SCALING CONVEX NEURAL NETWORKS WITH BURER-MONTEIRO FACTORIZATION

Arda Sahiner
Arcus Inc.
Stanford University

Tolga Ergen
LG AI Research

Batu Ozturkler
Stanford University

John Pauly
Stanford University

Morteza Mardani
NVIDIA Corporation

Mert Pilanci
Stanford University

ABSTRACT

It has been demonstrated that the training problem for a variety of (non) linear two-layer neural networks (such as two-layer perceptrons, convolutional networks, and self-attention) can be posed as equivalent convex optimization problems, with an induced regularizer which encourages low rank. However, this regularizer becomes prohibitively expensive to compute at moderate scales, impeding training convex neural networks. To this end, we propose applying the Burer-Monteiro factorization to convex neural networks, which for the first time enables a Burer-Monteiro perspective on neural networks with non-linearities. This factorization leads to an equivalent yet computationally tractable non-convex alternative with no spurious local minima. We develop a novel relative optimality bound of stationary points of the Burer-Monteiro factorization, providing verifiable conditions under which any stationary point is a global optimum. For the first time, we show that linear self-attention with sufficiently many heads has no spurious local minima.

1 INTRODUCTION

It has been demonstrated that the training problem for (non-linear) two-layer neural networks are equivalent to convex programs (Pilanci & Ergen, 2020; Ergen & Pilanci, 2020; Sahiner et al., 2021b; Ergen et al., 2021; Sahiner et al., 2021a). This has been observed for a variety of architectures, including multi-layer perceptrons (MLPs) (Pilanci & Ergen, 2020; Sahiner et al., 2021b), convolutional neural networks (CNNs) (Ergen & Pilanci, 2020; Sahiner et al., 2021c), and self-attention based transformers (Sahiner et al., 2022). A major benefit of convex training of neural networks is that global optimality is guaranteed, which brings transparency to training neural networks.

The convex formulation of neural networks induces biases by regularization of the network weights. For linear activation, the convex model directly imposes nuclear-norm regularization which is well-known to encourage low-rank solutions (Recht et al., 2010). For ReLU activation, however, the convex model induces a type of nuclear norm which promotes sparse factorization while the left factor is *constrained* to an affine space (Sahiner et al., 2021b). This constrained nuclear-norm is NP-hard to compute. This impedes the utility of convex neural networks for ReLU activation. To address this computational challenge, we seek a method which (i) inherits the per-iteration complexity of non-convex training of neural network, and (ii) inherits the optimality guarantees and transparency of convex training. To find a solution, we leverage the well-studied Burer-Monteiro (BM) factorization (Burer & Monteiro, 2003).

BM has been applied as an efficient solution strategy for problems ranging from matrix factorization (Zheng & Lafferty, 2016; Park et al., 2017; Ge et al., 2017; Gillis, 2017) to rank minimization (Mardani et al., 2013; Recht et al., 2010; Wang et al., 2017) and matrix completion (Mardani et al., 2015; Ge et al., 2017). BM has also been used for over-simplified neural networks such as (Kawaguchi, 2016; Haeffele & Vidal, 2017; Du & Lee, 2018), where optimality conditions for local minima are provided. However, no work has deployed BM factorization for practical non-linear neural networks, and no guarantees are available about the optimality of stationary points.

Thus, our focus in this work is to adapt BM for practical two-layer (non-linear) convex neural networks. We consider three common architectures, namely MLPs, CNNs, and self-attention networks. For these scenarios, we develop verifiable relative optimality bounds for all local minima and stationary points, which are easy and interpretable. In light of these conditions, we identify useful insights about the nature of neural networks contributing to optimality. In particular, we observe that for self-attention networks all local minima coincide with the global optima if there are sufficiently many heads. The optimality guarantees also provide useful algorithmic insights, allowing one to verify whether the light-weight first-order methods such as SGD achieve the global optimum for the non-convex training of neural networks. Our experiments with image classification task indicate that this BM factorization enables layerwise training of convex CNNs, which allows for convex networks for the first time to match the performance of multi-layer end-to-end trained non-convex CNNs.

1.1 Contributions

All in all, our contributions are summarized as follows:

- We propose the BM factorization for efficiently solving convex neural networks with ReLU activation for moderate and large scales. This is the first time BM theory has been applied to the non-linear neural network setting to the best of our knowledge.
- We derive a novel bound on the relative optimality of the stationary points of the BM factorization for neural networks.
- We identify simple and verifiable conditions which guarantee a stationary point of the non-convex BM formulation achieves the global optimum of the convex neural network.
- We provide insights into the fundamental building blocks of neural networks that contribute to optimality; e.g. that linear self-attention has no spurious local minima if it has sufficiently many heads.
- Our experiments verify the proposed relative optimality bound of stationary points from the BM factorization, and uncovers cases where SGD converges to saddle points, even in two-layer neural networks.

1.2 Related Work

Burer-Monteiro factorization. The Burer-Monteiro (BM) factorization was first introduced in (Burer & Monteiro, 2003; 2005). There has been a long line of work studying the use of this factorization for solving SDPs (Boumal et al., 2016; Cifuentes & Moitra, 2019; Waldspurger & Waters, 2020; Erdogdu et al., 2021). In the rectangular matrix case, gradient descent converges to a global optimum of the matrix factorization problem with high probability for certain classes of matrices (Zheng & Lafferty, 2016). The BM factorization has been also studied in the rectangular case in more generic settings (Bach et al., 2008; Haeffele et al., 2014; Haeffele & Vidal, 2017).

Nuclear norm and rank minimization. The ability of nuclear norm regularization to induce low rank has been studied extensively in compressed sensing (Candès & Recht, 2009; Recht et al., 2010; Candès & Tao, 2010). BM factorization has been applied to scale up nuclear-norm minimization (Mardani et al., 2015; 2013). It has also been deployed for low-rank matrix factorization (Cabral et al., 2013; Zhu et al., 2017; Park et al., 2017; Ge et al., 2017). The results show that all second-order critical points of the BM factorization are global optima if certain qualification conditions are met.

SGD for non-convex neural networks. It has been shown that for over-parameterized two-layer linear networks, all local minima are global minima (Kawaguchi, 2016). Accordingly, a line of work has attempted to show that gradient descent or its modifications provably find local minima and escape saddle points (Ge et al., 2015; Lee et al., 2016; Jin et al., 2017; Daneshmand et al., 2018). However, these works assume Lipschitz gradients and Hessians of the non-convex objective, which is not typically satisfied. Another line of work shows that gradient descent converges to global optima for sufficiently highly over-parameterized neural networks, with either the parameter count being a high-order polynomial of the sample count (Du et al., 2018; 2019; Arora et al., 2019), or the network architecture being simple (Du & Lee, 2018). In practice, it has been empirically observed that SGD can converge to local maxima, or get stuck in saddle points (Du et al., 2017; Ziyin et al., 2021). For

unregularized matrix factorization, it has also recently been shown that randomly initialized gradient descent provably converges to global minima (Ye & Du, 2021).

Convex neural networks. There is a long history of architecting convex optimization problems that mimic the performance of neural networks Zhang et al. (2016; 2017). It has recently been found that ReLU neural networks have equivalent convex programs for training, such as networks with scalar outputs (Pilanci & Ergen, 2020), vector-outputs (Sahiner et al., 2021b), convolutional networks (Ergen & Pilanci, 2020; Sahiner et al., 2021c), polynomial-activation networks (Bartan & Pilanci, 2021), batch-norm based networks (Ergen et al., 2021), Wasserstein GANs (Sahiner et al., 2021a), and self-attention networks (Sahiner et al., 2022). Despite efforts in developing efficient solvers, convex networks are only effectively trainable at small scales (Bai et al., 2022; Mishkin et al., 2022). Our novelty is to adapt BM factorization as a fast and scalable solution for training convex networks, with simple, verifiable conditions for global optimality.

2 PRELIMINARIES

We denote $(\cdot)_+ := \max\{0, \cdot\}$ as the ReLU non-linearity. We use superscripts, say $\mathbf{A}^{(i_i, i_2)}$, to denote blocks of matrices, and brackets, say $\mathbf{A}[i_1, i_2]$, to denote elements of matrices. We let $\mathbf{1}$ be the vector of ones of appropriate size, $\|\cdot\|_H$ be the ℓ-p norm with $p = H$, $\|\cdot\|_F$ be the Frobenius norm, and $\mathcal{B}_H$ be the unit H-norm ball, $\{\mathbf{u} : \|\mathbf{u}\|_H \leq 1\}$, . Unless otherwise stated, let F be a convex, differentiable function. We use n to denote the number of samples, and c to denote the output dimension of each network. All proofs are presented in Appendix A.

2.1 TWO-LAYER NEURAL NETWORKS AS CONVEX PROGRAMS

A line of work has demonstrated that two-layer neural networks are equivalent to convex optimization problems. We consider a data matrix $\mathbf{X} \in \mathbb{R}^{n \times d}$ and consider two-layer σ-activation network with c outputs, m neurons, weight-decay parameter $\beta > 0$:

$$p^*_{MLP} := \min_{\substack{\mathbf{W}_1 \in \mathbb{R}^{d \times m} \\ \mathbf{W}_2 \in \mathbb{R}^{c \times m}}} F(\sigma(\mathbf{X}\mathbf{W}_1)\mathbf{W}_2^\top) + \frac{\beta}{2} \sum_{j=1}^{m} \left(\|\mathbf{w}_{1j}\|_2^2 + \|\mathbf{w}_{2j}\|_2^2 \right). \tag{1}$$

When σ is a linear activation and $m \geq m^*$ for some $m^* \leq \min\{d, c\}$, this problem is equivalent to ((Rennie & Srebro, 2005), Section 2.2)

$$p^*_{LMLP} = \min_{\mathbf{Z} \in \mathbb{R}^{d \times c}} F(\mathbf{X}\mathbf{Z}) + \beta \|\mathbf{Z}\|_*, \tag{2}$$

whereas for a ReLU activation and $m \geq m^*$ for some unknown, problem-dependent $m^* \leq nc$ ((Sahiner et al., 2021b), Thm. 3.1),

$$p^*_{RMLP} = \min_{\mathbf{Z}_j \in \mathbb{R}^{d \times c}} F(\sum_{j=1}^{P} \mathbf{D}_j \mathbf{X} \mathbf{Z}_j) + \beta \sum_{j=1}^{P} \|\mathbf{Z}_j\|_{*,\mathrm{K}_j}, \ \mathbf{K}_j := (2\mathbf{D}_j - \mathbf{I}_n)\mathbf{X} \tag{3}$$

where $\{\mathbf{D}_j\}_{j=1}^P = \{\mathrm{diag}\left(\mathbb{1}\{\mathbf{X}\mathbf{u} \geq 0\}\right) : \mathbf{u} \in \mathbb{R}^d\}$ enumerates the possible activation patterns generated from $\mathbf{X}$, and the number of such patterns satisfies $P \leq 2r \left(\frac{e(n-1)}{r}\right)^r$, where $r := \mathrm{rank}(\mathbf{X})$ (Stanley et al., 2004; Pilanci & Ergen, 2020). The expression (3) also involves a constrained nuclear norm expression, which is defined as

$$\|\mathbf{Z}\|_{*,\mathrm{K}} := \min_{t \geq 0} t \text{ s.t. } \mathbf{Z} \in t\mathcal{C}, \ \mathcal{C} := \mathrm{conv}\{\mathbf{Z} = \mathbf{u}\mathbf{v}^\top : \ \mathbf{K}\mathbf{u} \geq 0, \ \|\mathbf{u}\|_2 \leq 1, \|\mathbf{v}\|_2 \leq 1\}. \tag{4}$$

