}_y\rangle + \langle\mathbf{z}_{y,i},\boldsymbol{\Theta}_y^\perp\tilde{\boldsymbol{\theta}}_y\rangle$, corresponding to dual variables $\mathbf{u}_y$ to get:

$$\begin{aligned}\min_{\substack{\tilde{\boldsymbol{\theta}}_y\in\mathbb{R}^p \\ \mathbf{v}_y\in\mathbb{R}^{n_y}}}\max_{\substack{\mathbf{w}\in\mathbb{R}^p \\ \mathbf{u}_y\in\mathbb{R}^{n_y}}} \quad & r\|\boldsymbol{\Sigma}_1^{-1/2}\tilde{\boldsymbol{\theta}}_1\|_2^2 + \langle\mathbf{w},\boldsymbol{\Sigma}_1^{-1/2}\tilde{\boldsymbol{\theta}}_1 - \boldsymbol{\Sigma}_{-1}^{-1/2}\tilde{\boldsymbol{\theta}}_{-1}\rangle \\ & + \frac{1}{n}\sum_{y\in\{-1,1\}}\left[\mathbf{1}^\top\mathcal{L}(\mathbf{v}_y) + \langle\mathbf{u}_y,\mathbf{1}\rangle\langle\tilde{\boldsymbol{\theta}}_y^*,\tilde{\boldsymbol{\theta}}_y\rangle + \langle\mathbf{u}_y,\mathbf{Z}_y\boldsymbol{\Theta}_y\tilde{\boldsymbol{\theta}}_y\rangle + \langle\mathbf{u}_y,\mathbf{Z}_y\boldsymbol{\Theta}_y^\perp\tilde{\boldsymbol{\theta}}_y\rangle - \langle\mathbf{u}_y,\mathbf{v}_y\rangle\right]\end{aligned} \tag{4}$$

Here, $\mathcal{L}(\mathbf{v}_y)\in\mathbb{R}^{n_y}$ is a vector containing $\mathcal{L}(\mathbf{v}_{y,i})$ and $\mathbf{Z}_y\in\mathbb{R}^{n_y\times p}$ is a matrix with the $\mathbf{z}_{y,i}$ as rows. We call this problem the Primary Optimization (PO). We note here that for each y, $\mathbf{Z}_y\boldsymbol{\Theta}_y \perp \mathbf{Z}_y\boldsymbol{\Theta}_y^\perp$. Furthermore, $\mathbf{Z}_1 \perp \mathbf{Z}_{-1}$. So, for each y, we can replace $\mathbf{Z}_y\boldsymbol{\Theta}_y^\perp$ by $\hat{\mathbf{Z}}_y\boldsymbol{\Theta}_y^\perp$ for an independent Gaussian matrix $\hat{\mathbf{Z}}_y$. Conditioning on the $\mathbf{Z}_y$, we can see that the objective function can be written in the form

$$\sum_{y\in\{-1,1\}}\langle\mathbf{u}_y,\hat{\mathbf{Z}}_y\boldsymbol{\Theta}_y^\perp\tilde{\boldsymbol{\theta}}_y\rangle + \psi((\tilde{\boldsymbol{\theta}}_y,\mathbf{v}_y),(\mathbf{w},\mathbf{u}_y))$$

for a convex-concave function ψ. Here, we can apply Theorem 3 of (Dhifallah & Lu, 2021), which formalizes the repeated application of the CGMT in this setting and allows us to write a corresponding Auxiliary Optimization (AO) problem. It is important to note that using this theorem requires that the optimization be over compact sets. This can be relaxed as in (Thrampoulidis et al., 2018) by arguing that the optimal solution in the limit is bounded so we can equivalently consider the PO to be over a sufficiently large bounded set without changing its solution.

This theorem states that if the optimal solution of the AO converges in probability to some value ϕ, then the optimal solution of the PO converges to the same value. Moreover, if we restrict the AO decision variables to lie outside of some set $\mathcal{S}$ and the resultant problem converges to some $\phi' > \phi$, we can conclude that the optimal minimizers/maximizers of the PO belong to $\mathcal{S}$ with probability tending to 1.

Applying the CGMT, we arrive at the following AO:

$$\begin{aligned}\min_{\substack{\tilde{\boldsymbol{\theta}}_y\in\mathbb{R}^p\\ \mathbf{v}_y\in\mathbb{R}^{n_y}}}\max_{\substack{\mathbf{w}\in\mathbb{R}^p\\ \mathbf{u}_y\in\mathbb{R}^{n_y}}} \quad & r\|\boldsymbol{\Sigma}_1^{-1/2}\tilde{\boldsymbol{\theta}}_1\|_2^2 + \langle \mathbf{w}, \boldsymbol{\Sigma}_1^{-1/2}\tilde{\boldsymbol{\theta}}_1 - \boldsymbol{\Sigma}_{-1}^{-1/2}\tilde{\boldsymbol{\theta}}_{-1}\rangle \\ & + \frac{1}{n}\sum_{y\in\{-1,1\}}\Big[\mathbf{1}^\top\mathcal{L}(\mathbf{v}_y) + \langle \mathbf{u}_y, \mathbf{1}\rangle\langle\tilde{\boldsymbol{\theta}}_y^*, \tilde{\boldsymbol{\theta}}_y\rangle + \langle \mathbf{u}_y, \mathbf{Z}_y\boldsymbol{\Theta}_y\tilde{\boldsymbol{\theta}}_y\rangle + \|\boldsymbol{\Theta}_y^\perp\tilde{\boldsymbol{\theta}}_y\|_2\mathbf{g}_y^\top\mathbf{u}_y \\ & \qquad + \|\mathbf{u}_y\|_2\mathbf{h}_y^\top\boldsymbol{\Theta}_y^\perp\tilde{\boldsymbol{\theta}}_y - \langle \mathbf{u}_y, \mathbf{v}_y\rangle\Big],\end{aligned} \tag{5}$$

where $\mathbf{g}_y \sim \mathcal{N}(0, \mathbf{I}_{n_y})$ and $\mathbf{h}_y \sim \mathcal{N}(0, \mathbf{I}_p)$ are independent random vectors. We will now perform a series of simplifications that show that this min-max problem converges in probability to the solution of the scalar problem provided in Theorem 1.

First, observe that the terms inside the summation are identical to the terms which appear in the AO which is studied in (Taheri et al., 2020) (ignoring the terms in their analysis which correspond to adversarial training). So, we can "scalarize" the AO problem by applying the same analysis separately over each y. Due to the similarity of the steps of this part of the proof, we omit some of the details and emphasize places where the simplification of the AO differs.

