Unraveling Attention via Convex Duality:
Analysis and Interpretations of Vision Transformers

Arda Sahiner[1] **Tolga Ergen**[1] **Batu Ozturkler**[1] **John Pauly**[1] **Morteza Mardani**[1] **Mert Pilanci**[1]

Abstract

Vision transformers using self-attention or its proposed alternatives have demonstrated promising results in many image related tasks. However, the underpinning inductive bias of attention is not well understood. To address this issue, this paper analyzes attention through the lens of convex duality. For the non-linear dot-product self-attention, and alternative mechanisms such as MLP-mixer and Fourier Neural Operator (FNO), we derive equivalent finite-dimensional convex problems that are interpretable and solvable to global optimality. The convex programs lead to *block nuclear-norm regularization* that promotes low rank in the latent feature and token dimensions. In particular, we show how self-attention networks implicitly clusters the tokens, based on their latent similarity. We conduct experiments for transferring a pre-trained transformer backbone for CIFAR-100 classification by fine-tuning a variety of convex attention heads. The results indicate the merits of the bias induced by attention compared with the existing MLP or linear heads.

1. Introduction

Transformers have recently delivered tremendous success for representation learning in language and vision. This is primarily due to the attention mechanism that effectively mixes the tokens' representation over the layers to learn the semantics present in the input[1]. After the inception of dot-product self-attention (Vaswani et al., 2017), there have been several efficient alternatives that scale nicely with the sequence size for large pretraining tasks; see e.g., (Wang et al., 2020; Shen et al., 2021; Kitaev et al., 2020; Panahi et al., 2021; Xiong et al., 2021). However, the learnable inductive bias of attention is not explored well. A strong theoretical understanding of attention's inductive bias can motivate designing more efficient architectures, and can explain the generalization ability of these networks.

Self-attention was the fundamental building block in the first proposed vision transformer (ViT) (Dosovitskiy et al., 2020). It consists of an outer product of two linear functions, followed by a non-linearity and a product with another linear function, which makes it non-convex and non-interpretable. One approach to understand attention has been to design new alternatives to self-attention which perform similarly well, which may help explain its underlying mechanisms. One set of work pertains to Multi-Layer Perceptron (MLP) based architectures, (Tolstikhin et al., 2021; Tatsunami & Taki, 2021; Touvron et al., 2021; Liu et al., 2021; Yu et al., 2021), while another line of work proposes Fourier based models (Lee-Thorp et al., 2021; Rao et al., 2021; Li et al., 2020; Guibas et al., 2021). Others have proposed replacing self-attention with matrix decomposition (Geng et al., 2021). While all of these works have appealing applications that leverage general concepts about the structure of attention, they lack any fine-tuned theoretical analysis on these architectures from an optimization perspective.

To address this shortcoming, we leverage convex duality to analyze a single block of self-attention with ReLU activation. Since self-attention incurs quadratic complexity in the sequence, we alternatively analyze more efficient modules. As representatives for more efficient modules, we focus on MLP Mixers (Tolstikhin et al., 2021) and Fourier Neural Operators (FNO) (Li et al., 2020). MLP-mixer mixes tokens (purely) using MLP projections in both the token and feature dimensions. In contrast, FNO mixes tokens using circular convolution that is efficiently implemented based on 2D Fourier transforms.

We find that all three of these analyzed modules are equivalent to finite-dimensional convex optimization problems, indicating that there are guarantees to provably optimize them to their global optima. Furthermore, we make novel observations about the bias induced by the convex models. In particular, convexified equivalents of both self-attention

[1] Department of Electrical Engineering, Stanford University, Stanford, CA, USA. Correspondence to: Arda Sahiner <sahiner@stanford.edu>.

Proceedings of the 39^{th} International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022. Copyright 2022 by the author(s).

[1] We use the terms "attention", "token mixing", and "mixing" interchangeably throughout this paper.

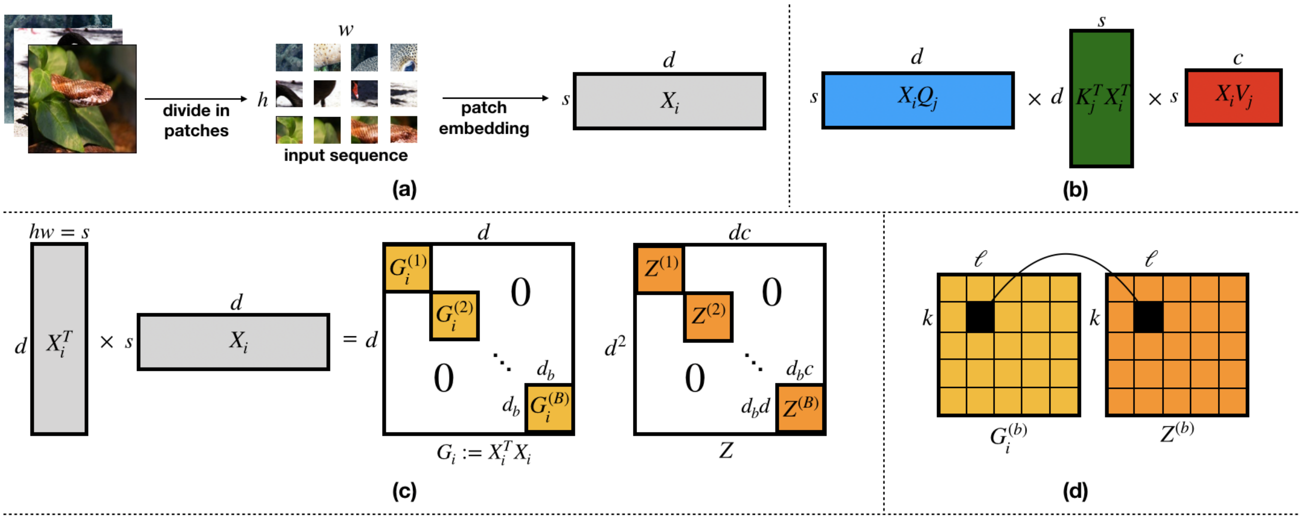

Figure 1. Illustration of the implicit convexity of (linear activation) multi-head self-attention. (a) Input image is first divided into $hw = s$ patches, where each patch is represented by a latent vector of dimension d. (b) The (non-convex) scaled dot-product self-attention applies learnable weights $\mathbf{Q}_j, \mathbf{K}_j, \mathbf{V}_j$ to the patch embeddings X_i as in (10). (c) In the equivalent convex optimization problem for the self-attention training objective (14), the Gram matrix G_i is formed that groups latent features in B different blocks, (d) and accordingly the nuclear norm regularization is imposed on the dual variables $\mathbf{Z}^{(b,k,l)} \in \mathbb{R}^{d\times c}$ based on the similarity scores $\mathbf{G}_i^{(b)}[k, l]$.

and MLP-Mixer modules resemble weighted combinations of MLPs, but with additional degrees of freedom (e.g. higher dimensional induced parameters), and a unique block nuclear norm regularization which ties their individual submodules together to encourage global cohesiveness. In contrast, the convexified FNO mixer amounts to circular convolution, while slight modifications to the FNO architecture can induce an equivalent group-wise convolution. We experimentally test and compare these convex attention heads for transfer learning of a pre-trained vision transformer to CIFAR-100 classification upon finetuning a single convex head. We observe that the inductive bias of these attention modules outperforms traditional convex models.

Our main contributions are summarized as follows:

- We provide guarantees that self-attention, MLP-Mixer, and FNO with linear and ReLU activation can be solved to their global optima by demonstrating their equivalence to convex optimization problems.
- By analyzing these equivalent convex programs, we provide interpretability to the optimization objectives of these attention modules.
- Our experiments validate the (convex) vision transformers perform better than baseline convex methods in a transfer learning task.

1.1. Related Work

This work is primarily related to two lines of research.

Interpreting attention. One approach has been to experimentally observe the properties of attention networks to understand them. For example, DINO proposes a contrastive self-supervised learning method for ViTs (Caron et al., 2021). It is observed that learned attention maps preserve semantic regions of the images. Another work compares the alignment across layers of trained ViTs and CNNs, concluding that ViTs have more a uniform representation structure across layers of the network (Raghu et al., 2021). Another work uses a Deep Taylor Decomposition approach to visualize portions of input image leading to a particular ViT prediction (Chefer et al., 2021).

Another approach is to analyze the expressivity of attention networks. One work interprets multi-head self-attention as a Bayesian inference, and provides tools to decide how many heads to use, and how to enforce distinct representations in different heads (An et al., 2020). Other analysis has demonstrated that sparse transformers can universally approximate any function (Yun et al., 2020), that multi-head self-attention is at least as expressive as a convolutional layer (Cordonnier et al., 2019), and that dot-product self-attention is not Lipschitz continuous (Kim et al., 2021).

Convex neural networks. Starting with (Pilanci & Ergen, 2020), there has been a long line of work demonstrating that various ReLU-activation neural network architectures have equivalent convex optimization problems. These include two-layer convolutional and vector-output networks (Ergen & Pilanci, 2020; Sahiner et al., 2020a;b), as well as deeper ones (Ergen & Pilanci, 2021), networks with Batch Normalization (Ergen et al., 2021), and Wasserstein GANs (Sahiner et al., 2021). Recent work has also demonstrated how to efficiently optimize the simplest forms of these equivalent convex networks, and incorporate additional constraints for adversarial robustness (Mishkin et al., 2022; Bai et al., 2022). However, none of these works have analyzed the building blocks of transformers, which are a leading method in many state-of-the-art vision and language processing tasks.

2. Preliminaries

In general, we analyze supervised learning problems where input training data $\{\mathbf{X}_i \in \mathbb{R}^{s\times d}\}_{i=1}^n$ are the result of a patch embedding layer, and we have corresponding labels of arbitrary size $\{\mathbf{Y}_i \in \mathbb{R}^{r\times c}\}_{i=1}^n$. For arbitrary convex loss function $\mathcal{L}(\cdot,\cdot)$, we solve the optimization problem

$$p^* := \min_\theta \sum_{i=1}^n \mathcal{L}\left(f_\theta(\mathbf{X}_i), \mathbf{Y}_i\right) + \mathcal{R}(\theta) \qquad (1)$$

for some learnable parameters θ, neural network $f_\theta(\cdot)$, and regularizer $\mathcal{R}(\cdot)$. Note that this formulation encapsulates both denoising and classification scenarios: in the classification setting of $r=1$, one can absorb global average pooling into the convex loss $\mathcal{L}$, whereas if $r=s$, one can directly use squared loss or other convex loss functions. One may also use this formulation to apply to both supervised and self-supervised learning.

In this paper, we denote $(\cdot)_+ := \max\{0,\cdot\}$ as the ReLU non-linearity. We use superscripts, say $\mathbf{A}^{(i_1,i_2)}$, to denote blocks of matrices, and brackets, say $\mathbf{A}[i_1,i_2]$, to denote elements of matrices, where the arguments refer to row (or block of rows) i_1 and column (or block of columns) i_2.

2.1. Implicit Convexity of Linear and ReLU MLPs

Previously, it has been demonstrated that standard two-layer ReLU MLPs are equivalent to convex optimization problems. We briefly describe the relevant background to provide context for much of the analysis in this paper. In particular, we are presented with a network with m neurons in the hidden layer, weight-decay parameter $\beta > 0$, and data $\mathbf{X}\in\mathbb{R}^{n\times d}$, $\mathbf{Y}\in\mathbb{R}^{n\times c}$:

$$p^*_{RMLP} := \min_{\mathbf{w}_{2j},\mathbf{w}_{2j}} \mathcal{L}\left(\sum_{j=1}^m (\mathbf{X}\mathbf{w}_{1j})_+\mathbf{w}_{2j}^\top, \mathbf{Y}\right) + \frac{\beta}{2}\sum_{j=1}^m \left(\|\mathbf{w}_{1j}\|_2^2 + \|\mathbf{w}_{2j}\|_2^2\right). \qquad (2)$$

While this problem is non-convex as stated, it has been demonstrated that the objective is equivalent to the solution of an equivalent convex optimization problem, and there is a one-to-one mapping between the two problems' solutions (Sahiner et al., 2020a). In particular, this analysis makes use of *hyperplane arrangements*, which enumerate all possible activation patterns of the ReLU non-linearity:

$$\mathcal{D} := \{\mathrm{diag}\left(\mathbb{1}\{\mathbf{X}\mathbf{u}\ge 0\}\right) : \mathbf{u}\in\mathbb{R}^d\} \qquad (3)$$

The set $\mathcal{D}$ is clearly finite, and its cardinality is bounded as $P := |\mathcal{D}| \le 2r\left(\frac{e(n-1)}{r}\right)^r$, where $r := \mathrm{rank}(\mathbf{X})$ (Stanley et al., 2004; Pilanci & Ergen, 2020). Through convex duality analysis, we can express an equivalent convex optimization problem by enumerating over the finite set of arrangements $\{\mathbf{D}_j\}_{j=1}^P$. We define the following norm

$$\|\mathbf{Z}\|_{*,\mathrm{K}} := \min_{t\ge 0} t \text{ s.t. } \mathbf{Z}\in t\mathcal{C} \qquad (4)$$
$$\mathcal{C} := \mathrm{conv}\{\mathbf{Z} = \mathbf{u}\mathbf{v}^\top : \mathrm{K}\mathbf{u}\ge 0, \|\mathbf{Z}\|_*\le 1, \mathbf{v}\in\mathbb{R}^c\}.$$

This norm is a quasi-nuclear norm, which differs from the standard nuclear norm in that the factorization upon which it relies puts a constraint on its left factors, which in our case will be an affine constraint. In convex ReLU neural networks, K is chosen to enforce the existence of $\{\mathbf{u}_k,\mathbf{v}_k\}$ such that $\mathbf{Z} = \sum_k \mathbf{u}_k\mathbf{v}_k^\top$ and $\mathbf{D}_j\mathbf{X}\mathbf{Z} = \sum_k (\mathbf{X}\mathbf{u}_k)_+\mathbf{v}_k^\top$, and penalizes $\sum_k \|\mathbf{u}_k\|_2\|\mathbf{v}_k\|_2$ [2].

With this established, it can be shown that

$$p^*_{RMLP} = \min_{\{\mathbf{Z}_j\}_{j=1}^P} \mathcal{L}\left(\sum_{j=1}^P \mathbf{D}_j\mathbf{X}\mathbf{Z}_j, \mathbf{Y}\right) + \beta\sum_{j=1}^P \|\mathbf{Z}_j\|_{*,\mathrm{K}_j}$$
$$\mathbf{K}_j := (2\mathbf{D}_j - \mathbf{I})\mathbf{X}. \qquad (5)$$

The two-layer ReLU MLP optimization problem is thus expressed as a piece-wise linear model with a constrained nuclear norm regularization. This contrasts with the two-layer linear activation MLP, whose convex equivalent is

$$p^*_{LMLP} = \min_{\mathbf{Z}} \mathcal{L}\left(\mathbf{X}\mathbf{Z}, \mathbf{Y}\right) + \beta\|\mathbf{Z}\|_*. \qquad (6)$$

This nuclear norm penalty is known to encourage low-rank solutions (Candès & Tao, 2010; Recht et al., 2010) and appears in matrix factorization problems (Gunasekar et al., 2017). One can also define the *gated ReLU* activation, where the ReLU gates are fixed to some $\{\mathbf{h}_j\}_{j=1}^m$ (Fiat et al., 2019)

$$g(\mathbf{X}\mathbf{w}_{1j}) := \mathrm{diag}\left(\mathbb{1}\{\mathbf{X}\mathbf{h}_j\ge 0\}\right)(\mathbf{X}\mathbf{w}_{1j}). \qquad (7)$$

Then, defining $\{\mathbf{D}_j\}_{j=1}^m := \{\mathrm{diag}\left(\mathbb{1}\{\mathbf{X}\mathbf{h}_j\ge 0\}\right)\}_{j=1}^m$, the corresponding convex gated ReLU activation two-layer network objective follows directly from the ReLU and linear cases, and is given by

$$p^*_{GMLP} = \min_{\{\mathbf{Z}_j\}_{j=1}^m} \mathcal{L}\left(\sum_{j=1}^m \mathbf{D}_j\mathbf{X}\mathbf{Z}_j, \mathbf{Y}\right) + \beta\sum_{j=1}^m \|\mathbf{Z}_j\|_*. \qquad (8)$$

[2] We illustrate an example of $\|\mathbf{Z}\|_{*,\mathrm{K}}$ in the Appendix C.1.

We note that for both linear and gated ReLU formulations, the regularization on the convex weights becomes the standard nuclear norm, since ReLU constraints no longer have to be enforced. It has been demonstrated that there is a small approximation gap between Gated ReLU and ReLU networks, and ReLU networks can be formed from the solutions to Gated ReLU problems (Mishkin et al., 2022).

In terms of efficient algorithms to solve these problems, using an accelerated proximal gradient descent algorithm applied to the convex linear and gated ReLU programs, one can achieve an ϵ-optimal solution in $\mathcal{O}(1/\sqrt{\epsilon})$ iterations (Toh & Yun, 2010). For the convex ReLU formulation,(Sahiner et al., 2020a) proposes a Frank-Wolfe algorithm for convex MLPs which can be adapted to this case, which in the general case requires $\mathcal{O}(1/\epsilon)$ iterations for ϵ-optimality (Jaggi, 2013).

In the subsequent sections, we will demonstrate how common vision transformer blocks with linear and ReLU activations can be related to equivalent convex optimization problems through similar convex duality techniques[3].

3. Implicit Convexity of Self-Attention

The canonical Vision Transformer (ViT) uses self-attention and MLPs as its backbone (Dosovitskiy et al., 2020). In particular, a single "head" of a self-attention network is given by the following:

$$f_j(\mathbf{X}_i) := \sigma\left(\frac{\mathbf{X}_i\mathbf{Q}_j\mathbf{K}_j^\top\mathbf{X}_i^\top}{\sqrt{d}}\right)\mathbf{X}_i\mathbf{V}_j, \tag{9}$$

where $\mathbf{Q}, \mathbf{K}, \mathbf{V}$ are all learnable parameters and $\sigma(\cdot)$ typically (but not always) represents the softmax non-linearity. In practice, one typically uses m "heads" of attention which are concatenated along the feature dimension, and this is followed by a "channel-mixing" layer, or alternatively a classification head:

$$\begin{aligned} f_{MHSA}(\mathbf{X}_i) &:= \begin{bmatrix} f_1(\mathbf{X}_i) & \cdots & f_m(\mathbf{X}_i)\end{bmatrix}\mathbf{W} \\ &= \sum_{j=1}^m \sigma\left(\frac{\mathbf{X}_i\mathbf{Q}_j\mathbf{K}_j^\top\mathbf{X}_i^\top}{\sqrt{d}}\right)\mathbf{X}_i\mathbf{V}_j\mathbf{W}_j \end{aligned} \tag{10}$$

For the purpose of our analysis, noting that both $\mathbf{Q}_j\mathbf{K}_j^\top$ and $\mathbf{V}_j\mathbf{W}_j$ can be expressed by a single linear layer, we model the multi-head self-attention network as

$$f_{SA}(\mathbf{X}_i) := \sum_{j=1}^m \sigma\left(\frac{\mathbf{X}_i\mathbf{W}_{1j}\mathbf{X}_i^\top}{\sqrt{d}}\right)\mathbf{X}_i\mathbf{W}_{2j}. \tag{11}$$

We then define the multi-head self-attention training problem as follows

$$\begin{aligned} p_{SA}^* := \min_{\mathbf{W}_{1j},\mathbf{W}_{2j}} &\sum_{i=1}^n \mathcal{L}(f_{SA}(\mathbf{X}_i), \mathbf{Y}_i) \\ &+ \frac{\beta}{2}\sum_{j=1}^m \|\mathbf{W}_{1j}\|_F^2 + \|\mathbf{W}_{2j}\|_F^2. \end{aligned} \tag{12}$$

We thus employ any generic convex loss function and standard weight-decay in our formulation. While direct convex analysis when $\sigma(\cdot)$ represents the softmax activation is intractable, we can analyze this architecture for many other activation functions. In particular, self-attention with both linear and ReLU activation functions has been proposed with performance on par to standard softmax activation networks (Shen et al., 2021; Yorsh et al., 2021; Yorsh & Kovalenko, 2021; Zhang et al., 2021). Thus, we will analyze linear, ReLU, and gated ReLU activation variants of the multi-head self-attention.

Theorem 3.1. *For the linear activation multi-head self-attention training problem (12), for $m \geq m^*$ where $m^* \leq \min\{d^2, dc\}$, the standard non-convex training objective is equivalent to a convex optimization problem, given by*

$$\begin{aligned} p_{SA}^* = \min_{\mathbf{Z}\in\mathbb{R}^{d^2\times dc}} &\sum_{i=1}^n \mathcal{L}\left(\sum_{k=1}^d\sum_{\ell=1}^d \mathbf{G}_i[k,\ell]\mathbf{X}_i\mathbf{Z}^{(k,\ell)}, \mathbf{Y}_i\right) \\ &+ \beta\|\mathbf{Z}\|_* \end{aligned} \tag{13}$$

where $\mathbf{G}_i := \mathbf{X}_i^\top\mathbf{X}_i$ and $\mathbf{Z}^{(k,\ell)} \in \mathbb{R}^{d\times c}$.

Our result demonstrates that a linear activation self-attention model consists of a Gram (feature correlation) matrix weighted linear model, with a nuclear norm penalty which groups the individual models to each other[4].

One may also view the convex model as a set of linear models with a weighted nuclear norm, where each block $\mathbf{Z}^{(k,\ell)}$ has a corresponding weight of $1/\mathbf{G}_i[k,\ell]$. Thus, features with high correlation will have corresponding linear weights with larger norm. We note that when $\beta = 0$, the linear self-attention model (13) is equivalent to the linear two-layer MLP (6).

While typically the nuclear norm penalty on $\mathbf{Z}$ has no corresponding norm on each individual linear model $\mathbf{Z}^{(k,\ell)}$, the following result summarizes an instance where the nuclear norm could decompose into smaller blocks.

Corollary 3.2. *Assume some of the features of $\mathbf{X}_i$ are entirely uncorrelated for all i, i.e. $\mathbf{G}_i$ is block diagonal with*

[3] Gated ReLU analysis is also provided in Appendix C.2

[4] Furthermore, there is a one-to-one mapping between the solutions of the convex and non-convex programs, which we describe in Appendix A.

blocks $\{\mathbf{G}_i^{(b)} \in \mathbb{R}^{d_b \times d_b}\}_{b=1}^B$ for all i. Then, the convex program (13) reduces to the following convex program

$$p_{SA}^* = \min_{\mathbf{Z}^{(b)}} \sum_{i=1}^n \mathcal{L}\left(\sum_{b=1}^B \sum_{k=1}^{d_b} \sum_{\ell=1}^{d_b} \mathbf{G}_i^{(b)}[k,\ell]\mathbf{X}_i\mathbf{Z}^{(b,k,\ell)}, \mathbf{Y}_i\right) + \beta \sum_{b=1}^B \|\mathbf{Z}^{(b)}\|_*,\ \mathbf{Z}^{(b)} \in \mathbb{R}^{d_b d \times d_b c}. \tag{14}$$

This corollary thus demonstrates that under the assumption of sets of uncorrelated features, a linear self-attention block separates over these sets. In particular, the blocks of $\mathbf{Z}$ corresponding to values of 0 in the Gram matrix $\mathbf{G}_i$ will be set to 0, eliminating interactions between uncorrelated features. This phenomenon is illustrated in Figure 1.

While this linear model provides a simple, elegant explanation for the underpinnings of self-attention, we can also analyze self-attention blocks with non-linearities. We thus provide an analysis of ReLU-activation self-attention.