This norm is a quasi-nuclear norm, which differs from the standard nuclear norm in that the factorization upon which it relies imposes a constraint on its left factors. In convex ReLU neural networks, this norm enforces 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 $\sum_k \|\mathbf{u}_k \mathbf{v}_k^\top\|_*$. This norm is NP-hard to compute (Sahiner et al., 2021b). A variant of these ReLU activations, called *gated ReLU* activations, achieves the piecewise linearity of ReLU activations without enforcing the constraints (Fiat et al., 2019). Specifically, the ReLU gates are fixed to some $\{\mathbf{h}_j\}_{j=1}^P$ to form

$$\sigma(\mathbf{X}\mathbf{w}_{1j}) := \mathrm{diag}\left(\mathbb{1}\{\mathbf{X}\mathbf{h}_j \geq 0\}\right)(\mathbf{X}\mathbf{w}_{1j}) = \mathbf{D}_j \mathbf{X} \mathbf{w}_{1j}.$$

With gated ReLU activation, the equivalent convex program is given by ((Sahiner et al., 2022))

$$p^*_{GMLP} = \min_{\mathbf{Z}_j \in \mathbb{R}^{d\times c}} F(\sum_{j=1}^{P} \mathbf{D}_j \mathbf{X} \mathbf{Z}_j) + \beta \sum_{j=1}^{P} \|\mathbf{Z}_j\|_*, \tag{5}$$

which thereby converts the constrained nuclear norm penalty to a standard nuclear norm penalty, improving the complexity of the ReLU network. In addition to the multi-layer perceptron (MLP) formulation, two-layer ReLU-activation convolutional neural networks (CNNs) with global average pooling have been demonstrated to be equivalent to convex programs as well (Sahiner et al., 2021b;c; Ergen & Pilanci, 2020). The non-convex formulation is given by

$$p^*_{RCNN} := \min_{\substack{\mathbf{w}_{1j}\in\mathbb{R}^h \\ \mathbf{w}_{2j}\in\mathbb{R}^c}} \sum_{i=1}^{n} F\Big(\sum_{j=1}^{m} \mathbf{w}_{2j} \mathbf{1}^\top (\mathbf{X}_i \mathbf{w}_{1j})_+\Big) + \frac{\beta}{2} \sum_{j=1}^{m} \|\mathbf{w}_{1j}\|_2^2 + \|\mathbf{w}_{2j}\|_2^2, \tag{6}$$

where samples $\mathbf{X}_i \in \mathbb{R}^{K\times h}$ are represented by patch matrices, which hold a convolutional patch of size h in each of their K rows. It has been shown (Sahiner et al., 2021b) that as long as $m \geq m^*$ where $m^* \leq nc$, this is equivalent to a convex program ((Sahiner et al., 2021b), Cor. 5.1)

$$p^*_{RCNN} = \min_{\mathbf{Z}_j \in \mathbb{R}^{h\times c}} \sum_{i=1}^{n} F((\sum_{j=1}^{P} \mathbf{1}^\top \mathbf{D}_j^{(i)} \mathbf{X}_i \mathbf{Z}_j)^\top) + \beta \sum_{j=1}^{P} \|\mathbf{Z}_j\|_{*,\mathrm{K}_j}$$

$$\mathbf{K}_j := (2\mathbf{D}_j - \mathbf{I}_{nK})\mathbf{X},\ \mathbf{X} := \begin{bmatrix} \mathbf{X}_1 \\ \cdots \\ \mathbf{X}_n \end{bmatrix} \tag{7}$$

where $\{\mathbf{D}_j\}_{j=1}^P = \left\{\mathrm{diag}\left(\mathbb{1}\{\mathbf{X}\mathbf{u} \geq 0\}\right) : \mathbf{u} \in \mathbb{R}^h\right\}$ and $\mathbf{D}_j^{(i)} \in \mathbb{R}^{K\times K}$. The only exponential dependence of P is on h, which is typically fixed.

Lastly, we review existing convexity results for self-attention transformers (Sahiner et al., 2022). We have the following non-convex objective for a single block of multi-head self-attention with m heads, where $\mathbf{X}_i \in \mathbb{R}^{s\times d}$ with s tokens and d features

$$p^*_{SA} := \min_{\substack{\mathbf{W}_{1j}\in\mathbb{R}^{d\times d} \\ \mathbf{W}_{2j}\in\mathbb{R}^{d\times c}}} \sum_{i=1}^{n} F\left(\sum_{j=1}^{m} \sigma\left(\mathbf{X}_i \mathbf{W}_{1j} \mathbf{X}_i^\top\right) \mathbf{X}_i \mathbf{W}_{2j}\right) + \frac{\beta}{2} \sum_{j=1}^{m} \|\mathbf{W}_{1j}\|_F^2 + \|\mathbf{W}_{2j}\|_F^2, \tag{8}$$

for which a variety of objectives F can be posed, including classification (e.g. F incorporates global average pooling followed by softmax-cross-entropy with labels) or denoising (e.g. F is a squared loss). For linear, gated ReLU, and ReLU activation, this is equivalent to a convex program (see Appendices A.12, B.3). Here, we show that these architectures are amenable to the BM factorization.

2.2 The Burer-Monteiro Factorization

First proposed by Burer & Monteiro (2003), the Burer-Monteiro (BM) factorization proposes to solve SDPs over some square matrix $\mathbf{Q}$ in terms of rectangular factors $\mathbf{R}$ where $\mathbf{Q}$ is substituted by $\mathbf{R}\mathbf{R}^\top$. It was first demonstrated that solving over $\mathbf{R}$ does not introduce spurious local minima rank-constrained SDPs, given $\mathrm{rank}(\mathbf{R}) \geq \mathrm{rank}(\mathbf{Q}^*)$ for optimal solution to the original SDP $\mathbf{Q}^*$ (Burer & Monteiro, 2005). We seek applications where we optimize over a non-square matrix $\mathbf{Z}$, i.e.

$$p^*_{CVX} := \min_{\mathbf{Z}\in\mathbb{R}^{d\times c}} F(\mathbf{Z}) \tag{9}$$

for a convex, differentiable function F. One may approach this by factoring $\mathbf{Z} = \mathbf{U}\mathbf{V}^\top$, where $\mathbf{U} \in \mathbb{R}^{d\times m}$, $\mathbf{V} \in \mathbb{R}^{c\times m}$ for some arbitrary choice m. Then, we have an equivalent non-convex problem over $\mathbf{R} := \begin{bmatrix} \mathbf{U} \\ \mathbf{V} \end{bmatrix}$, for $f(\mathbf{R}) = F(\mathbf{U}\mathbf{V}^\top)$:

$$p^*_{CVX} = \min_{\mathbf{R}} f(\mathbf{R}). \tag{10}$$

Noting that (9) is convex over $\mathbf{R}\mathbf{R}^\top = \begin{bmatrix} \mathbf{U}\mathbf{U}^\top & \mathbf{U}\mathbf{V}^\top \\ \mathbf{V}\mathbf{U}^\top & \mathbf{V}\mathbf{V}^\top \end{bmatrix}$, one may apply directly the result of Boumal et al. (2020) to conclude that as long as $m \geq d + c$, all local minima of (10) are global

minima of (9) (see Appendix A.2). Further, work from Bach et al. (2008) and Haeffele et al. (2014) demonstrates that all rank-deficient local minimizers of (10) achieve the global minimum p^*_{CVX} [1].

A long line of work has analyzed the conditions where known non-convex optimization algorithms will converge to second-order critical points (local minima) (Ge et al., 2015; Jin et al., 2017; Daneshmand et al., 2018). Under the assumption of a bounded f and its Hessian, a second-order critical point can be found by noisy gradient descent (Ge et al., 2015), or other second-order algorithms (Sun et al., 2015). Even vanilla gradient descent with random initialization has been demonstrated to almost surely converge to a local minimum for f with Lipschitz gradient (Lee et al., 2016). However, if the gradient of f is not Lipschitz-continuous, there are no guarantees that gradient descent will find a second-order critical point of (10): one may encounter a stationary point which is a saddle. For example, in the linear regression setting, i.e.

$$f(\mathbf{R}) = \|\mathbf{X}\mathbf{U}\mathbf{V}^\top - \mathbf{Y}\|_F^2, \tag{11}$$

the gradient of f is Lipschitz continuous with respect to $\mathbf{U}$ when $\mathbf{V}$ is fixed and vice-versa, but not Lipschitz continuous with respect to $\mathbf{R}$ (Mukkamala & Ochs, 2019). Thus, one may not directly apply the results of Ge et al. (2015); Sun et al. (2015); Lee et al. (2016) in this case. Instead, we seek to understand the conditions under which stationary points to (10) correspond to global optima of (9). One such condition is given in Mardani et al. (2013; 2015).

Theorem 2.1 (From (Mardani et al., 2013))**.** *Stationary points $\hat{\mathbf{U}}, \hat{\mathbf{V}}$ of the optimization problem*

$$p^* := \min_{\mathbf{U},\mathbf{V}} \frac{1}{2}\|\mathbf{U}\mathbf{V}^\top - \mathbf{Y}\|_F^2 + \frac{\beta}{2}\left(\|\mathbf{U}\|_F^2 + \|\mathbf{V}\|_F^2\right) \tag{12}$$

correspond to global optima $\mathbf{Z}^* = \hat{\mathbf{U}}\hat{\mathbf{V}}^\top$ of the equivalent convex optimization problem

$$p^* = \min_{\mathbf{Z}} \frac{1}{2}\|\mathbf{Z} - \mathbf{Y}\|_F^2 + \beta\|\mathbf{Z}\|_* \tag{13}$$

provided that $\|\mathbf{Y} - \hat{\mathbf{U}}\hat{\mathbf{V}}^\top\|_2 \leq \beta$.

3 Burer-Monteiro Factorization for Convex Neural Networks

3.1 MLPs

We first seek to compare the convex formulations of the MLP training problem (2), (3), and (5) to their BM factorizations. We describe how to find the BM factorization for any convex MLP.