To begin, we can decouple the optimization over $\mathbf{u}_y$ into the optimization over its norm and its unit direction and explicitly solve the maximization over its direction. Hence, we are left only with the maximization over its norm, which we denote through the variables $\beta_y = \|\mathbf{u}_y\|_2/\sqrt{n}$. As in (Taheri et al., 2020), we introduce new variables $\tilde{\boldsymbol{\rho}}_y = \tilde{\boldsymbol{\theta}}_y$ (enforced via the dual variables $\boldsymbol{\lambda}_y$) to allow us to separate the terms inside the summation from the terms outside the summation:

$$\begin{aligned}\min_{\substack{\tilde{\boldsymbol{\rho}}_y,\tilde{\boldsymbol{\theta}}_y\in\mathbb{R}^p\\ \mathbf{v}_y\in\mathbb{R}^{n_y}}}\max_{\substack{\mathbf{w},\boldsymbol{\lambda}_y\in\mathbb{R}^p\\ \beta_y\in\mathbb{R}_+}} \quad & r\|\boldsymbol{\Sigma}_1^{-1/2}\tilde{\boldsymbol{\rho}}_1\|_2^2 + \langle \mathbf{w}, \boldsymbol{\Sigma}_1^{-1/2}\tilde{\boldsymbol{\rho}}_1 - \boldsymbol{\Sigma}_{-1}^{-1/2}\tilde{\boldsymbol{\rho}}_{-1}\rangle \\ & + \sum_{y\in\{-1,1\}}\Big[\frac{1}{n}\mathbf{1}^\top\mathcal{L}(\mathbf{v}_y) + \frac{\beta_y}{\sqrt{n}}\Big\|-\mathbf{v}_y + \langle\tilde{\boldsymbol{\theta}}_y^*, \tilde{\boldsymbol{\theta}}_y\rangle\mathbf{1} + \mathbf{Z}_y\boldsymbol{\Theta}_y\tilde{\boldsymbol{\theta}}_y + \mathbf{g}_y\|\boldsymbol{\Theta}_y^\perp\tilde{\boldsymbol{\theta}}_y\|_2\Big\|_2 \\ & \qquad + \frac{\beta_y}{\sqrt{n}}\langle \mathbf{h}_y, \boldsymbol{\Theta}_y^\perp\tilde{\boldsymbol{\theta}}_y\rangle + \frac{1}{\sqrt{p}}\langle\boldsymbol{\lambda}_y, \tilde{\boldsymbol{\rho}}_y - \tilde{\boldsymbol{\theta}}_y\rangle\Big]\end{aligned} \tag{6}$$

Next, we explicitly solve the minimization over $\boldsymbol{\Theta}_y^\perp\tilde{\boldsymbol{\theta}}_y$ and let $\alpha_y = \|\boldsymbol{\Theta}_y^\perp\tilde{\boldsymbol{\theta}}_y\|_2$ to get

$$\begin{aligned}\min_{\substack{\tilde{\boldsymbol{\rho}}_y,\boldsymbol{\Theta}_y\tilde{\boldsymbol{\theta}}_y\in\mathbb{R}^p\\ \mathbf{v}_y\in\mathbb{R}^{n_y}\\ \alpha_y\in\mathbb{R}_+}}\max_{\substack{\mathbf{w},\boldsymbol{\lambda}_y\in\mathbb{R}^p\\ \beta_y\in\mathbb{R}_+}} \quad & r\|\boldsymbol{\Sigma}_1^{-1/2}\tilde{\boldsymbol{\rho}}_1\|_2^2 + \langle \mathbf{w}, \boldsymbol{\Sigma}_1^{-1/2}\tilde{\boldsymbol{\rho}}_1 - \boldsymbol{\Sigma}_{-1}^{-1/2}\tilde{\boldsymbol{\rho}}_{-1}\rangle \\ & + \sum_{y\in\{-1,1\}}\Big[\frac{1}{n}\mathbf{1}^\top\mathcal{L}(\mathbf{v}_y) + \frac{\beta_y}{\sqrt{n}}\Big\|-\mathbf{v}_y + \langle\tilde{\boldsymbol{\theta}}_y^*, \boldsymbol{\Theta}_y\tilde{\boldsymbol{\theta}}_y\rangle\mathbf{1} + \mathbf{Z}_y\boldsymbol{\Theta}_y\tilde{\boldsymbol{\theta}}_y + \mathbf{g}_y\alpha_y\Big\|_2 \\ & \qquad - \alpha_y\Big\|-\frac{\boldsymbol{\Theta}_y^\perp\boldsymbol{\lambda}_y}{\sqrt{p}} + \frac{\beta_y\boldsymbol{\Theta}_y^\perp\mathbf{h}_y}{\sqrt{n}}\Big\|_2 + \frac{1}{\sqrt{p}}\langle\boldsymbol{\lambda}_y, \boldsymbol{\Theta}_y(\tilde{\boldsymbol{\rho}}_y - \tilde{\boldsymbol{\theta}}_y\rangle + \frac{1}{\sqrt{p}}\langle\boldsymbol{\lambda}_y, \boldsymbol{\Theta}_y^\perp\tilde{\boldsymbol{\rho}}_y\rangle\Big].\end{aligned} \tag{7}$$

Note that this step requires switching the $\min$ and $\max$ (which is allowed by the compactification argument discussed earlier). The next step is to rewrite the $\|\cdot\|_2$ as $\|\cdot\|_2^2$ using the trick $x = \min_{\tau\geq 0}\frac{x^2}{2\tau} + \frac{\tau}{2}$. This results in new decision variables $\tau_y, \eta_y \geq 0$:

$$\min_{\substack{\tilde{\boldsymbol{\rho}}_y, \boldsymbol{\Theta}_y \tilde{\boldsymbol{\theta}}_y \in \mathbb{R}^p \\ \mathbf{v}_y \in \mathbb{R}^{n_y} \\ \alpha_y, \tau_y \geq 0}} \max_{\substack{\mathbf{w}, \boldsymbol{\lambda}_y \in \mathbb{R}^p \\ \beta_y, \gamma_y \geq 0}} \quad r\|\boldsymbol{\Sigma}_1^{-1/2}\tilde{\boldsymbol{\rho}}_1\|_2^2 + \langle \mathbf{w}, \boldsymbol{\Sigma}_1^{-1/2}\tilde{\boldsymbol{\rho}}_1 - \boldsymbol{\Sigma}_{-1}^{-1/2}\tilde{\boldsymbol{\rho}}_{-1}\rangle$$
$$+ \sum_{y\in\{-1,1\}} \left[\frac{1}{n}\mathbf{1}^\top \mathcal{L}(\mathbf{v}_y) + \frac{\beta_y}{2\tau_y n}\left\| -\mathbf{v}_y + \langle \tilde{\boldsymbol{\theta}}_y^*, \boldsymbol{\Theta}_y\tilde{\boldsymbol{\theta}}_y\rangle \mathbf{1} + \mathbf{Z}_y\boldsymbol{\Theta}_y\tilde{\boldsymbol{\theta}}_y + \mathbf{g}_y\alpha_y \right\|_2^2 \right. \tag{8}$$
$$\left. + \frac{\beta_y\tau_y}{2} - \frac{\alpha_y}{2\gamma_y p}\left\| -\frac{\boldsymbol{\Theta}_y^\perp \boldsymbol{\lambda}_y}{\sqrt{p}} + \frac{\beta_y \boldsymbol{\Theta}_y^\perp \mathbf{h}_y}{\sqrt{n}} \right\|_2^2 - \frac{\alpha_y\gamma_y}{2} + \frac{1}{\sqrt{p}}\langle \boldsymbol{\lambda}_y, \boldsymbol{\Theta}_y(\tilde{\boldsymbol{\rho}}_y - \tilde{\boldsymbol{\theta}}_y\rangle + \frac{1}{\sqrt{p}}\langle \boldsymbol{\lambda}_y, \boldsymbol{\Theta}_y^\perp \tilde{\boldsymbol{\rho}}_y\rangle \right].$$