Theorem 3.3. *For the ReLU activation multi-head self-attention training problem (12), we define*

$$\mathbf{X} := \begin{bmatrix} \mathbf{X}_1 \otimes \mathbf{X}_1 \\ \cdots \\ \mathbf{X}_n \otimes \mathbf{X}_n \end{bmatrix}$$

$$\{\mathbf{D}_j\}_{j=1}^P := \{\operatorname{diag}\left(\mathbb{1}\{\mathbf{X}\mathbf{u}_j \geq 0\}\right):\ \mathbf{u}_j \in \mathbb{R}^{d^2}\},$$

where $P \leq 2r\left(\frac{e(n-1)}{r}\right)^r$ and $r := \operatorname{rank}(\mathbf{X})$. Then, for $m \geq m^*$ where $m^* \leq n\min\{d^2, dc\}$, the standard non-convex training objective is equivalent to a convex optimization problem, given by

$$p_{SA}^* = \min_{\mathbf{Z}_j \in \mathbb{R}^{d^2 \times dc}} \sum_{i=1}^n \mathcal{L}\left(\sum_{j=1}^P \sum_{\ell=1}^d \sum_{k=1}^d \mathbf{G}_{i,j}^{(k,\ell)}\mathbf{X}_i\mathbf{Z}_j^{(k,\ell)}, \mathbf{Y}_i\right) + \beta \sum_{j=1}^m \|\mathbf{Z}_j\|_{*,\mathrm{K}_j},\ \mathbf{K}_j := (2\mathbf{D}_j - \mathbf{I}_{ns^2})\mathbf{X}, \tag{15}$$

where $\mathbf{G}_{i,j} := (\mathbf{X}_i \otimes \mathbf{I}_s)^\top \mathbf{D}_j^{(i)}(\mathbf{X}_i \otimes \mathbf{I}_s)$, $\mathbf{G}_{i,j}^{(k,\ell)} \in \mathbb{R}^{s \times s}$ and $\mathbf{Z}_j^{(k,\ell)} \in \mathbb{R}^{d \times c}$.

Interestingly, while the hyperplane arrangements for a standard ReLU MLP depend only on the data matrix $\mathbf{X}$, for a self-attention network they are more complex, instead depending on $\mathbf{X}_i \otimes \mathbf{X}_i$. These hyperplane arrangements define the constraints for the constrained nuclear norm penalty. One could potentially see a ReLU activation self-attention model as a fusion of two models–one which uses $\mathbf{X}_i \otimes \mathbf{X}_i$ for generating hyperplane arrangements, and one which uses $\mathbf{X}_i$ for linear predictions. Thus, unlike the linear self-attention case, even in the case of $\beta = 0$, the ReLU self-attention network (15) is not equivalent to the ReLU MLP model (5).

Furthermore, while in the linear-activation case in (13), each linear model was scaled by a single entry in $\mathbf{G}_i$, in the ReLU case each linear model is scaled by a diagonal matrix $\mathbf{G}_{i,j}^{(k,\ell)}$ which combines second-order information from $\mathbf{X}_i$ with the hyperplane arrangements induced by the ReLU activation function. One may, for example, note the identity

$$\mathbf{G}_{i,j}^{(k,\ell)} = \sum_{t=1}^s \mathbf{X}_i[t,k]\mathbf{X}_i[t,\ell]\mathbf{D}_j^{(i,t)}, \tag{16}$$

for diagonal $\mathbf{D}_j^{(i,t)} \in \{0,1\}^{s \times s}$. Therefore, $\mathbf{G}_{i,j}^{(k,\ell)}$ can be viewed as a correlation between features k and ℓ weighted by diagonal $\{0,1\}$-valued hyperplane arrangements for each corresponding row t, in other words a type of "local" correlation, where locality is achieved by the $\{0,1\}$ values in $\mathbf{D}_j^{(i,t)}$. This local correlation scales each token of the prediction, essentially giving weight to tokens which have been not been masked away by $\mathbf{D}_j^{(i,t)}$.

4. Alternative Mixing Mechanisms

While self-attention is the original proposed token mixer used for vision transformers, there are many other alternative approaches which have shown to produce similar results while being more computationally efficient. We tackle two such architectures here.

4.1. MLP Mixer

We begin by analyzing the MLP-Mixer architecture, an all-MLP alternative to self-attention networks with competitive performance on image classification benchmarks (Tolstikhin et al., 2021). The proposal is simple–apply an MLP along one dimension of the input, followed by an MLP along the opposite dimension. The simplest form of this MLP-Mixer architecture can thus be written as

$$p_{MM}^* := \min_{\mathbf{W}_{1j}, \mathbf{W}_{2j}} \sum_{i=1}^n \mathcal{L}(\sum_{j=1}^m \sigma(\mathbf{W}_{1j}\mathbf{X}_i)\mathbf{W}_{2j}, \mathbf{Y}_i) + \frac{\beta}{2}\sum_{j=1}^m \|\mathbf{W}_{1j}\|_F^2 + \|\mathbf{W}_{2j}\|_F^2, \tag{17}$$

where σ is an activation function. While (Tolstikhin et al., 2021) use the GeLU non-linearity (Hendrycks & Gimpel, 2016), we analyze the simpler linear and ReLU activation counterparts, which shed important insights into the underlying structure of the MLP-Mixer architecture.

Theorem 4.1. *For the linear activation MLP-Mixer training problem (17), for $m \geq m^*$ where $m^* \leq \min\{s^2, dc\}$, the* *standard non-convex training objective is equivalent to a convex optimization problem, given by*

$$p^*_{MM} = \min_{\mathbf{Z}\in\mathbb{R}^{s^2\times dc}} \sum_{i=1}^{n} \mathcal{L}\left(\begin{bmatrix} f_1(\mathbf{X}_i) & \cdots & f_c(\mathbf{X}_i)\end{bmatrix}, \mathbf{Y}_i\right) + \beta\|\mathbf{Z}\|_*,\ f_p(\mathbf{X}_i) := \mathbf{Z}^{(p)}\mathbf{vec}(\mathbf{X}_i) \tag{18}$$

where $\mathbf{Z}^{(p)} \in \mathbb{R}^{s\times sd}$ *for* $p \in [c]$, *for* $\mathbf{Z}^{(p,t)} \in \mathbb{R}^{s\times d}$ *for* $t \in [s]$, *and*

$$\mathbf{Z}^{(p)} = \begin{bmatrix}\mathbf{Z}^{(p,1)} & \cdots & \mathbf{Z}^{(p,s)}\end{bmatrix}, \mathbf{Z} = \begin{bmatrix}\mathbf{Z}^{(1,1)} & \cdots & \mathbf{Z}^{(c,1)} \\ & \cdots & \\ \mathbf{Z}^{(1,s)} & \cdots & \mathbf{Z}^{(c,s)}\end{bmatrix}.$$

We can contrast the fitting term of the linear MLP-Mixer to a standard linear MLP (6), where each column k of the network output is given by

$$\mathbf{X}_i\mathbf{Z}^{(k)} = (\mathbf{Z}^{(k)^\top} \otimes \mathbf{I}_s)\mathbf{vec}(\mathbf{X}_i),$$

where $\mathbf{Z}^{(k)} \in \mathbb{R}^d$. Thus, the MLP-Mixer gives the network s^2 more degrees of freedom for fitting each column of $\mathbf{Y}_i$ than a standard linear MLP. This demonstrates that unlike the linear self-attention network, a linear MLP-Mixer model is not equivalent to a linear standard MLP even when $\beta = 0$. One may speculate that this additional implicit degrees of freedom allows mixer-like MLP models to fit complex distributions more easily compared to standard MLPs. While it appears from the fitting term that each output class of $\mathbf{Y}_i$ is fit independently, we note that these outputs are coupled together by the nuclear norm on $\mathbf{Z}$, which encourages $\{\ \mathbf{Z}^{(k)}\}_{k=1}^{c}$ to be similar to one another.

Another interpretation of the convexified linear MLP-Mixer architecture may be achieved by simply permuting the columns of $\mathbf{Z}$ to form $\tilde{\mathbf{Z}}$, which does not affect the nuclear norm and thus does not impact the optimal solution. If one partitions the columns of $\tilde{\mathbf{Z}}$ according to blocks $\tilde{\mathbf{Z}}^{(t,k)} \in \mathbb{R}^{s\times c}$ for $t \in [s]$, $k \in [d]$, one can also write

$$p^*_{MM} = \min_{\tilde{\mathbf{Z}}} \sum_{i=1}^{n} \mathcal{L}\left(\sum_{t=1}^{s}\sum_{k=1}^{d} \mathbf{X}_i[t,k]\tilde{\mathbf{Z}}^{(t,k)}, \mathbf{Y}_i\right) + \beta\|\tilde{\mathbf{Z}}\|_* \tag{19}$$

Here, the connections to the linear self-attention network (13) become more clear. While (13) is a weighted summation of linear models, where the weights correspond to Gram matrix entries, (19) is a weighted summation of predictions, where the weights correspond to data matrix entries. We also note that in most networks, typically $s < d$, so the MLP-Mixer block has a lower order complexity to solve compared to the self-attention block. We can also extend these results to ReLU activation MLP-Mixers.

Theorem 4.2. *For the ReLU activation MLP-Mixer training problem* (17), *we define*

$$\mathbf{X} := \begin{bmatrix}\mathbf{X}_1^\top \otimes \mathbf{I}_s \\ \cdots \\ \mathbf{X}_n^\top \otimes \mathbf{I}_s\end{bmatrix}$$

$$\{\mathbf{D}_j\}_{j=1}^{P} := \{\mathrm{diag}\left(\mathbb{1}\{\mathbf{X}\mathbf{u}_j \geq 0\}\right) :\ \mathbf{u}_j \in \mathbb{R}^{s^2}\},$$

where $P \leq 2r\left(\frac{e(n-1)}{r}\right)^r$ *and* $r := \mathrm{rank}(\mathbf{X})$. *Then, for* $m \geq m^*$ *where* $m^* \leq n\min\{s^2, dc\}$, *the standard non-convex training objective is equivalent to a convex optimization problem, given by*

$$p^*_{MM} = \min_{\mathbf{Z}_j\in\mathbb{R}^{s^2\times dc}} \sum_{i=1}^{n} \mathcal{L}\left(\begin{bmatrix} f_1(\mathbf{X}_i) & \cdots & f_c(\mathbf{X}_i)\end{bmatrix}, \mathbf{Y}_i\right) + \beta\sum_{j=1}^{P}\|\mathbf{Z}_j\|_{*,\mathrm{K}_j}, \mathbf{K}_j := (2\mathbf{D}_j - \mathbf{I}_{nd})\mathbf{X} \tag{20}$$

where

$$f_p(\mathbf{X_i}) := \sum_{j=1}^{P}\left[\mathbf{D}_j^{(i,1)}\mathbf{Z}_j^{(p,1)} \cdots \mathbf{D}_j^{(i,d)}\mathbf{Z}_j^{(p,d)}\right]\mathbf{vec}(\mathbf{X}_i)$$

for $\mathbf{D}_j^{(i,k)} \in \mathbb{R}^{s\times s}$ *and* $\mathbf{Z}_j^{(p,k)} \in \mathbb{R}^{s\times s}$.

Now, unlike the self-attention model, where the effective data matrix for hyperplane arrangements was $\mathbf{X}_i \otimes \mathbf{X}_i$, the MLP-mixer's arrangements use $\mathbf{X}_i^\top \otimes \mathbf{I}_s$, providing additional degrees of freedom for partitioning the data while still incorporating only first-order information about the data. Using the same column permutation trick as in (19), one may write (20) as

$$p^*_{MM} = \min_{\tilde{\mathbf{Z}}_j} \sum_{i=1}^{n} \mathcal{L}\left(\sum_{j=1}^{P}\sum_{t=1}^{s}\sum_{k=1}^{d} \mathbf{X}_i[t,k]\mathbf{D}_j^{(i,k)}\tilde{\mathbf{Z}}_j^{(t,k)}, \mathbf{Y}_i\right) + \beta\sum_{j=1}^{P}\|\tilde{\mathbf{Z}}_j\|_{*,\mathrm{K}_j}, \tag{21}$$

where again now we see clearly the differences with ReLU self-attention (15), with the diagonal arrangements weighted by $\mathbf{X}_i$ rather than the Gram matrix, and instead of $\tilde{\mathbf{Z}}_j^{(t,k)}$ being weights of a linear model, they are simply predictions.

4.2. Fourier Neural Operator

In contrast to self-attention or MLP-like attention mechanisms, there is also a family of Fourier-based alternatives to self-attention which have recently shown promise in vision. We present Fourier Neural Operator (FNO) (Li et al., 2020; Guibas et al., 2021), which works as follows: i) 2D DFT is applied first over the spatial tokens; ii) each token is multiplied by its own weight matrix; and iii) inverse DFT returns the Fourier tokens back to the original (spatial) domain.

To express FNO in a compact matrix form, note that in addition to standard MLP weights $\mathbf{W}_1 \in \mathbb{R}^{d\times m}$, $\mathbf{W}_2 \in \mathbb{R}^{m\times c}$, FNO blocks has a third set of weights $\mathbf{L} \in \mathbb{R}^{s\times d\times d}$. Let us define the Fourier transform $\mathbf{F} := \mathbf{F}_h \otimes \mathbf{F}_w$, that is a vectorized version of $h \times w$ 2D Fourier transform. It is more convenient to work with the weights $\mathbf{L}$ in the Fourier space as $\mathbf{V}$, where the 2D Fourier transform has been applied on the first dimension, namely $\mathbf{V}[:, i, j]$ for every i, j.

Now, define $\mathbf{V}^{(j)} = \mathbf{V}[j, :, :]$. Each row of $\mathbf{F}\mathbf{X}_i$ is then multiplied by $d \times d$ weight matrix $\mathbf{V}^{(j)}$, and converted back to the image domain as follows

$$f_{FN}(\mathbf{X}_i) := \sigma\left(\left(\mathbf{F}^{-1}\begin{bmatrix}(\mathbf{F}\mathbf{X}_i)_1^\top \mathbf{V}^{(1)}\\ \cdots \\ (\mathbf{F}\mathbf{X}_i)_s^\top \mathbf{V}^{(s)}\end{bmatrix}\right)\mathbf{W}_1\right)\mathbf{W}_2. \tag{22}$$

This representation can be heavily simplified.

Lemma 4.3. *For weights $\mathbf{W}_1 \in \mathbb{R}^{sd\times m}$, $\mathbf{W}_2 \in \mathbb{R}^{m\times c}$, the FNO block (22) can be equivalently represented as*

$$f_{FN}(\mathbf{X}_i) = \sum_{j=1}^{m} \sigma\left(\mathrm{circ}(\mathbf{X}_i)\mathbf{w}_{1j}\right)\mathbf{w}_{2j}^\top \tag{23}$$

where $\mathrm{circ}(\mathbf{X}_i) \in \mathbb{R}^{s\times sd}$ denotes a matrix composed of all s circulant shifts of $\mathbf{X}_i$ along its first dimension.

We can accordingly write the FNO training objective as

$$p^*_{FN} := \min_{\mathbf{w}_{1j}, \mathbf{w}_{2j}} \sum_{i=1}^{n} \mathcal{L}(f_{FN}(\mathbf{X}_i), \mathbf{Y}_i) + \frac{\beta}{2}\sum_{j=1}^{m} \|\mathbf{w}_{1j}\|_2^2 + \|\mathbf{w}_{2j}\|_2^2. \tag{24}$$

We note that FNO actually starkly resembles a two-layer CNN, where the first layer consists of a convolutional layer with full circulant padding and a *global* convolutional kernel. Unlike a typical CNN, in which the kernel size is usually small and the convolution is *local*, the convolution here is much larger, which means there are many more parameters than a typical CNN. Similar CNN architectures have previously been analyzed via convex duality (Ergen & Pilanci, 2020; Sahiner et al., 2020b). Accordingly, for both linear and ReLU activation, (24) is equivalent to a convex optimization problem.

Theorem 4.4. *For the linear activation FNO training problem (24), for $m \geq m^*$ where $m^* \leq \min\{sd, c\}$, the standard non-convex training objective is equivalent to a convex optimization problem, given by*

$$p^*_{FN} = \min_{\mathbf{Z}\in\mathbb{R}^{sd\times c}} \sum_{i=1}^{n} \mathcal{L}\left(\mathrm{circ}(\mathbf{X}_i)\mathbf{Z}, \mathbf{Y}_i\right) + \beta\|\mathbf{Z}\|_*. \tag{25}$$

Theorem 4.5. *For the ReLU activation FNO training problem (24), we define*

$$\mathbf{X} := \begin{bmatrix}\mathrm{circ}(\mathbf{X}_1)\\ \cdots \\ \mathrm{circ}(\mathbf{X}_n)\end{bmatrix}$$

$$\{\mathbf{D}_j\}_{j=1}^{P} := \{\mathrm{diag}\left(\mathbb{1}\{\mathbf{X}\mathbf{u}_j \geq 0\}\right) : \ \mathbf{u}_j \in \mathbb{R}^{sd}\},$$

where $P \leq 2r\left(\frac{e(n-1)}{r}\right)^r$ and $r := \mathrm{rank}(\mathbf{X})$. Then, for $m \geq m^*$ where $m^* \leq n\min\{sd, c\}$, the standard non-convex training objective is equivalent to a convex optimization problem, given by

$$p^*_{FN} = \min_{\mathbf{Z}_j\in\mathbb{R}^{sd\times c}} \sum_{i=1}^{n} \mathcal{L}\left(\sum_{j=1}^{P} \mathbf{D}_j^{(i)}\mathrm{circ}(\mathbf{X}_i)\mathbf{Z}_j, \mathbf{Y}_i\right) + \beta\sum_{j=1}^{P}\|\mathbf{Z}_j\|_{*,\mathrm{K}_j}, \ \mathbf{K}_j := (2\mathbf{D}_j - \mathbf{I}_{ns})\mathbf{X}. \tag{26}$$

4.2.1. Block Diagonal FNO

While the FNO formulation is quite elegant, it requires many parameters (d^2 for each token). Accordingly, modifications have been proposed in the form of Adaptive Fourier Neural Operator (AFNO) (Guibas et al., 2021). One important modification pertains to enforcing the token weights to obey a block diagonal structure. This has been significantly improved the training and generalization ability of AFNO compared with the standard FNO [5]. We call this architecture B-FNO, which boils down to

$$f_{BFN}(\mathbf{X}_i) := \sigma\left(\mathbf{F}^{-1}\begin{bmatrix}(\mathbf{F}\mathbf{X}_i)_1^\top \mathbf{V}^{(1)}\\ \cdots \\ (\mathbf{F}\mathbf{X}_i)_s^\top \mathbf{V}^{(s)}\end{bmatrix}\right)\mathbf{W}_2 \tag{27}$$

$$\mathbf{L}^{(l)} := \begin{bmatrix}\mathbf{L}^{(l,1)} & & \\ & \ddots & \\ & & \mathbf{L}^{(l,B)}\end{bmatrix} \in \mathbb{R}^{d\times m}, \ l \in [s]$$

$$\mathbf{W}_2 := \begin{bmatrix}\mathbf{W}_2^{(1)} & & \\ & \ddots & \\ & & \mathbf{W}_2^{(B)}\end{bmatrix} \in \mathbb{R}^{m\times c},$$

for B blocks. This can be simplified as

[5] A standard AFNO network also includes additional steps, including a soft-thresholding operator, which for simplicity we do not analyze here.

Lemma 4.6. *For weights $\mathbf{W}_{1b} \in \mathbb{R}^{sd/B \times m/B}$ and $\mathbf{W}_{2b} \in \mathbb{R}^{m/B \times c/B}$, assuming σ operates element-wise, the B-FNO model (27) can be equivalently represented as*

$$f_{BFN}(\mathbf{X}_i) = \begin{bmatrix} f_{BFN}^{(1)}(\mathbf{X}_i) & \cdots & f_{BFN}^{(B)}(\mathbf{X}_i) \end{bmatrix} \tag{28}$$
$$f_{BFN}^{(b)}(\mathbf{X}_i) = \sum_{j=1}^{m} \sigma\left(\mathrm{circ}(\mathbf{X}_i^{(b)})\mathbf{w}_{1bj}\right)\mathbf{w}_{2bj}^\top$$

where $\mathrm{circ}(\mathbf{X}_i^{(b)}) \in \mathbb{R}^{s \times sd/B}$ is a matrix composed of all s circulant shifts of $\mathbf{X}_i^{(b)} \in \mathbb{R}^{s \times d/B}$ along its first dimension.

Interestingly, the block-diagonal weights of AFNO contrast the *local* convolution in CNNs with a *global* and *group-wise* convolution with B groups. We thus define

$$p_{BFN}^* := \min_{\mathbf{W}_{1bj}, \mathbf{W}_{2bj}} \sum_{i=1}^{n} \mathcal{L}(f_{BFN}(\mathbf{X}_i), \mathbf{Y}_i)$$
$$+ \frac{\beta}{2}\sum_{b=1}^{B}\sum_{j=1}^{m} \|\mathbf{w}_{1bj}\|_2^2 + \|\mathbf{w}_{2bj}\|_2^2. \tag{29}$$

Theorem 4.7. *For the linear activation B-FNO training problem (29), for $m \geq m^*$ where $m^* \leq 1/B \min\{sd, c\}$, the standard non-convex training objective is equivalent to a convex optimization problem, given by*

$$p_{BFN}^* = \min_{\mathbf{Z}_b} \sum_{i=1}^{n} \mathcal{L}\left(\begin{bmatrix} f^{(1)}(\mathbf{X}_i) & \cdots & f^{(B)}(\mathbf{X}_i)\end{bmatrix}, \mathbf{Y}_i\right),$$
$$+ \beta \sum_{b=1}^{B} \|\mathbf{Z}_b\|_* \tag{30}$$
$$\mathbf{Z}_b \in \mathbb{R}^{sd/B \times c/B},\ f^{(b)} := \mathrm{circ}(\mathbf{X}_i^{(b)})\mathbf{Z}_b.$$

Theorem 4.8. *For the ReLU activation B-FNO training problem (29), we define*

$$\mathbf{X}_b := \begin{bmatrix} \mathrm{circ}(\mathbf{X}_1^{(b)}) \\ \cdots \\ \mathrm{circ}(\mathbf{X}_n^{(b)}) \end{bmatrix}$$
$$\{\mathbf{D}_{b,j}\}_{j=1}^{P_b} := \{\mathrm{diag}\left(\mathbb{1}\{\mathbf{X}_b\mathbf{u}_j \geq 0\}\right) :\ \mathbf{u}_j \in \mathbb{R}^{sd/B}\},$$

where $P_b \leq 2r_b\left(\frac{e(n-1)}{r_b}\right)^{r_b}$ and $r_b := \mathrm{rank}(\mathbf{X}_b)$. Then, for $m \geq m^*$ where $m^* \leq n/B \min\{sd, c\}$, the standard non-convex training objective is equivalent to a convex optimization problem, given by

$$p_{BFN}^* = \min_{\mathbf{Z}_{b,j}} \sum_{i=1}^{n} \mathcal{L}\left(\begin{bmatrix} f^{(1)}(\mathbf{X}_i) & \cdots & f^{(B)}(\mathbf{X}_i)\end{bmatrix}, \mathbf{Y}_i\right)$$
$$+ \beta \sum_{b=1}^{B}\sum_{j=1}^{P_b} \|\mathbf{Z}_{j,b}\|_{*,\mathrm{K}_{b,j}}, \tag{31}$$

where

$$\mathbf{Z}_{b,j} \in \mathbb{R}^{sd/B \times c/B},\ f^{(b)}(\mathbf{X}_i) := \sum_{j=1}^{P_b} \mathbf{D}_{b,j}\mathrm{circ}(\mathbf{X}_i^{(b)})\mathbf{Z}_{b,j}$$
$$\mathbf{K}_{b,j} := (2\mathbf{D}_{b,j} - \mathbf{I}_{ns})\mathbf{X}_b.$$

5. Numerical Results

In this section, we seek to compare the performance of the transformer heads we have analyzed in this work to baseline convex optimization methods. This comparison allows us to illustrate the implicit biases imposed by these novel heads in a practical example. In particular, we consider the task of training a single new block of these convex heads for performing an image classification task. This is essentially the task of transfer learning *without fine-tuning existing weights of the backbone network*, which may be essential in computation and memory-constrained settings at the edge. For few-shot fine-tuning transformer tasks, non-convex optimization has observed to be unstable under different random initializations (Mosbach et al., 2020). Furthermore, fine-tuning only the final layer a network is a common practice, which performs very well in spurious correlation benchmarks (Kirichenko et al., 2022).

Specifically, we seek to classify images from the CIFAR-100 dataset (Krizhevsky et al., 2009). We first generate embeddings from a pretrained gMLP-S model (Liu et al., 2021) on 224 × 224 images from the ImageNet-1k dataset (Deng et al., 2009) with 16 × 16 patches ($s = 196$, $d = 256$). We then finetune the *single convex head* to classify images from CIFAR-100, while leaving the pre-trained backbone fixed.