Lemma 3.1. *For any matrix $\mathbf{M} \in \mathbb{R}^{n \times d_c}$, let $f(\mathbf{U}, \mathbf{V}) := F(\mathbf{M}\mathbf{U}\mathbf{V}^\top)$ be a differentiable function. For any $\beta > 0$ and arbitrary vector norms $\|\cdot\|_R$ and $\|\cdot\|_C$, we define the* Burer-Monteiro factorization

$$p^* := \min_{\substack{\mathbf{U}\in\mathbb{R}^{d_c\times m}\\ \mathbf{V}\in\mathbb{R}^{d_r\times m}}} f(\mathbf{U}, \mathbf{V}) + \frac{\beta}{2}\left(\sum_{j=1}^{m} \|\mathbf{u}_j\|_C^2 + \|\mathbf{v}_j\|_R^2\right). \tag{14}$$

For the matrix norm $\|\cdot\|_D$ defined as

$$\|\mathbf{Z}\|_D := \max_{\mathbf{R}} \operatorname{trace}(\mathbf{R}^\top\mathbf{Z}) \text{ s.t. } \mathbf{u}^\top\mathbf{R}\mathbf{v} \leq 1 \; \forall \mathbf{u} \in \mathcal{B}_C, \; \forall \mathbf{v} \in \mathcal{B}_R, \tag{15}$$

the problem (14) is equivalent to the convex optimization problem

$$p^* = \min_{\mathbf{Z}\in\mathbb{R}^{d_c\times d_r}} F(\mathbf{M}\mathbf{Z}) + \beta\|\mathbf{Z}\|_D. \tag{16}$$

Remark 3.2. In the case of a linear MLP, $\mathbf{M} = \mathbf{X}$, $d_c = d$, $d_r = c$, and $\|\cdot\|_D = \|\cdot\|_*$, so using the definition of $\|\cdot\|_D$, in the corresponding BM factorization, $R = 2$ and $C = 2$ (Bach et al., 2008). For a gated ReLU network, the regularizer is still the nuclear norm, and thus the same $R = C = 2$ regularization appears in the BM factorization. In the case of the ReLU MLP, the nuclear norm is

[1] Under mild conditions, see Appendix A.3.

replaced by $\|\cdot\|_D = \sum_{j=1}^P \|\cdot_j\|_{*,\mathrm{K}_j}$, which in the BM factorization amounts to having the constraint $\mathbf{K}_j\mathbf{U}_j \geq \mathbf{0}$. We express the BM factorization of convex MLPs below.

$$p^*_{LMLP} = \min_{\substack{\mathbf{U}\in\mathbb{R}^{d\times m}\\ \mathbf{V}\in\mathbb{R}^{c\times m}}} F(\mathbf{X}\mathbf{U}\mathbf{V}^\top) + \frac{\beta}{2}\left(\|\mathbf{U}\|_F^2 + \|\mathbf{V}\|_F^2\right) \tag{17}$$

$$p^*_{GMLP} = \min_{\substack{\mathbf{U}_j\in\mathbb{R}^{d\times m}\\ \mathbf{V}_j\in\mathbb{R}^{c\times m}}} F(\sum_{j=1}^P \mathbf{D}_j\mathbf{X}\mathbf{U}_j\mathbf{V}_j^\top) + \frac{\beta}{2}\sum_{j=1}^P\left(\|\mathbf{U}_j\|_F^2 + \|\mathbf{V}_j\|_F^2\right) \tag{18}$$

$$p^*_{RMLP} = \min_{\substack{\mathbf{U}_j\in\mathbb{R}^{d\times m}:(2\mathbf{D}_j-\mathbf{I}_n)\mathbf{X}\mathbf{U}_j\geq\mathbf{0}\\ \mathbf{V}_j\in\mathbb{R}^{c\times m}}} F(\sum_{j=1}^P \mathbf{D}_j\mathbf{X}\mathbf{U}_j\mathbf{V}_j^\top) + \frac{\beta}{2}\sum_{j=1}^P\left(\|\mathbf{U}_j\|_F^2 + \|\mathbf{V}_j\|_F^2\right) \tag{19}$$

To the best of our knowledge, (19) presents the first application of BM factorization to a non-linear neural network, which is enabled by the convex model (3).

In the linear case, the BM factorization (17) is identical to the original non-convex formulation of a linear MLP with m neurons. In the case of gated ReLU, the BM factorization when $m = 1$ is equivalent to the original non-convex formulation. However, for ReLU activation two-layer networks, the BM factorization even when $m = 1$ corresponds to a different (i.e. constrained, rather than ReLU activation) model than the non-convex formulation. While the original convex program is NP-hard due to the quasi-nuclear norm (Sahiner et al., 2021b), the per-iteration complexity of the BM factorization is much lower than for the convex ReLU MLP.

The BM factorizations of these convex MLPs are non-convex, hence finding a global minimum appears intractable. However, the following theorem demonstrates that as long as a rank-deficient local minimum to the BM factorization is obtained, it corresponds to a global optimum.

Theorem 3.3. *If $m \geq d_c + d_r$, all local minima of the BM factorization (14) are global minima. Furthermore, if F is twice-differentiable, any rank-deficient local minimum $\hat{\mathbf{R}} := \begin{bmatrix}\hat{\mathbf{U}}\\ \hat{\mathbf{V}}\end{bmatrix}$ of (14) corresponds to a global minimizer $\mathbf{Z}^* = \hat{\mathbf{U}}\hat{\mathbf{V}}^\top$ of (16).*

This result demonstrates that these two-layer convex MLPs have no spurious local minima under mild conditions. However, there remains an algorithmic challenge: it is not straightforward to obtain a guaranteed local minima when the gradients of f are not Lipschitz continuous. The following result provides a general condition under which stationary points of the (14) are global optima of (16).

Theorem 3.4. *For any non-negative objective function F, for a stationary $(\hat{\mathbf{U}}, \hat{\mathbf{V}})$ of (14) with corresponding $\hat{\mathbf{Z}} = \hat{\mathbf{U}}\hat{\mathbf{V}}^\top$ with objective $\hat{p}$ for (16), the relative optimality gap $\frac{\hat{p}-p^*}{p^*}$ satisfies*

$$\frac{\hat{p}-p^*}{p^*} \leq \left(\frac{\|\nabla_{\mathbf{Z}}F(\mathbf{M}\hat{\mathbf{Z}})\|_D^*}{\beta} - 1\right)_+ \tag{20}$$

where $\|\cdot\|_D^*$ is the dual norm of $\|\cdot\|_D$.

This bound can be calculated by taking the gradient of the unregularized objective function, evaluated at candidate solution $\hat{\mathbf{Z}}$ to the convex problem (16), which is formed by the stationary point of BM problem (14). In the case of a linear MLP with $\mathbf{X} = \mathbf{I}_d$, F a squared-loss objective, and $\|\nabla_{\mathbf{Z}}F(\mathbf{M}\hat{\mathbf{Z}})\|_D^* \leq \beta$, our result exactly recovers the result of Theorem 2.1 from Mardani et al. (2013). When this condition is not exactly satisfied, (20) provides a novel result in the form of an optimality gap bound. To our knowledge, this is the first result that generalizes the optimality conditions for stationary points from any BM factorization of a neural network. This provides an easily computable bound after solving (14) which quantifies how close a solution is to the global minimum. In the case of a ReLU MLP, the relative optimality gap is given by

$$\frac{\hat{p}-p^*}{p^*} \leq \left(\max_{\substack{j\in[P]\\ \mathbf{u}\in\mathcal{B}_2\\ \mathbf{K}_j\mathbf{u}\geq 0}} \frac{1}{\beta}\|\nabla_{\mathbf{Z}_j}F(\sum_{j'=1}^P \mathbf{D}_{j'}\mathbf{X}\hat{\mathbf{Z}}_{j'})\mathbf{u}\|_2 - 1\right)_+ .$$

Computing this quantity requires solving a cone-constrained PCA problem (Deshpande et al., 2014). In certain cases, the optimality gap of stationary points (20) is always zero.

Theorem 3.5. *A stationary point* $(\hat{\mathbf{U}}, \hat{\mathbf{V}})$ *of (14) is a global minimizer of (16) if* $R = C = 2$ *and*

$$\text{rank}(\hat{\mathbf{U}}) = \text{rank}(\hat{\mathbf{V}}) = \min\{d_c, d_r\}. \tag{21}$$

Thus, for linear and gated ReLU MLPs, we can ensure that if the Burer-Monteiro factorization achieves a stationary point with full rank, it is corresponds with the global optimum of the convex program. We now can further extend these results to CNNs and self-attention architectures.

3.2 CNNs

Before proceeding to explore the BM factorization in the context of two-layer CNNs, we first provide a new result on an equivalent convex program for two-layer ReLU CNNs with arbitrary linear pooling operations, which extends the results of Sahiner et al. (2021b); Ergen & Pilanci (2020) on Global Average Pooling CNNs. Define $\mathbf{P}_a \in \mathbb{R}^{a \times K}$ to be a linear pooling matrix which pools the K spatial dimensions to an arbitrary size a. Then, we express the non-convex two-layer CNN problem as

$$p^*_{CNN} := \min_{\substack{\mathbf{w}_{1j} \in \mathbb{R}^h \\ \mathbf{W}_{2j} \in \mathbb{R}^{c \times a}}} \sum_{i=1}^n F\left(\sum_{j=1}^m \mathbf{W}_{2j} \mathbf{P}_a \sigma(\mathbf{X}_i \mathbf{w}_{1j})\right) + \frac{\beta}{2} \sum_{j=1}^m \|\mathbf{w}_{1j}\|_2^2 + \|\mathbf{W}_{2j}\|_F^2. \tag{22}$$

Theorem 3.6. *For* $\beta > 0$ *and ReLU activation* $\sigma(\cdot) = (\cdot)_+$*, if* $m \geq m^*$ *where* $m^* \leq nac$*, then (22) is equivalent to a convex optimization problem, given by*

$$p^*_{CNN} = \min_{\mathbf{Z}_k \in \mathbb{R}^{h \times ac}} \sum_{i=1}^n F\left(\sum_{k=1}^P \begin{bmatrix} \text{trace}(\mathbf{P}_a \mathbf{D}_k^{(i)} \mathbf{X}_i \mathbf{Z}_k^{(1)}) \\ \vdots \\ \text{trace}(\mathbf{P}_a \mathbf{D}_k^{(i)} \mathbf{X}_i \mathbf{Z}_k^{(c)}) \end{bmatrix}\right) + \beta \sum_{k=1}^P \|\mathbf{Z}_k\|_{*, \mathrm{K}_k}, \tag{23}$$

$$\mathbf{K}_k := (2\mathbf{D}_k - \mathbf{I}_{nK}) \begin{bmatrix} \mathbf{X}_1 \\ \cdots \\ \mathbf{X}_n \end{bmatrix}, \ \mathbf{Z}_k^{(c')} \in \mathbb{R}^{h \times a} \ \forall c' \in [c].$$

Thus, we provide a novel result which characterizes two-layer CNNs with arbitrary linear pooling operations as a convex program. Similar results can be shown for the linear and gated-ReLU activation cases[2]. With this established, we present our main results on the BM factorization for CNNs.

Lemma 3.7. *The BM factorization of the convex CNN problem with ReLU activation is given as:*

$$p^*_{RCNN} = \min_{\substack{\mathbf{u}_{jk} \in \mathbb{R}^h \\ \mathbf{V}_{jk} \in \mathbb{R}^{c \times a}}} \sum_{i=1}^n F\left(\sum_{k=1}^P \sum_{j=1}^m \mathbf{V}_{jk} \mathbf{P}_a \mathbf{D}_k^{(i)} \mathbf{X}_i \mathbf{u}_{jk}\right) + \frac{\beta}{2} \sum_{k=1}^P \sum_{j=1}^m \left(\|\mathbf{u}_{jk}\|_F^2 + \|\mathbf{V}_{jk}\|_F^2\right),$$
$$\text{s.t. } (2\mathbf{D}_k^{(i)} - \mathbf{I})\mathbf{X}_i \mathbf{u}_{jk} \geq 0 \, \forall j \in [m], \forall k \in [P] \tag{24}$$

The BM factorization closely resembles the original non-convex formulation (22). Generally, (24) inherits the results of Theorems 3.3, 3.4, and 3.5; we present one such corollary here.