Then, we optimize over $\boldsymbol{\lambda}_y$. This step involves a few intermediate simplifications (completing the square and separately optimizing over $\boldsymbol{\Theta}\boldsymbol{\lambda}$ and $\boldsymbol{\Theta}^\perp\boldsymbol{\lambda}$ separately. These steps are identical to those in (Taheri et al., 2020), and hence the details are omitted here. This step results in the additional constraint $\mu_y = \frac{\langle \tilde{\boldsymbol{\theta}}_y^*, \tilde{\boldsymbol{\rho}}_y\rangle}{\|\tilde{\boldsymbol{\theta}}_y^*\|_2^2}$, which enforces $\boldsymbol{\Theta}_y\tilde{\boldsymbol{\theta}}_y = \boldsymbol{\Theta}_y\tilde{\boldsymbol{\rho}}_y$. Again using strong duality, we write this as a maximization over the dual variables η_y and ultimately obtain

$$\min_{\substack{\tilde{\boldsymbol{\rho}}_y \in \mathbb{R}^p \\ \mathbf{v}_y \in \mathbb{R}^{n_y} \\ \alpha_y, \tau_y \in \mathbb{R}_+ \\ \mu_y \in \mathbb{R}}} \max_{\substack{\mathbf{w} \in \mathbb{R}^p \\ \beta_y, \gamma_y \in \mathbb{R}_+ \\ \eta_y \in \mathbb{R}}} \quad r\|\boldsymbol{\Sigma}_1^{-1/2}\tilde{\boldsymbol{\rho}}_1\|_2^2 + \langle \mathbf{w}, \boldsymbol{\Sigma}_1^{-1/2}\tilde{\boldsymbol{\rho}}_1 - \boldsymbol{\Sigma}_{-1}^{-1/2}\tilde{\boldsymbol{\rho}}_{-1}\rangle$$
$$+ \sum_{y\in\{-1,1\}} \left[\frac{1}{n}\mathbf{1}^\top \mathcal{L}(\mathbf{v}_y) + \frac{\beta_y}{2\tau_y n}\left\| -\mathbf{v}_y + \langle \tilde{\boldsymbol{\theta}}_y^*, \boldsymbol{\Theta}_y\tilde{\boldsymbol{\theta}}_y\rangle \mathbf{1} + \mathbf{Z}_y\boldsymbol{\Theta}_y\tilde{\boldsymbol{\theta}}_y + \mathbf{g}_y\alpha_y \right\|_2^2 \right.$$
$$+ \frac{\beta_y\tau_y}{2} - \frac{\alpha_y\gamma_y}{2} + \frac{\gamma_y}{2p\alpha_y}\left\| \tilde{\boldsymbol{\rho}}_y\sqrt{p} + \frac{\alpha_y\beta_y}{\gamma_y\sqrt{\delta}}\mathbf{h}_y \right\|_2^2 - \frac{\mu_y^2\gamma_y\|\tilde{\boldsymbol{\theta}}_y\|_2^2}{2\alpha_y} - \frac{\alpha_y\beta_y^2\|\boldsymbol{\Theta}_y^\perp\mathbf{h}_y\|^2}{2n\gamma_y} \tag{9}$$
$$\left. - \frac{\alpha_y\beta_y^2}{2p\gamma_y\delta}\|\boldsymbol{\Theta}_y\mathbf{h}_y\|_2^2 - \frac{2\beta_y}{\sqrt{n}}\langle \mathbf{h}, \boldsymbol{\Theta}_y\tilde{\boldsymbol{\rho}}_y\rangle + \eta_y\left(\mu_y - \frac{\langle \tilde{\boldsymbol{\theta}}_y^*, \tilde{\boldsymbol{\rho}}_y\rangle}{\|\tilde{\boldsymbol{\theta}}_y^*\|_2^2}\right) \right]$$

Note here, as in (Taheri et al., 2020), that the terms $\frac{\alpha_y\beta_y^2}{2p\gamma_y\delta}\|\boldsymbol{\Theta}_y\mathbf{h}_y\|_2^2$ and $\frac{2\beta_y}{\sqrt{n}}\langle \mathbf{h}, \boldsymbol{\Theta}_y\tilde{\boldsymbol{\rho}}_y\rangle$ tend to 0 in the limit.

Next, optimization over $\mathbf{v}_1$ and $\mathbf{v}_{-1}$ is done by noting that

$$\min_{\mathbf{v}_1, \mathbf{v}_{-1}} \sum_{y\in\{-1,1\}} \left[\frac{1}{n}\mathbf{1}^\top \mathcal{L}(\mathbf{v}_y) + \frac{\beta_y}{2\tau_y n}\left\| -\mathbf{v}_y + \langle \tilde{\boldsymbol{\theta}}_y^*, \boldsymbol{\Theta}_y\tilde{\boldsymbol{\theta}}_y\rangle \mathbf{1} + \mathbf{Z}_y\boldsymbol{\Theta}_y\tilde{\boldsymbol{\theta}}_y + \mathbf{g}_y\alpha_y \right\|_2^2 \right]$$
$$= \sum_{y\in\{-1,1\}} \pi_y \min_{\mathbf{v}_y \in \mathbb{R}^{n_y}} \left[\frac{1}{n_y}\mathbf{1}^\top \mathcal{L}(\mathbf{v}_y) + \frac{\beta_y}{2\tau_y n_y}\left\| -\mathbf{v}_y + \langle \tilde{\boldsymbol{\theta}}_y^*, \boldsymbol{\Theta}_y\tilde{\boldsymbol{\theta}}_y\rangle \mathbf{1} + \mathbf{Z}_y\boldsymbol{\Theta}_y\tilde{\boldsymbol{\theta}}_y + \mathbf{g}_y\alpha_y \right\|_2^2 \right]$$
$$= \sum_{y\in\{-1,1\}} \pi_y \frac{1}{n_y}\sum_{i=1}^{n_y} \mathcal{M}_{\mathcal{L}}^{(1)}\left(\langle \tilde{\boldsymbol{\theta}}_y^*, \boldsymbol{\Theta}_y\tilde{\boldsymbol{\theta}}_y\rangle + \langle \mathbf{z}_{y,i}, \boldsymbol{\Theta}_y\tilde{\boldsymbol{\theta}}_y\rangle + \alpha_y\mathbf{g}_{y,i}; \frac{\tau_y}{\beta_y} \right)$$
$$= \sum_{y\in\{-1,1\}} \pi_y \frac{1}{n_y}\sum_{i=1}^{n_y} \mathcal{M}_{\mathcal{L}}^{(1)}\left(\mu_y\|\tilde{\boldsymbol{\theta}}_y\|_2^2 + \langle \mathbf{z}_{y,i}, \boldsymbol{\Theta}_y\tilde{\boldsymbol{\theta}}_y\rangle + \alpha_y\mathbf{g}_{y,i}; \frac{\tau_y}{\beta_y} \right)$$