For the backbone gMLP architecture, we reduce the feature dimension to $d = 100$ with average pooling as a pre-processing step before training with the convex heads. Similarly, for computational efficiency, we train the Gated ReLU variants of the standard ReLU architectures, since these Gated ReLU activation networks are unconstrained. For BFNO, we choose $B = 5$. All the heads use identical dimensions ($d = 100, s = 196, c = 100$), and we choose the number of neurons in ReLU heads to be $m = 100$, except in the case of self-attention, where we choose $m = 5$ to have the parameter count be roughly equal across heads (see Table 2 in Appendix B). As our baseline, we compare to a simple linear model (i.e. logistic regression) and to convex equivalents of MLPs as discussed in Section 2.1.

We summarize the results in Table 1. Here, we demonstrate that the attention variants outperform the standard convex MLP and linear baselines. This suggests that the higher-order information and additional degrees of freedom of the attention architectures provide advantages for difficult

Table 1. CIFAR-100 classification accuracy for training a single *convex* head. Embeddings are generated from gMLP-S pre-trained on ImageNet. Note that the backbone is not fine-tuned.

CONVEX HEAD	ACT.	TOP-1	TOP-5
SELF-ATTENTION	LINEAR	73.81	92.87
MLP-MIXER		**78.11**	**94.79**
B-FNO		68.68	90.62
FNO		72.29	93.03
MLP		65.95	89.33
LINEAR		66.42	89.27
SELF-ATTENTION	RELU	74.74	93.45
MLP-MIXER		**80.22**	**95.79**
B-FNO		77.65	94.97
FNO		72.93	92.71
MLP		73.05	92.52

vision tasks. Surprisingly, for self-attention, FNO, and MLP-Mixer, there is only a marginal gap between the linear and ReLU activation performance, suggesting most of the benefit of these architectures is in their fundamental structure, rather than the nonlinearity which is applied. In contrast, for B-FNO, there is a very large gap between ReLU and linear activation accuracies, suggesting that this nonlinearity is more crucial when group convolutions are applied. These convexified architectures thus pave the way towards stable and transparent models for transfer learning.

6. Conclusion

We demonstrated that blocks of self-attention and common alternatives such as MLP-Mixer, FNO, and B-FNO are equivalent to convex optimization problems in the case of linear and ReLU activations. These equivalent convex formulations implicitly cluster correlated features, and are penalized with a block nuclear norm regularizer which ensures a global representation. For future work, it remains to craft efficient approximate solvers of these networks by leveraging the structure of these unique regularizers. Faster solvers may implemented, such as FISTA or related algorithms (some of which were explored in the context of convex MLPs (Mishkin et al., 2022)). In the long term for practical adoption, future theoretical work would also require analysis of deeper networks as are often used in practice. One may use this work in designing new network architectures by specifying the desired convex formulation.

ACKNOWLEDGMENTS

This work was partially supported by the National Science Foundation under grants ECCS-2037304, DMS-2134248, the Army Research Office. Research reported in this work was supported by the National Institutes of Health under award numbers R01EB009690 and U01EB029427. The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Institutes of Health.

References

An, B., Lyu, J., Wang, Z., Li, C., Hu, C., Tan, F., Zhang, R., Hu, Y., and Chen, C. Repulsive attention: Rethinking multi-head attention as bayesian inference. *arXiv preprint arXiv:2009.09364*, 2020.

Bai, Y., Gautam, T., and Sojoudi, S. Efficient global optimization of two-layer relu networks: Quadratic-time algorithms and adversarial training. *arXiv preprint arXiv:2201.01965*, 2022.

Burer, S. and Monteiro, R. D. Local minima and convergence in low-rank semidefinite programming. *Mathematical programming*, 103(3):427–444, 2005.

Candès, E. J. and Tao, T. The power of convex relaxation: Near-optimal matrix completion. *IEEE Transactions on Information Theory*, 56(5):2053–2080, 2010.

Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., and Joulin, A. Emerging properties in self-supervised vision transformers. *arXiv preprint arXiv:2104.14294*, 2021.

Chefer, H., Gur, S., and Wolf, L. Transformer interpretability beyond attention visualization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 782–791, 2021.

Cordonnier, J.-B., Loukas, A., and Jaggi, M. On the relationship between self-attention and convolutional layers. *arXiv preprint arXiv:1911.03584*, 2019.

Cubuk, E. D., Zoph, B., Mane, D., Vasudevan, V., and Le, Q. V. Autoaugment: Learning augmentation policies from data. *arXiv preprint arXiv:1805.09501*, 2018.

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.

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.

Ergen, T. and Pilanci, M. Implicit convex regularizers of cnn architectures: Convex optimization of two-and three-layer networks in polynomial time. *arXiv preprint arXiv:2006.14798*, 2020.

Ergen, T. and Pilanci, M. Global optimality beyond two layers: Training deep relu networks via convex programs. In *International Conference on Machine Learning*, pp. 2993–3003. PMLR, 2021.

Ergen, T., Sahiner, A., Ozturkler, B., Pauly, J., Mardani, M., and Pilanci, M. Demystifying batch normalization in relu networks: Equivalent convex optimization models and implicit regularization. *arXiv preprint arXiv:2103.01499*, 2021.

Fiat, J., Malach, E., and Shalev-Shwartz, S. Decoupling gating from linearity. *arXiv preprint arXiv:1906.05032*, 2019.

Geng, Z., Guo, M.-H., Chen, H., Li, X., Wei, K., and Lin, Z. Is attention better than matrix decomposition? *arXiv preprint arXiv:2109.04553*, 2021.

Guibas, J., Mardani, M., Li, Z., Tao, A., Anandkumar, A., and Catanzaro, B. Adaptive fourier neural operators: Efficient token mixers for transformers. *arXiv preprint arXiv:2111.13587*, 2021.

Gunasekar, S., Woodworth, B. E., Bhojanapalli, S., Neyshabur, B., and Srebro, N. Implicit regularization in matrix factorization. *Advances in Neural Information Processing Systems*, 30, 2017.

Hendrycks, D. and Gimpel, K. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016.

Jaggi, M. Revisiting frank-wolfe: Projection-free sparse convex optimization. In *International Conference on Machine Learning*, pp. 427–435. PMLR, 2013.

Kim, H., Papamakarios, G., and Mnih, A. The lipschitz constant of self-attention. In *International Conference on Machine Learning*, pp. 5562–5571. PMLR, 2021.

Kirichenko, P., Izmailov, P., and Wilson, A. G. Last layer re-training is sufficient for robustness to spurious correlations. *arXiv preprint arXiv:2204.02937*, 2022.

Kitaev, N., Kaiser, Ł., and Levskaya, A. Reformer: The efficient transformer. *arXiv preprint arXiv:2001.04451*, 2020.

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

Lee-Thorp, J., Ainslie, J., Eckstein, I., and Ontanon, S. Fnet: Mixing tokens with fourier transforms. *arXiv preprint arXiv:2105.03824*, 2021.

Li, Z., Kovachki, N., Azizzadenesheli, K., Liu, B., Bhattacharya, K., Stuart, A., and Anandkumar, A. Fourier neural operator for parametric partial differential equations. *arXiv preprint arXiv:2010.08895*, 2020.

Liu, H., Dai, Z., So, D. R., and Le, Q. V. Pay attention to mlps, 2021.

Loshchilov, I. and Hutter, F. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Magnus, J. R. and Neudecker, H. *Matrix differential calculus with applications in statistics and econometrics*. John Wiley & Sons, 2019.

Mardani, M., Mateos, G., and Giannakis, G. B. Decentralized sparsity-regularized rank minimization: Algorithms and applications. *IEEE Transactions on Signal Processing*, 61(21):5374–5388, 2013.

Mardani, M., Mateos, G., and Giannakis, G. B. Subspace learning and imputation for streaming big data matrices and tensors. *IEEE Transactions on Signal Processing*, 63 (10):2663–2677, 2015.

Mishkin, A., Sahiner, A., and Pilanci, M. Fast convex optimization for two-layer relu networks: Equivalent model classes and cone decompositions. *arXiv preprint arXiv:2202.01331*, 2022.

Mosbach, M., Andriushchenko, M., and Klakow, D. On the stability of fine-tuning bert: Misconceptions, explanations, and strong baselines. *arXiv preprint arXiv:2006.04884*, 2020.

Panahi, A., Saeedi, S., and Arodz, T. Shapeshifter: a parameter-efficient transformer using factorized reshaped matrices. *Advances in Neural Information Processing Systems*, 34, 2021.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32:8026–8037, 2019.

Pilanci, M. and Ergen, T. Neural networks are convex regularizers: Exact polynomial-time convex optimization formulations for two-layer networks. In *International Conference on Machine Learning*, pp. 7695–7705. PMLR, 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, 2021.

Rao, Y., Zhao, W., Zhu, Z., Lu, J., and Zhou, J. Global filter networks for image classification. *Advances in Neural Information Processing Systems*, 34, 2021.

Recht, B., Fazel, M., and Parrilo, P. A. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM review*, 52(3):471–501, 2010.

Sahiner, A., Ergen, T., Pauly, J., and Pilanci, M. Vector-output relu neural network problems are copositive programs: Convex analysis of two layer networks and polynomial-time algorithms. *arXiv preprint arXiv:2012.13329*, 2020a.

Sahiner, A., Mardani, M., Ozturkler, B., Pilanci, M., and Pauly, J. Convex regularization behind neural reconstruction. *arXiv preprint arXiv:2012.05169*, 2020b.

Sahiner, A., Ergen, T., Ozturkler, B., Bartan, B., Pauly, J., Mardani, M., and Pilanci, M. Hidden convexity of wasserstein gans: Interpretable generative models with closed-form solutions. *arXiv preprint arXiv:2107.05680*, 2021.

Savarese, P., Evron, I., Soudry, D., and Srebro, N. How do infinite width bounded norm networks look in function space? In *Conference on Learning Theory*, pp. 2667–2690. PMLR, 2019.

Shapiro, A. Semi-infinite programming, duality, discretization and optimality conditions. *Optimization*, 58(2):133–161, 2009.

Shen, Z., Zhang, M., Zhao, H., Yi, S., and Li, H. Efficient attention: Attention with linear complexities. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 3531–3539, 2021.

Stanley, R. P. et al. An introduction to hyperplane arrangements. *Geometric combinatorics*, 13(389-496):24, 2004.

Tatsunami, Y. and Taki, M. Raftmlp: Do mlp-based models dream of winning over computer vision? *arXiv preprint arXiv:2108.04384*, 2021.

Toh, K.-C. and Yun, S. An accelerated proximal gradient algorithm for nuclear norm regularized linear least squares problems. *Pacific Journal of optimization*, 6(615-640): 15, 2010.

Tolstikhin, I., Houlsby, N., Kolesnikov, A., Beyer, L., Zhai, X., Unterthiner, T., Yung, J., Keysers, D., Uszkoreit, J., Lucic, M., et al. Mlp-mixer: An all-mlp architecture for vision. *arXiv preprint arXiv:2105.01601*, 2021.

Touvron, H., Bojanowski, P., Caron, M., Cord, M., El-Nouby, A., Grave, E., Izacard, G., Joulin, A., Synnaeve, G., Verbeek, J., et al. Resmlp: Feedforward networks for image classification with data-efficient training. *arXiv preprint arXiv:2105.03404*, 2021.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. In *Advances in neural information processing systems*, pp. 5998–6008, 2017.

Wang, S., Li, B. Z., Khabsa, M., Fang, H., and Ma, H. Linformer: Self-attention with linear complexity. *arXiv preprint arXiv:2006.04768*, 2020.

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

Xiong, Y., Zeng, Z., Chakraborty, R., Tan, M., Fung, G., Li, Y., and Singh, V. Nystr\" omformer: A nystr\" om-based algorithm for approximating self-attention. *arXiv preprint arXiv:2102.03902*, 2021.

Yorsh, U. and Kovalenko, A. Pureformer: Do we even need attention? *arXiv preprint arXiv:2111.15588*, 2021.

Yorsh, U., Kordík, P., and Kovalenko, A. Simpletron: Eliminating softmax from attention computation. *arXiv preprint arXiv:2111.15588*, 2021.

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. *arXiv preprint arXiv:2111.11418*, 2021.

Yun, C., Chang, Y.-W., Bhojanapalli, S., Rawat, A. S., Reddi, S. J., and Kumar, S. $o(n)$ connections are expressive enough: Universal approximability of sparse transformers. *arXiv preprint arXiv:2006.04862*, 2020.

Zhang, B., Titov, I., and Sennrich, R. Sparse attention with linear units. *arXiv preprint arXiv:2104.07012*, 2021.

Appendix

A. Proofs

A.1. Preliminaries

Here, we describe the general technique which allows for convex duality of self-attention and similar architectures. In particular, all of the proofs for the theorems in the subsequent sections follow the following general form:

1. Re-scale the weights such that the optimization problem has a Frobenius norm penalty on the second layer weights with a norm constraint on the first layer weights.
2. Form the dual problem over the second-layer weights, creating a dual constraint which depends on the first layer weights.
3. Solve the dual constraint over the first layer weights.
4. Form the Lagrangian problem and solve over the dual weights.
5. Make any simplifications as necessary.

We describe the first two steps here, and use it in the following proofs.

Lemma A.1. *Suppose we are given an optimization problem of the form*

$$p^* := \min_{\mathbf{W}_{1j}, \mathbf{W}_{2j}} \sum_{i=1}^{n} \mathcal{L}(\sum_{j=1}^{m} g(\mathbf{X}_i; \mathbf{W}_{1j})\mathbf{W}_{2j}, \mathbf{Y}_i) + \frac{\beta}{2} \sum_{j=1}^{m} \|\mathbf{W}_{1j}\|_F^2 + \|\mathbf{W}_{2j}\|_F^2, \tag{32}$$

where $g(\mathbf{X}_i; \mathbf{W}_{1j})$ is any function such that $g(\mathbf{X}_i; \alpha_j \mathbf{W}_{1j}) = \alpha_j g(\mathbf{X}_i; \mathbf{W}_{1j})$ for $\alpha_j > 0$. Then, this problem is equivalent to

$$p^* = \min_{\|\mathbf{W}_{1j}\|_F \leq 1, \mathbf{W}_{2j}} \sum_{i=1}^{n} \mathcal{L}(\sum_{j=1}^{m} g(\mathbf{X}_i; \mathbf{W}_{1j})\mathbf{W}_{2j}, \mathbf{Y}_i) + \beta \sum_{j=1}^{m} \|\mathbf{W}_{2j}\|_F. \tag{33}$$

Proof. Due to the form of $g(\mathbf{X}_i; \mathbf{W}_{1j})$, we can rescale the parameters as $\bar{\mathbf{W}}_{1j} = \alpha_j \bar{\mathbf{W}}_{1j}$, $\bar{\mathbf{W}}_{2j} = \bar{\mathbf{W}}_{2j}/\alpha_j$ for $\alpha_j > 0$ without changing the network output. Then, to minimize the regularization term, we can write the problem as

$$p^* := \min_{\mathbf{W}_{1j}, \mathbf{W}_{2j}} \min_{\alpha_j > 0} \sum_{i=1}^{n} \mathcal{L}(\sum_{j=1}^{m} g(\mathbf{X}_i; \mathbf{W}_{1j})\mathbf{W}_{2j}, \mathbf{Y}_i) + \frac{\beta}{2} \sum_{j=1}^{m} \alpha_j^2 \|\mathbf{W}_{1j}\|_F^2 + \|\mathbf{W}_{2j}\|_F^2/\alpha_j^2. \tag{34}$$

Solving this minimization problem over α_j (Savarese et al., 2019; Sahiner et al., 2020a), we obtain

$$p^* := \min_{\mathbf{W}_{1j}, \mathbf{W}_{2j}} \sum_{i=1}^{n} \mathcal{L}(\sum_{j=1}^{m} g(\mathbf{X}_i; \mathbf{W}_{1j})\mathbf{W}_{2j}, \mathbf{Y}_i) + \beta \sum_{j=1}^{m} \|\mathbf{W}_{1j}\|_F \|\mathbf{W}_{2j}\|_F. \tag{35}$$

We can thus set $\|\mathbf{W}_{1j}\|_F = 1$ without loss of generality, and further relaxing this to $\|\mathbf{W}_{1j}\|_F \leq 1$ does not change the optimal solution. Thus, we are left with the desired result:

$$p^* = \min_{\|\mathbf{W}_{1j}\|_F \leq 1, \mathbf{W}_{2j}} \sum_{i=1}^{n} \mathcal{L}(\sum_{j=1}^{m} g(\mathbf{X}_i; \mathbf{W}_{1j})\mathbf{W}_{2j}, \mathbf{Y}_i) + \beta \sum_{j=1}^{m} \|\mathbf{W}_{2j}\|_F. \tag{36}$$

□

Note that the assumption on g encapsulates all architectures studied in this work: self-attention, MLP-Mixer, FNO, B-FNO, and other extensions with ReLU, gated ReLU, or linear activation functions all satisfy this property.

Lemma A.2. *Suppose that*

$$p^* := \min_{\|\mathbf{W}_{1j}\|_F \leq 1, \mathbf{W}_{2j}} \sum_{i=1}^n \mathcal{L}(\sum_{j=1}^m g(\mathbf{X}_i; \mathbf{W}_{1j})\mathbf{W}_{2j}, \mathbf{Y}_i) + \beta \sum_{j=1}^m \|\mathbf{W}_{2j}\|_F. \tag{37}$$

Then, for all $\beta > 0$, if $m \geq m^*$ for some m^*, this optimization problem is equivalent to

$$\begin{aligned} p^* = &\max_{\mathbf{V}_i} -\sum_{i=1}^n \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i) \\ \textit{s.t. } &\max_{\|\mathbf{W}_1\|_F \leq 1} \|\sum_{i=1}^n \mathbf{V}_i^\top g(\mathbf{X}_i; \mathbf{W}_1)\|_F \leq \beta, \end{aligned} \tag{38}$$

where $\mathcal{L}^*$ is the Fenchel conjugate of $\mathcal{L}$.

Proof. We first can re-write the problem as

$$p^* = \min_{\|\mathbf{W}_{1j}\|_F \leq 1} \min_{\mathbf{W}_{2j}, \mathbf{R}_i} \sum_{i=1}^n \mathcal{L}(\mathbf{R}_i, \mathbf{Y}_i) + \beta \sum_{j=1}^m \|\mathbf{W}_{2j}\|_F \text{ s.t. } \mathbf{R}_i = \sum_{j=1}^m g(\mathbf{X}_i; \mathbf{W}_{1j})\mathbf{W}_{2j}. \tag{39}$$

Then, we form the Lagrangian of this problem as

$$p^* = \min_{\|\mathbf{W}_{1j}\|_F \leq 1} \min_{\mathbf{W}_{2j}, \mathbf{R}_i} \max_{\mathbf{V}_i} \sum_{i=1}^n \mathcal{L}(\mathbf{R}_i, \mathbf{Y}_i) + \beta \sum_{j=1}^m \|\mathbf{W}_{2j}\|_F + \sum_{i=1}^n \text{trace}(\mathbf{V}_i^\top \mathbf{R}_i) - \sum_{i=1}^n \text{trace}\left(\mathbf{V}_i^\top \sum_{j=1}^m g(\mathbf{X}_i; \mathbf{W}_{1j})\mathbf{W}_{2j}\right). \tag{40}$$

By Sion's minimax theorem, we can reverse the order of the outer maximum and minimum, and minimize this problem over $\mathbf{W}_{2j}$ and $\mathbf{R}_i$. Defining the Fenchel conjugate of $\mathcal{L}$ as $\mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i) := \max_{\mathbf{R}_i} -\mathcal{L}(\mathbf{R}_i, \mathbf{Y}_i) + \text{trace}(\mathbf{V}_i^\top \mathbf{R}_i)$, we have

$$p^* = \min_{\|\mathbf{W}_{1j}\|_F \leq 1} \max_{\mathbf{V}_i} \min_{\mathbf{W}_{2j}} -\sum_{i=1}^n \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i) + \beta \sum_{j=1}^m \|\mathbf{W}_{2j}\|_F - \sum_{i=1}^n \text{trace}\left(\mathbf{V}_i^\top \sum_{j=1}^m g(\mathbf{X}_i; \mathbf{W}_{1j})\mathbf{W}_{2j}\right). \tag{41}$$

Now, we solve over $\mathbf{W}_{2j}$ to obtain

$$p^* = \min_{\|\mathbf{W}_{1j}\|_F \leq 1} \max_{\mathbf{V}_i : \|\sum_{i=1}^n \mathbf{V}_i^\top g(\mathbf{X}_i; \mathbf{W}_{1j})\|_F \leq \beta} -\sum_{i=1}^n \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i) \tag{42}$$

Now, since Slater's condition holds because $\beta > 0$, as long as $m \geq m^*$ where m^* is the dimension of the constraints (see the individual cases for examples) we are permitted to switch the order of the maximum and minimum to obtain the desired result (Shapiro, 2009):

$$\begin{aligned} p^* = &\max_{\mathbf{V}_i} -\sum_{i=1}^n \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i) \\ \text{s.t. } &\max_{\|\mathbf{W}_1\|_F \leq 1} \|\sum_{i=1}^n \mathbf{V}_i^\top g(\mathbf{X}_i; \mathbf{W}_1)\|_F \leq \beta. \end{aligned} \tag{43}$$

□

These two lemmas will prove invaluable in the subsequent proofs.