Corollary 3.8. *A stationary point* $((\hat{\mathbf{u}}_{jk}, \hat{\mathbf{V}}_{jk})_{j=1}^m)_{k=1}^P$ *of (24) corresponds to a global minimizer* $\hat{\mathbf{Z}}_k = \sum_{j=1}^m \hat{\mathbf{u}}_{jk} \text{vec}\left(\hat{\mathbf{V}}_{jk}\right)^\top$ *of (23) provided that*

$$\|\sum_{i=1}^n \nabla_{\mathbf{Z}_k} F\left(\sum_{k'=1}^P \begin{bmatrix} \text{trace}(\mathbf{P}_a \mathbf{D}_{k'}^{(i)} \mathbf{X}_i \mathbf{Z}_{k'}^{(1)}) \\ \vdots \\ \text{trace}(\mathbf{P}_a \mathbf{D}_{k'}^{(i)} \mathbf{X}_i \mathbf{Z}_{k'}^{(c)}) \end{bmatrix}\right) \mathbf{u}\|_2 \leq \beta, \quad \forall k \in [P], \forall \mathbf{u} \in \mathcal{B}_2 : (2\mathbf{D}_k^{(i)} - \mathbf{I})\mathbf{X}_i \mathbf{u} \geq 0. \tag{25}$$

[2]We examine linear and gated ReLU activations for CNNs in Appendix B.2.

3.3 Multi-Head Self-Attention

We now for the first time extend BM factorization theory to self-attention networks.

Lemma 3.9. *The BM factorization of the convex linear-activation*[3] *self-attention problem is given as:*

$$p^*_{LSA} = \min_{\substack{\mathbf{U}_j \in \mathbb{R}^{d\times d} \\ \mathbf{V}_j \in \mathbb{R}^{d\times c}}} \sum_{i=1}^{n} F\left(\sum_{j=1}^{m} \mathbf{X}_i \mathbf{U}_j \mathbf{X}_i^\top \mathbf{X}_i \mathbf{V}_j\right) + \frac{\beta}{2}\sum_{j=1}^{m} \|\mathbf{U}_j\|_F^2 + \|\mathbf{V}_j\|_F^2. \tag{26}$$

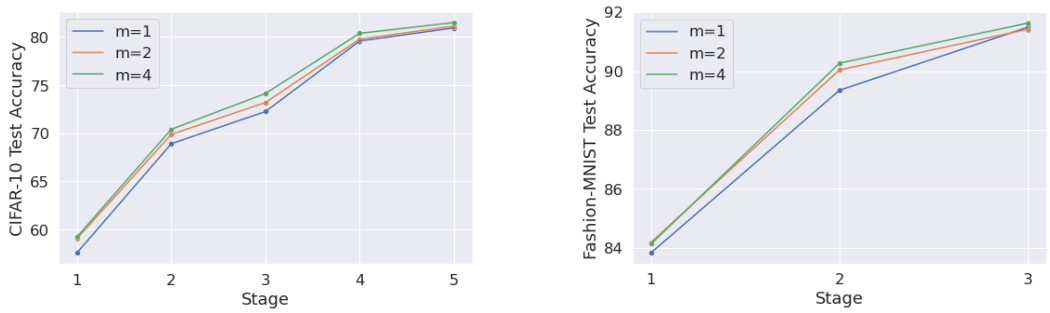
In addition to inheriting all of the results of Theorems 3.3, 3.4, and 3.5, noting the equivalence of the BM factorization with the original non-convex program (8), we are the first to show conditions under which there are no spurious local minima for self-attention networks.

Corollary 3.10. *The linear-activation self-attention network (26) has no spurious local minima as long as the number of heads satisfies $m \geq m^*$ where $m^* \leq d^2 + dc$. Furthermore, for any twice-differentiable objective F, if for any local minimum $(\hat{\mathbf{U}}_j, \hat{\mathbf{V}}_j)_{j=1}^m$ of (26), the matrix*

$$\hat{\mathbf{R}} := \begin{bmatrix} \mathrm{vec}(\hat{\mathbf{U}}_1) & \cdots & \mathrm{vec}(\hat{\mathbf{U}}_m) \\ \mathrm{vec}(\hat{\mathbf{V}}_1) & \cdots & \mathrm{vec}(\hat{\mathbf{V}}_m) \end{bmatrix} \in \mathbb{R}^{d(d+c)\times m} \tag{27}$$

is rank-deficient, then this local minimum is also a global minimum of (104).

4 Experimental Results



(a) CIFAR-10 (b) Fashion-MNIST

Figure 1: BM enables layerwise training of convex ReLU CNNs, which are competitive with end-to-end ReLU networks of the same depth. For CIFAR-10, with $m = 4$ we achieve a test accuracy of 81.5% compared to 81.6% for end-to-end non-convex training, and for Fashion-MNIST, with $m = 4$ we achieve a test accuracy of 91.6% compared to 91.2% for non-convex training (Kiliçarslan & Celik, 2021; Bhatnagar et al., 2017).

4.1 The Relative Optimality Gap Bound

In this section, we illustrate the utility of our proposed relative optimality bound for stationary points in the setting of two-layer fully-connected networks. We also seek to examine how this bound changes with respect to the number of samples n, the regularization parameter β (which controls the sparsity of the convex solution), and the number of factors in the BM factorization m. We initialize a class-balanced three-class spiral data set with varied number of samples n (see Figure 3 in Appendix C.1 for examples). For this dataset, we then train the gated ReLU MLP BM factorization (18) with varying number of factors m. We then compare the stationary points of these BM factorizations found by gradient descent (GD) to the global optimum, which we compute from (5).

For each stationary point of the BM factorization, we compute the relative optimality gap bound provided in our result in Theorem 3.4. We note that since $d = 2$, $c = 3$ in this case, for all j, as long as $m \geq 5$ all local minima of the BM factorization are global minima (Burer & Monteiro, 2005; Haeffele et al., 2014). While Lee et al. (2016) demonstrated that gradient descent

[3] We examine gated ReLU and ReLU activations for self-attention in Appendix B.3.

with a random initialization converges to a local optimum almost surely for losses whose gradient is Lipschitz continuous, we use squared loss with one-hot-encoded class labels, which is not Lipschitz continuous (Mukkamala & Ochs, 2019). Thus, there is no guarantee that GD will find the global minimum. We display results over β in Figure 2. Our bound gives a useful proxy for whether the BM factorization converges to the global minimum. For larger values of β, it becomes much easier for GD to find an optimal solution, but GD almost never finds the global minimum.

We find that GD applied to the BM factorization finds saddle points that are not quite local minima, but close. Interestingly, there is only a minor relationship between the optimality gap and the rank of the BM factorization m. This experiment further validates the need to consider stationary points of the BM factorization, rather than just local minima.

4.2 BM Enables Layerwise Training of Convex CNNs

We consider the task of leveraging the theory of two-layer convex ReLU neural networks for training deep image classifiers. Following the approach of (Belilovsky et al., 2019), we seek to train two-layer convex CNNs greedily to mimick the performance of a deep network. In the non-convex setting, the greedy approach proceeds by training a single two-layer CNN, then freezing the weights of this CNN, using the latent representation of this CNN as the features for another two-layer CNN, and repeating this process for a specified number of stages. We leverage the result of Theorem 3.6 to convert this non-convex layerwise training procedure to a convex one, training stages of convex two-layer gated ReLU CNNs. We apply this procedure to the CIFAR-10 (Krizhevsky et al., 2009) and Fashion-MNIST (Xiao et al., 2017) datasets, using the architecture of (Belilovsky et al., 2019) (see Appendix C.2).

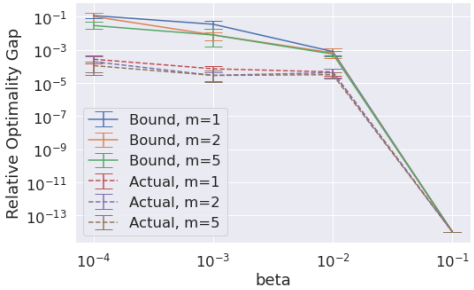


Figure 2: Relative optimality gap of the non-convex BM factorization of a gated-ReLU two-layer MLP for three-class spiral data classification ($n = 150$, $d = 2$, $c = 3$). We demonstrate how β and m affect relative optimality gap, both in in terms of the proposed bound and the actual gap, where the global minimum is found by convex optimization.

In a memory-limited setting, layerwise training with the convex model equation 23 is impossible, because the latent representation to be used as input for the second stage, given by $\{\{\mathbf{D}_j^{(i)}\mathbf{X}_i\mathbf{Z}_j^{(c')})\}_{j=1}^P\}_{c'=1}^c$, has Pac channels, which for reasonable choices of $P = 256$, $a = 4$, $c = 10$ yields upwards of 10^4 channels for the input to the second CNN stage. Accordingly, we employ the BM factorization of size m, so the latent representation only consists of mP channels.

Figure 1 demonstrates that this BM scheme for layerwise training allows for performance to improve one stage the next of the layerwise training procedure, reaching the performance of much deeper networks while enabling a convex optimization procedure. Training five stages of a BM factorized convex two-layer gated ReLU CNN on CIFAR-10 resulted in a final test accuracy of 80.9%, 81.1%, and 81.5% for $m \in [1, 2, 4]$ respectively. Previously, it has been demonstrated that a six-layer ReLU CNN achieves 81.6% on CIFAR-10 when trained end-to-end (Kiliçarslan & Celik, 2021). Interestingly, we find that increasing m generally improves performance. The three-stage trained BM factorized convex two-layer gated ReLU CNN on Fasion-MNIST achieved a final test accuracy of 91.5%, 91.4%, and 91.6% for $m \in [1, 2, 4]$ respectively, compared to 91.2% for a four-layer ReLU CNN trained end-to-end (Bhatnagar et al., 2017). On this dataset, we observe that the impact of increasing m on performance is less pronounced than for CIFAR-10.

The BM factorization is essential for convex neural networks to match deep ReLU networks. Without the BM factorization, the induced regularizer of convex CNNs is intractable to compute, and the latent representation used for layerwise learning is prohibitively large. While inheriting the guarantees of Theorems 3.3, 3.4, and 3.5, these layerwise trained BM networks match the performance of end-to-end, highly non-convex ReLU deep networks.

REFERENCES

Erling D Andersen and Knud D Andersen. The mosek interior point optimizer for linear programming: an implementation of the homogeneous algorithm. In *High performance optimization*, pp. 197–232. Springer, 2000.

Sanjeev Arora, Simon Du, Wei Hu, Zhiyuan Li, and Ruosong Wang. Fine-grained analysis of optimization and generalization for overparameterized two-layer neural networks. In *International Conference on Machine Learning*, pp. 322–332. PMLR, 2019.

Francis Bach, Julien Mairal, and Jean Ponce. Convex sparse matrix factorizations. *arXiv preprint arXiv:0812.1869*, 2008.

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

Burak Bartan and Mert Pilanci. Neural spectrahedra and semidefinite lifts: Global convex optimization of polynomial activation neural networks in fully polynomial-time. *arXiv preprint arXiv:2101.02429*, 2021.