Finally, we optimize over $\tilde{\boldsymbol{\rho}}$. This step differs from that in (Taheri et al., 2020) since it accounts for the extra constraint that $\boldsymbol{\Sigma}_1^{-1/2}\tilde{\boldsymbol{\rho}}_1 = \boldsymbol{\Sigma}_{-1}^{-1/2}\tilde{\boldsymbol{\rho}}_{-1}$.

$$\min_{\substack{\tilde{\boldsymbol{\rho}}_1,\tilde{\boldsymbol{\rho}}_{-1} \\ \text{s.t. } \boldsymbol{\Sigma}_1^{-1/2}\tilde{\boldsymbol{\rho}}_1=\boldsymbol{\Sigma}_{-1}^{-1/2}\tilde{\boldsymbol{\rho}}_{-1}}} r\|\boldsymbol{\Sigma}_1^{-1/2}\tilde{\boldsymbol{\rho}}_1\|_2^2 + \sum_y \left[\frac{\gamma_y}{2p\alpha_y}\left\|\tilde{\boldsymbol{\rho}}_y\sqrt{p} + \frac{\alpha_y\beta_y}{\gamma_y\sqrt{\delta}}\mathbf{h}_y\right\|_2^2 - \frac{\eta_y\langle\tilde{\boldsymbol{\theta}}_y^*, \tilde{\boldsymbol{\rho}}_y\rangle}{\|\tilde{\boldsymbol{\theta}}_y^*\|_2^2}\right]$$

$$= \min_{\substack{\tilde{\boldsymbol{\rho}}_1,\tilde{\boldsymbol{\rho}}_{-1} \\ \text{s.t. } \boldsymbol{\Sigma}_1^{-1/2}\tilde{\boldsymbol{\rho}}_1=\boldsymbol{\Sigma}_{-1}^{-1/2}\tilde{\boldsymbol{\rho}}_{-1}}} r\|\boldsymbol{\Sigma}_1^{-1/2}\tilde{\boldsymbol{\rho}}_1\|_2^2 + \sum_y \left[\frac{\gamma_y}{2p\alpha_y}\left\|\tilde{\boldsymbol{\rho}}_y\sqrt{p} + \frac{\alpha_y\beta_y}{\gamma_y\sqrt{\delta}}\mathbf{h}_y - \frac{\eta_y\alpha_y\sqrt{p}}{\gamma_y\|\tilde{\boldsymbol{\theta}}_y^*\|_2^2}\tilde{\boldsymbol{\theta}}_y^*\right\|_2^2 - \frac{\eta_y^2\alpha_y}{2\gamma_y\|\tilde{\boldsymbol{\theta}}_y^*\|_2^2}\right]$$

Here, we have rewritten the term involving $\mathbf{w}$ as an explicit constraint on the $\tilde{\boldsymbol{\rho}}_y$, using strong duality. Now, note that this is equivalent to optimizing over a single variable $\boldsymbol{\rho} = \boldsymbol{\Sigma}_1^{-1/2}\tilde{\boldsymbol{\rho}}_1 = \boldsymbol{\Sigma}_{-1}^{-1/2}\tilde{\boldsymbol{\rho}}_{-1}$, so we can write this as

$$\min_{\boldsymbol{\rho}} r\|\boldsymbol{\rho}\|_2^2 + \sum_y \left[\frac{\gamma_y}{2p\alpha_y}\left\|\boldsymbol{\Sigma}_y^{1/2}\left(\boldsymbol{\rho}\sqrt{p} + \frac{\alpha_y\beta_y}{\gamma_y\sqrt{\delta}}\boldsymbol{\Sigma}_y^{-1/2}\mathbf{h}_y - \frac{\eta_y\alpha_y\sqrt{p}}{\gamma_y\|\tilde{\boldsymbol{\theta}}_y^*\|_2^2}\boldsymbol{\Sigma}_y^{-1}\boldsymbol{\theta}^*\right)\right\|_2^2 - \frac{\eta_y^2\alpha_y}{2\gamma_y\|\tilde{\boldsymbol{\theta}}_y^*\|_2^2}\right]$$

$$= \frac{r}{p}\sum_{i=1}^p \mathcal{M}_{\ell_2^2}^{(2)}\left(\frac{\alpha_y\beta_y}{\gamma_y\sqrt{\delta\lambda_{y,i}}}\mathbf{h}_{y,i} + \frac{\eta_y\alpha_y\sqrt{p}}{\gamma_y\|\tilde{\boldsymbol{\theta}}_y^*\|_2^2\lambda_{y,i}}\boldsymbol{\theta}_i^*; \frac{\alpha_y r}{\gamma_y\lambda_{y,i}}\right) - \sum_y \frac{\eta_y^2\alpha_y}{2\gamma_y\|\tilde{\boldsymbol{\theta}}_y^*\|_2^2}$$

We have now expressed the AO entirely in terms of a scalar optimization problem. We then can consider the asymptotics of each term in the objective function of the scalarized AO, using the fact that $\|\tilde{\boldsymbol{\theta}}_y^*\|_2 \to \zeta_y$ and the Moreau envelope terms converge in probability to their expectations as $n, p \to \infty$. This shows pointwise convergence of the objective function. We can use the fact that pointwise convergence of continuous, convex functions over compact sets is uniform (e.g., Cor II.1 in (Andersen & Gill, 1982)) to conclude that the optimal value of the AO converges to the optimal value of the scalar min-max problem. Hence, by the CGMT, the optimal value of the original ERM problem (the PO) also converges in probability to this value.

We now need to extend this to a statement about the *test error*, as given in Lemma 2. By Lemma 7 in (Taheri et al., 2020), the desired result follows if we can show that the solution $\tilde{\boldsymbol{\theta}}_y$ of the POE satisfies

$$\frac{\langle\tilde{\boldsymbol{\theta}}_y, \tilde{\boldsymbol{\theta}}_y^*\rangle}{\|\tilde{\boldsymbol{\theta}}_y^*\|_2^2} \xrightarrow{P} \mu_y^*, \quad \|\boldsymbol{\Theta}_y^\perp\tilde{\boldsymbol{\theta}}_y\|_2 \xrightarrow{P} \alpha_y^*,$$

where μ_y^* and α_y^* are the optimal decision variables from 1. The proof of these two statements are identical, so we only show the latter. First fix $\epsilon > 0$ and define the sets $\mathcal{S}_y = \left\{\tilde{\boldsymbol{\theta}}_y : \left|\|\boldsymbol{\Theta}_y^\perp\tilde{\boldsymbol{\theta}}_y\|_2 - \alpha_y^*\right| < \epsilon\right\}$. Now we consider the same AO but with the additional constraint $\tilde{\boldsymbol{\theta}}_y \in \mathcal{S}_y^C$. By assumption that the scalar min-max in Theorem 1 has a unique solution, this can only strictly increase the optimal value with probability approaching 1. Hence, we can apply the third part of Theorem 3 in (Dhifallah & Lu, 2021) to conclude that $\mathbb{P}\{\tilde{\boldsymbol{\theta}}_y \notin \mathcal{S}_y\} \to 0$ and the desired convergence in probability to α_y^* holds. Substituting these values into the expression for the POE in Lemma 7 of (Taheri et al., 2020), we can conclude that the per-class POE converges to the desired quantity.