A.2. Proof of Theorem 3.1

We first note that for self-attention we remove the $1/\sqrt{d}$ factor for simplicity. We apply Lemmas A.1 and A.2 to (12) with the linear activation function to obtain

$$p^*_{SA} = \max_{\mathbf{V}_i} - \sum_{i=1}^{n} \mathcal{L}^*\left(\mathbf{V}_i, \mathbf{Y}_i\right)$$
$$\text{s.t.} \max_{\|\mathbf{W}_1\|_F \leq 1} \left\| \sum_{i=1}^{n} \mathbf{V}_i^\top (\mathbf{X}_i \mathbf{W}_1 \mathbf{X}_i^\top) \mathbf{X}_i \right\|_F \leq \beta. \tag{44}$$

Due to the identity $\mathbf{vec}(\mathbf{ABC}) = (\mathbf{C}^\top \otimes \mathbf{A})\mathbf{vec}(\mathbf{B})$ (Magnus & Neudecker, 2019), we can write this as

$$p^*_{SA} = \max_{\mathbf{V}_i} - \sum_{i=1}^{n} \mathcal{L}^*\left(\mathbf{V}_i, \mathbf{Y}_i\right)$$
$$\text{s.t.} \max_{\|\mathbf{w}_1\|_2 \leq 1} \left\| \sum_{i=1}^{n} ((\mathbf{X}_i^\top \otimes \mathbf{V}_i^\top)(\mathbf{X}_i \otimes \mathbf{X}_i)) \mathbf{w}_1 \right\|_2 \leq \beta, \tag{45}$$

where $\mathbf{w}_1 = \mathbf{vec}(\mathbf{W}_1)$. We note that the norm constraint here has dimension dc and $\mathbf{w}_1$ has dimension d^2, so by (Shapiro, 2009) this strong duality result from Lemma A.2 requires that $m^* \leq \min\{d^2, dc\}$. We can further write this as

$$p^*_{SA} = \max_{\mathbf{V}_i} - \sum_{i=1}^{n} \mathcal{L}^*\left(\mathbf{V}_i, \mathbf{Y}_i\right)$$
$$\text{s.t.} \max_{\|\mathbf{Z}\|_* \leq 1} \operatorname{trace} \left(\sum_{i=1}^{n} (\mathbf{X}_i^\top \otimes \mathbf{V}_i^\top)(\mathbf{X}_i \otimes \mathbf{X}_i) \mathbf{Z} \right) \leq \beta. \tag{46}$$

Now, we form the Lagrangian, given by

$$p^*_{SA} = \max_{\mathbf{V}_i} \min_{\|\mathbf{Z}\|_* \leq 1} \min_{\lambda \geq 0} - \sum_{i=1}^{n} \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i) + \lambda \left(\beta - \sum_{i=1}^{n} \mathbf{vec}((\mathbf{X}_i \otimes \mathbf{X}_i)\mathbf{Z})^\top \mathbf{vec}\left(\mathbf{X}_i \otimes \mathbf{V}_i\right) \right). \tag{47}$$

By Sion's minimax theorem, we are permitted to change the order of the maxima and minima, to obtain

$$p^*_{SA} = \min_{\lambda \geq 0} \min_{\|\mathbf{Z}\|_* \leq 1} \max_{\mathbf{V}_i} - \sum_{i=1}^{n} \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i) + \lambda \left(\beta - \sum_{i=1}^{n} \mathbf{vec}((\mathbf{X}_i \otimes \mathbf{X}_i)\mathbf{Z})^\top \mathbf{vec}\left(\mathbf{X}_i \otimes \mathbf{V}_i\right) \right). \tag{48}$$

Now, defining $\mathbf{K}_{c,s}$ as the (c, s) commutation matrix we have the following identity (Magnus & Neudecker, 2019)

$$\mathbf{vec}(\mathbf{X}_i \otimes \mathbf{V}_i) = ((\mathbf{I}_d \otimes \mathbf{K}_{c,s})(\mathbf{vec}(\mathbf{X}_i) \otimes \mathbf{I}_c) \otimes \mathbf{I}_s)\, \mathbf{vec}(\mathbf{V}_i).$$

Using this identity and maximizing over $\mathbf{V}_i$, we obtain

$$p^*_{SA} = \min_{\|\mathbf{Z}\|_* \leq 1} \min_{\lambda \geq 0} \sum_{i=1}^{n} \mathcal{L}\left(\left((\mathbf{vec}(\mathbf{X}_i)^\top \otimes \mathbf{I}_c)(\mathbf{I}_d \otimes \mathbf{K}_{s,c}) \otimes \mathbf{I}_s \right) \mathbf{vec}((\mathbf{X}_i \otimes \mathbf{X}_i)(\lambda \mathbf{Z})), \mathbf{vec}(\mathbf{Y}_i) \right) + \beta\lambda. \tag{49}$$

Rescaling such that $\tilde{\mathbf{Z}} = \lambda \mathbf{Z}$, we obtain

$$p^*_{SA} = \min_{\mathbf{Z} \in \mathbb{R}^{d^2 \times dc}} \sum_{i=1}^{n} \mathcal{L}\left(\left((\mathbf{vec}(\mathbf{X}_i)^\top \otimes \mathbf{I}_c)(\mathbf{I}_d \otimes \mathbf{K}_{s,c}) \otimes \mathbf{I}_s \right) \mathbf{vec}((\mathbf{X}_i \otimes \mathbf{X}_i)\mathbf{Z}), \mathbf{vec}(\mathbf{Y}_i) \right) + \beta \|\mathbf{Z}\|_*. \tag{50}$$

It appears as though this is a very complicated function, but it actually simplifies greatly. In particular, one can write this as

$$p^*_{SA} = \min_{\mathbf{Z} \in \mathbb{R}^{d^2 \times dc}} \sum_{i=1}^{n} \mathcal{L}\left(\hat{\mathbf{Y}}_i, \mathbf{Y}_i \right) + \beta \|\mathbf{Z}\|_*$$
$$\hat{\mathbf{Y}}_i[o, p] := \sum_{k=1}^{d} \sum_{l=1}^{d} \sum_{t=1}^{s} \mathbf{X}_i[t, l] \mathbf{X}_i[t, k] \mathbf{X}_i[o, :]^\top \mathbf{Z}^{(k,l)}. \tag{51}$$

Making any final simplifications, one obtains the desired result.

$$p^*_{SA} = \min_{\mathbf{Z} \in \mathbb{R}^{d^2 \times dc}} \sum_{i=1}^n \mathcal{L}\left(\sum_{k=1}^d \sum_{\ell=1}^d \mathbf{G}_i[k, \ell] \mathbf{X}_i \mathbf{Z}^{(k,\ell)}, \mathbf{Y}_i \right) + \beta \|\mathbf{Z}\|_*. \tag{52}$$

Lastly, we also demonstrate that there is a one-to-one mapping between the solution to (13) and (12). In particular, imagine we have a solution $\mathbf{Z}^*$ to (13) with optimal value p^*_{CVX}. Let $r := \operatorname{rank}(\mathbf{Z}^*)$, and take the SVD of $\mathbf{Z}^*$ as $\sum_{j=1}^r \sigma_j \mathbf{u}_j \mathbf{v}_j^\top$, where $\mathbf{u}_j \in \mathbb{R}^{d^2}$ and $\mathbf{v}_j \in \mathbb{R}^{dc}$. Let $\text{vec}^{-1}(\mathbf{u_j}) \in \mathbb{R}^{d \times d}$ be the result of taking chunks of d-length vectors from $\mathbf{u}_j$ and stacking them in columns. Similarly, let $\text{vec}^{-1}(\mathbf{v_j}) \in \mathbb{R}^{c \times d}$ be the result of taking chunks of c-length vectors from $\mathbf{v}_j$ and stacking them in columns. Furthermore we will let $\text{vec}^{-1}(\mathbf{u_j})_k$ be the kth column of $\text{vec}^{-1}(\mathbf{u_j})$. Then, recognize that

$$\mathbf{Z}^{*(k,\ell)} = \sum_{j=1}^r \sigma_j \text{vec}^{-1}(\mathbf{u_j})_k \text{vec}^{-1}(\mathbf{v_j})_\ell^\top$$

Thus, given $\mathbf{Z}^*$, we can form a candidate solution to (12) as follows:

$$\begin{aligned} \mathbf{Z}^* &= \sum_{j=1}^r \sigma_j \mathbf{u}_j \mathbf{v}_j^\top \\ \hat{\mathbf{W}}_{1j} &= \sqrt{\sigma_j} \text{vec}^{-1}(\mathbf{u_j}) \\ \hat{\mathbf{W}}_{2j} &= \sqrt{\sigma_j} \text{vec}^{-1}(\mathbf{v_j})^\top \end{aligned}$$

We then have

$$\begin{aligned} \hat{p}_{NCVX} &= \sum_{i=1}^n \mathcal{L}\left(\sum_{j=1}^r \mathbf{X}_i \hat{\mathbf{W}}_{1j} \mathbf{X}_i^\top \mathbf{X}_i \hat{\mathbf{W}}_{2j}, \mathbf{Y}_i \right) + \frac{\beta}{2} \sum_{j=1}^r \|\hat{\mathbf{W}}_{1j}\|_F^2 + \|\hat{\mathbf{W}}_{2j}\|_F^2 \\ &= \sum_{i=1}^n \mathcal{L}\left(\mathbf{X}_i \sum_{j=1}^r \sigma_j \text{vec}^{-1}(\mathbf{u_j}) \mathbf{G}_i \text{vec}^{-1}(\mathbf{v_j})^\top, \mathbf{Y}_i \right) + \frac{\beta}{2} \sum_{j=1}^r \|\sqrt{\sigma_j} \mathbf{u_j}\|_2^2 + \|\sqrt{\sigma_j} \mathbf{v}_j\|_2^2 \\ &= \sum_{i=1}^n \mathcal{L}\left(\sum_{k=1}^d \sum_{\ell=1}^d \mathbf{X}_i \sum_{j=1}^r \sigma_j \text{vec}^{-1}(\mathbf{u_j})_k \mathbf{G}_i[k, \ell] \text{vec}^{-1}(\mathbf{v_j})_\ell^\top, \mathbf{Y}_i \right) + \frac{\beta}{2} \sum_{j=1}^r \sigma_j + \sigma_j \\ &= \sum_{i=1}^n \mathcal{L}\left(\sum_{k=1}^d \sum_{\ell=1}^d \mathbf{G}_i[k, \ell] \mathbf{X}_i \mathbf{Z}^{*(k,\ell)}, \mathbf{Y}_i \right) + \beta \|\mathbf{Z}^*\|_* \\ &= p^*_{CVX} \end{aligned}$$

Thus, the two solutions match. Similarly, if we have the solution $(\mathbf{W}_{1j}^*, \mathbf{W}_{2j}^*)$ to (12), we can form the equivalent optimal convex weights to (13) as

$$\mathbf{Z}^* = \sum_{j=1}^m \text{vec}(\mathbf{W}_{1j}^*) \text{vec}(\mathbf{W}_{2j}^*)^\top$$

and the same proof can be demonstrated in reverse.

□

A.3. Proof of Corollary 3.2

We suppose that $\mathbf{G}$ is block diagonal with B such blocks of size d_b such that $\sum_b d_b = d$. Then, we have

$$p^*_{SA} = \min_{\mathbf{Z} \in \mathbb{R}^{d^2 \times dc}} \sum_{i=1}^n \mathcal{L}\left(\sum_{b=1}^B \sum_{j=1}^{d_b} \sum_{k=1}^{d_b} \mathbf{G}_i^{(b)}[j, k] \mathbf{X}_i \mathbf{Z}^{(b,j,k)}, \mathbf{Y}_i \right) + \beta \|\mathbf{Z}\|_* \tag{53}$$

Then, we have, for blocks $\mathbf{Z}^{(b)} \in \mathbb{R}^{d_b d \times d_b c}$,

$$\|\mathbf{Z}\|_* = \max_{\|\mathbf{A}\|_2 \leq 1} \langle \mathbf{Z}, \mathbf{A} \rangle \geq \max_{\|\mathbf{A}^{(b)}\|_2 \leq 1} \sum_{b=1}^{b} \langle \mathbf{Z}^{(b)}, \mathbf{A}^{(b)} \rangle = \sum_{b=1}^{B} \|\mathbf{Z}^{(b)}\|_*. \tag{54}$$

This lower bound for $\mathbf{Z}$ is achievable without changing the fitting term, by letting

$$\mathbf{Z} = \begin{bmatrix} \mathbf{Z}^{(1)} & 0 & 0 \\ 0 & \mathbf{Z}^{(2)} & 0 \\ & \cdots & \\ 0 & 0 & \mathbf{Z}^{(B)} \end{bmatrix} \tag{55}$$

Thus, in the block diagonal case, we have

$$\begin{aligned} p_{SA}^* = \min_{\mathbf{Z}} \sum_{i=1}^{n} \mathcal{L}\left(\sum_{b=1}^{B} \sum_{k=1}^{d_b} \sum_{\ell=1}^{d_b} \mathbf{G}_i^{(b)}[k,\ell] \mathbf{X}_i \mathbf{Z}^{(b,k,\ell)}, \mathbf{Y}_i \right) \\ + \beta \sum_{b=1}^{B} \|\mathbf{Z}^{(b)}\|_* \end{aligned} \tag{56}$$

as desired. □

A.4. Proof of Theorem 3.3

We first note that for self-attention we remove the $1/\sqrt{d}$ factor for simplicity. We apply Lemmas A.1 and A.2 to (12) with the ReLU activation function to obtain

$$\begin{aligned} p_{SA}^* = \max_{\mathbf{V}_i} & - \sum_{i=1}^{n} \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i) \\ \text{s.t.} & \max_{\|\mathbf{W}_1\|_F \leq 1} \left\| \sum_{i=1}^{N} \mathbf{V}_i^\top (\mathbf{X}_i \mathbf{W}_1 \mathbf{X}_i^\top)_+ \mathbf{X}_i \right\|_F . \end{aligned} \tag{57}$$

We again apply $\mathbf{vec}(\mathbf{ABC}) = (\mathbf{C}^\top \otimes \mathbf{A})\mathbf{vec}(\mathbf{B})$ (Magnus & Neudecker, 2019) to obtain

$$\begin{aligned} p_{SA}^* = \max_{\mathbf{V}_i} & - \sum_{i=1}^{n} \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i) \\ \text{s.t.} & \max_{\|\mathbf{w}_1\|_2 \leq 1} \left\| \sum_{i=1}^{N} (\mathbf{X}_i \otimes \mathbf{V}_i^\top)((\mathbf{X}_i \otimes \mathbf{X}_i)\mathbf{w}_1)_+ \right\|_2 . \end{aligned} \tag{58}$$

Now, let $\mathbf{D}_j^{(i)} \in \mathbb{R}^{s^2 \times s^2}$ be the ith block of $\mathbf{D}_j$, and enumerate over all possible hyperplane arrangements j. Then, we have

$$\begin{aligned} p_{SA}^* = \max_{\mathbf{V}_i} & - \sum_{i=1}^{n} \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i) \\ \text{s.t.} & \max_{\substack{\|\mathbf{w}_1\|_2 \leq 1 \\ j \in [P] \\ \mathbf{K}_j \mathbf{w}_1 \geq 0}} \left\| \sum_{i=1}^{N} (\mathbf{X}_i^\top \otimes \mathbf{V}_i^\top) \mathbf{D}_j^{(i)} (\mathbf{X}_i \otimes \mathbf{X}_i) \mathbf{w}_1 \right\|_2 . \end{aligned} \tag{59}$$

We note that the norm constraint here has dimension dc and $\mathbf{w}_1$ has dimension d^2, so by (Shapiro, 2009; Pilanci & Ergen, 2020) this strong duality result from Lemma A.2 requires that $m^* \leq n \min\{d^2, dc\}$. Now, using the concept of dual norm, this is equal to

$$p_{SA}^* = \max_{\mathbf{V}_i} - \sum_{i=1}^{n} \mathcal{L}^* \left(\mathbf{V}_i, \mathbf{Y}_i\right)$$
$$\text{s.t.} \max_{\substack{\|\mathbf{g}\|_2 \leq 1 \\ \|\mathbf{w}_1\|_2 \leq 1 \\ j \in [P] \\ \mathbf{K}_j \mathbf{w}_1 \geq 0}} \mathbf{g}^\top \sum_{i=1}^{N} (\mathbf{X}_i^\top \otimes \mathbf{V}_i^\top) \mathbf{D}_j^{(i)} (\mathbf{X}_i \otimes \mathbf{X}_i) \mathbf{w}_1 \tag{60}$$

We can also define sets $\mathcal{C}_j := \{\mathbf{Z} = \mathbf{u}\mathbf{g}^\top \in \mathbb{R}^{d^2 \times dc} : \mathbf{K}_j \mathbf{u} \geq 0 \; \forall i, \; \|\mathbf{Z}\|_* \leq 1\}$. Then, we have

$$p_{SA}^* = \max_{\mathbf{V}_i} - \sum_{i=1}^{n} \mathcal{L}^* \left(\mathbf{V}_i, \mathbf{Y}_i\right)$$
$$\text{s.t.} \max_{\substack{j \in [P] \\ \mathbf{Z} \in \mathcal{C}_j}} \text{trace} \left(\sum_{i=1}^{n} (\mathbf{X}_i^\top \otimes \mathbf{V}_i^\top) \mathbf{D}_j^{(i)} (\mathbf{X}_i \otimes \mathbf{X}_i) \mathbf{Z} \right) \tag{61}$$

Now, we simply need to form the Lagrangian and solve. The Lagrangian is given by

$$p_{SA}^* = \max_{\mathbf{V}_i} \min_{\lambda \geq 0} \min_{\mathbf{Z}_j \in \mathcal{C}_j} - \sum_{i=1}^{n} \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i) + \sum_{j=1}^{P} \lambda_j \left(\beta - \sum_{i=1}^{n} \mathbf{vec} \left(\mathbf{D}_j^{(i)} (\mathbf{X}_i \otimes \mathbf{X}_i) \mathbf{Z}_j \right)^\top \mathbf{vec} \left(\mathbf{X}_i \otimes \mathbf{V}_i \right) \right) \tag{62}$$

We now can switch the order of max and min via Sion's minimax theorem and maximize over $\mathbf{V}_i$. Defining $\mathbf{K}_{c,s}$ as the (c, s) commutation matrix (Magnus & Neudecker, 2019):

$$\mathbf{vec}(\mathbf{X}_i \otimes \mathbf{V}_i) = ((\mathbf{I}_d \otimes \mathbf{K}_{c,s})(\mathbf{vec}(\mathbf{X}_i) \otimes \mathbf{I}_c) \otimes \mathbf{I}_s) \, \mathbf{vec}(\mathbf{V}_i)$$

Maximizing over $\mathbf{V}_i$, we have

$$p_{SA}^* = \min_{\lambda \geq 0} \min_{\mathbf{Z}_j \in \mathcal{C}_j} \sum_{i=1}^{n} \mathcal{L} \left(\sum_{j=1}^{P} \left((\mathbf{vec}(\mathbf{X}_i)^\top \otimes \mathbf{I}_c)(\mathbf{I}_d \otimes \mathbf{K}_{s,c}) \otimes \mathbf{I}_s \right) \mathbf{vec} \left(\mathbf{D}_j^{(i)} (\mathbf{X}_i \otimes \mathbf{X}_i) \lambda_j \mathbf{Z}_j \right), \mathbf{vec}(\mathbf{Y}_i) \right) + \beta \sum_{j=1}^{m} \lambda_j. \tag{63}$$

Again rescaling $\tilde{\mathbf{Z}}_j = \lambda_j \mathbf{Z}_j$, we have

$$p_{SA}^* = \min_{\mathbf{Z}_j \in \mathbb{R}^{d^2 \times dc}} \sum_{i=1}^{n} \mathcal{L} \left(\sum_{j=1}^{P} \left((\mathbf{vec}(\mathbf{X}_i)^\top \otimes \mathbf{I}_c)(\mathbf{I}_d \otimes \mathbf{K}_{s,c}) \otimes \mathbf{I}_s \right) \mathbf{vec} \left(\mathbf{D}_j^{(i)} (\mathbf{X}_i \otimes \mathbf{X}_i) \mathbf{Z}_j \right), \mathbf{vec}(\mathbf{Y}_i) \right) + \beta \sum_{j=1}^{m} \|\mathbf{Z}_j\|_{*,\mathrm{K}_j}. \tag{64}$$

It appears as though this is a very complicated function, but it actually simplifies greatly. In particular, one can write this as

$$p_{SA}^* = \min_{\mathbf{Z}_j \in \mathbb{R}^{d^2 \times dc}} \sum_{i=1}^{n} \mathcal{L} \left(\hat{\mathbf{Y}}_i, \mathbf{Y}_i \right) + \beta \|\mathbf{Z}\|_{*,\mathrm{K}_j}$$
$$\hat{\mathbf{Y}}_i[o,p] := \sum_{j=1}^{P} \sum_{k=1}^{d} \sum_{l=1}^{d} \sum_{t=1}^{s} \mathbf{X}_i[t,l] \mathbf{X}_i[t,k] \mathbf{D}_j^{(t,m)} \mathbf{X}_i[o,:]^\top \mathbf{Z}_j^{(k,l)}. \tag{65}$$

Making any final simplifications, one obtains the desired result.

$$p_{SA}^* = \min_{\mathbf{Z}_j \in \mathbb{R}^{d^2 \times dc}} \sum_{i=1}^{n} \mathcal{L} \left(\sum_{j=1}^{P} \sum_{k=1}^{d} \sum_{\ell=1}^{d} \mathbf{G}_{i,j}^{(k,\ell)} \mathbf{X}_i \mathbf{Z}_j^{(k,\ell)}, \mathbf{Y}_i \right) + \beta \sum_{j=1}^{m} \|\mathbf{Z}_j\|_{*,\mathrm{K}_j}. \tag{66}$$

Lastly, we also demonstrate that there is a one-to-one mapping between the solution to (15) and (12). In particular, imagine we have a solution $\{\mathbf{Z}_j^*\}_{j=1}^{P}$ to (15) with optimal value p_{CVX}^*, where $m^* \leq n$ are non-zero. Take the cone-constrained SVD of $\mathbf{Z}_j^*$ as $\sum_{x=1}^{r_j} \sigma_{jx} \mathbf{u}_{jx} \mathbf{v}_{jx}^\top$, where $\mathbf{u}_{jx} \in \mathbb{R}^{d^2}$ and $(2\mathbf{D}_j - \mathbf{I}_{ns^2})\mathbf{X}\mathbf{u}_{jx} \geq 0$, and $\mathbf{v}_j \in \mathbb{R}^{dc}$. Let $\text{vec}^{-1}(\mathbf{u_{jx}}) \in \mathbb{R}^{d \times d}$ be the result of taking chunks of d-length vectors from $\mathbf{u}_{jx}$ and stacking them in columns. Similarly, let $\text{vec}^{-1}(\mathbf{v_{jx}}) \in \mathbb{R}^{c \times d}$ be the result of taking chunks of c-length vectors from $\mathbf{v}_j$ and stacking them in columns. Furthermore we will let $\text{vec}^{-1}(\mathbf{u_j})_k$ be the kth column of $\text{vec}^{-1}(\mathbf{u_j})$. Then, recognize that

$$\mathbf{Z}^{*(k,\ell)}_{j} = \sum_{x=1}^{r} \sigma_{jx} \text{vec}^{-1}(\mathbf{u_{jx}})_k \text{vec}^{-1}(\mathbf{v_{jx}})_\ell^\top$$

Thus, given $\mathbf{Z}^*$, we can form a candidate solution to (12) as follows:

$$\begin{aligned}
\mathbf{Z}_j^* &= \sum_{x=1}^{r_j} \sigma_{jx} \mathbf{u}_{jx} \mathbf{v}_{jx}^\top,\ (2\mathbf{D}_j - \mathbf{I}_{ns^2})\mathbf{X}\mathbf{u}_{jx} \geq 0,\ \|\mathbf{u}_{jx}\|_2 = 1,\ \|\mathbf{v}_{jx}\|_2 = 1 \\
\hat{\mathbf{W}}_{1jx} &= \sqrt{\sigma_{jx}} \text{vec}^{-1}(\mathbf{u_{jx}}) \\
\hat{\mathbf{W}}_{2jx} &= \sqrt{\sigma_{jx}} \text{vec}^{-1}(\mathbf{v_{jx}})^\top
\end{aligned}$$

We then have

$$\begin{aligned}
\hat{p}_{NCVX} &= \sum_{i=1}^{n} \mathcal{L}\left(\sum_{j=1}^{m^*} \sum_{x=1}^{r_j} (\mathbf{X}_i \hat{\mathbf{W}}_{1jx} \mathbf{X}_i^\top)_+ \mathbf{X}_i \hat{\mathbf{W}}_{2jx}, \mathbf{Y}_i \right) + \frac{\beta}{2} \sum_{j=1}^{m^*} \sum_{x=1}^{r_j} \|\hat{\mathbf{W}}_{1jx}\|_F^2 + \|\hat{\mathbf{W}}_{2jx}\|_F^2 \\
&= \sum_{i=1}^{n} \mathcal{L}\left(\sum_{j=1}^{m^*} \sum_{x=1}^{r_j} \sigma_{jx} \left(\text{diag}^{-1}(\mathbf{D}_j^{(i)}) \odot (\mathbf{X}_i \text{vec}^{-1}(\mathbf{u_{jx}}) \mathbf{X}_i^\top) \right) \mathbf{X}_i \text{vec}^{-1}(\mathbf{v_{jx}})^\top, \mathbf{Y}_i \right) + \frac{\beta}{2} \sum_{j=1}^{m^*} \sum_{x=1}^{r_j} \|\sqrt{\sigma_{jx}} \mathbf{u}_{jx}\|_2^2 + \|\sqrt{\sigma_{jx}} \mathbf{v}_{jx}\|_2^2 \\
&= \sum_{i=1}^{n} \mathcal{L}\left(\sum_{k=1}^{d} \sum_{\ell=1}^{d} \sum_{j=1}^{m^*} \sum_{x=1}^{r_j} \sigma_{jx} \mathbf{G}_{i,j}^{(k,\ell)} \mathbf{X}_i \text{vec}^{-1}(\mathbf{u_j})_k \text{vec}^{-1}(\mathbf{v_j})_\ell^\top, \mathbf{Y}_i \right) + \frac{\beta}{2} \sum_{j=1}^{m^*} \sum_{x=1}^{r_j} \sigma_{jx} + \sigma_{jx} \\
&= \sum_{i=1}^{n} \mathcal{L}\left(\sum_{k=1}^{d} \sum_{\ell=1}^{d} \mathbf{G}_{i,j}^{(k,\ell)} \mathbf{X}_i \mathbf{Z}_j^{*(k,\ell)}, \mathbf{Y}_i \right) + \beta \sum_{j=1}^{m^*} \|\mathbf{Z}_j^*\|_{*,\mathrm{K}_j} \\
&= p_{CVX}^*
\end{aligned}$$