Eugene Belilovsky, Michael Eickenberg, and Edouard Oyallon. Greedy layerwise learning can scale to imagenet. In *International conference on machine learning*, pp. 583–593. PMLR, 2019.

Shobhit Bhatnagar, Deepanway Ghosal, and Maheshkumar H Kolekar. Classification of fashion article images using convolutional neural networks. In *2017 Fourth International Conference on Image Information Processing (ICIIP)*, pp. 1–6. IEEE, 2017.

Nicolas Boumal, Vlad Voroninski, and Afonso Bandeira. The non-convex burer-monteiro approach works on smooth semidefinite programs. *Advances in Neural Information Processing Systems*, 29, 2016.

Nicolas Boumal, Vladislav Voroninski, and Afonso S Bandeira. Deterministic guarantees for burer-monteiro factorizations of smooth semidefinite programs. *Communications on Pure and Applied Mathematics*, 73(3):581–608, 2020.

Samuel Burer and Renato DC Monteiro. A nonlinear programming algorithm for solving semidefinite programs via low-rank factorization. *Mathematical Programming*, 95(2):329–357, 2003.

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

Ricardo Cabral, Fernando De la Torre, João P Costeira, and Alexandre Bernardino. Unifying nuclear norm and bilinear factorization approaches for low-rank matrix decomposition. In *Proceedings of the IEEE international conference on computer vision*, pp. 2488–2495, 2013.

Emmanuel J Candès and Benjamin Recht. Exact matrix completion via convex optimization. *Foundations of Computational mathematics*, 9(6):717–772, 2009.

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

Diego Cifuentes and Ankur Moitra. Polynomial time guarantees for the burer-monteiro method. *arXiv preprint arXiv:1912.01745*, 2019.

Hadi Daneshmand, Jonas Kohler, Aurelien Lucchi, and Thomas Hofmann. Escaping saddles with stochastic gradients. In *International Conference on Machine Learning*, pp. 1155–1164. PMLR, 2018.

Yash Deshpande, Andrea Montanari, and Emile Richard. Cone-constrained principal component analysis. *Advances in Neural Information Processing Systems*, 27, 2014.

Steven Diamond and Stephen Boyd. Cvxpy: A python-embedded modeling language for convex optimization. *The Journal of Machine Learning Research*, 17(1):2909–2913, 2016.

Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp. *arXiv preprint arXiv:1605.08803*, 2016.

Simon Du and Jason Lee. On the power of over-parametrization in neural networks with quadratic activation. In *International conference on machine learning*, pp. 1329–1338. PMLR, 2018.

Simon Du, Jason Lee, Yuandong Tian, Aarti Singh, and Barnabas Poczos. Gradient descent learns one-hidden-layer cnn: Don't be afraid of spurious local minima. In *International Conference on Machine Learning*, pp. 1339–1348. PMLR, 2018.

Simon Du, Jason Lee, Haochuan Li, Liwei Wang, and Xiyu Zhai. Gradient descent finds global minima of deep neural networks. In *International conference on machine learning*, pp. 1675–1685. PMLR, 2019.

Simon S Du, Chi Jin, Jason D Lee, Michael I Jordan, Aarti Singh, and Barnabas Poczos. Gradient descent can take exponential time to escape saddle points. *Advances in neural information processing systems*, 30, 2017.

Murat A Erdogdu, Asuman Ozdaglar, Pablo A Parrilo, and Nuri Denizcan Vanli. Convergence rate of block-coordinate maximization burer–monteiro method for solving large sdps. *Mathematical Programming*, pp. 1–39, 2021.

Tolga Ergen and Mert Pilanci. Implicit convex regularizers of cnn architectures: Convex optimization of two-and three-layer networks in polynomial time. In *International Conference on Learning Representations*, 2020.

Tolga Ergen, Arda Sahiner, Batu Ozturkler, John M Pauly, Morteza Mardani, and Mert Pilanci. Demystifying batch normalization in relu networks: Equivalent convex optimization models and implicit regularization. In *International Conference on Learning Representations*, 2021.

Jonathan Fiat, Eran Malach, and Shai Shalev-Shwartz. Decoupling gating from linearity. *arXiv preprint arXiv:1906.05032*, 2019.

Rong Ge, Furong Huang, Chi Jin, and Yang Yuan. Escaping from saddle points—online stochastic gradient for tensor decomposition. In *Conference on learning theory*, pp. 797–842. PMLR, 2015.

Rong Ge, Chi Jin, and Yi Zheng. No spurious local minima in nonconvex low rank problems: A unified geometric analysis. In *International Conference on Machine Learning*, pp. 1233–1242. PMLR, 2017.

Nicolas Gillis. Introduction to nonnegative matrix factorization. *arXiv preprint arXiv:1703.00663*, 2017.

Benjamin Haeffele, Eric Young, and Rene Vidal. Structured low-rank matrix factorization: Optimality, algorithm, and applications to image processing. In *International conference on machine learning*, pp. 2007–2015. PMLR, 2014.

Benjamin D Haeffele and René Vidal. Global optimality in neural network training. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7331–7339, 2017.

Chi Jin, Rong Ge, Praneeth Netrapalli, Sham M Kakade, and Michael I Jordan. How to escape saddle points efficiently. In *International Conference on Machine Learning*, pp. 1724–1732. PMLR, 2017.

Kenji Kawaguchi. Deep learning without poor local minima. *Advances in neural information processing systems*, 29, 2016.

Serhat Kiliçarslan and Mete Celik. Rsigelu: A nonlinear activation function for deep neural networks. *Expert Systems with Applications*, 174:114805, 2021.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Jason D Lee, Max Simchowitz, Michael I Jordan, and Benjamin Recht. Gradient descent only converges to minimizers. In *Conference on learning theory*, pp. 1246–1257. PMLR, 2016.

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

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

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

Aaron Mishkin, Arda Sahiner, and Mert Pilanci. Fast convex optimization for two-layer relu networks: Equivalent model classes and cone decompositions. In *Proceedings of the 39th International Conference on Machine Learning*, 2022.

Mahesh Chandra Mukkamala and Peter Ochs. Beyond alternating updates for matrix factorization with inertial bregman proximal gradient algorithms. *Advances in Neural Information Processing Systems*, 32, 2019.

Dohyung Park, Anastasios Kyrillidis, Constantine Carmanis, and Sujay Sanghavi. Non-square matrix sensing without spurious local minima via the burer-monteiro approach. In *Artificial Intelligence and Statistics*, pp. 65–74. PMLR, 2017.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32: 8026–8037, 2019.

Mert Pilanci and Tolga Ergen. 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.

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

Jasson DM Rennie and Nathan Srebro. Fast maximum margin matrix factorization for collaborative prediction. In *Proceedings of the 22nd international conference on Machine learning*, pp. 713–719, 2005.

Arda Sahiner, Tolga Ergen, Batu Ozturkler, Burak Bartan, John M Pauly, Morteza Mardani, and Mert Pilanci. Hidden convexity of wasserstein gans: Interpretable generative models with closed-form solutions. In *International Conference on Learning Representations*, 2021a.

Arda Sahiner, Tolga Ergen, John M Pauly, and Mert Pilanci. Vector-output relu neural network problems are copositive programs: Convex analysis of two layer networks and polynomial-time algorithms. In *ICLR*, 2021b.

Arda Sahiner, Morteza Mardani, Batu Ozturkler, Mert Pilanci, and John M Pauly. Convex regularization behind neural reconstruction. In *ICLR*, 2021c.

Arda Sahiner, Tolga Ergen, Batu Ozturkler, John Pauly, Morteza Mardani, and Mert Pilanci. Unraveling attention via convex duality: Analysis and interpretations of vision transformers. In *Proceedings of the 39th International Conference on Machine Learning*, 2022.

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

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

Ju Sun, Qing Qu, and John Wright. When are nonconvex problems not scary? *arXiv preprint arXiv:1510.06096*, 2015.

Irene Waldspurger and Alden Waters. Rank optimality for the burer–monteiro factorization. *SIAM journal on Optimization*, 30(3):2577–2602, 2020.

Lingxiao Wang, Xiao Zhang, and Quanquan Gu. A unified computational and statistical framework for nonconvex low-rank matrix estimation. In *Artificial Intelligence and Statistics*, pp. 981–990. PMLR, 2017.

Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.

Tian Ye and Simon S Du. Global convergence of gradient descent for asymmetric low-rank matrix factorization. *Advances in Neural Information Processing Systems*, 34, 2021.

Gavin Zhang, Salar Fattahi, and Richard Y Zhang. Preconditioned gradient descent for overparameterized nonconvex burer-monteiro factorization with global optimality certification. *J. Mach. Learn. Res.*, 24:163–1, 2023.

Richard Y Zhang. Improved global guarantees for the nonconvex burer–monteiro factorization via rank overparameterization. *arXiv preprint arXiv:2207.01789*, 2022.

Yuchen Zhang, Jason D Lee, and Michael I Jordan. l1-regularized neural networks are improperly learnable in polynomial time. In *International Conference on Machine Learning*, pp. 993–1001. PMLR, 2016.

Yuchen Zhang, Percy Liang, and Martin J Wainwright. Convexified convolutional neural networks. In *International Conference on Machine Learning*, pp. 4044–4053. PMLR, 2017.

Qinqing Zheng and John Lafferty. Convergence analysis for rectangular matrix completion using burer-monteiro factorization and gradient descent. *arXiv preprint arXiv:1605.07051*, 2016.

Zhihui Zhu, Qiuwei Li, Gongguo Tang, and Michael B Wakin. The global optimization geometry of low-rank matrix optimization. *arXiv preprint arXiv:1703.01256*, 2017.

Liu Ziyin, Botao Li, James B Simon, and Masahito Ueda. Sgd can converge to local maxima. In *International Conference on Learning Representations*, 2021.

A PROOFS

A.1 COMPLEXITY AND CONVERGENCE RESULTS OF THE BM FACTORIZATION

In this subsection, we provide some additional details on the complexity and convergence of the BM factorization (19). We consider the problem dimensions of $\mathbf{X} \in \mathbb{R}^{n \times d}$, BM factorization size m, and network output dimension c.

First, we can compare the per-iteration time complexity of a naive implementation of projected gradient descent on (19) to the the per-iteration time complexity of a Frank-Wolfe algorithm designed to solve (3) as proposed in Sahiner et al. (2021b). Consider the case of squared loss, e.g. $F(\hat{\mathbf{Y}}, \mathbf{Y}) = \frac{1}{2}\|\hat{\mathbf{Y}} - \mathbf{Y}\|_F^2$. For $r := \operatorname{rank}(\mathbf{X})$, it is shown in Appendix A.4.2 of Sahiner et al. (2021b) that a single iteration of Frank-Wolfe has time complexity $\mathcal{O}(Pn^r)$, where P is the number of hyperplane arrangements.