□

A.2. Additional details for the numerical simulations

In this section, we provide additional details about the results displayed in Figure 2.

In each of the three spectral imbalance settings, we apply Theorem 1 with $\pi_y = 0.5$, the overparameterization ratio $\delta = 2$, regularization parameter $r = 0.5$ and the loss as the squared hinge loss, $\mathcal{L}(t) = \max(0, 1-t)^2$. The scalar min-max problem is solved using gradient descent/ascent with learning rate 0.01. The choice of Π used for each of the three settings is given below:

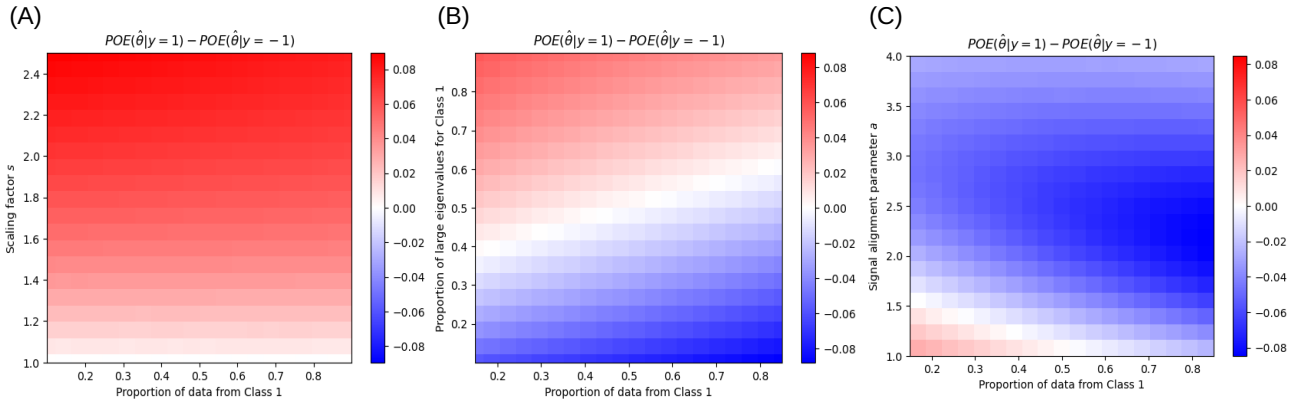

Figure 1. Interplay of sample and spectral imbalance: Heatmap of the class gap across different amounts of sample imbalance and spectral imbalance. Settings (A), (B), and (C) correspond to the same three types of spectral imbalance considered in Section 3.3 and the simulation details provided in Appendix A.2.

1. **Impact of eigenvalue scaling (Setting (A)):**

$$(\sqrt{p}\boldsymbol{\theta}_i^*, \lambda_i^{(1)}, \lambda_i^{(-1)}) \overset{\text{i.i.d.}}{\sim} \begin{cases} (1, 2s, 2) & \text{with prob. } 0.5 \\ (1, 0.5s, 0.5) & \text{with prob. } 0.5 \end{cases}$$

2. **Impact of eigenvalue decay (Setting (B)):**

$$(\sqrt{p}\boldsymbol{\theta}_i^*, \lambda_i^{(1)}, \lambda_i^{(-1)}) \overset{\text{i.i.d.}}{\sim} \begin{cases} (1, 2, 2) & \text{with prob. } 0.5p \\ (1, 2, 0.5) & \text{with prob. } 0.5p \\ (1, 0.5, 2) & \text{with prob. } 0.5(1-p) \\ (1, 0.5, 0.5) & \text{with prob. } 0.5(1-p) \end{cases}$$

3. **Impact of alignment with target signal (Setting (C)):**

$$(\sqrt{p}\boldsymbol{\theta}_i^*, \lambda_i^{(1)}, \lambda_i^{(-1)}) \overset{\text{i.i.d.}}{\sim} \begin{cases} (1, 2, 0.5) & \text{with prob. } 0.5 \\ (a, 0.5, 2) & \text{with prob. } 0.5 \end{cases}$$

In each case, the numerical simulations are averaged over 50 independent draws of $n = 1000$ training examples (so $p = 500$) from the GMM. For each trial, we solve the ERM objective function using gradient descent.

A.3. The interplay of sample imbalance and spectral imbalance

In Figure 2, we aimed to isolate the effect of spectral imbalance, so we assume that the training data is *sample-balanced*, i.e., $\pi_1 = \pi_{-1} = \frac{1}{2}$ and an equal proportion of the training data comes from each class. The asymptotic predictions in Theorem 1, however, apply for general π_y, allowing for a careful study of the interplay between these aspects of the model setting.

In Figure 1, we provide the exact asymptotic predictions for this interplay in the three spectrally-imbalanced settings we introduce in Section 3.3. These results reveal a nuanced relationship between these factors: for a fixed amount of spectral imbalance, data imbalance can either reduce or exacerbate the class gap, depending on the "direction" of the imbalance. Interestingly, we find that this relationship depends crucially on the type of spectral imbalance in the dataset. For example, in setting (A), sample imbalance seems to have little asymptotic effect on the class gap for a fixed amount of spectral imbalance. By contrast, in settings (B) and (C), the sample imbalance and spectral imbalance both have a pronounced role in determining the class gap.

B. Additional Experimental Details

In Section B.1, we first provide additional visualizations to supplement the visualizations presented in the main text. In Section B.2, we record the Pearson correlation coefficient score and other statistical metrics for each encoder. In Section B.4, we record the performance and details of the pre-trained models studied. In Section B.5, we record the details of the ImageNet-C dataset created and used in Section 4.4 and also expand on the results in Section 4.4.

B.1. Additional visualizations

In Figure 2, we visualize the eigenspectrum in VIT-B as an additional example to complement Figure 1(A), which visualized the eigenspectrum for ResNet-50. Specifically, the left panel plots the eigenspectrum on regular scale (as in Figure 1(A)) for VIT-B; the right panel plots the eigenspectrum on log-log scale to reveal the power law behavior, which was also observed and analyzed for ResNet-50 in Section 4.

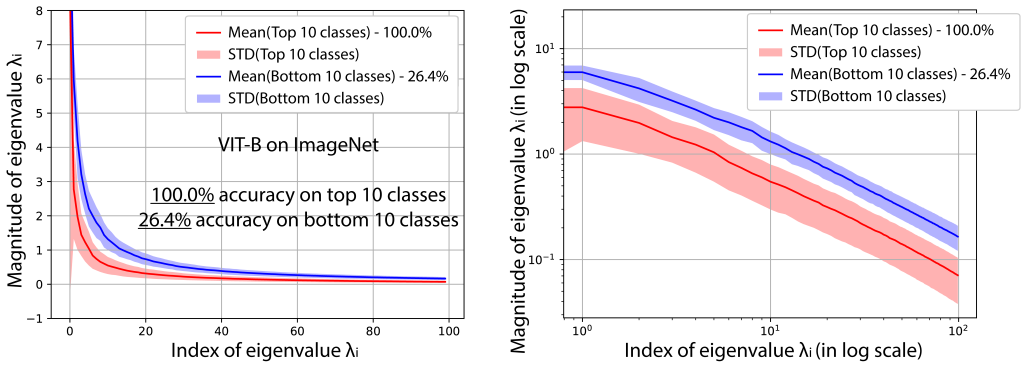

Figure 2. Eigenspectrum visualization plots in normal scale and log scale for VIT-B.