Thus, the two solutions match. □

A.5. Proof of Theorem 4.1

We apply Lemmas A.1 and A.2 to (17) with the linear activation function to obtain

$$\begin{aligned}
p_{MM}^* = \max_{\mathbf{V}_i} &- \sum_{i=1}^{n} \mathcal{L}^*\left(\mathbf{V}_i, \mathbf{Y}_i\right) \\
\text{s.t.} &\max_{\|\mathbf{W}_1\|_F \leq 1} \|\sum_{i=1}^{n} \mathbf{V}_i^T \mathbf{W}_1 \mathbf{X}_i\|_F \leq \beta
\end{aligned} \tag{67}$$

We again apply $\mathbf{vec}(\mathbf{ABC}) = (\mathbf{C}^\top \otimes \mathbf{A})\mathbf{vec}(\mathbf{B})$ (Magnus & Neudecker, 2019) and maximize over $\mathbf{vec}(\mathbf{W}_1)$ to obtain

$$\begin{aligned}
p_{MM}^* &= -\max_{\mathbf{V}_i} \sum_{i=1}^{n} \mathcal{L}^*\left(\mathbf{V}_i, \mathbf{Y}_i\right) \\
&\text{s.t. } \|\sum_{i=1}^{n} \mathbf{X}_i^\top \otimes \mathbf{V}_i^\top\|_2 \leq \beta.
\end{aligned}$$

We note that the norm constraint here has dimension dc and $\mathbf{vec}(\mathbf{W}_1)$ has dimension s^2, so by (Shapiro, 2009) this strong duality result from Lemma A.2 requires that $m^* \leq \min\{s^2, dc\}$. The Lagrangian is given by

$$p_{MM}^* = \max_{\mathbf{V}_i} \min_{\lambda \geq 0} \min_{\|\mathbf{Z}\|_* \leq 1} - \sum_{i=1}^{n} \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i) + \sum_{j=1}^{m} \lambda_j \left(\beta - \sum_{i=1}^{n} \text{trace}\left((\mathbf{I}_d \otimes \mathbf{V}_i^T)(\mathbf{X}_i^\top \otimes \mathbf{I}_s) \mathbf{Z} \right) \right) \tag{68}$$

which by Sion's minimax theorem and simplification can also be written as

$$p^*_{MM} = \min_{\lambda\geq 0} \min_{\|\mathbf{Z}\|_*\leq 1} \max_{\mathbf{V}_i} -\sum_{i=1}^{n} \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i) + \sum_{j=1}^{m} \lambda_j \left(\beta - \sum_{i=1}^{n} \mathbf{vec}((\mathbf{X}_i^\top \otimes \mathbf{I}_s)\mathbf{Z})^\top \mathbf{vec}(\mathbf{I}_d \otimes \mathbf{V}_i)\right) \tag{69}$$

We define $\mathbf{K}_{c,d}$ as the (c, d) commutation matrix (Magnus & Neudecker, 2019):

$$\mathbf{vec}(\mathbf{I}_d \otimes \mathbf{V}_i) = ((\mathbf{I}_d \otimes \mathbf{K}_{c,d})(\mathbf{vec}(\mathbf{I}_d) \otimes \mathbf{I}_c) \otimes \mathbf{I}_s)\, \mathbf{vec}(\mathbf{V}_i)$$

Maximizing over $\mathbf{V}_i$, followed by re-scaling $\tilde{\mathbf{Z}} = \lambda \mathbf{Z}$ gives us

$$p^*_{MM} = \min_{\mathbf{Z}\in\mathbb{R}^{s^2\times dc}} \sum_{i=1}^{n} \mathcal{L}\left(\left((\mathbf{vec}(\mathbf{I}_d)^\top \otimes \mathbf{I}_c)(\mathbf{I}_d \otimes \mathbf{K}_{d,c}) \otimes \mathbf{I}_s\right) \mathbf{vec}\left((\mathbf{X}_i^\top \otimes \mathbf{I}_s)\mathbf{Z}\right), \mathbf{vec}(\mathbf{Y}_i)\right) + \beta\|\mathbf{Z}\|_* \tag{70}$$

Making any final simplifications, one obtains the desired result.

$$\begin{aligned} p^*_{MM} &= \min_{\mathbf{Z}\in\mathbb{R}^{s^2\times dc}} \sum_{i=1}^{n} \mathcal{L}\left(\begin{bmatrix} f_1(\mathbf{X}_i) & \cdots & f_c(\mathbf{X}_i)\end{bmatrix}, \mathbf{Y}_i\right) + \beta\|\mathbf{Z}\|_* \\ f_p(\mathbf{X}_i) &:= \mathbf{Z}^{(p)}\mathbf{vec}(\mathbf{X}_i). \end{aligned} \tag{71}$$

Lastly, we also demonstrate that there is a one-to-one mapping between the solution to (18) and (17). In particular, we have a solution $\mathbf{Z}^*$ to (18) with optimal value p^*_{CVX}. First, rearrange the solution to be of the form $\tilde{\mathbf{Z}}^*$ of (19). Let $r := \mathrm{rank}(\tilde{\mathbf{Z}}^*)$, and take the SVD of $\tilde{\mathbf{Z}}^*$ as $\sum_{j=1}^{r} \sigma_j \mathbf{u}_j \mathbf{v}_j^\top$, where $\mathbf{u}_j \in \mathbb{R}^{s^2}$ and $\mathbf{v}_j \in \mathbb{R}^{dc}$. Let $\mathrm{vec}^{-1}(\mathbf{u_j}) \in \mathbb{R}^{s\times s}$ be the result of taking chunks of s-length vectors from $\mathbf{u}_j$ and stacking them in columns. Similarly, let $\mathrm{vec}^{-1}(\mathbf{v_j}) \in \mathbb{R}^{c\times d}$ be the result of taking chunks of c-length vectors from $\mathbf{v}_j$ and stacking them in columns. Furthermore we will let $\mathrm{vec}^{-1}(\mathbf{u_j})_t$ be the tth column of $\mathrm{vec}^{-1}(\mathbf{u_j})$. Then, recognize that

$$\tilde{\mathbf{Z}}^{*^{(t,k)}} = \sum_{j=1}^{r} \sigma_j \mathrm{vec}^{-1}(\mathbf{u_j})_t \mathrm{vec}^{-1}(\mathbf{v_j})_k^\top$$

Thus, given $\mathbf{Z}^*$, we can form a candidate solution to (12) as follows:

$$\begin{aligned} \tilde{\mathbf{Z}}^* &= \sum_{j=1}^{r} \sigma_j \mathbf{u}_j \mathbf{v}_j^\top \\ \hat{\mathbf{W}}_{1j} &= \sqrt{\sigma_j}\mathrm{vec}^{-1}(\mathbf{u_j}) \\ \hat{\mathbf{W}}_{2j} &= \sqrt{\sigma_j}\mathrm{vec}^{-1}(\mathbf{v_j})^\top \end{aligned}$$

We then have

$$\begin{aligned} \hat{p}_{NCVX} &= \sum_{i=1}^{n} \mathcal{L}\left(\sum_{j=1}^{r} \hat{\mathbf{W}}_{1j}\mathbf{X}_i\hat{\mathbf{W}}_{2j}, \mathbf{Y}_i\right) + \frac{\beta}{2}\sum_{j=1}^{r} \|\hat{\mathbf{W}}_{1j}\|_F^2 + \|\hat{\mathbf{W}}_{2j}\|_F^2 \\ &= \sum_{i=1}^{n} \mathcal{L}\left(\mathbf{X}_i \sum_{j=1}^{r} \sigma_j \mathrm{vec}^{-1}(\mathbf{u_j})\mathbf{X}_i\mathrm{vec}^{-1}(\mathbf{v_j})^\top, \mathbf{Y}_i\right) + \frac{\beta}{2}\sum_{j=1}^{r} \|\sqrt{\sigma_j}\mathbf{u}_j\|_2^2 + \|\sqrt{\sigma_j}\mathbf{v}_j\|_2^2 \\ &= \sum_{i=1}^{n} \mathcal{L}\left(\sum_{t=1}^{s}\sum_{k=1}^{d}\sum_{j=1}^{r} \sigma_j \mathrm{vec}^{-1}(\mathbf{u_j})_t \mathbf{X}_i[t,k]\mathrm{vec}^{-1}(\mathbf{v_j})_k^\top, \mathbf{Y}_i\right) + \frac{\beta}{2}\sum_{j=1}^{r} \sigma_j + \sigma_j \\ &= \sum_{i=1}^{n} \mathcal{L}\left(\sum_{t=1}^{s}\sum_{k=1}^{d} \mathbf{X}_i[k,\ell]\tilde{\mathbf{Z}}^{*^{(t,k)}}, \mathbf{Y}_i\right) + \beta\|\tilde{\mathbf{Z}}^*\|_* \\ &= p^*_{CVX} \end{aligned}$$

Thus, the two solutions match. □

A.6. Proof of Theorem 4.2

We apply Lemmas A.1 and A.2 to (17) with the ReLU activation function to obtain

$$p^*_{MM} = \max_{\mathbf{V}_i} - \sum_{i=1}^n \mathcal{L}^*\left(\mathbf{V}_i, \mathbf{Y}_i\right)$$
$$\text{s.t.} \max_{\|\mathbf{W}_1\|_F \le 1} \left\| \sum_{i=1}^n \mathbf{V}_i^T (\mathbf{W}_1 \mathbf{X}_i)_+ \right\|_F \le \beta \tag{72}$$

This is equivalent to (Magnus & Neudecker, 2019)

$$p^*_{MM} = \max_{\mathbf{V}_i} - \sum_{i=1}^n \mathcal{L}^*\left(\mathbf{V}_i, \mathbf{Y}_i\right)$$
$$\text{s.t.} \max_{\|\mathbf{W}_1\|_F \le 1} \left\| \sum_{i=1}^n (\mathbf{I}_d \otimes \mathbf{V}_i^T)((\mathbf{X}_i^\top \otimes \mathbf{I}_s)\mathbf{vec}(\mathbf{W}_1)_+ \right\|_F \le \beta \tag{73}$$

We note that the norm constraint here has dimension dc and $\mathbf{vec}(\mathbf{W}_1)$ has dimension s^2, so by (Shapiro, 2009; Pilanci & Ergen, 2020) this strong duality result from Lemma A.2 requires that $m^* \le n \min\{s^2, dc\}$. Now, let $\mathbf{D}_j^{(i)} \in \mathbb{R}^{sd \times sd}$ be the ith block of $\mathbf{D}_j$, and enumerate over all possible hyperplane arrangements j. Then, we have

$$p^*_{MM} = \max_{\mathbf{V}_i} - \sum_{i=1}^n \mathcal{L}^*\left(\mathbf{V}_i, \mathbf{Y}_i\right)$$
$$\text{s.t.} \max_{\substack{\|\mathbf{w}_1\|_2 \le 1 \\ j \in [P] \\ \mathbf{K}_j \mathbf{w}_1 \ge 0}} \left\| \sum_{i=1}^n (\mathbf{I}_d \otimes \mathbf{V}_i^T) \mathbf{D}_j^{(i)} (\mathbf{X}_i^\top \otimes \mathbf{I}_s) \mathbf{w}_1 \right\|_2 \le \beta. \tag{74}$$

Now, using the concept of dual norm, this is equal to

$$p^*_{MM} = \max_{\mathbf{V}_i} - \sum_{i=1}^n \mathcal{L}^*\left(\mathbf{V}_i, \mathbf{Y}_i\right)$$
$$\text{s.t.} \max_{\substack{\|\mathbf{g}\|_2 \le 1 \\ \|\mathbf{w}_1\|_2 \le 1 \\ j \in [P] \\ \mathbf{K}_j \mathbf{w}_1 \ge 0}} \mathbf{g}^\top \sum_{i=1}^n (\mathbf{I}_d \otimes \mathbf{V}_i^T) \mathbf{D}_j^{(i)} (\mathbf{X}_i^\top \otimes \mathbf{I}_s) \mathbf{w}_1 \le \beta \tag{75}$$

We can also define sets $\mathcal{C}_j := \{\mathbf{Z} = \mathbf{u}\mathbf{g}^\top \in \mathbb{R}^{s^2 \times dc} : \mathbf{K}_j \mathbf{u} \ge 0, \ \|\mathbf{Z}\|_* \le 1\}$. Then, we have

$$p^*_{MM} = \max_{\mathbf{V}_i} - \sum_{i=1}^n \mathcal{L}^*\left(\mathbf{V}_i, \mathbf{Y}_i\right)$$
$$\text{s.t.} \max_{\substack{j \in [P] \\ \mathbf{Z} \in \mathcal{C}_j}} \operatorname{trace}\left(\sum_{i=1}^n (\mathbf{I}_d \otimes \mathbf{V}_i^T) \mathbf{D}_j^{(i)} (\mathbf{X}_i^\top \otimes \mathbf{I}_s) \mathbf{Z} \right) \le \beta \tag{76}$$

Now, we simply need to form the Lagrangian and solve. The Lagrangian is given by

$$p^*_{MM} = \max_{\mathbf{V}_i} \min_{\lambda \ge 0} \min_{\mathbf{Z}_j \in \mathcal{C}_j} - \sum_{i=1}^n \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i) + \sum_{j=1}^P \lambda_j \left(\beta - \sum_{i=1}^n \operatorname{trace}\left((\mathbf{I}_d \otimes \mathbf{V}_i^T) \mathbf{D}_j^{(i)} (\mathbf{X}_i^\top \otimes \mathbf{I}_s) \mathbf{Z} \right) \right) \tag{77}$$

We now can switch the order of max and min via Sion's minimax theorem and maximize over $\mathbf{V}_i$:

$$p^*_{MM} = \min_{\lambda \ge 0} \min_{\mathbf{Z}_j \in \mathcal{C}_j} \max_{\mathbf{V}_i} - \sum_{i=1}^n \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i) + \sum_{j=1}^P \lambda_j \left(\beta - \sum_{i=1}^n \mathbf{vec}\left(\mathbf{D}_j^{(i)} (\mathbf{X}_i^\top \otimes \mathbf{I}_s) \mathbf{Z}_j \right)^\top \mathbf{vec}\left(\mathbf{I}_d \otimes \mathbf{V}_i \right) \right) \tag{78}$$

Now, defining $\mathbf{K}_{c,d}$ as the (c,d) commutation matrix:

$$\mathbf{vec}(\mathbf{I}_d \otimes \mathbf{V}_i) = ((\mathbf{I}_d \otimes \mathbf{K}_{c,d})(\mathbf{vec}(\mathbf{I}_d) \otimes \mathbf{I}_c) \otimes \mathbf{I}_s)\, \mathbf{vec}(\mathbf{V}_i)$$

Solving over $\mathbf{V}_i$ yields

$$p^*_{MM} = \min_{\lambda \geq 0} \min_{\mathbf{Z}_j \in \mathcal{C}_j} \sum_{i=1}^{n} \mathcal{L}\left(\sum_{j=1}^{P} \left((\mathbf{vec}(\mathbf{I}_d)^\top \otimes \mathbf{I}_c)(\mathbf{I}_d \otimes \mathbf{K}_{d,c}) \otimes \mathbf{I}_s\right) \mathbf{vec}\left(\mathbf{D}_j^{(i)}(\mathbf{X}_i^\top \otimes \mathbf{I}_s)\lambda_j \mathbf{Z}_j\right), \mathbf{vec}(\mathbf{Y}_i) \right) + \beta \sum_{j=1}^{P} \lambda_j \tag{79}$$

Re-scaling $\tilde{\mathbf{Z}}_j = \lambda_j \mathbf{Z}_j$ gives us

$$p^*_{MM} = \min_{\mathbf{Z}_j} \sum_{i=1}^{n} \mathcal{L}\left(\sum_{j=1}^{P} \left((\mathbf{vec}(\mathbf{I}_d)^\top \otimes \mathbf{I}_c)(\mathbf{I}_d \otimes \mathbf{K}_{d,c}) \otimes \mathbf{I}_s\right) \mathbf{vec}\left(\mathbf{D}_j^{(i)}(\mathbf{X}_i^\top \otimes \mathbf{I}_s)\mathbf{Z}_j\right), \mathbf{vec}(\mathbf{Y}_i) \right) + \beta \sum_{j=1}^{P} \|\mathbf{Z}_j\|_{*,\mathrm{K}_j}. \tag{80}$$

One can actually greatly simplify this result, and can re-write this as

$$p^*_{MM} = \min_{\mathbf{Z}_j \in \mathbb{R}^{s^2 \times dc}} \sum_{i=1}^{n} \mathcal{L}\left(\begin{bmatrix} f_1(\mathbf{X}_i) & \cdots & f_c(\mathbf{X}_i)\end{bmatrix}, \mathbf{Y}_i\right) + \beta \sum_{j=1}^{P} \|\mathbf{Z}_j\|_{*,\mathrm{K}_j} \tag{81}$$

$$f_p(\mathbf{X}_i) := \sum_{j=1}^{P} \left[\mathbf{D}_j^{(i,1)}\mathbf{Z}_j^{(p,1)} \cdots \mathbf{D}_j^{(i,d)}\mathbf{Z}_j^{(p,d)}\right] \mathbf{vec}(\mathbf{X}_i). \tag{82}$$

Lastly, we also demonstrate that there is a one-to-one mapping between the solution to (20) and (17). In particular, we have a solution $\{\mathbf{Z}_j^*\}_{j=1}^{m^*}$ to (20) with optimal value p^*_{CVX}, where $m^* \leq n$ are non-zero. First, rearrange the solution to be of the form $\tilde{\mathbf{Z}}_j^*$ of (21). Take the cone-constrained SVD of $\tilde{\mathbf{Z}}_j^*$ as $\sum_{x=1}^{r_j} \sigma_{jx}\mathbf{u}_{jx}\mathbf{v}_{jx}^\top$, where $\mathbf{u}_{jx} \in \mathbb{R}^{s^2}$ and $\mathbf{v}_{jx} \in \mathbb{R}^{dc}$. Let $\mathrm{vec}^{-1}(\mathbf{u_{jx}}) \in \mathbb{R}^{s \times s}$ be the result of taking chunks of s-length vectors from $\mathbf{u}_{jx}$ and stacking them in columns. Similarly, let $\mathrm{vec}^{-1}(\mathbf{v_{jx}}) \in \mathbb{R}^{c \times d}$ be the result of taking chunks of c-length vectors from $\mathbf{v}_{jx}$ and stacking them in columns. Furthermore we will let $\mathrm{vec}^{-1}(\mathbf{u_{jx}})_t$ be the tth column of $\mathrm{vec}^{-1}(\mathbf{u_{jx}})$. Then, recognize that

$$\tilde{\mathbf{Z}}_j^{*^{(t,k)}} = \sum_{x=1}^{r_j} \sigma_{jx}\mathrm{vec}^{-1}(\mathbf{u_{jx}})_t \mathrm{vec}^{-1}(\mathbf{v_{jx}})_k^\top$$

Thus, given $\mathbf{Z}_j^*$, we can form a candidate solution to (12) as follows:

$$\begin{aligned}
\tilde{\mathbf{Z}}_j^* &= \sum_{x=1}^{rx} \sigma_{jx}\mathbf{u}_{jx}\mathbf{v}_{jx}^\top,\ (2\mathbf{D}_j - \mathbf{I}_{nd})\mathbf{X}\mathbf{u}_{jx} \geq 0,\ \|\mathbf{u}_{jx}\|_2 = 1,\ \|\mathbf{v}_{jx}\|_2 = 1 \\
\hat{\mathbf{W}}_{1jx} &= \sqrt{\sigma_j}\mathrm{vec}^{-1}(\mathbf{u_{jx}}) \\
\hat{\mathbf{W}}_{2jx} &= \sqrt{\sigma_j}\mathrm{vec}^{-1}(\mathbf{v_{jx}})^\top
\end{aligned}$$

We then have

$$\begin{aligned}
\hat{p}_{NCVX} &= \sum_{i=1}^{n} \mathcal{L}\left(\sum_{j=1}^{m^*}\sum_{x=1}^{r_j} (\hat{\mathbf{W}}_{1jx}\mathbf{X}_i)_+ \hat{\mathbf{W}}_{2jx}, \mathbf{Y}_i \right) + \frac{\beta}{2}\sum_{j=1}^{m^*}\sum_{x=1}^{r_j} \|\hat{\mathbf{W}}_{1j}\|_F^2 + \|\hat{\mathbf{W}}_{2j}\|_F^2 \\
&= \sum_{i=1}^{n} \mathcal{L}\left(\mathbf{X}_i \sum_{j=1}^{m^*}\sum_{x=1}^{r_j} \sigma_j (\mathrm{vec}^{-1}(\mathbf{u_j})\mathbf{X}_i)_+ \mathrm{vec}^{-1}(\mathbf{v_j})^\top, \mathbf{Y}_i \right) + \frac{\beta}{2}\sum_{j=1}^{m^*}\sum_{x=1}^{r_j} \|\sqrt{\sigma_j}\mathbf{u}_j\|_2^2 + \|\sqrt{\sigma_j}\mathbf{v}_j\|_2^2 \\
&= \sum_{i=1}^{n} \mathcal{L}\left(\sum_{t=1}^{s}\sum_{k=1}^{d}\sum_{j=1}^{m^*}\sum_{x=1}^{r_j} \sigma_j (\mathrm{vec}^{-1}(\mathbf{u_j})_t \mathbf{X}_i[t,k])_+ \mathrm{vec}^{-1}(\mathbf{v_j})_k^\top, \mathbf{Y}_i \right) + \frac{\beta}{2}\sum_{j=1}^{m^*}\sum_{x=1}^{r_j} \sigma_j + \sigma_j \\
&= \sum_{i=1}^{n} \mathcal{L}\left(\sum_{j=1}^{m^*}\sum_{t=1}^{s}\sum_{k=1}^{d} \mathbf{X}_i[k,\ell]\mathbf{D}_j^{(i,k)}\tilde{\mathbf{Z}}_j^{*^{(t,k)}}, \mathbf{Y}_i \right) + \beta\sum_{j=1}^{m^*} \|\tilde{\mathbf{Z}}_j^*\|_* \\
&= p^*_{CVX}
\end{aligned}$$

Thus, the two solutions match. □

A.7. Proof of Lemma 4.3

The expression (22) can equivalently be written as

$$f_{FN}(\mathbf{X}_i) = \sigma\left(\left(\mathbf{F}^{-1}\begin{bmatrix}\mathbf{F}_1^\top\mathbf{X}_i & & \\ & \ddots & \\ & & \mathbf{F}_s^\top\mathbf{X}_i\end{bmatrix}\begin{bmatrix}\mathbf{V}^{(1)} \\ \cdots \\ \mathbf{V}^{(s)}\end{bmatrix}\right)\mathbf{W}_1\right)\mathbf{W}_2 \tag{83}$$

and further as

$$f_{FN}(\mathbf{X}_i) = \sigma\left(\left(\mathbf{F}^{-1}\begin{bmatrix}\mathbf{F}_1^\top\mathbf{X}_i & & \\ & \ddots & \\ & & \mathbf{F}_s^\top\mathbf{X}_i\end{bmatrix}(\mathbf{F}\otimes\mathbf{I}_d)\begin{bmatrix}\mathbf{L}^{(1)} \\ \cdots \\ \mathbf{L}^{(s)}\end{bmatrix}\right)\mathbf{W}_1\right)\mathbf{W}_2, \tag{84}$$

which simplifies to

$$f_{FN}(\mathbf{X}_i) = \sigma\left(\left(\begin{bmatrix}\mathbf{X}_i & \mathbf{X}_{i(1)}\cdots & \mathbf{X}_{i(s)}\end{bmatrix}\mathbf{L}\right)\mathbf{W}_1\right)\mathbf{W}_2, \tag{85}$$

where $\mathbf{X}_{i(u)}$ is $\mathbf{X}_i$ circularly shifted by u spots in its first dimension, and $\mathbf{K}$ has been reshaped to the form $\mathbb{R}^{sd\times d}$. Noting that we can merge $\mathbf{L}\mathbf{W}_1$ into one matrix without losing any expressibility, we obtain

$$f_{FN}(\mathbf{X}_i) = \sigma\left(\mathrm{circ}(\mathbf{X}_i)\mathbf{W}_1\right)\mathbf{W}_2 = \sum_{j=1}^{m}\sigma\left(\mathrm{circ}(\mathbf{X}_i)\mathbf{w}_{1j}\right)\mathbf{w}_{2j}^\top \tag{86}$$

as desired. □

A.8. Proof of Theorem 4.4

We start with the problem

$$p^*_{FN} = \min_{\mathbf{w}_{1j},\mathbf{w}_{2j}}\sum_{i=1}^{n}\mathcal{L}\left(\sum_{j=1}^{m}\mathrm{circ}(\mathbf{X}_i)\mathbf{w}_{1j}\mathbf{w}_{2j}^\top, \mathbf{Y}_i\right) + \frac{\beta}{2}\sum_{j=1}^{m}\|\mathbf{w}_{1j}\|_2^2 + \|\mathbf{w}_{2j}\|_2^2 \tag{87}$$