In comparison, projected gradient descent applied to (19) takes two steps:

1. Gradient descent steps on $\{(\mathbf{U}_j, \mathbf{V}_j)\}_{j=1}^P$. The gradient of the objective of (19) with respect to $\mathbf{U}_j$ is given by
$$(\mathbf{D}_j\mathbf{X})^\top \left(\sum_{j'=1}^{P} \mathbf{D}_{j'}\mathbf{X}\mathbf{U}_{j'}\mathbf{V}_{j'}^\top - \mathbf{Y} \right) \mathbf{V}_j + \beta \mathbf{U}_j. \tag{28}$$
Noting that the residual $\mathbf{R} := \sum_{j'=1}^{P} \mathbf{D}_{j'}\mathbf{X}\mathbf{U}_{j'}\mathbf{V}_{j'}^\top - \mathbf{Y}$ can be computed once in $\mathcal{O}(P(n^2c + ndc + dcm))$ time and re-used in all gradient computations, the total complexity of computing all gradients and taking a gradient descent step is given by $\mathcal{O}\left(P(n^2(d+c) + ndc + dcm)\right)$.
2. A projection step of $\mathbf{U}_j$ onto the space $(2\mathbf{D}_j - \mathbf{I})\,\mathbf{X}\mathbf{U}_j \geq 0$ for each $j \in [P]$– this naively can be solved in $\mathcal{O}(P(n^3 + n^2dm))$ time.

Overall, then, using projected gradient descent on (19) is of time complexity $\mathcal{O}(P(n^3 + n^2(d + c + dm) + dcm))$, which unless m is exponential in r is significantly more computationally efficient than solving (3) with Frank-Wolfe in $\mathcal{O}(Pn^r)$ time.

For the convergence rates of naive gradient descent applied to BM factorizations (17), (18), and (19), we point to Zhang et al. (2023), which suggests that these formulations may converge sub-linearly to a stationary point in the case that m exceeds the rank of the optimal solution $\mathbf{Z}^*$. However, Zhang et al. (2023) suggest a pre-conditioning method to gradient descent applied to BM formulations which guarantees linear convergence and can be applied here.

A.2 RESULT OF (BURER & MONTEIRO, 2005) AND ITS APPLICATIONS TO RECTANGULAR MATRICES

In this subsection, we outline the precise theoretical statement of (Burer & Monteiro, 2005) and describe exactly how it corresponds to our summary in Section 2.2, and thus the application to the later derivations in our work. We first describe the following result, without proof, from (Burer & Monteiro, 2005).

Lemma A.1 (Lemma 2.1 of (Burer & Monteiro, 2005))**.** *Suppose* $\mathbf{R} \in \mathbb{R}^{(d+c)\times r}$, $\mathbf{S} \in \mathbb{R}^{(d+c)\times r}$ *satisfy* $\mathbf{R}\mathbf{R}^\top = \mathbf{S}\mathbf{S}^\top$. *Then,* $\mathbf{S} = \mathbf{R}\mathbf{Q}$ *for some orthogonal* $\mathbf{Q} \in \mathbb{R}^{r\times r}$.

Now, we proceed to prove analogs of Theorem 2.3 of (Burer & Monteiro, 2005) for general SDPs with a rank constraint.

Lemma A.2 (Analog of Lemma 2.2 of (Burer & Monteiro, 2005))**.** *Consider the problem*
$$\min_{\mathbf{R}\in\mathbb{R}^{(d+c)\times r}} F'(\mathbf{R}\mathbf{R}^\top). \tag{29}$$
$\hat{\mathbf{R}}$ *is a local minimum of (29) if and only if* $\hat{\mathbf{R}}\mathbf{Q}$ *is a local minimum of (29) for all orthogonal* $\mathbf{Q} \in \mathbb{R}^{r\times r}$.

Proof. Since $\mathbf{Q}$ is orthogonal, $(\hat{\mathbf{R}}\mathbf{Q})(\hat{\mathbf{R}}\mathbf{Q})^\top = \hat{\mathbf{R}}\mathbf{Q}\mathbf{Q}^\top\hat{\mathbf{R}}^\top = \hat{\mathbf{R}}\hat{\mathbf{R}}^\top$. Thus, $\hat{\mathbf{R}}' := \hat{\mathbf{R}}\mathbf{Q}$ attains the same objective value, gradients, and higher order derivatives as $\hat{\mathbf{R}}$. Thus, $\hat{\mathbf{R}}$ is a local minimum if and only if $\hat{\mathbf{R}}'$ is a local minimum. □

Theorem A.3 (Analog of Theorem 2.3 of (Burer & Monteiro, 2005))**.** *Consider the following two problems.*

$$\min_{\substack{\mathbf{X}\in\mathbb{R}^{(d+c)\times(d+c)}\\ \mathbf{X}\succeq 0\\ \operatorname{rank}(\mathbf{X})\leq r}} F'(\mathbf{X}), \tag{30}$$

$$\min_{\mathbf{R}\in\mathbb{R}^{(d+c)\times r}} F'(\mathbf{R}\mathbf{R}^\top). \tag{31}$$

Then, for any continuous function F', a feasible solution $\hat{\mathbf{X}}$ is a local minimizer of (30) if and only if, for $\hat{\mathbf{X}} = \hat{\mathbf{R}}\hat{\mathbf{R}}^\top$, $\hat{\mathbf{R}}$ is a local minimizer of (31).

Proof. We follow the exact same lines as (Burer & Monteiro, 2005). By continuity of the map $\mathbf{R} \to \mathbf{R}\mathbf{R}^\top$, we know that if $\hat{\mathbf{X}}$ is a local minimizer of (30), then $\hat{\mathbf{R}}$ is a local minimizer of (31). Now, we must prove the other direction, namely that if $\hat{\mathbf{X}} = \hat{\mathbf{R}}\hat{\mathbf{R}}^\top$ is not local minimizer of (30), then $\hat{\mathbf{R}}$ is not a local minimizer of (31).

Suppose that $\hat{\mathbf{X}}$ is not a local minimum. By continuity of F', then, there must be a sequence of feasible solutions $\{\mathbf{X}^k\}$ of (30) converging to $\hat{\mathbf{X}}$ such that $F'(\mathbf{X}^k) < F'(\hat{\mathbf{X}})$ for all k. For each k, choose $\mathbf{R}^k$ such that $\mathbf{X}^k = \mathbf{R}^k{\mathbf{R}^k}^\top$. Since $\{\mathbf{X}^k\}$ is bounded, it follows that $\{\mathbf{R}^k\}$ is bounded and hence has a subsequence $\{\mathbf{R}^k\}_{k\in\mathcal{K}}$ converging to some $\mathbf{R}$ such that $\hat{\mathbf{X}} = \mathbf{R}\mathbf{R}^\top$. Since $F'(\mathbf{R}^k{\mathbf{R}^k}^\top) = F'(\mathbf{X}^k) < F'(\hat{\mathbf{X}}) = F'(\mathbf{R}\mathbf{R}^\top)$, we see that $\mathbf{R}$ is not a local minimum of (31). Using the fact that $\hat{\mathbf{X}} = \hat{\mathbf{R}}\hat{\mathbf{R}}^\top = \mathbf{R}\mathbf{R}^\top$ together with Lemmas A.1 and A.2, we conclude that $\hat{\mathbf{R}}$ is not a local minimum of (31). □

With this established, we now describe the setting described in Section 2.2, i.e. the rectangular matrix, non-SDP case.

Lemma A.4. *Consider the optimization problems*

$$p_1^* := \min_{\substack{\mathbf{X}\in\mathbb{R}^{(d+c)\times(d+c)}\\ \mathbf{X}\succeq 0\\ \operatorname{rank}(\mathbf{X})\leq m^*}} F'(\mathbf{X}), \tag{32}$$

and

$$p_2^* := \min_{\mathbf{Z}\in\mathbb{R}^{d\times c}} F(\mathbf{Z}), \tag{33}$$

where F is convex. Define $F' : \mathbb{R}^{(d+c)\times(d+c)} \to \mathbb{R}$ such that

$$F'(\begin{bmatrix}\mathbf{X}_1 & \mathbf{X}_2\\ \mathbf{X}_3 & \mathbf{X}_4\end{bmatrix}) := F(\mathbf{X}_2) \tag{34}$$

for $\mathbf{X}_1 \in \mathbb{R}^{d\times d}$, $\mathbf{X}_2 \in \mathbb{R}^{d\times c}$, $\mathbf{X}_3 \in \mathbb{R}^{c\times d}$, $\mathbf{X}_4 \in \mathbb{R}^{c\times c}$. Then, any local minimizer $\hat{\mathbf{X}}$ to (32) corresponds a global minimizer $\mathbf{Z}^* = \hat{\mathbf{X}}_2$ of (33) for some $m^* \leq d + c$.

Proof. This follows from Boumal et al. (2020), Corollary 3.2. □

Lemma A.5 (Used in Section 2.2)**.** *For any convex, continuous function F, a local minimum $(\hat{\mathbf{U}}, \hat{\mathbf{V}})$ of*

$$\min_{\mathbf{U}\in\mathbb{R}^{d\times m},\mathbf{V}\in\mathbb{R}^{c\times m}} F(\mathbf{U}\mathbf{V}^\top), \tag{35}$$

corresponds to a global minimum $\mathbf{Z}^* = \hat{\mathbf{U}}\hat{\mathbf{V}}^\top$ of (33) as long as $m \geq m^*$ for some $m^* \leq d + c$.

Proof. Define $F' : \mathbb{R}^{(d+c)\times(d+c)} \to \mathbb{R}$ such that

$$F'(\begin{bmatrix} \mathbf{X}_1 & \mathbf{X}_2 \\ \mathbf{X}_3 & \mathbf{X}_4 \end{bmatrix}) := F(\mathbf{X}_2) \tag{36}$$

for $\mathbf{X}_1 \in \mathbb{R}^{d\times d}$, $\mathbf{X}_2 \in \mathbb{R}^{d\times c}$, $\mathbf{X}_3 \in \mathbb{R}^{c\times d}$, $\mathbf{X}_4 \in \mathbb{R}^{c\times c}$. Then, let $\mathbf{R} = \begin{bmatrix} \mathbf{U} \\ \mathbf{V} \end{bmatrix} \in \mathbb{R}^{(d+c)\times m}$. One can re-write $F(\mathbf{U}\mathbf{V}^\top)$ as $F'(\mathbf{R}\mathbf{R}^\top)$. Then, we see that (35) can be expressed as (31), and any local minimizer to (35) is a local minimizer to (31).

Next, note that from Theorem A.3 that any local minimizer of (31) is a local minimizer of (30). Lastly, note from Lemma A.4 that any local minimizer of (30) corresponds to a global minimizer of (33) for $m^* \leq d + c$. Thus, any local minimizer of (35) is a global minimizer of (33) as long as $m \geq m^*$, where $m^* \leq d + c$. □

A.3 Result of (Haeffele et al., 2014) and its application to rectangular matrices

In this subsection, we outline the precise theoretical statement of (Haeffele et al., 2014) and describe exactly how it corresponds to our summary in Section 2.2, and thus the application to the later derivations in our work. We first describe the following result, without proof, from (Haeffele et al., 2014).

Theorem A.6 (Theorem 2 of (Haeffele et al., 2014))**.** *Let $F' : S_n^+ \to \mathbb{R}$ be of the form such that $F'(\mathbf{X}) = G'(\mathbf{X}) + H'(\mathbf{X})$, where G' is convex, twice differentiable with compact level sets, and H' is a proper convex function such that F' is lower semi-continuous. Then, if $\hat{\mathbf{R}}$ is a rank-deficient local minimizer of*

$$\min_{\mathbf{R}\in\mathbb{R}^{(d+c)\times m}} F'(\mathbf{R}\mathbf{R}^\top), \tag{37}$$

then $\mathbf{X}^* = \hat{\mathbf{R}}\hat{\mathbf{R}}^\top$ is a global minimizer of

$$\min_{\substack{\mathbf{X}\in\mathbb{R}^{(d+c)\times(d+c)} \\ \mathbf{X}\succeq 0}} F'(\mathbf{X}). \tag{38}$$

We now describe how this theorem applies to the setting described in Section 2.2, i.e. the rectangular matrix, non-SDP case.