Additionally, in Figure 3, we show the direct visualization of eigenspectrum in other encoders. The distributional differences across classes are consistent and strong in all observed representation spaces.

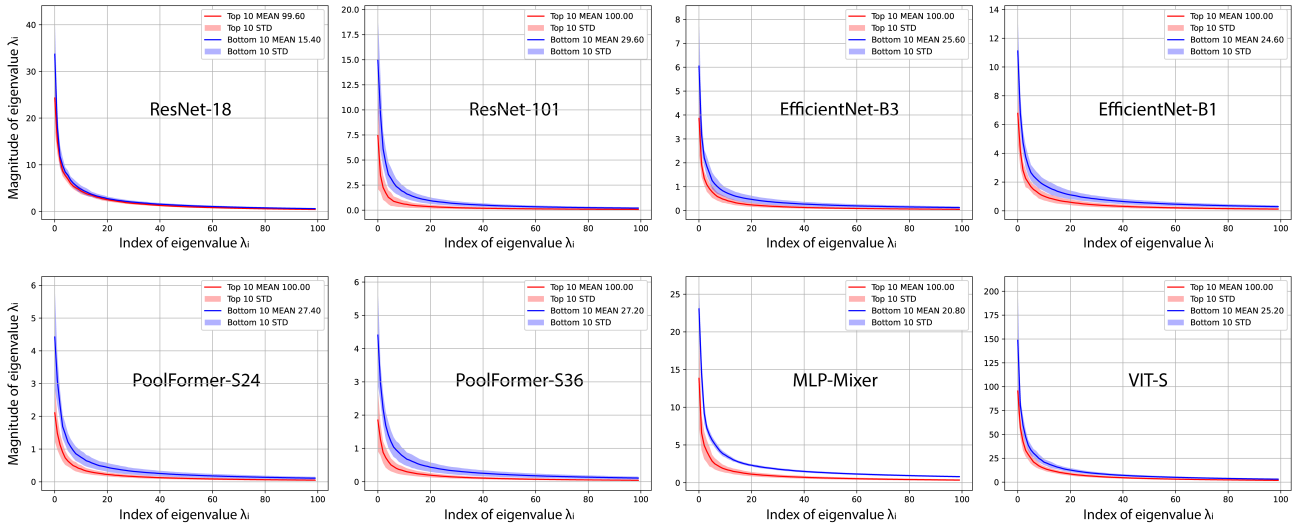

Figure 3. Eigenspectrum visualization plots for other encoders.

In Figure 4, we show that the distribution of eigenvalues across encoders are different for additional eigenvalue positions. The same figure as in Figure 1(B) (which was plotted for λ_5) is plotted for λ_{50} and λ_{100}. We observed similar trends for all other eigenvalue indexes as well.

Finally, in Figure 5, we show the Pearson correlation coefficient plots corresponding to Figure 3(B) (which corresponded to

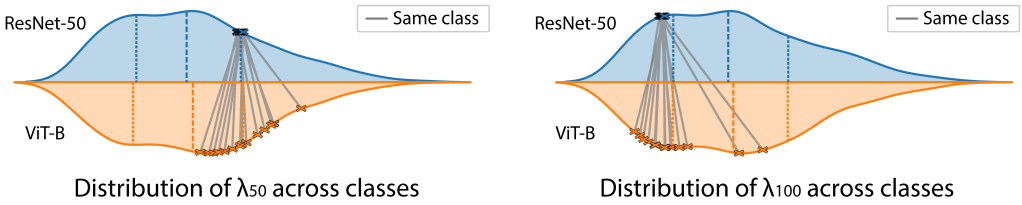

Figure 4. Comparing the eigenvalue distributions across encoders for λ_{50} and λ_{100} (for all classes $C = 1000$).

ResNet-50) for all the other models.

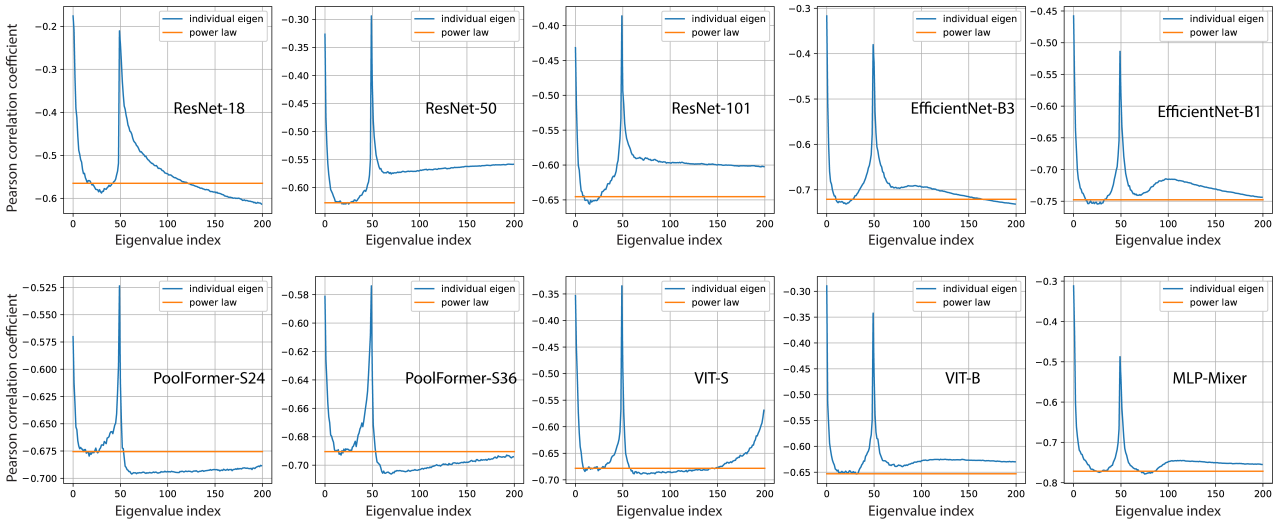

Figure 5. Correlation plots for other encoders.

B.2. Correlation analysis of all the encoders

First, we systematically record the average accuracy of the bottom 10, bottom 100, top 100, and top 10 classes of each encoder in Table 1 to highlight the extent of the class bias.

Table 1. The prevalence of class bias across encoders.