We now apply Lemmas A.1 and A.2 to obtain

$$\begin{aligned} p^*_{FN} = \max_{\mathbf{V}_i} & -\sum_{i=1}^{n}\mathcal{L}^*(\mathbf{V}_i,\mathbf{Y}_i) \\ \text{s.t.} \max_{\|\mathbf{w}_1\|_2\le 1} & \|\sum_{i=1}^{n}\mathbf{V}_i^\top\mathrm{circ}(\mathbf{X}_i)\mathbf{w}_1\|_2 \le \beta \end{aligned} \tag{88}$$

We note that the norm constraint here has dimension c and $\mathbf{w}_1$ has dimension sd, so by (Shapiro, 2009; Pilanci & Ergen, 2020) this strong duality result from Lemma A.2 requires that $m^* \le \min\{sd, c\}$. This is equivalent to

$$\begin{aligned} p^*_{FN} = \max_{\mathbf{V}_i} & -\sum_{i=1}^{n}\mathcal{L}^*(\mathbf{V}_i,\mathbf{Y}_i) \\ \text{s.t. } & \|\sum_{i=1}^{n}\mathbf{V}_i^\top\mathrm{circ}(\mathbf{X}_i)\|_2 \le \beta \end{aligned} \tag{89}$$

We form the Lagrangian as

$$p^*_{FN} = \max_{\mathbf{V}_i}\min_{\lambda\ge 0}\min_{\|\mathbf{Z}\|_*\le 1} -\sum_{i=1}^{n}\mathcal{L}^*(\mathbf{V}_i,\mathbf{Y}_i) + \lambda(\beta - \mathrm{trace}(\mathbf{Z}^\top\sum_{i=1}^{n}\mathrm{circ}(\mathbf{X}_i)^\top\mathbf{V}_i)). \tag{90}$$

We switch the order of the maximum and minimum using Sion's minimax theorem and maximize over $\mathbf{V}_i$

$$p^*_{FN} = \min_{\lambda \geq 0} \min_{\|\mathbf{Z}\|_* \leq 1} \sum_{i=1}^{n} \mathcal{L}(\text{circ}(\mathbf{X}_i)\mathbf{Z}, \mathbf{Y}_i) + \beta\lambda. \tag{91}$$

Lastly, we rescale $\tilde{\mathbf{Z}} = \lambda \mathbf{Z}$ to obtain

$$p^*_{FN} = \min_{\mathbf{Z} \in \mathbb{R}^{sd \times c}} \sum_{i=1}^{n} \mathcal{L}\left(\text{circ}(\mathbf{X}_i)\mathbf{Z}, \mathbf{Y}_i\right) + \beta\|\mathbf{Z}\|_*. \tag{92}$$

as desired. □

A.9. Proof of Theorem 4.5

We now apply Lemmas A.1 and A.2 with the ReLU activation function to obtain

$$\begin{aligned} p^*_{FN} = &\max_{\mathbf{V}_i} - \sum_{i=1}^{n} \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i) \\ \text{s.t.} &\max_{\|\mathbf{w}_1\|_2 \leq 1} \|\sum_{i=1}^{n} \mathbf{V}_i^\top (\text{circ}(\mathbf{X}_i)\mathbf{w}_1)_+\|_2 \leq \beta \end{aligned} \tag{93}$$

We note that the norm constraint here has dimension c and $\mathbf{w}_1$ has dimension sd, so by (Shapiro, 2009; Pilanci & Ergen, 2020) this strong duality result from Lemma A.2 requires that $m^* \leq n\min\{sd, c\}$. We introduce hyperplane arrangements $\mathbf{D}_j$ and enumerate over all of them, yielding

$$\begin{aligned} p^*_{FN} = &\max_{\mathbf{V}_i} - \sum_{i=1}^{n} \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i) \\ \text{s.t.} &\max_{\substack{\|\mathbf{w}_1\|_2 \leq 1 \\ j \in [P] \\ \mathbf{K}_j\mathbf{w}_1 \geq 0}} \|\sum_{i=1}^{n} \mathbf{V}_i^\top \mathbf{D}_j^{(i)} \text{circ}(\mathbf{X}_i)\mathbf{w}_1\|_2 \leq \beta. \end{aligned} \tag{94}$$

Using the concept of dual norm, this is equivalent to

$$\begin{aligned} p^*_{FN} = &\max_{\mathbf{V}_i} - \sum_{i=1}^{n} \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i) \\ \text{s.t.} &\max_{\substack{\|\mathbf{g}\|_2 \leq 1 \\ \|\mathbf{w}_1\|_2 \leq 1 \\ j \in [P] \\ \mathbf{K}_j\mathbf{w}_1 \geq 0}} \mathbf{g}^\top \sum_{i=1}^{n} \mathbf{V}_i^\top \mathbf{D}_j^{(i)} \text{circ}(\mathbf{X}_i)\mathbf{w}_1 \leq \beta \end{aligned} \tag{95}$$

We can also define sets $\mathcal{C}_j := \{\mathbf{Z} = \mathbf{u}\mathbf{g}^\top \in \mathbb{R}^{s^2 \times dc} : \mathbf{K}_j\mathbf{u} \geq 0,\ \|\mathbf{Z}\|_* \leq 1\}$. Then, we have

$$\begin{aligned} p^*_{FN} = &\max_{\mathbf{V}_i} - \sum_{i=1}^{n} \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i) \\ \text{s.t.} &\max_{\substack{j \in [P] \\ \mathbf{Z} \in \mathcal{C}_j}} \text{trace}\left(\sum_{i=1}^{n} \mathbf{V}_i^\top \mathbf{D}_j^{(i)} \text{circ}(\mathbf{X}_i)\mathbf{Z}\right) \leq \beta \end{aligned} \tag{96}$$

We form the Lagrangian as

$$p^*_{FN} = \max_{\mathbf{V}_i} \min_{\lambda \geq 0} \min_{\mathbf{Z}_j \in \mathcal{C}_j} - \sum_{i=1}^{n} \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i) + \sum_{j=1}^{P} \lambda_j (\beta - \text{trace}(\mathbf{Z}_j^\top \sum_{i=1}^{n} \mathbf{D}_j^{(i)} \text{circ}(\mathbf{X}_i)^\top \mathbf{V}_i)). \tag{97}$$

We switch the order of the maximum and minimum using Sion's minimax theorem and maximize over $\mathbf{V}_i$

$$p^*_{FN} = \min_{\lambda_j \geq 0} \min_{\mathbf{Z}_j \in \mathcal{C}_j} \sum_{i=1}^{n} \mathcal{L}(\sum_{j=1}^{P} \mathbf{D}_j^{(i)} \mathrm{circ}(\mathbf{X}_i)\mathbf{Z}_j, \mathbf{Y}_i) + \beta \sum_{j=1}^{P} \lambda_j. \tag{98}$$

Lastly, we rescale $\tilde{\mathbf{Z}}_j = \lambda_j \mathbf{Z}_j$ to obtain

$$p^*_{FN} = \min_{\mathbf{Z}_j \in \mathbb{R}^{sd \times c}} \sum_{i=1}^{n} \mathcal{L}\left(\sum_{j=1}^{P} \mathbf{D}_j^{(i)} \mathrm{circ}(\mathbf{X}_i)\mathbf{Z}_j, \mathbf{Y}_i\right) + \beta \sum_{j=1}^{P} \|\mathbf{Z}_j\|_{*,\mathrm{K}_j}. \tag{99}$$

as desired. □

A.10. Proof of Lemma 4.6

The expression (28) can equivalently be written as

$$f_{BFN}(\mathbf{X}_i) = \sigma\left(\mathbf{F}^{-1} \begin{bmatrix} \mathbf{F}_1^\top \mathbf{X}_i & & \\ & \ddots & \\ & & \mathbf{F}_s^\top \mathbf{X}_i \end{bmatrix} \begin{bmatrix} \mathbf{V}^{(1)} \\ \cdots \\ \mathbf{V}^{(s)} \end{bmatrix}\right) \mathbf{W}_2 \tag{100}$$

and further as

$$f_{BFN}(\mathbf{X}_i) = \sigma\left(\mathbf{F}^{-1} \begin{bmatrix} \mathbf{F}_1^\top \mathbf{X}_i & & \\ & \ddots & \\ & & \mathbf{F}_s^\top \mathbf{X}_i \end{bmatrix} (\mathbf{F} \otimes \mathbf{I}_d) \begin{bmatrix} \mathbf{L}^{(1)} \\ \cdots \\ \mathbf{L}^{(s)} \end{bmatrix}\right) \mathbf{W}_2, \tag{101}$$

which simplifies to

$$f_{BFN}(\mathbf{X}_i) = \sigma\left(\begin{bmatrix} \mathbf{X}_i & \mathbf{X}_{i(1)} \cdots & \mathbf{X}_{i(s)} \end{bmatrix} \mathbf{L}\right) \mathbf{W}_2, \tag{102}$$

where $\mathbf{X}_{i(u)}$ is $\mathbf{X}_i$ circularly shifted by u spots in its first dimension, and $\mathbf{L}$ has been reshaped to the form $\mathbb{R}^{sd \times d}$. Without loss of generality, noting the block diagonal structure over blocks of $\mathbf{L}$, we permute the rows and corresponding columns of $\mathbf{L}$ so that $\mathbf{L}$ is a block diagonal matrix

$$f_{BFN}(\mathbf{X}_i) = \sigma\left(\left[\mathrm{circ}(\mathbf{X}_i^{(1)}) \cdots \quad \mathrm{circ}(\mathbf{X}_i^{(B)})\right] \mathbf{L}\right) \mathbf{W}_2. \tag{103}$$

Simplifying the term inside, we have

$$f_{BFN}(\mathbf{X}_i) = \sigma\left(\left[\mathrm{circ}(\mathbf{X}_i^{(1)})\mathbf{L}^{(1)} \cdots \quad \mathrm{circ}(\mathbf{X}_i^{(B)})\mathbf{L}^{(B)}\right]\right) \mathbf{W}_2. \tag{104}$$

Now, assuming σ is applied elementwise, we have

$$f_{BFN}(\mathbf{X}_i) = \left[\sigma(\mathrm{circ}(\mathbf{X}_i^{(1)})\mathbf{L}^{(1)}) \cdots \quad \sigma(\mathrm{circ}(\mathbf{X}_i^{(B)})\mathbf{L}^{(B)})\right] \mathbf{W}_2. \tag{105}$$

Lastly, we use the block structure on $\mathbf{W}_2$ to obtain

$$f_{BFN}(\mathbf{X}_i) = \left[\sigma(\mathrm{circ}(\mathbf{X}_i^{(1)})\mathbf{L}^{(1)})\mathbf{W}_2^{(1)} \cdots \quad \sigma(\mathrm{circ}(\mathbf{X}_i^{(B)})\mathbf{L}^{(B)})\mathbf{W}_2^{(B)}\right] \tag{106}$$

which we can write equivalently as

$$\begin{aligned} f_{BFN}(\mathbf{X}_i) &= \left[f_{BFN}^{(1)}(\mathbf{X}_i) \quad \cdots \quad f_{BFN}^{(B)}(\mathbf{X}_i)\right] \\ f_{BFN}^{(b)}(\mathbf{X}_i) &= \sum_{j=1}^{m} \sigma\left(\mathrm{circ}(\mathbf{X}_i^{(b)})\mathbf{w}_{1bj}\right) \mathbf{w}_{2bj}^\top \end{aligned} \tag{107}$$

as desired. □

A.11. Proof of Theorem 4.7

We now apply Lemmas A.1 and A.2 with ReLU activation obtain

$$\begin{aligned} p^*_{BFN} &= \max_{\mathbf{V}_i} -\sum_{i=1}^n \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i) \\ \text{s.t.} & \max_{\|\mathbf{w}_{1b}\|_2 \leq 1} \|\sum_{i=1}^n {\mathbf{V}_i^{(b)}}^\top \text{circ}(\mathbf{X}_i^{(b)})\mathbf{w}_{1b}\|_2 \leq \beta \; \forall b \in [B] \end{aligned} \tag{108}$$

We note that the norm constraint here has dimension c/B and $\mathbf{w}_{1b}$ has dimension sd/B, so by (Shapiro, 2009; Pilanci & Ergen, 2020) this strong duality result from Lemma A.2 requires that $m^* \leq 1/B \min\{sd, c\}$. This is equivalent to

$$\begin{aligned} p^*_{BFN} &= \max_{\mathbf{V}_i} -\sum_{i=1}^n \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i) \\ \text{s.t.} & \|\sum_{i=1}^n {\mathbf{V}_i^{(b)}}^\top \text{circ}(\mathbf{X}_i^{(b)})\|_2 \leq \beta \; \forall b \in [B] \end{aligned} \tag{109}$$

We form the Lagrangian as

$$p^*_{BFN} = \max_{\mathbf{V}_i} \min_{\lambda \geq 0} \min_{\|\mathbf{Z}_b\|_* \leq 1} -\sum_{i=1}^n \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i) + \sum_{b=1}^B \lambda_b(\beta - \text{trace}(\mathbf{Z}_b^\top \sum_{i=1}^n \text{circ}(\mathbf{X}_i^{(b)})^\top \mathbf{V}_i^{(b)})). \tag{110}$$

We switch the order of the maximum and minimum using Sion's minimax theorem and maximize over $\mathbf{V}_i^{(b)}$

$$p^*_{BFN} = \min_{\lambda \geq 0} \min_{\|\mathbf{Z}_b\|_* \leq 1} \sum_{i=1}^n \mathcal{L}(\begin{bmatrix} \lambda_1 \text{circ}(\mathbf{X}_i^{(1)})\mathbf{Z}_1 & \cdots & \lambda_B \text{circ}(\mathbf{X}_i^{(B)})\mathbf{Z}_B \end{bmatrix}, \mathbf{Y}_i) + \beta \sum_{j=1}^B \lambda_b. \tag{111}$$

Lastly, we rescale $\tilde{\mathbf{Z}}_b = \lambda_b \mathbf{Z}_b$ to obtain

$$p^*_{BFN} = \min_{\mathbf{Z} \in \mathbb{R}^{sd \times c}} \sum_{i=1}^n \mathcal{L}\left(\begin{bmatrix} \text{circ}(\mathbf{X}_i^{(1)})\mathbf{Z}_1 & \cdots & \text{circ}(\mathbf{X}_i^{(B)})\mathbf{Z}_B \end{bmatrix}, \mathbf{Y}_i\right) + \beta\|\mathbf{Z}\|_*. \tag{112}$$

as desired. □

A.12. Proof of Theorem 4.8

We now apply Lemmas A.1 and A.2 with the ReLU activation function to obtain

$$\begin{aligned} p^*_{BFN} &= \max_{\mathbf{V}_i} -\sum_{i=1}^n \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i) \\ \text{s.t.} & \max_{\|\mathbf{w}_{1b}\|_2 \leq 1} \|\sum_{i=1}^n {\mathbf{V}_i^{(b)}}^\top (\text{circ}(\mathbf{X}_i^{(b)})\mathbf{w}_{1b})_+\|_2 \leq \beta \; \forall b \in [B] \end{aligned} \tag{113}$$

We note that the norm constraint here has dimension c/B and $\mathbf{w}_{1b}$ has dimension sd/B, so by (Shapiro, 2009; Pilanci & Ergen, 2020) this strong duality result from Lemma A.2 requires that $m^* \leq n/B \min\{sd, c\}$. We introduce hyperplane arrangements $\mathbf{D}_{b,j}$ and enumerate over all of them, yielding

$$\begin{aligned} p^*_{BFN} &= \max_{\mathbf{V}_i} -\sum_{i=1}^n \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i) \\ \text{s.t.} & \max_{\substack{\|\mathbf{w}_{1b}\|_2 \leq 1 \\ j \in [P_b] \\ \mathbf{K}_{b,j}\mathbf{w}_{1b} \geq 0}} \|\sum_{i=1}^n {\mathbf{V}_i^{(b)}}^\top \mathbf{D}_{b,j}^{(i)} \text{circ}(\mathbf{X}_i^{(b)})\mathbf{w}_{1b}\|_2 \leq \beta \; \forall b \in [B]. \end{aligned} \tag{114}$$

Using the concept of dual norm, this is equivalent to

$$\begin{aligned} p^*_{BFN} = \max_{\mathbf{V}_i} & - \sum_{i=1}^n \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i) \\ \text{s.t.} & \max_{\substack{\|\mathbf{g}_b\|_2 \leq 1 \\ \|\mathbf{w}_{1b}\|_2 \leq 1 \\ j \in [P_b] \\ \mathbf{K}_{b,j}\mathbf{w}_{1b} \geq 0}} \mathbf{g}^\top \sum_{i=1}^n {\mathbf{V}_i^{(b)}}^\top \mathbf{D}_{b,j}^{(i)} \text{circ}(\mathbf{X}_i^{(b)}) \mathbf{w}_{1b} \leq \beta \; \forall b \in [B]. \end{aligned} \tag{115}$$

We can also define sets $\mathcal{C}_{b,j} := \{\mathbf{Z} = \mathbf{u}\mathbf{g}^\top \in \mathbb{R}^{s^2 \times dc} : \mathbf{K}_{b,j}\mathbf{u} \geq 0, \; \|\mathbf{Z}\|_* \leq 1\}$. Then, we have

$$\begin{aligned} p^*_{BFN} = \max_{\mathbf{V}_i} & - \sum_{i=1}^n \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i) \\ \text{s.t.} & \max_{\substack{j \in [P_b] \\ \mathbf{Z} \in \mathcal{C}_{b,j}}} \text{trace} \left(\sum_{i=1}^n {\mathbf{V}_i^{(b)}}^\top \mathbf{D}_{b,j}^{(i)} \text{circ}(\mathbf{X}_i^{(b)}) \mathbf{Z} \right) \leq \beta \; \forall b \in [B]. \end{aligned} \tag{116}$$

We form the Lagrangian as

$$p^*_{BFN} = \max_{\mathbf{V}_i} \min_{\lambda \geq 0} \min_{\mathbf{Z}_{b,j} \in \mathcal{C}_{b,j}} - \sum_{i=1}^n \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i) + \sum_{b=1}^B \sum_{j=1}^{P_b} \lambda_{b,j} (\beta - \text{trace}(\mathbf{Z}_{b,j}^\top \sum_{i=1}^n \mathbf{D}_{b,j}^{(i)} \text{circ}(\mathbf{X}_i^{(b)})^\top \mathbf{V}_i^{(b)})). \tag{117}$$

We switch the order of the maximum and minimum using Sion's minimax theorem and maximize over $\mathbf{V}_i$

$$p^*_{BFN} = \min_{\lambda_j \geq 0} \min_{\mathbf{Z}_{b,j} \in \mathcal{C}_{b,j}} \sum_{i=1}^n \mathcal{L}(\begin{bmatrix} \sum_{j=1}^{P_1} \lambda_{1,j} \mathbf{D}_{1,j} \text{circ}(\mathbf{X}_i^{(1)}) \mathbf{Z}_{1,j} & \cdots & \sum_{j=1}^{P_B} \lambda_{B,j} \mathbf{D}_{B,j} \text{circ}(\mathbf{X}_i^{(B)}) \mathbf{Z}_{B,j} \end{bmatrix}, \mathbf{Y}_i) + \beta \sum_{b=1}^B \sum_{j=1}^{P_b} \lambda_{b,j}. \tag{118}$$

Lastly, we rescale $\tilde{\mathbf{Z}}_{b,j} = \lambda_{b,j} \mathbf{Z}_{b,j}$ to obtain

$$p^*_{BFN} = \min_{\mathbf{Z}_{b,j}} \sum_{i=1}^n \mathcal{L} \left(\begin{bmatrix} \sum_{j=1}^{P_1} \mathbf{D}_{1,j} \text{circ}(\mathbf{X}_i^{(1)}) \mathbf{Z}_{1,j} & \cdots & \sum_{j=1}^{P_B} \mathbf{D}_{B,j} \text{circ}(\mathbf{X}_i^{(B)}) \mathbf{Z}_{B,j} \end{bmatrix}, \mathbf{Y}_i \right) + \beta \sum_{b=1}^B \sum_{j=1}^{P_b} \|\mathbf{Z}_{j,b}\|_{*, \mathrm{K}_{b,j}}, \tag{119}$$

as desired. □

B. Experimental Details

All heads were trained on two NVIDIA 1080 Ti GPUs using the Pytorch deep learning library (Paszke et al., 2019). For our backbone, we used pre-trained weights from the Pytorch Image Models library (Wightman, 2019). For all experiments, we trained each head for 70 epochs, and used a regularization parameter of $\beta = 2 \times 10^{-2}$, the AdamW optimizer (Loshchilov & Hutter, 2017), and a cosine learning rate schedule with a warmup of three epochs with warmup learning rate of 2×10^{-7}, an initial learning rate chosen based on training accuracy of either 5×10^{-3} or 10^{-4}, and a final learning rate of 2×10^{-2} times the initial learning rate. Data augmentation was performed using AutoAugment (Cubuk et al., 2018), along with color jittering, label smoothing, and training data interpolation. All heads aside from the self-attention head were trained using a batch size of 100, whereas the self-attention head was trained with a batch size of 20. For all ReLU heads, we choose a number of neurons such that the number of parameters across FNO, MLP-Mixer, self-attention, and MLPs are roughly equal. We provide information about the number of parameters in each head in Table 2.

As an ablation, we also studied the effect of changing the backbone architecture on the results of this experiment. In particular, as a backbone, we also tried using a ViT-base model (Dosovitskiy et al., 2020) with 16 × 16 patches pre-trained on ImageNet-1k images of size 224 × 224 ($s = 196$, $d = 768$). Then, we followed the same average pooling approach as

Table 2. Parameter count for convex heads used for the experiments in Table 1 .

CONVEX HEAD	ACT.	PARAMS
SELF-ATTENTION	LINEAR	50.5M
MLP-MIXER		98.9M
B-FNO		392K
FNO		1.96M
MLP		1.96M
LINEAR		10K
SELF-ATTENTION	RELU	253M
MLP-MIXER		196M
B-FNO		39.2M
FNO		196M
MLP		196M

for the gMLP backbone experiments, and kept all other network parameters the same. The results of this experiment are summarized in Table 3.

We see from this table that some of the general results from Table 1 still hold, though in this case MLP-Mixer architectures and self-attention architectures are roughly equivalent in performance. We suspect that one major reason for the larger gap between the two methods in Table 1 could be due to the backbone architecture, since the gMLP architecture is MLP-based, as opposed to ViT which is self-attention based. Thus, we speculate that adding an additional MLP-Mixer head to gMLP may be more concordant with the features extracted from the gMLP backbone, whereas the inverse is true for the ViT backbone.

In order to avoid the heavy computational cost of nuclear norm minimization for all considered convex models (besides "linear", which is just a logistic regression with standard weight decay), we rely on the Burer-Monteiro factorization (Burer & Monteiro, 2005). In particular, if one has the problem

$$\min_{\mathbf{Z}\in\mathbb{R}^{a\times c}} \sum_{i=1}^{n} \mathcal{L}(f_i(\mathbf{Z}), \mathbf{Y}_i) + \beta\|\mathbf{Z}\|_*, \tag{120}$$

we know that this is equivalent to

$$\min_{\mathbf{U}\in\mathbb{R}^{a\times b}, \mathbf{V}\in\mathbb{R}^{c\times b}} \sum_{i=1}^{n} \mathcal{L}(f_i(\mathbf{U}\mathbf{V}^\top), \mathbf{Y}_i) + \frac{\beta}{2}\left(\|\mathbf{U}\|_F^2 + \|\mathbf{V}\|_F^2\right), \tag{121}$$

granted that the solution $\mathbf{Z}^*$ to (120) has a rank less than the new latent dimension b (Recht et al., 2010). However, while (120) is convex, (121) is not. One can also show that if the solution $\mathbf{Z}^*$ to (120) has a rank less than the new latent dimension b, the problem (121) has no spurious local minima (Burer & Monteiro, 2005). We can also show that any stationary point $\hat{\mathbf{Z}} = \hat{\mathbf{U}}\hat{\mathbf{V}}^\top$ of (121), achieved e.g., with gradient descent, is a global optimum for (120) if it satisfies the following qualification condition

$$\|\sum_{i=1}^{n} \nabla_{\mathbf{Z}}\mathcal{L}(f_i(\hat{\mathbf{Z}}), \mathbf{Y}_i)\|_2 \leq \beta, \tag{122}$$

see (Mardani et al., 2013; 2015) for the proof and more details. We thus employ the Burer-Monteiro factorization for all problems, except linear. For the transformer architectures, we choose the latent dimension such that the total model size is unchanged, whereas for the MLP architecture, we choose the latent dimension to have parameters on the same order as the other transformer architectures (e.g. $b = sc/2$).