Lemma A.7 (Used in Section 2.2)**.** *Let $F : \mathbb{R}^{d\times c} \to \mathbb{R}$ be of the form such that $F(\mathbf{Z}) = G(\mathbf{Z}) + H(\mathbf{Z})$, where G is convex, twice differentiable with compact level sets, and H is a proper convex function such that F is lower semi-continuous. Then, if $\hat{\mathbf{R}} = \begin{bmatrix} \hat{\mathbf{U}} \\ \hat{\mathbf{V}} \end{bmatrix}$ is a rank-deficient local minimizer of*

$$\min_{\substack{\mathbf{U}\in\mathbb{R}^{d\times m} \\ \mathbf{V}\in\mathbb{R}^{d\times m}}} F(\mathbf{U}\mathbf{V}^\top), \tag{39}$$

then $\mathbf{Z}^* = \hat{\mathbf{U}}\hat{\mathbf{V}}^\top$ is a global minimizer of

$$\min_{\mathbf{Z}\in\mathbb{R}^{d\times c}} F(\mathbf{Z}). \tag{40}$$

Proof. Define $F' : \mathbb{R}^{(d+c)\times(d+c)} \to \mathbb{R}$ such that

$$F'(\begin{bmatrix} \mathbf{X}_1 & \mathbf{X}_2 \\ \mathbf{X}_3 & \mathbf{X}_4 \end{bmatrix}) := F(\mathbf{X}_2) \tag{41}$$

for $\mathbf{X}_1 \in \mathbb{R}^{d\times d}$, $\mathbf{X}_2 \in \mathbb{R}^{d\times c}$, $\mathbf{X}_3 \in \mathbb{R}^{c\times d}$, $\mathbf{X}_4 \in \mathbb{R}^{c\times c}$. Clearly, if $F = G + H$, where G is twice-differentiable and H is proper convex, then, $F' = G' + H'$ where G' is twice-differentiable and H' is proper convex. From the proof of Lemma A.5, we know that (39) is exactly the same as (37) for $\mathbf{R} = \begin{bmatrix} \mathbf{U} \\ \mathbf{V} \end{bmatrix}$. Furthermore, we know from Lemma A.4 that any

global minimum $\mathbf{Z}^*$ of (40) corresponds to a global minimum $\mathbf{X}^*$ of (38). Lastly, we know from Theorem A.6 that any rank-deficient local minimum of (37) corresponds to a global minimizer of (38).

Putting it all together, we have that a rank-deficient local minimizer $\hat{\mathbf{R}} = \begin{bmatrix} \hat{\mathbf{U}} \\ \hat{\mathbf{V}} \end{bmatrix}$ of (39) is a rank-deficient local minimizer of (37), which corresponds to a global optimizer $\mathbf{X}^* = \hat{\mathbf{R}}\hat{\mathbf{R}}^\top$ of (38) (Theorem A.6), which corresponds to a global optimizer $\mathbf{Z}^* = \hat{\mathbf{U}}\hat{\mathbf{V}}^\top$ of (40) (Lemma A.4). □

A.4 Additional Results on Size of BM factorization

We note the additional result from Zhang (2022). In the case that F is twice-differentiable, L-smooth, and μ-strongly convex, a local minimum $(\hat{\mathbf{U}}, \hat{\mathbf{V}})$ of (35) corresponds to a global minimum of (33) as long as $m \geq \frac{1}{4}(L/\mu - 1)^2 m^*$, where $m^* = \operatorname{rank}(\mathbf{Z}^*)$ and $\mathbf{Z}^*$ is the optimal solution to (33).

A.5 Proof of Lemma 3.1

Proof. We first analyze the solution of the following optimization problem

$$f^* = \min_{\mathbf{u}_j, \mathbf{v}_j} \frac{1}{2} \left(\sum_{j=1}^m \|\mathbf{u}_j\|_C^2 + \|\mathbf{v}_j\|_R^2 \right) \text{ s.t. } \mathbf{U}\mathbf{V}^\top = \mathbf{Z}. \tag{42}$$

We can write this as an equivalent problem here (Bach et al., 2008; Pilanci & Ergen, 2020):

$$f^* = \min_{\mathbf{u}_j \in \mathcal{B}_C, \mathbf{v}_j} \left(\sum_{j=1}^m \|\mathbf{v}_j\|_R \right) \text{ s.t. } \mathbf{U}\mathbf{V}^\top = \mathbf{Z}. \tag{43}$$

We can form the Lagrangian of this as

$$f^* = \min_{\mathbf{u}_j \in \mathcal{B}_C, \mathbf{v}_j} \max_{\mathbf{R}} \left(\sum_{j=1}^m \|\mathbf{v}_j\|_R \right) - \operatorname{trace}\left(\mathbf{R}^\top \mathbf{U}\mathbf{V}^\top\right) + \operatorname{trace}\left(\mathbf{R}^\top \mathbf{Z}\right). \tag{44}$$

By Sion's minimax theorem, we can switch the maximum over $\mathbf{R}$ and minimum over $\mathbf{V}$ and minimize over $\mathbf{V}$ to obtain

$$f^* = \min_{\mathbf{u} \in \mathcal{B}_C} \max_{\mathbf{R}} \operatorname{trace}\left(\mathbf{R}^\top \mathbf{Z}\right) \text{ s.t.} \|\mathbf{u}^\top \mathbf{R}\|_R^* \leq 1. \tag{45}$$

As long as $m \geq \operatorname{rank}(\mathbf{Z})$, by Slater's condition, we can switch the minimum and maximum (Shapiro, 2009) to obtain

$$f^* = \max_{\mathbf{R}} \operatorname{trace}\left(\mathbf{R}^\top \mathbf{Z}\right) \text{ s.t.} \|\mathbf{u}^\top \mathbf{R}\|_R^* \leq 1 \; \forall \mathbf{u} \in \mathcal{B}_C. \tag{46}$$

By the definition of dual norm, we can also write this as

$$f^* = \max_{\mathbf{R}} \operatorname{trace}\left(\mathbf{R}^\top \mathbf{Z}\right) \text{ s.t.} \mathbf{u}^\top \mathbf{R}\mathbf{v} \leq 1 \; \forall \mathbf{u} \in \mathcal{B}_C, \; \forall \mathbf{v} \in \mathcal{B}_R = \|\mathbf{Z}\|_D. \tag{47}$$

Thus, with this result, we have

$$p^* := \min_{\substack{\mathbf{U} \in \mathbb{R}^{d_c \times m} \\ \mathbf{V} \in \mathbb{R}^{d_r \times m}}} f(\mathbf{U}, \mathbf{V}) + \frac{\beta}{2} \left(\sum_{j=1}^m \|\mathbf{u}_j\|_C^2 + \|\mathbf{v}_j\|_R^2 \right), \tag{48}$$

equivalently as

$$p^* = \min_{\substack{\mathbf{U} \in \mathbb{R}^{d_c \times m} \\ \mathbf{V} \in \mathbb{R}^{d_r \times m}}} F(\mathbf{M}\mathbf{U}\mathbf{V}^\top) + \frac{\beta}{2} \left(\sum_{j=1}^m \|\mathbf{u}_j\|_C^2 + \|\mathbf{v}_j\|_R^2 \right), \tag{49}$$

or also as

$$p^* = \min_{\mathbf{Z}: \operatorname{rank}(\mathbf{Z}) \leq m} F(\mathbf{M}\mathbf{Z}) + \min_{\substack{\mathbf{U} \in \mathbb{R}^{d_c \times m} \\ \mathbf{V} \in \mathbb{R}^{d_r \times m} \\ \mathbf{U}\mathbf{V}^\top = \mathbf{Z}}} \frac{\beta}{2} \left(\sum_{j=1}^m \|\mathbf{u}_j\|_C^2 + \|\mathbf{v}_j\|_R^2 \right), \tag{50}$$

where we now apply our previous result to obtain

$$p^* = \min_{\mathbf{Z}:\text{rank}(\mathbf{Z})\leq m} F(\mathbf{M}\mathbf{Z}) + \beta\|\mathbf{Z}\|_D, \tag{51}$$

which if $\text{rank}(\mathbf{Z}^*) \geq m$ is equivalent to

$$p^* = \min_{\mathbf{Z}} F(\mathbf{M}\mathbf{Z}) + \beta\|\mathbf{Z}\|_D. \tag{52}$$

□

A.6 Proof of Theorem 3.3

Proof. We simply note that (14) is the Burer-Monteiro factorization of (16). Thus, from Lemma A.5, as long as $m \geq d_c + d_r$, all local minima of (14) are global minima. Furthermore, note that (16) is composed of two components, one of which is a twice-differentiable function, and the other is a proper convex function. Thus, by Lemma A.7, all rank-deficient local minima are global minima. □

A.7 Proof of Theorem 3.4

Proof. Stationary points of (14) satisfy

$$\mathbf{0} \in \nabla_{\mathbf{U}} f(\hat{\mathbf{U}}, \hat{\mathbf{V}}) + \beta\|\hat{\mathbf{U}}\|_C \partial\|\hat{\mathbf{U}}\|_C$$
$$\mathbf{0} \in \nabla_{\mathbf{V}} f(\hat{\mathbf{U}}, \hat{\mathbf{V}}) + \beta\|\hat{\mathbf{V}}\|_R \partial\|\hat{\mathbf{V}}\|_R,$$

where we define

$$\|\hat{\mathbf{U}}\|_C := \sum_{j=1}^{m} \|\hat{\mathbf{u}}_j\|_C$$

and the same for $\|\hat{\mathbf{V}}\|_R$. By the definition of the subgradient, this stationarity condition can be written as

$$\exists \mathbf{U}' \text{ s.t. } \text{trace}(\hat{\mathbf{U}}^\top \mathbf{U}') = \|\hat{\mathbf{U}}\|_C,\ \|\mathbf{U}'\|_C^* \leq 1,\ \mathbf{0} = \nabla_{\mathbf{U}} f(\hat{\mathbf{U}}, \hat{\mathbf{V}}) + \beta\|\hat{\mathbf{U}}\|_C \mathbf{U}'$$
$$\exists \mathbf{V}' \text{ s.t. } \text{trace}(\hat{\mathbf{V}}^\top \mathbf{V}') = \|\hat{\mathbf{V}}\|_R,\ \|\mathbf{V}'\|_R^* \leq 1,\ \mathbf{0} = \nabla_{\mathbf{V}} f(\hat{\mathbf{U}}, \hat{\mathbf{V}}) + \beta\|\hat{\mathbf{V}}\|_R \mathbf{V}'.$$

By the chain rule, we have that

$$\mathbf{0} = \nabla_{\mathbf{Z}} F(\mathbf{M}\hat{\mathbf{Z}})\hat{\mathbf{V}} + \beta\|\hat{\mathbf{U}}\|_C \mathbf{U}'$$
$$\mathbf{0} = \nabla_{\mathbf{Z}} F(\mathbf{M}\hat{\mathbf{Z}})^\top \hat{\mathbf{U}} + \beta\|\hat{\mathbf{V}}\|_R \mathbf{V}'$$