MODEL	BOTTOM 10	BOTTOM 100	TOP 100	TOP 10
DENSENET	21.40%	40.76%	96.44%	99.80%
EFFICIENTNET-B1	24.60%	44.00%	97.64%	100.00%
EFFICIENTNET-B3	25.60%	44.64%	98.00%	100.00%
RESNET-18	15.40%	34.06%	94.34%	99.60%
RESNET-50	28.80%	46.94%	98.40%	100.00%
RESNET-101	29.60%	49.44%	98.64%	100.00%
VIT-S	25.20%	41.22%	96.02%	100.00%
VIT-B	26.40%	48.86%	98.54%	100.00%
MLPMIXER	20.80%	38.52%	96.08%	100.00%
POOLFORMER-S24	27.40%	48.10%	98.60%	100.00%
POOLFORMER-S36	27.20%	49.04%	98.80%	100.00%
AVG	24.76%	44.14%	97.41%	99.95%

Second, we record the Pearson correlation coefficient (PCC) between per-class accuracy and class-dependent eigenspectrum properties for each encoder in Table 2. Aside from the individual eigenvalue scaling and power law offset, we also record

the power law slope and two effective ranks that measure the *rate* of eigenvalue decay in different ways. The definition of the studied two effective ranks are as below:

Definition 3 (**Effective Rank**). For any covariance matrix (spectrum) $\mathbf{\Sigma}$, we define its two effective ranks:

$$\rho_k(\mathbf{\Sigma}) = \frac{\sum_{i>k} \lambda_i}{\lambda_{k+1}}, \quad R_k(\mathbf{\Sigma}) := \frac{(\sum_{i>k} \lambda_i)^2}{\sum_{i>k} \lambda_i^2}. \tag{10}$$

This definition was proposed in (Bartlett et al., 2020) and has been shown to constitute a "sufficient" statistic to explain average generalization error across regression and classification tasks (Bartlett et al., 2020; Tsigler & Bartlett, 2020; Muthukumar et al., 2021; Wang & Thrampoulidis, 2021). We examine the relationship between effective ranks for $k = 0$ and defer a more detailed study for other values of k to future work.

Table 2. Examining the correlation between spectral properties and per-class accuracy on different pre-trained encoders.

MODEL	MAX CORR.	PL CORR a.	PL CORR b.	EFFECTIVE RANK 1	EFFECTIVE RANK 2
DENSENET	-0.6677	-0.6636	-0.0598	-0.5107	-0.5384
EFFICIENTNET-B1	-0.7599	-0.7476	-0.0107	-0.6283	-0.6131
EFFICIENTNET-B3	-0.7505	-0.7210	0.0007	-0.6209	-0.5816
RESNET-18	-0.6229	-0.5649	0.0968	-0.4434	-0.4722
RESNET-50	-0.6293	-0.6270	0.0447	-0.4943	-0.4952
RESNET-101	-0.6560	-0.6454	0.0170	-0.4869	-0.4896
VIT-S	-0.6891	-0.6783	-0.0571	-0.5195	-0.5399
VIT-B	-0.6536	-0.6531	-0.1493	-0.4824	-0.4934
MLPMIXER	-0.7785	-0.7715	0.0460	-0.7074	-0.6707
POOLFORMER-S24	-0.6961	-0.6755	0.0708	-0.5544	-0.5895
POOLFORMER-S36	-0.7062	-0.6904	0.0006	-0.5652	-0.5638
AVG	-0.6918	-0.6762	-0.0002	-0.5466	-0.5497

Interestingly, we observe across models that the spectral metrics that measure *relative rate* of eigenvalue decay — such as the power law decay parameter b and the effective ranks — do not strongly correlate with class-wise accuracy, while the spectral metrics that are more absolute — i.e. the individual eigenvalues and the power law parameter a — do strongly correlate.

B.3. Correlation analysis on smaller datasets

Aside from ImageNet, we also perform the same correlation analysis on smaller-scale datasets like CIFAR-10 and FASHION-MNIST to demonstrate the versatility of the proposed method. As shown in Table 3, the negative correlation is stronger in smaller-scale datasets with fewer classes.

Table 3. Examining the correlation between spectral properties and per-class accuracy on smaller datasets.

MODEL	CIFAR-10	FASHION-MNIST	IMAGENET
DENSENET	-0.8750	-0.8045	-0.6636
EFFICIENTNET-B1	-0.8104	-0.7330	-0.7476
EFFICIENTNET-B3	-0.8906	-0.5591	-0.7210
RESNET-18	-0.8311	-0.7634	-0.5649
RESNET-34	-0.8334	-0.6677	-0.5772
RESNET-50	-0.8859	-0.8605	-0.6270
VIT-S	-0.7293	-0.7922	-0.6783
VIT-B	-0.8049	-0.4837	-0.6531
MLPMIXER	-0.7228	-0.9127	-0.7715
POOLFORMER-S24	-0.8477	-0.6522	-0.6755
POOLFORMER-S36	-0.8173	-0.7057	-0.6904
AVG	-0.8226	-0.7213	-0.6700

B.4. Details of pretrained networks

In this section we record the details of the pre-trained networks that we used and evaluated to allow for reproducibility of our results.

- **Residual Networks.** ResNet is instrumental in many visual recognition tasks, as it addresses the vanishing gradient problem, allowing very deep networks to be trained effectively. In this paper, we used the pre-trained weights for ResNet from TorchVision. We list their performance, model card name, and parameters below.
  - For ResNet-101, we used the v2 weight from the new training recipe of torchvision. The Top1 accuracy is 81.886% and Top5 accuracy is 95.78% on ImageNet-1K, with 44.5M parameters. Its representation dimensionality is equal to 2048.
  - For ResNet-50, we used the v2 weight from the new training recipe of torchvision. The Top1 accuracy is 80.858% and Top5 accuracy is 95.434% on ImageNet-1K, with 25.6M parameters. Its representation dimensionality is equal to 2048.
  - For ResNet-18, we used the default weight (or v1 weight from torchvision). The Top1 accuracy is 69.758% and Top5 accuracy is 89.078% on ImageNet-1K, with 11.7M parameters. Its representation dimensionality is equal to 512.

- **Improved CNNs.** Many research works propose variants of convolutional architectures to improve ResNets. Among these efforts, DenseNet (Huang et al., 2017) and EfficientNet (Tan & Le, 2019) are two representative architectures that are more parameter-efficient. They design different strategies to connect layers differently to improve the parameter-efficiency as well as information flow inside the networks. DenseNet improves the flow of information and gradients throughout the network by densely connecting all layers directly with each other. EfficientNet designs a compound coefficient that uniformly scales the depth, width, and resolution of CNNs and thus improves performance with fewer parameters.
  - For DenseNet, we used the weights from timm (model card `densenet121.ra_in1k`). The Top1 accuracy is 75.57% and Top5 accuracy is 92.61%, with 8.0M parameters. Its representation dimensionality is equal to 1024.
  - For EfficientNet-B3, we used the weights from timm (model card `efficientnet_b3.ra2_in1k`). The Top1 accuracy is 77.60% and Top5 accuracy is 93.59%, with 12.2M parameters. Its representation dimensionality is equal to 1536.
  - For EfficientNet-B1, we used the weights from timm (model card `efficientnet_b1.ft_in1k`), The Top1 accuracy is 78.54% and Top5 accuracy is 94.38%, with 7.8M parameters. Its representation dimensionality is equal to 1280.