C. Additional Theoretical Results

C.1. Visualizing the Constrained Nuclear Norm

Here, we seek to provide some additional intuition around the constrained nuclear norm $\|\mathbf{Z}\|_{*,\mathrm{K}}$ for a simple case, to contrast the regularization for convex ReLU against convex linear and gated ReLU networks. We provide a visualization for

Table 3. CIFAR-100 classification accuracy for training a single *convex* head. Embeddings are generated from gMLP-S pre-trained on ImageNet. Note that the backbone is not fine-tuned.

CONVEX HEAD	ACT.	TOP-1	TOP-5
SELF-ATTENTION	LINEAR	**67.24**	88.63
MLP-MIXER		66.55	**88.70**
B-FNO		36.04	64.28
FNO		63.88	87.20
MLP		56.68	81.79
LINEAR		57.00	81.96
SELF-ATTENTION	RELU	**68.16**	88.74
MLP-MIXER		67.84	**89.24**
B-FNO		66.66	87.93
FNO		67.87	88.97
MLP		64.14	86.57

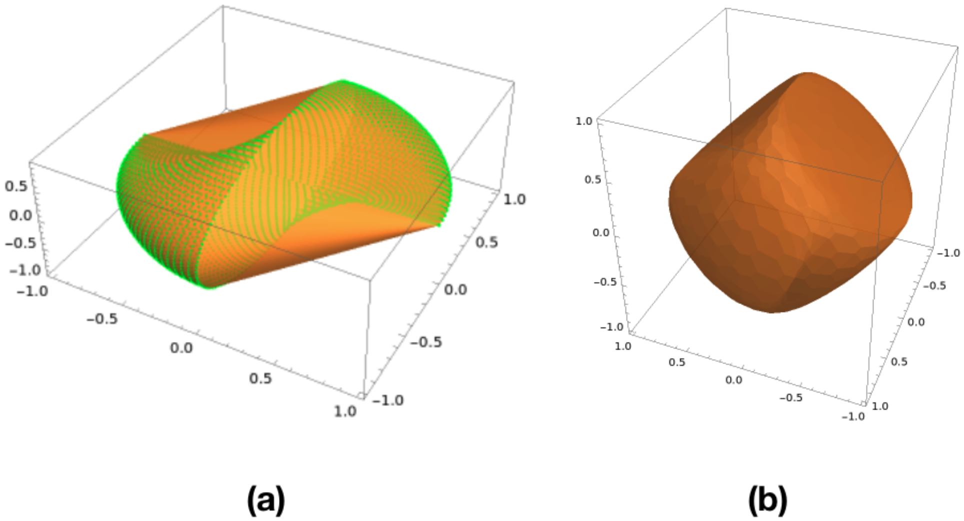

Figure 2. Visualization of the constrained nuclear norm $\|\mathbf{Z}\|_{*,\mathrm{K}}$ defined in Eq. (4). Here, we visualize $\mathbf{Z} = \begin{bmatrix} z_1 & z_2 \\ z_3 & z_4 \end{bmatrix} \in \mathbb{R}^{2\times 2}$ in terms of three of its coordinates, z_1, z_2, and z_4, with the final coordinate z_3 fixed as 0. **(a)** Constrained nuclear norm space arises in the convex formulation of ReLU networks (see (4)). Here, the dotted green region illustrates the set $\mathcal{C}' = \{\mathbf{Z} = \mathbf{u}\mathbf{v}^\top : \mathbf{u} \geq 0, \|\mathbf{Z}\|_* \leq 1, \mathbf{v} \in \mathbb{R}^c\}$, and the opaque orange region shows its convex hull $\operatorname{conv}(\mathcal{C}')$. **(b)** In the case of linear activation, the constrained nuclear norm reduces to the standard nuclear norm $\|\mathbf{Z}\|_*$. Notice that introducing ReLU nonlinearity breaks the symmetry present in **(b)** and yields a complicated non-convex space, i.e., green dots in **(a)**. However, our convex analytic approach relaxes this set as the convex hull, i.e., orange solid space in **(a)**, by keeping the extreme points, which are the points that play a crucial role in the optimal solution, intact. Therefore, we are able to convert standard non-convex ReLU network training problems into polynomial-time trainable convex optimization problems.

$\|\mathbf{Z}\|_{*,\mathrm{K}}$ compared to $\|\mathbf{Z}\|_*$ in the case that $\mathbf{K} = \mathbf{I}$. This would occur in a ReLU network when $\mathbf{X} = \mathbf{I}$ and we encounter a particular hyperplane arrangement $\mathbf{D}_j = \mathbf{I}$. One can also note that in ReLU MLPs where $n \leq d$ and the data $\mathbf{X}$ is whitened, the convex optimization objective will reduce to a linear model with this norm as its regularization (Sahiner et al., 2020a).

In particular, in the case that $n = d = c = 2$, we can visualize the coordinates of $\mathbf{Z} \in \mathbb{R}^{2\times 2}$ which satisfy $\|\mathbf{Z}\|_{*,\mathrm{K}} \leq 1$, contrasted with the coordinates of $\mathbf{Z}$ which satisfy $\|\mathbf{Z}\|_* \leq 1$. We do so in Figure 2, illustrating the complicated, yet still convex, regularization in the case of a ReLU neural network.

C.2. Gated ReLU activation extensions

C.2.1. SELF-ATTENTION

Theorem C.1. *For the Gated ReLU activation multi-head self-attention training problem* (12)*, we define*

$$\mathbf{X} := \begin{bmatrix} \mathbf{X}_1 \otimes \mathbf{X}_1 \\ \cdots \\ \mathbf{X}_n \otimes \mathbf{X}_n \end{bmatrix}$$
$$\{\mathbf{D}_j\}_{j=1}^m := \{\operatorname{diag}\left(\mathbb{1}\{\mathbf{X}\mathbf{h}_j \geq 0\}\right)\}_{j=1}^m,$$

for fixed gates $\{\mathbf{h}_j \in \mathbb{R}^{d^2}\}_{j=1}^m$*. Then, the standard non-convex training objective is equivalent to a convex optimization problem, given by*

$$p_{SA}^* = \min_{\mathbf{Z}_j \in \mathbb{R}^{d^2 \times dc}} \sum_{i=1}^n \mathcal{L}\left(\sum_{j=1}^m \sum_{k=1}^d \sum_{\ell=1}^d \mathbf{G}_{i,j}^{(k,\ell)} \mathbf{X}_i \mathbf{Z}_j^{(k,\ell)}, \mathbf{Y}_i\right) + \beta \sum_{j=1}^m \|\mathbf{Z}_j\|_*, \tag{123}$$

where

$$\mathbf{G}_{i,j} := (\mathbf{X}_i \otimes \mathbf{I}_s)^\top \mathbf{D}_j^{(i)} (\mathbf{X}_i \otimes \mathbf{I}_s),$$

for $\mathbf{G}_{i,j}^{(k,\ell)} \in \mathbb{R}^{s \times s}$ *and* $\mathbf{Z}_j^{(k,\ell)} \in \mathbb{R}^{d \times c}$*.*

We note here that instead of the constrained nuclear norm penalty, we have a standard nuclear norm penalty on $\mathbf{Z}$.

Proof. We apply Lemmas A.1 and A.2 to (12) with the gated ReLU activation function to obtain

$$\begin{aligned} p_{SA}^* = \max_{\mathbf{V}_i} & - \sum_{i=1}^n \mathcal{L}^*\left(\mathbf{V}_i, \mathbf{Y}_i\right) \\ \text{s.t.} & \max_{\|\mathbf{W}_{1j}\|_F \leq 1} \left\| \sum_{i=1}^N \mathbf{V}_i^\top \mathbf{D}_j^{(i)} \mathbf{X}_i \mathbf{W}_{1j} \mathbf{X}_i^\top)_+ \mathbf{X}_i \right\|_F \quad \forall j \in [m]. \end{aligned} \tag{124}$$

We again apply $\mathbf{vec}(\mathbf{ABC}) = (\mathbf{C}^\top \otimes \mathbf{A})\mathbf{vec}(\mathbf{B})$ (Magnus & Neudecker, 2019) to obtain

$$\begin{aligned} p_{SA}^* = \max_{\mathbf{V}_i} & - \sum_{i=1}^n \mathcal{L}^*\left(\mathbf{V}_i, \mathbf{Y}_i\right) \\ \text{s.t.} & \max_{\|\mathbf{w}_{1j}\|_2 \leq 1} \left\| \sum_{i=1}^N (\mathbf{X}_i \otimes \mathbf{V}_i^\top) \mathbf{D}_j^{(i)} (\mathbf{X}_i \otimes \mathbf{X}_i) \mathbf{w}_1 \right\|_2 \leq \beta \; \forall j \in [m]. \end{aligned} \tag{125}$$

Now, using the concept of dual norm, this is equal to

$$\begin{aligned} p_{SA}^* = \max_{\mathbf{V}_i} & - \sum_{i=1}^n \mathcal{L}^*\left(\mathbf{V}_i, \mathbf{Y}_i\right) \\ \text{s.t.} & \max_{\substack{\|\mathbf{g}\|_2 \leq 1 \\ \|\mathbf{w}_{1j}\|_2 \leq 1 \\ j \in [m]}} \mathbf{g}^\top \sum_{i=1}^N (\mathbf{X}_i^\top \otimes \mathbf{V}_i^\top) \mathbf{D}_j^{(i)} (\mathbf{X}_i \otimes \mathbf{X}_i) \mathbf{w}_{1j} \leq \beta \end{aligned} \tag{126}$$

Then, we have

$$\begin{aligned} p_{SA}^* = \max_{\mathbf{V}_i} & - \sum_{i=1}^n \mathcal{L}^*\left(\mathbf{V}_i, \mathbf{Y}_i\right) \\ \text{s.t.} & \max_{\substack{j \in [m] \\ \|\mathbf{Z}\|_*}} \operatorname{trace}\left(\sum_{i=1}^n (\mathbf{X}_i^\top \otimes \mathbf{V}_i^\top) \mathbf{D}_j^{(i)} (\mathbf{X}_i \otimes \mathbf{X}_i) \mathbf{Z}\right) \leq \beta \end{aligned} \tag{127}$$

Now, we simply need to form the Lagrangian and solve. The Lagrangian is given by

$$p^*_{SA} = \max_{\mathbf{V}_i} \min_{\lambda \geq 0} \min_{\|\mathbf{Z}_j\|_* \in \mathcal{C}_j} -\sum_{i=1}^n \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i) + \sum_{j=1}^m \lambda_j \left(\beta - \sum_{i=1}^n \mathbf{vec}\left(\mathbf{D}_j^{(i)}(\mathbf{X}_i \otimes \mathbf{X}_i)\mathbf{Z}_j\right)^\top \mathbf{vec}\left(\mathbf{X}_i \otimes \mathbf{V}_i\right) \right) \tag{128}$$

We now can switch the order of max and min via Sion's minimax theorem and maximize over $\mathbf{V}_i$. Defining $\mathbf{K}_{c,s}$ as the (c, s) commutation matrix (Magnus & Neudecker, 2019):

$$\mathbf{vec}(\mathbf{X}_i \otimes \mathbf{V}_i) = ((\mathbf{I}_d \otimes \mathbf{K}_{c,s})(\mathbf{vec}(\mathbf{X}_i) \otimes \mathbf{I}_c) \otimes \mathbf{I}_s)\, \mathbf{vec}(\mathbf{V}_i)$$

Maximizing over $\mathbf{V}_i$, we have

$$p^*_{SA} = \min_{\lambda \geq 0} \min_{\|\mathbf{Z}_j\|_* \leq 1} \sum_{i=1}^n \mathcal{L}\left(\sum_{j=1}^m \left((\mathbf{vec}(\mathbf{X}_i)^\top \otimes \mathbf{I}_c)(\mathbf{I}_d \otimes \mathbf{K}_{s,c}) \otimes \mathbf{I}_s\right) \mathbf{vec}\left(\mathbf{D}_j^{(i)}(\mathbf{X}_i \otimes \mathbf{X}_i)\lambda_j \mathbf{Z}_j\right), \mathbf{vec}(\mathbf{Y}_i) \right) + \beta \sum_{j=1}^m \lambda_j. \tag{129}$$

Again rescaling $\tilde{\mathbf{Z}}_j = \lambda_j \mathbf{Z}_j$, we have

$$p^*_{SA} = \min_{\mathbf{Z}_j \in \mathbb{R}^{d^2 \times dc}} \sum_{i=1}^n \mathcal{L}\left(\sum_{j=1}^m \left((\mathbf{vec}(\mathbf{X}_i)^\top \otimes \mathbf{I}_c)(\mathbf{I}_d \otimes \mathbf{K}_{s,c}) \otimes \mathbf{I}_s\right) \mathbf{vec}\left(\mathbf{D}_j^{(i)}(\mathbf{X}_i \otimes \mathbf{X}_i)\mathbf{Z}_j\right), \mathbf{vec}(\mathbf{Y}_i) \right) + \beta \sum_{j=1}^m \|\mathbf{Z}_j\|_*. \tag{130}$$

It appears as though this is a very complicated function, but it actually simplifies greatly. In particular, one can write this as

$$\begin{aligned} p^*_{SA} &= \min_{\mathbf{Z}_j \in \mathbb{R}^{d^2 \times dc}} \sum_{i=1}^n \mathcal{L}\left(\hat{\mathbf{Y}}_i, \mathbf{Y}_i\right) + \beta \|\mathbf{Z}\|_* \\ \hat{\mathbf{Y}}_i[o, p] &:= \sum_{j=1}^m \sum_{k=1}^d \sum_{l=1}^d \sum_{t=1}^s \mathbf{X}_i[t, l]\mathbf{X}_i[t, k]\mathbf{D}_j^{(t,m)} \mathbf{X}_i[o, :]^\top \mathbf{Z}_j^{(k,l)}. \end{aligned} \tag{131}$$

Making any final simplifications, one obtains the desired result.

$$p^*_{SA} = \min_{\mathbf{Z}_j \in \mathbb{R}^{d^2 \times dc}} \sum_{i=1}^n \mathcal{L}\left(\sum_{j=1}^m \sum_{k=1}^d \sum_{\ell=1}^d \mathbf{G}_{i,j}^{(k,\ell)} \mathbf{X}_i \mathbf{Z}_j^{(k,\ell)}, \mathbf{Y}_i \right) + \beta \sum_{j=1}^m \|\mathbf{Z}_j\|_*. \tag{132}$$

□

C.2.2. MLP-MIXER

Theorem C.2. *For the Gated ReLU activation MLP-Mixer training problem* (17)*, we define*

$$\mathbf{X} := \begin{bmatrix} \mathbf{X}_1^\top \otimes \mathbf{I}_s \\ \cdots \\ \mathbf{X}_n^\top \otimes \mathbf{I}_s \end{bmatrix}$$

$$\{\mathbf{D}_j\}_{j=1}^m := \{\mathrm{diag}\left(\mathbb{1}\{\mathbf{X}\mathbf{h}_j \geq 0\}\right)\},$$

for fixed gates $\{\mathbf{h}_j \in \mathbb{R}^{s^2}\}_{j=1}^m$. Then, the standard non-convex training objective is equivalent to a convex optimization problem, given by

$$p^*_{MM} = \min_{\mathbf{Z}_j \in \mathbb{R}^{s^2 \times dc}} \sum_{i=1}^n \mathcal{L}\left(\begin{bmatrix} f_1(\mathbf{X}_i) & \cdots & f_c(\mathbf{X}_i) \end{bmatrix}, \mathbf{Y}_i\right) + \beta \sum_{j=1}^m \|\mathbf{Z}_j\|_* \tag{133}$$

where

$$f_p(\mathbf{X_i}) := \sum_{j=1}^m \left[\mathbf{D}_j^{(i,1)} \mathbf{Z}_j^{(p,1)} \cdots \mathbf{D}_j^{(i,d)} \mathbf{Z}_j^{(p,d)}\right] \mathbf{vec}(\mathbf{X}_i)$$

for $\mathbf{D}_j^{(i,k)} \in \mathbb{R}^{s \times s}$ and $\mathbf{Z}_j^{(p,k)} \in \mathbb{R}^{s \times s}$.

Proof. We apply Lemmas A.1 and A.2 to (17) with the Gated ReLU activation function to obtain

$$\begin{aligned} p_{MM}^{*} &= \max_{\mathbf{V}_i} - \sum_{i=1}^{n} \mathcal{L}^{*}\left(\mathbf{V}_i, \mathbf{Y}_i\right) \\ \text{s.t.} & \max_{\substack{\|\mathbf{W}_1\|_F \le 1 \\ j \in [m]}} \left\| \sum_{i=1}^{n} \mathbf{V}_i^T \sigma_j(\mathbf{W}_1 \mathbf{X}_i)_+ \right\|_F \le \beta \end{aligned} \tag{134}$$

This is equivalent to (Magnus & Neudecker, 2019)

$$\begin{aligned} p_{MM}^{*} &= \max_{\mathbf{V}_i} - \sum_{i=1}^{n} \mathcal{L}^{*}\left(\mathbf{V}_i, \mathbf{Y}_i\right) \\ \text{s.t.} & \max_{\substack{\|\mathbf{W}_1\|_F \le 1 \\ j \in [m]}} \left\| \sum_{i=1}^{n} (\mathbf{I}_d \otimes \mathbf{V}_i^T) \mathbf{D}_j^{(i)} (\mathbf{X}_i^\top \otimes \mathbf{I}_s) \mathbf{vec}(\mathbf{W}_1) \right\|_F \le \beta \end{aligned} \tag{135}$$

Now, using the concept of dual norm, this is equal to

$$\begin{aligned} p_{MM}^{*} &= \max_{\mathbf{V}_i} - \sum_{i=1}^{n} \mathcal{L}^{*}\left(\mathbf{V}_i, \mathbf{Y}_i\right) \\ \text{s.t.} & \max_{\substack{\|\mathbf{g}\|_2 \le 1 \\ \|\mathbf{w}_1\|_2 \le 1 \\ j \in [m]}} \mathbf{g}^\top \sum_{i=1}^{n} (\mathbf{I}_d \otimes \mathbf{V}_i^T) \mathbf{D}_j^{(i)} (\mathbf{X}_i^\top \otimes \mathbf{I}_s) \mathbf{w}_1 \le \beta \end{aligned} \tag{136}$$

Then, we have

$$\begin{aligned} p_{MM}^{*} &= \max_{\mathbf{V}_i} - \sum_{i=1}^{n} \mathcal{L}^{*}\left(\mathbf{V}_i, \mathbf{Y}_i\right) \\ \text{s.t.} & \max_{\substack{j \in [m] \\ \|\mathbf{Z}\|_* \le 1}} \operatorname{trace}\left(\sum_{i=1}^{n} (\mathbf{I}_d \otimes \mathbf{V}_i^T) \mathbf{D}_j^{(i)} (\mathbf{X}_i^\top \otimes \mathbf{I}_s) \mathbf{Z} \right) \le \beta \end{aligned} \tag{137}$$

Now, we simply need to form the Lagrangian and solve. The Lagrangian is given by

$$p_{MM}^{*} = \max_{\mathbf{V}_i} \min_{\lambda \ge 0} \min_{\|\mathbf{Z}_j\|_* \le 1} - \sum_{i=1}^{n} \mathcal{L}^{*}(\mathbf{V}_i, \mathbf{Y}_i) + \sum_{j=1}^{m} \lambda_j \left(\beta - \sum_{i=1}^{n} \operatorname{trace}\left((\mathbf{I}_d \otimes \mathbf{V}_i^T) \mathbf{D}_j^{(i)} (\mathbf{X}_i^\top \otimes \mathbf{I}_s) \mathbf{Z} \right) \right) \tag{138}$$

We now can switch the order of max and min via Sion's minimax theorem and maximize over $\mathbf{V}_i$:

$$p_{MM}^{*} = \min_{\lambda \ge 0} \min_{|\mathbf{Z}_j\|_* \le 1} \max_{\mathbf{V}_i} - \sum_{i=1}^{n} \mathcal{L}^{*}(\mathbf{V}_i, \mathbf{Y}_i) + \sum_{j=1}^{m} \lambda_j \left(\beta - \sum_{i=1}^{n} \mathbf{vec}\left(\mathbf{D}_j^{(i)} (\mathbf{X}_i^\top \otimes \mathbf{I}_s) \mathbf{Z}_j \right)^\top \mathbf{vec}\left(\mathbf{I}_d \otimes \mathbf{V}_i \right) \right) \tag{139}$$

Now, defining $\mathbf{K}_{c,d}$ as the (c, d) commutation matrix:

$$\mathbf{vec}(\mathbf{I}_d \otimes \mathbf{V}_i) = \left((\mathbf{I}_d \otimes \mathbf{K}_{c,d})(\mathbf{vec}(\mathbf{I}_d) \otimes \mathbf{I}_c) \otimes \mathbf{I}_s \right) \mathbf{vec}(\mathbf{V}_i)$$

Solving over $\mathbf{V}_i$ yields

$$p_{MM}^{*} = \min_{\lambda \ge 0} \min_{\|\mathbf{Z}_j\|_* \le 1} \sum_{i=1}^{n} \mathcal{L}\left(\sum_{j=1}^{m} \left((\mathbf{vec}(\mathbf{I}_d)^\top \otimes \mathbf{I}_c)(\mathbf{I}_d \otimes \mathbf{K}_{d,c}) \otimes \mathbf{I}_s \right) \mathbf{vec}\left(\mathbf{D}_j^{(i)} (\mathbf{X}_i^\top \otimes \mathbf{I}_s) \lambda_j \mathbf{Z}_j \right), \mathbf{vec}(\mathbf{Y}_i) \right) + \beta \sum_{j=1}^{m} \lambda_j \tag{140}$$

Re-scaling $\tilde{\mathbf{Z}}_j = \lambda_j \mathbf{Z}_j$ gives us

$$p^*_{MM} = \min_{\mathbf{Z}_j} \sum_{i=1}^n \mathcal{L}\left(\sum_{j=1}^m \left((\mathbf{vec}(\mathbf{I}_d)^\top \otimes \mathbf{I}_c)(\mathbf{I}_d \otimes \mathbf{K}_{d,c}) \otimes \mathbf{I}_s\right) \mathbf{vec}\left(\mathbf{D}_j^{(i)}(\mathbf{X}_i^\top \otimes \mathbf{I}_s)\mathbf{Z}_j\right), \mathbf{vec}(\mathbf{Y}_i)\right) + \beta \sum_{j=1}^m \|\mathbf{Z}_j\|_*. \tag{141}$$

One can actually greatly simplify this result, and can re-write this as

$$p^*_{MM} = \min_{\mathbf{Z}_j \in \mathbb{R}^{s^2 \times dc}} \sum_{i=1}^n \mathcal{L}\left(\begin{bmatrix} f_1(\mathbf{X}_i) & \cdots & f_c(\mathbf{X}_i)\end{bmatrix}, \mathbf{Y}_i\right) + \beta \sum_{j=1}^m \|\mathbf{Z}_j\|_* \tag{142}$$

$$f_p(\mathbf{X}_i) := \sum_{j=1}^m \left[\mathbf{D}_j^{(i,1)}\mathbf{Z}_j^{(p,1)} \cdots \mathbf{D}_j^{(i,d)}\mathbf{Z}_j^{(p,d)}\right] \mathbf{vec}(\mathbf{X}_i). \tag{143}$$

as desired. □

C.2.3. FNO

Theorem C.3. *For the Gated ReLU activation FNO training problem (24), we define*

$$\mathbf{X} := \begin{bmatrix} \mathrm{circ}(\mathbf{X}_1) \\ \cdots \\ \mathrm{circ}(\mathbf{X}_n) \end{bmatrix}$$
$$\{\mathbf{D}_j\}_{j=1}^m := \{\mathrm{diag}\left(\mathbb{1}\{\mathbf{X}\mathbf{h}_j \geq 0\}\right)\},$$

for fixed gates $\{\mathbf{h}_j \in \mathbb{R}^{sd}\}_{j=1}^m$. Then, the standard non-convex training objective is equivalent to a convex optimization problem, given by

$$p^*_{FN} = \min_{\mathbf{Z}_j \in \mathbb{R}^{sd \times c}} \sum_{i=1}^n \mathcal{L}\left(\sum_{j=1}^m \mathbf{D}_j^{(i)} \mathrm{circ}(\mathbf{X}_i)\mathbf{Z}_j, \mathbf{Y}_i\right) + \beta \sum_{j=1}^m \|\mathbf{Z}_j\|_*. \tag{144}$$

Proof. We now apply Lemmas A.1 and A.2 with the Gated ReLU activation function to obtain

$$\begin{aligned} p^*_{FN} = \max_{\mathbf{V}_i} & -\sum_{i=1}^n \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i) \\ \text{s.t. } & \max_{\substack{\|\mathbf{w}_1\|_2 \leq 1 \\ j \in [m]}} \|\sum_{i=1}^n \mathbf{V}_i^\top \mathbf{D}_j^{(i)} \mathrm{circ}(\mathbf{X}_i)\mathbf{w}_1\|_2 \leq \beta. \end{aligned} \tag{145}$$

Using the concept of dual norm, this is equivalent to

$$\begin{aligned} p^*_{FN} = \max_{\mathbf{V}_i} & -\sum_{i=1}^n \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i) \\ \text{s.t. } & \max_{\substack{\|\mathbf{g}\|_2 \leq 1 \\ \|\mathbf{w}_1\|_2 \leq 1 \\ j \in [m]}} \mathbf{g}^\top \sum_{i=1}^n \mathbf{V}_i^\top \mathbf{D}_j^{(i)} \mathrm{circ}(\mathbf{X}_i)\mathbf{w}_1 \leq \beta \end{aligned} \tag{146}$$

Then, we have

$$\begin{aligned} p^*_{FN} = \max_{\mathbf{V}_i} & -\sum_{i=1}^n \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i) \\ \text{s.t. } & \max_{\substack{j \in [m] \\ \|\mathbf{Z}\|_* \leq 1}} \mathrm{trace}\left(\sum_{i=1}^n \mathbf{V}_i^\top \mathbf{D}_j^{(i)} \mathrm{circ}(\mathbf{X}_i)\mathbf{Z}\right) \leq \beta \end{aligned} \tag{147}$$

We form the Lagrangian as

$$p^*_{FN} = \max_{\mathbf{V}_i} \min_{\lambda \geq 0} \min_{\|\mathbf{Z}_j\|_* \leq 1} -\sum_{i=1}^{n} \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i) + \sum_{j=1}^{m} \lambda_j(\beta - \mathrm{trace}(\mathbf{Z}_j^\top \sum_{i=1}^{n} \mathbf{D}_j^{(i)} \mathrm{circ}(\mathbf{X}_i)^\top \mathbf{V}_i)). \tag{148}$$

We switch the order of the maximum and minimum using Sion's minimax theorem and maximize over $\mathbf{V}_i$

$$p^*_{FN} = \min_{\lambda_j \geq 0} \min_{\|\mathbf{Z}_j\|_* \leq 1} \sum_{i=1}^{n} \mathcal{L}(\sum_{j=1}^{m} \mathbf{D}_j^{(i)} \mathrm{circ}(\mathbf{X}_i)\mathbf{Z}_j, \mathbf{Y}_i) + \beta \sum_{j=1}^{m} \lambda_j. \tag{149}$$

Lastly, we rescale $\tilde{\mathbf{Z}}_j = \lambda_j \mathbf{Z}_j$ to obtain

$$p^*_{FN} = \min_{\mathbf{Z}_j \in \mathbb{R}^{sd \times c}} \sum_{i=1}^{n} \mathcal{L}\left(\sum_{j=1}^{m} \mathbf{D}_j^{(i)} \mathrm{circ}(\mathbf{X}_i)\mathbf{Z}_j, \mathbf{Y}_i\right) + \beta \sum_{j=1}^{m} \|\mathbf{Z}_j\|_*. \tag{150}$$

as desired. □

C.2.4. B-FNO

Theorem C.4. *For the Gated ReLU activation B-FNO training problem* (29)*, we define*

$$\mathbf{X}_b := \begin{bmatrix} \mathrm{circ}(\mathbf{X}_1^{(b)}) \\ \cdots \\ \mathrm{circ}(\mathbf{X}_n^{(b)}) \end{bmatrix}$$
$$\{\mathbf{D}_{b,j}\}_{j=1}^m := \{\mathrm{diag}\left(\mathbb{1}\{\mathbf{X}_b \mathbf{h}_{b,j} \geq 0\}\right)\},$$

for fixed gates $\{\mathbf{h}_{b,j} \in \mathbb{R}^{sd/B}\}_{j=1}^m$. Then, the standard non-convex training objective is equivalent to a convex optimization problem, given by

$$p^*_{BFN} = \min_{\mathbf{Z}_{b,j}} \sum_{i=1}^{n} \mathcal{L}\left(\begin{bmatrix} f^{(1)}(\mathbf{X}_i) & \cdots & f^{(B)}(\mathbf{X}_i) \end{bmatrix}, \mathbf{Y}_i\right) + \beta \sum_{b=1}^{B} \sum_{j=1}^{m} \|\mathbf{Z}_{j,b}\|_*, \tag{151}$$

where

$$f^{(b)} := \sum_{j=1}^{m} \mathbf{D}_{b,j} \mathrm{circ}(\mathbf{X}_i^{(b)})\mathbf{Z}_{b,j}$$

Proof. We now apply Lemmas A.1 and A.2 with the Gated ReLU activation function to obtain

$$\begin{aligned} p^*_{BFN} = \max_{\mathbf{V}_i} & -\sum_{i=1}^{n} \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i) \\ \text{s.t.} & \max_{\substack{\|\mathbf{w}_{1b}\|_2 \leq 1 \\ b \in [B] \\ j \in [m]}} \|\sum_{i=1}^{n} {\mathbf{V}_i^{(b)}}^\top \mathbf{D}_{b,j}^{(i)} \mathrm{circ}(\mathbf{X}_i^{(b)})\mathbf{w}_{1b}\|_2 \leq \beta \end{aligned} \tag{152}$$

Using the concept of dual norm, this is equivalent to

$$\begin{aligned} p^*_{BFN} = \max_{\mathbf{V}_i} & -\sum_{i=1}^{n} \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i) \\ \text{s.t.} & \max_{\substack{\|\mathbf{g}_b\|_2 \leq 1 \\ \|\mathbf{w}_{1b}\|_2 \leq 1 \\ j \in [m] \\ \mathbf{K}_{b,j}\mathbf{w}_{1b} \geq 0}} \mathbf{g}^\top \sum_{i=1}^{n} {\mathbf{V}_i^{(b)}}^\top \mathbf{D}_{b,j}^{(i)} \mathrm{circ}(\mathbf{X}_i^{(b)})\mathbf{w}_{1b} \leq \beta \; \forall b \in [B]. \end{aligned} \tag{153}$$

Then, we have

$$
\begin{aligned}
p^*_{BFN} = \max_{\mathbf{V}_i} & -\sum_{i=1}^n \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i) \\
\text{s.t.} & \max_{\substack{j\in[m] \\ \|\mathbf{Z}\|_*\leq 1}} \operatorname{trace}\left(\sum_{i=1}^n {\mathbf{V}_i^{(b)}}^\top \mathbf{D}_{b,j}^{(i)} \operatorname{circ}(\mathbf{X}_i^{(b)})\mathbf{Z}\right) \leq \beta \; \forall b \in [B].
\end{aligned} \tag{154}
$$

We form the Lagrangian as

$$
p^*_{BFN} = \max_{\mathbf{V}_i} \min_{\lambda\geq 0} \min_{\|\mathbf{Z}_{b,j}\|_*\leq 1} -\sum_{i=1}^n \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i) + \sum_{b=1}^B \sum_{j=1}^m \lambda_{b,j}(\beta - \operatorname{trace}(\mathbf{Z}_{b,j}^\top \sum_{i=1}^n \mathbf{D}_{b,j}^{(i)} \operatorname{circ}(\mathbf{X}_i^{(b)})^\top \mathbf{V}_i^{(b)})). \tag{155}
$$

We switch the order of the maximum and minimum using Sion's minimax theorem and maximize over $\mathbf{V}_i$

$$
p^*_{BFN} = \min_{\lambda_j\geq 0} \min_{\|\mathbf{Z}_{b,j}\|_*\leq 1} \sum_{i=1}^n \mathcal{L}(\left[\sum_{j=1}^m \lambda_{1,j}\mathbf{D}_{1,j}\operatorname{circ}(\mathbf{X}_i^{(1)})\mathbf{Z}_{1,j} \quad \cdots \quad \sum_{j=1}^m \lambda_{B,j}\mathbf{D}_{B,j}\operatorname{circ}(\mathbf{X}_i^{(B)})\mathbf{Z}_{B,j}\right], \mathbf{Y}_i) + \beta \sum_{b=1}^B \sum_{j=1}^m \lambda_{b,j}. \tag{156}
$$

Lastly, we rescale $\tilde{\mathbf{Z}}_{b,j} = \lambda_{b,j}\mathbf{Z}_{b,j}$ to obtain

$$
p^*_{BFN} = \min_{\mathbf{Z}_{b,j}} \sum_{i=1}^n \mathcal{L}\left(\left[\sum_{j=1}^m \mathbf{D}_{1,j}\operatorname{circ}(\mathbf{X}_i^{(1)})\mathbf{Z}_{1,j} \quad \cdots \quad \sum_{j=1}^m \mathbf{D}_{B,j}\operatorname{circ}(\mathbf{X}_i^{(B)})\mathbf{Z}_{B,j}\right], \mathbf{Y}_i\right) + \beta \sum_{b=1}^B \sum_{j=1}^m \|\mathbf{Z}_{j,b}\|_*, \tag{157}
$$

as desired.

□

C.3. Additional Attention Alternatives: PoolFormer and FNet

In (Yu et al., 2021), the authors propose a simple alternative to the standard MLP-Mixer architecture. In particular, the forward function is given by

$$
f_{PF}(\mathbf{X}_i) = \sigma(\mathbf{P}\mathbf{X}_i\mathbf{W}_1)\mathbf{W}_2 \tag{158}
$$

where $\mathbf{P} \in \mathbb{R}^{s\times s}$ is a local pooling function. In this way, the PoolFormer architecture still mixes across different tokens, but in a non-learnable, deterministic fashion.

In (Lee-Thorp et al., 2021), the authors propose FNet, another alternative which resembles PoolFormer architecture. In particular, a 2D FFT is applied to the input $\mathbf{X}_i$ before being passed through an MLP

$$
f_{FNET}(\mathbf{X}_i) = \sigma(\mathbf{F}_s\mathbf{X}_i\mathbf{F}_d^\top\mathbf{W}_1)\mathbf{W}_2 \tag{159}
$$

One can use similar convex duality results as in the main body of this paper to generate convex dual forms for this architecture for linear, ReLU, and gated ReLU activation PoolFormers and FNets. To keep these results general, we will be analyzing networks of the form

$$
p^*_{PF} := \min_{\mathbf{w}_{1j},\mathbf{w}_{2j}} \sum_{i=1}^n \mathcal{L}\left(\sum_{j=1}^m \sigma(h(\mathbf{X}_i)\mathbf{w}_{1j})\mathbf{w}_{2j}^\top, \mathbf{Y}_i\right) + \frac{\beta}{2}\sum_{j=1}^m \|\mathbf{w}_{1j}\|_2^2 + \|\mathbf{w}_{2j}\|_2^2 \tag{160}
$$

for any generic function $h : \mathbb{R}^{s\times d} \to \mathbb{R}^{s\times d}$, which encapsulates both methods and more.

Theorem C.5. *For the linear activation network training problem* (160)*, for $m \geq m^*$ where $m^* \leq \min\{d, c\}$, the standard non-convex training objective is equivalent to a convex optimization problem, given by*

$$
p^*_{PF} = \min_{\mathbf{Z}\in\mathbb{R}^{d\times c}} \sum_{i=1}^n \mathcal{L}\left(h(\mathbf{X}_i)\mathbf{Z}, \mathbf{Y}_i\right) + \beta\|\mathbf{Z}\|_*. \tag{161}
$$

Proof. We now apply Lemmas A.1 and A.2 to obtain

$$p^*_{PF} = \max_{\mathbf{V}_i} -\sum_{i=1}^n \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i)$$

$$\text{s.t. } \max_{\|\mathbf{w}_1\|_2 \leq 1} \|\sum_{i=1}^n \mathbf{V}_i^\top h(\mathbf{X}_i)\mathbf{w}_1\|_2 \leq \beta \tag{162}$$

This is equivalent to

$$p^*_{PF} = \max_{\mathbf{V}_i} -\sum_{i=1}^n \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i)$$

$$\text{s.t. } \|\sum_{i=1}^n \mathbf{V}_i^\top h(\mathbf{X}_i)\|_2 \leq \beta \tag{163}$$

We form the Lagrangian as

$$p^*_{PF} = \max_{\mathbf{V}_i} \min_{\lambda \geq 0} \min_{\|\mathbf{Z}\|_* \leq 1} -\sum_{i=1}^n \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i) + \lambda(\beta - \text{trace}(\mathbf{Z}^\top \sum_{i=1}^n h(\mathbf{X}_i)^\top \mathbf{V}_i)). \tag{164}$$

We switch the order of the maximum and minimum using Sion's minimax theorem and maximize over $\mathbf{V}_i$

$$p^*_{PF} = \min_{\lambda \geq 0} \min_{\|\mathbf{Z}\|_* \leq 1} \sum_{i=1}^n \mathcal{L}(h(\mathbf{X}_i)\mathbf{Z}, \mathbf{Y}_i) + \beta\lambda. \tag{165}$$

Lastly, we rescale $\tilde{\mathbf{Z}} = \lambda\mathbf{Z}$ to obtain

$$p^*_{PF} = \min_{\mathbf{Z} \in \mathbb{R}^{d \times c}} \sum_{i=1}^n \mathcal{L}\left(h(\mathbf{X}_i)\mathbf{Z}, \mathbf{Y}_i\right) + \beta\|\mathbf{Z}\|_*. \tag{166}$$

as desired. □

Theorem C.6. *For the ReLU activation training problem* (160)*, we define*

$$\mathbf{X} := \begin{bmatrix} h(\mathbf{X}_1) \\ \cdots \\ h(\mathbf{X}_n) \end{bmatrix}$$

$$\{\mathbf{D}_j\}_{j=1}^P := \{\text{diag}\left(\mathbb{1}\{\mathbf{X}\mathbf{u}_j \geq 0\}\right) : \ \mathbf{u}_j \in \mathbb{R}^d\},$$

where $P \leq 2r\left(\frac{e(n-1)}{r}\right)^r$ *and* $r := \text{rank}(\mathbf{X})$*. Then, for* $m \geq m^*$ *where* $m^* \leq n\min\{d, c\}$*, the standard non-convex training objective is equivalent to a convex optimization problem, given by*

$$p^*_{PF} = \min_{\mathbf{Z}_j \in \mathbb{R}^{d \times c}} \sum_{i=1}^n \mathcal{L}\left(\sum_{j=1}^P \mathbf{D}_j^{(i)} h(\mathbf{X}_i)\mathbf{Z}_j, \mathbf{Y}_i\right) + \beta\sum_{j=1}^P \|\mathbf{Z}_j\|_{*,\mathrm{K}_j}, \tag{167}$$

where

$$\mathbf{K}_j := (2\mathbf{D}_j - \mathbf{I}_{ns})\mathbf{X}.$$

Proof. We now apply Lemmas A.1 and A.2 with the ReLU activation function to obtain

$$p^*_{PF} = \max_{\mathbf{V}_i} -\sum_{i=1}^n \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i)$$

$$\text{s.t. } \max_{\|\mathbf{w}_1\|_2 \leq 1} \|\sum_{i=1}^n \mathbf{V}_i^\top (h(\mathbf{X}_i)\mathbf{w}_1)_+\|_2 \leq \beta \tag{168}$$

We introduce hyperplane arrangements $\mathbf{D}_j$ and enumerate over all of them, yielding

$$p_{PF}^* = \max_{\mathbf{V}_i} - \sum_{i=1}^n \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i)$$
$$\text{s.t. } \max_{\substack{\|\mathbf{w}_1\|_2 \le 1 \\ j \in [P] \\ \mathbf{K}_j \mathbf{w}_1 \ge 0}} \| \sum_{i=1}^n \mathbf{V}_i^\top \mathbf{D}_j^{(i)} h(\mathbf{X}_i) \mathbf{w}_1 \|_2 \le \beta. \tag{169}$$

Using the concept of dual norm, this is equivalent to

$$p_{PF}^* = \max_{\mathbf{V}_i} - \sum_{i=1}^n \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i)$$
$$\text{s.t. } \max_{\substack{\|\mathbf{g}\|_2 \le 1 \\ \|\mathbf{w}_1\|_2 \le 1 \\ j \in [P] \\ \mathbf{K}_j \mathbf{w}_1 \ge 0}} \mathbf{g}^\top \sum_{i=1}^n \mathbf{V}_i^\top \mathbf{D}_j^{(i)} h(\mathbf{X}_i) \mathbf{w}_1 \le \beta \tag{170}$$

We can also define sets $\mathcal{C}_j := \{\mathbf{Z} = \mathbf{u}\mathbf{g}^\top \in \mathbb{R}^{d \times c} : \mathbf{K}_j \mathbf{u} \ge 0,\ \|\mathbf{Z}\|_* \le 1\}$. Then, we have

$$p_{PF}^* = \max_{\mathbf{V}_i} - \sum_{i=1}^n \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i)$$
$$\text{s.t. } \max_{\substack{j \in [P] \\ \mathbf{Z} \in \mathcal{C}_j}} \operatorname{trace} \left(\sum_{i=1}^n \mathbf{V}_i^\top \mathbf{D}_j^{(i)} h(\mathbf{X}_i) \mathbf{Z} \right) \le \beta \tag{171}$$

We form the Lagrangian as

$$p_{PF}^* = \max_{\mathbf{V}_i} \min_{\lambda \ge 0} \min_{\mathbf{Z}_j \in \mathcal{C}_j} - \sum_{i=1}^n \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i) + \sum_{j=1}^P \lambda_j (\beta - \operatorname{trace}(\mathbf{Z}_j^\top \sum_{i=1}^n \mathbf{D}_j^{(i)} h(\mathbf{X}_i)^\top \mathbf{V}_i)). \tag{172}$$

We switch the order of the maximum and minimum using Sion's minimax theorem and maximize over $\mathbf{V}_i$

$$p_{PF}^* = \min_{\lambda_j \ge 0} \min_{\mathbf{Z}_j \in \mathcal{C}_j} \sum_{i=1}^n \mathcal{L}(\sum_{j=1}^P \mathbf{D}_j^{(i)} h(\mathbf{X}_i) \mathbf{Z}_j, \mathbf{Y}_i) + \beta \sum_{j=1}^P \lambda_j. \tag{173}$$

Lastly, we rescale $\tilde{\mathbf{Z}}_j = \lambda_j \mathbf{Z}_j$ to obtain

$$p_{PF}^* = \min_{\mathbf{Z}_j \in \mathbb{R}^{d \times c}} \sum_{i=1}^n \mathcal{L} \left(\sum_{j=1}^P \mathbf{D}_j^{(i)} h(\mathbf{X}_i) \mathbf{Z}_j, \mathbf{Y}_i \right) + \beta \sum_{j=1}^P \|\mathbf{Z}_j\|_{*,\mathrm{K}_j}. \tag{174}$$

as desired. □

Theorem C.7. *For the Gated ReLU activation training problem* (160)*, we define*

$$\mathbf{X} := \begin{bmatrix} h(\mathbf{X}_1) \\ \cdots \\ h(\mathbf{X}_n) \end{bmatrix}$$
$$\{\mathbf{D}_j\}_{j=1}^P := \{\operatorname{diag}(\mathbb{1}\{\mathbf{X}\mathbf{h}_j \ge 0\})\},$$

for fixed gates $\{\mathbf{h}_j \in \mathbb{R}^d\}_{j=1}^m$. Then, the standard non-convex training objective is equivalent to a convex optimization problem, given by

$$p_{PF}^* = \min_{\mathbf{Z}_j \in \mathbb{R}^{d \times c}} \sum_{i=1}^n \mathcal{L} \left(\sum_{j=1}^m \mathbf{D}_j^{(i)} h(\mathbf{X}_i) \mathbf{Z}_j, \mathbf{Y}_i \right) + \beta \sum_{j=1}^m \|\mathbf{Z}_j\|_*. \tag{175}$$

Proof. We now apply Lemmas A.1 and A.2 with the Gated ReLU activation function to obtain

$$\begin{aligned} p_{PF}^* &= \max_{\mathbf{V}_i} - \sum_{i=1}^n \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i) \\ \text{s.t. } &\max_{\substack{\|\mathbf{w}_1\|_2 \leq 1 \\ j \in [m]}} \| \sum_{i=1}^n \mathbf{V}_i^\top \mathbf{D}_j^{(i)} h(\mathbf{X}_i)\mathbf{w}_1 \|_2 \leq \beta. \end{aligned} \tag{176}$$

Using the concept of dual norm, this is equivalent to

$$\begin{aligned} p_{PF}^* &= \max_{\mathbf{V}_i} - \sum_{i=1}^n \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i) \\ \text{s.t. } &\max_{\substack{\|\mathbf{g}\|_2 \leq 1 \\ \|\mathbf{w}_1\|_2 \leq 1 \\ j \in [m]}} \mathbf{g}^\top \sum_{i=1}^n \mathbf{V}_i^\top \mathbf{D}_j^{(i)} h(\mathbf{X}_i)\mathbf{w}_1 \leq \beta \end{aligned} \tag{177}$$

Then, we have

$$\begin{aligned} p_{PF}^* &= \max_{\mathbf{V}_i} - \sum_{i=1}^n \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i) \\ \text{s.t. } &\max_{\substack{j \in [m] \\ \|\mathbf{Z}\|_* \leq 1}} \operatorname{trace}\left(\sum_{i=1}^n \mathbf{V}_i^\top \mathbf{D}_j^{(i)} h(\mathbf{X}_i)\mathbf{Z} \right) \leq \beta \end{aligned} \tag{178}$$

We form the Lagrangian as

$$p_{PF}^* = \max_{\mathbf{V}_i} \min_{\lambda \geq 0} \min_{\|\mathbf{Z}_j\|_* \leq 1} - \sum_{i=1}^n \mathcal{L}^*(\mathbf{V}_i, \mathbf{Y}_i) + \sum_{j=1}^m \lambda_j (\beta - \operatorname{trace}(\mathbf{Z}_j^\top \sum_{i=1}^n \mathbf{D}_j^{(i)} h(\mathbf{X}_i)^\top \mathbf{V}_i)). \tag{179}$$

We switch the order of the maximum and minimum using Sion's minimax theorem and maximize over $\mathbf{V}_i$

$$p_{PF}^* = \min_{\lambda_j \geq 0} \min_{\|\mathbf{Z}_j\|_* \leq 1} \sum_{i=1}^n \mathcal{L}(\sum_{j=1}^m \mathbf{D}_j^{(i)} h(\mathbf{X}_i)\mathbf{Z}_j, \mathbf{Y}_i) + \beta \sum_{j=1}^m \lambda_j. \tag{180}$$

Lastly, we rescale $\tilde{\mathbf{Z}}_j = \lambda_j \mathbf{Z}_j$ to obtain

$$p_{PF}^* = \min_{\mathbf{Z}_j \in \mathbb{R}^{d \times c}} \sum_{i=1}^n \mathcal{L}\left(\sum_{j=1}^m \mathbf{D}_j^{(i)} h(\mathbf{X}_i)\mathbf{Z}_j, \mathbf{Y}_i \right) + \beta \sum_{j=1}^m \|\mathbf{Z}_j\|_*. \tag{181}$$

as desired. □