- **Vision Transformers.** Recently, the transformer architecture has achieved strong performance even in the vision domain (Dosovitskiy et al., 2020). The vision transformer splits an image into patches and uses self-attention to study the interaction of patches. By doing so, it aggregates global information at an early stage, creating significantly different representations than CNNs (Raghu et al., 2021).
  - For VIT-B, we used the weights from timm (`vit_base_patch16_224.augreg2_in21k_ft_in1k`), The Top1 accuracy is 81.10% and Top5 accuracy is 95.72%, with 86.6M parameters. Its representation dimensionality is equal to 768. We note that the model is pre-trained on ImageNet21k and later fine-tuned on ImageNet1k.
  - For VIT-S, we used the weights from timm (`vit_small_patch16_224.augreg_in21k_ft_in1k`). The Top1 accuracy is 74.63% and Top5 accuracy is 92.67%, with 22.1M parameters. Its representation dimensionality is equal to 384. We note that the model is pre-trained on ImageNet21k and later fine-tuned on ImageNet1k.

- **Transformers without Self-Attention.** Recent studies show that self-attention is not required for obtaining good representations in vision. We study in particular MLP-Mixer (Tolstikhin et al., 2021) and PoolFormer (Yu et al., 2022), which uses multi-layer perceptrons and average pooling to mix information across patches, respectively.
  - For MLP-Mixer, we used weights from timm (model card `mixer_b16_224.goog_in21k_ft_in1k`). The Top1 accuracy is 72.57% and Top5 accuracy is 90.06%, with 59.9M parameters. Its representation dimensionality is equal to 768. We note that it is pre-trained on ImageNet21k and later fine-tuned on ImageNet1k.

– For PoolFormer-S36, we used the weights from timm (model card `poolformerv2_s36.sail_in1k`). The Top1 accuracy is 81.54% and Top5 accuracy is 95.69%, with 30.8M parameters. Its representation dimensionality is equal to 512.
– For PoolFormer-S24, we used the weights from timm (model card `poolformerv2_s24.sail_in1k`). The Top1 accuracy is 80.76% and Top5 accuracy is 95.39%, with 21.3M parameters. Its representation dimensionality is equal to 512.

B.5. Details of ImageNet-C results

We created the "augmented" ImageNet-C datasets based on (Hendrycks & Dietterich, 2019). The preprocessing plans (corruptions) we selected are:

- Gaussian Noise, Shot Noise, Impulse Noise, Defocus Blur, Motion Blur, Zoom Blur, Snow, Fog, Brightness, Contrast, Elastic, Pixelate, JPEG, Speckle Noise, Spatter, and Saturate.

Note that these corruptions correspond to the 16 augmentations discussed in Section 4.4. While (Hendrycks & Dietterich, 2019) only augment the test dataset, we perform the image altering operations on the training set as well, as we wish to estimate the eigenvalues using the training data.

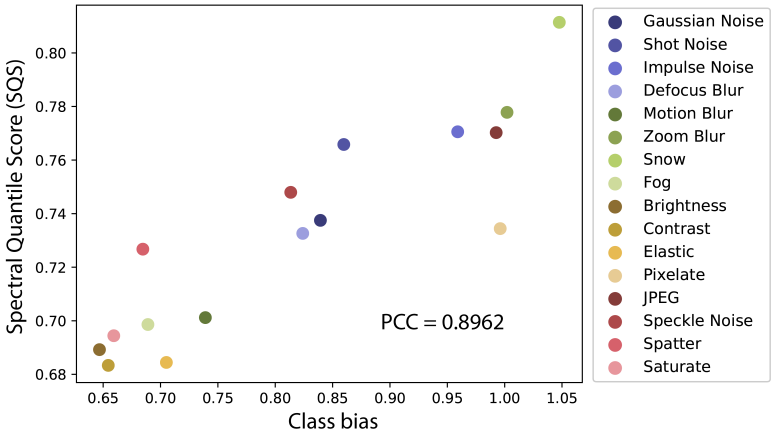

Figure 6. Spectral Quantile Score (SQS) of different data manipulation plans.

In Figure 6, we show the Spectral Quantile Score (SQS) of applying the same encoder on different data with different data manipulation plans (as discussed in Section 4.4). The PCC is 0.90, showing strong correlation (again) between class bias and the SQS.

For the statistical significance results for each class, we first compute the Pearson correlation coefficient r for each class, and calculate the p-value as below:

$$p = 2P(T_{n-2} > t), \text{ where } t = \frac{r\sqrt{n-2}}{\sqrt{1-r^2}} \tag{11}$$

where n is the number of samples, and T_{n-2} is a Student t's distribution with $n-2$ degrees of freedom.

Complete results We repeat the experiments to systematically demonstrate that the proposed ensemble method works across many different combinations and types of augmentations. Here, for each version of ImageNet-C (out of 16 total versions), we apply the ensemble method based on the other 15 types of corruption, creating 16x15 = 240 runs. For each column, we report the average and best improvement in overall accuracy obtained by ensembling. As shown in Table 4, across 240 runs, our proposed ensemble method can improve the average augmentation and worse augmentation by 2.6% and 5.9%, respectively.

B.6. Comparing spectral imbalance and semantic scale

The authors of (Ma et al., 2022) study the overall volume of each class (related to the product of eigenvalues of the class-covariances) and its correlation with class disparities. In contrast, our work reveals dependence on the eigenvalue

Table 4. Full improvement results on ImageNet-C.

Improvement	Gaussian Noise	Shot Noise	Impulse Noise	Defocus Blur	Motion Blur	Zoom Blur	Snow	Fog	Brightness
Δ(AVG, Ours)	0.019	0.022	0.031	0.019	0.017	0.039	0.040	0.024	0.034
Δ(MIN, Ours)	0.044	0.048	0.068	0.045	0.044	0.080	0.081	0.056	0.074

Improvement	Contrast	Elastic	Pixelate	JPEG	Speckle Noise	Spatter	Saturate	**Imp. Avg.**
Δ(AVG, Ours)	0.030	0.024	0.033	0.010	0.020	0.024	0.030	0.026
Δ(MIN, Ours)	0.066	0.054	0.075	0.075	0.045	0.057	0.064	0.059

distributions rather than the overall volume, which we believe provides a more fine-grained characterization of feature imbalances.

To demonstrate the difference between the two approaches, we compare our results to those reported in (Ma et al., 2022) for the case of sample-balanced data. Interestingly, we obtain similar overall levels of predictive accuracy (as measured by the Pearson correlation coefficient between the spectra and the per-class performance) to their proposed inter-cluster corrected measure (S), which takes into account inter-cluster distances in addition to the eigenvalue volume (S'). In contrast, the spectral volume (S') itself does not have as strong predictive performance. Thus, this shows that a more fine-grained analysis of the eigenvalues can have a big benefit when describing class gaps in the sample-balanced setting.

Table 5. Direct comparison with the semantic scale method on CIFAR-10.

Model	Method	Absolute correlation
ResNet-18	(Ma et al., 2022) (S')	0.5433
ResNet-18	(Ma et al., 2022) (Best S)	0.7850
ResNet-18	Ours	**0.8311**
ResNet-18	Δ	0.0461
ResNet-34	(Ma et al., 2022) (S')	0.5750
ResNet-34	(Ma et al., 2022) (Best S)	0.8056
ResNet-34	Ours	**0.8334**
ResNet-34	Δ	0.0278