We now right-multiply the top equation by $\hat{\mathbf{U}}^\top$ and the bottom equation by $\hat{\mathbf{V}}^\top$ to obtain

$$\mathbf{0} = \nabla_{\mathbf{Z}} F(\mathbf{M}\hat{\mathbf{Z}})\hat{\mathbf{V}}\hat{\mathbf{U}}^\top + \beta\|\hat{\mathbf{U}}\|_C \mathbf{U}'\hat{\mathbf{U}}^\top \tag{53}$$
$$\mathbf{0} = \nabla_{\mathbf{Z}} F(\mathbf{M}\hat{\mathbf{Z}})^\top \hat{\mathbf{U}}\hat{\mathbf{V}}^\top + \beta\|\hat{\mathbf{V}}\|_R \mathbf{V}'\hat{\mathbf{V}}^\top. \tag{54}$$

Taking the trace, we have

$$-\frac{1}{\beta}\text{trace}\left(\nabla_{\mathbf{Z}} F(\mathbf{M}\hat{\mathbf{Z}})^\top \hat{\mathbf{Z}}\right) = \|\hat{\mathbf{U}}\|_C \text{trace}\left(\mathbf{U}'\hat{\mathbf{U}}^\top\right) = \|\hat{\mathbf{U}}\|_R \text{trace}\left(\mathbf{V}'\hat{\mathbf{V}}^\top\right). \tag{55}$$

Noting the definitions of $\mathbf{U}'$ and $\mathbf{V}'$, we have

$$-\frac{1}{\beta}\text{trace}\left(\nabla_{\mathbf{Z}} F(\mathbf{M}\hat{\mathbf{Z}})^\top \hat{\mathbf{Z}}\right) = \|\hat{\mathbf{U}}\|_C^2 = \|\hat{\mathbf{V}}\|_R^2 \tag{56}$$

Furthermore, since clearly $F(\mathbf{M}\hat{\mathbf{Z}}) = f(\hat{\mathbf{U}}, \hat{\mathbf{V}})$, we have that

$$-\frac{1}{\beta}\text{trace}\left(\nabla_{\mathbf{Z}} F(\mathbf{M}\hat{\mathbf{Z}})^\top \hat{\mathbf{Z}}\right) = \|\hat{\mathbf{U}}\|_C^2 = \|\hat{\mathbf{V}}\|_R^2 = \frac{1}{2}(\|\hat{\mathbf{U}}\|_C^2 + \|\hat{\mathbf{V}}\|_R^2) = \|\hat{\mathbf{Z}}\|_D. \tag{57}$$

Now, we examine the optimality conditions for (16). In particular, we have

$$p^* = \min_{\mathbf{Z}} F(\mathbf{M}\mathbf{Z}) + \beta\|\mathbf{Z}\|_D, \tag{58}$$

which by definition of the dual norm, is equivalent to

$$p^* = \min_{\mathbf{Z}} \max_{\mathbf{Z}' \in \mathcal{B}_{D^*}} F(\mathbf{M}\mathbf{Z}) + \beta \mathrm{trace}(\mathbf{Z}^\top \mathbf{Z}'). \tag{59}$$

Now suppose we have an approximate saddle point $(\hat{\mathbf{Z}}, \hat{\mathbf{Z}}')$ with objective value $\hat{p}$ such that $\nabla_{\mathbf{Z}} F(\hat{\mathbf{Z}}) + \beta \hat{\mathbf{Z}}' = 0$, $\mathrm{trace}(\hat{\mathbf{Z}}^\top \hat{\mathbf{Z}}') = \|\hat{\mathbf{Z}}\|_D$, and $\hat{\mathbf{Z}}' \in (1+\epsilon)\mathcal{B}_{D^*}$ for some $\epsilon \geq 0$. Let $\tilde{\mathbf{Z}} = \frac{1}{1+\epsilon}\hat{\mathbf{Z}}'$. By strong duality and non-negativity of F, we have

$$\begin{aligned}
p^* &= \max_{\mathbf{Z}' \in \mathcal{B}_{D^*}} \min_{\mathbf{Z}} F(\mathbf{M}\mathbf{Z}) + \beta \mathrm{trace}(\mathbf{Z}^\top \mathbf{Z}') && (60)\\
&\geq \min_{\mathbf{Z}} F(\mathbf{M}\mathbf{Z}) + \beta \mathrm{trace}(\mathbf{Z}^\top \tilde{\mathbf{Z}}) && (61)\\
&= \min_{\mathbf{Z}} F(\mathbf{M}\mathbf{Z}) + \frac{\beta}{1+\epsilon} \mathrm{trace}(\mathbf{Z}^\top \hat{\mathbf{Z}}') && (62)\\
&\geq \frac{1}{1+\epsilon} \min_{\mathbf{Z}} \Big(F(\mathbf{M}\mathbf{Z}) + \beta \mathrm{trace}(\mathbf{Z}^\top \hat{\mathbf{Z}}') \Big) && (63)\\
&= \frac{1}{1+\epsilon} \hat{p} && (64)
\end{aligned}$$

Rearranging, we have that

$$\frac{\hat{p} - p^*}{p^*} \leq \epsilon. \tag{65}$$

For the assumptions of this inequality to hold, for any candidate solution $\hat{\mathbf{Z}}$, one must satisfy $\nabla_{\mathbf{Z}} F(\mathbf{M}\hat{\mathbf{Z}}) + \beta \hat{\mathbf{Z}}' = 0$ and $\mathrm{trace}(\hat{\mathbf{Z}}^\top \hat{\mathbf{Z}}') = \|\hat{\mathbf{Z}}\|_D$. Solving the former equality for $\hat{\mathbf{Z}}' = -\frac{1}{\beta}\nabla_{\mathbf{Z}} F(\mathbf{M}\hat{\mathbf{Z}})$, we have by (57) that the latter equality is satisfied at any stationary point of the BM factorization. Lastly, for (65) to hold for a particular ϵ, one must have

$$\hat{\mathbf{Z}}' = -\frac{1}{\beta}\nabla_{\mathbf{Z}} F(\mathbf{M}\hat{\mathbf{Z}}) \in (1+\epsilon)\mathcal{B}_{D^*}, \tag{66}$$

so $\epsilon = \left(\frac{\|\nabla_{\mathbf{Z}} F(\mathbf{M}\hat{\mathbf{Z}})\|_D^*}{\beta} - 1 \right)_+$, i.e.

$$\frac{\hat{p} - p^*}{p^*} \leq \left(\frac{\|\nabla_{\mathbf{Z}} F(\mathbf{M}\hat{\mathbf{Z}})\|_D^*}{\beta} - 1 \right)_+ \tag{67}$$

□

A.8 Proof of Theorem 3.5

Proof. From the stationary point condition, we have

$$\begin{aligned}
\mathbf{0} &= \nabla_{\mathbf{Z}} F(\mathbf{M}\hat{\mathbf{Z}})\hat{\mathbf{V}} + \beta \hat{\mathbf{U}} && (68)\\
\mathbf{0} &= \nabla_{\mathbf{Z}} F(\mathbf{M}\hat{\mathbf{Z}})^\top \hat{\mathbf{U}} + \beta \hat{\mathbf{V}}. && (69)
\end{aligned}$$

From (69) we can obtain

$$\hat{\mathbf{U}}^\top \nabla_{\mathbf{Z}} F(\mathbf{M}\hat{\mathbf{Z}}) = -\beta \hat{\mathbf{V}}^\top. \tag{70}$$

Substituting this into (68), we have

$$\begin{aligned}
\mathbf{0} &= -\beta \hat{\mathbf{V}}^\top \hat{\mathbf{V}} + \beta \hat{\mathbf{U}}^\top \hat{\mathbf{U}} && (71)\\
\hat{\mathbf{V}}^\top \hat{\mathbf{V}} &= \hat{\mathbf{U}}^\top \hat{\mathbf{U}}. && (72)
\end{aligned}$$

Thus, let $r := \mathrm{rank}(\hat{\mathbf{V}}) = \mathrm{rank}(\hat{\mathbf{U}}) \leq \min\{d_c, d_r\}$. We can write the compact SVD of the stationary point as

$$\begin{aligned}
\hat{\mathbf{U}} &= \mathbf{L}_{\mathbf{U}} \mathbf{\Lambda} \mathbf{R}^\top \\
\hat{\mathbf{V}} &= \mathbf{L}_{\mathbf{V}} \mathbf{\Lambda} \mathbf{R}^\top,
\end{aligned}$$

where $\mathbf{L_U} \in \mathbb{R}^{d_c \times r}$, $\mathbf{L_V} \in \mathbb{R}^{d_r \times r}$. Assume without loss of generality that $d_c > d_r$, so $d_r = \min\{d_c, d_r\}$. We have

$$\mathbf{0} = \nabla_{\mathbf{Z}} F(\mathbf{M}\hat{\mathbf{Z}})\hat{\mathbf{V}} + \beta\hat{\mathbf{U}} \tag{73}$$

$$\mathbf{0} = \nabla_{\mathbf{Z}} F(\mathbf{M}\hat{\mathbf{Z}})\mathbf{L_V}\mathbf{\Lambda}\mathbf{R}^\top + \beta\mathbf{L_U}\mathbf{\Lambda}\mathbf{R}^\top \tag{74}$$

$$-\beta\mathbf{L_U} = \nabla_{\mathbf{Z}} F(\mathbf{M}\hat{\mathbf{Z}})\mathbf{L_V}. \tag{75}$$

If $r = d_r$, $\mathbf{L_V}$ is square and therefore unitary, so

$$\|\nabla_{\mathbf{Z}} F(\mathbf{M}\hat{\mathbf{Z}})\|_2 = \|\nabla_{\mathbf{Z}} F(\mathbf{M}\hat{\mathbf{Z}})\mathbf{L_V}\|_2 \tag{76}$$

$$= \|-\beta\mathbf{L_U}\|_2 \tag{77}$$

$$= \beta. \tag{78}$$

Thus, we satisfy (20) with equality. □

In general, note that when $r < \min\{d_c, d_r\}$, we have

$$\|\nabla_{\mathbf{Z}} F(\mathbf{M}\hat{\mathbf{Z}})\|_2 = \|\nabla_{\mathbf{Z}} F(\mathbf{M}\hat{\mathbf{Z}})\|_2\|\mathbf{L_V}\|_2 \geq \|\nabla_{\mathbf{Z}} F(\mathbf{M}\hat{\mathbf{Z}})\mathbf{L_V}\|_2 \tag{79}$$

$$= \|-\beta\mathbf{L_U}\|_2 \tag{80}$$

$$= \beta, \tag{81}$$

so (20) is a lower bound, which depends on how $\nabla_{\mathbf{Z}} F(\hat{\mathbf{Z}})$ behaves when operating on vectors in $\text{null}(\mathbf{L}_\mathbf{V}^\top)$.

A.9 Proof of Theorem 3.6

Proof. We begin with the non-convex objective (22)

$$p^*_{CNN} := \min_{\substack{\mathbf{w}_{1j} \in \mathbb{R}^h \\ \mathbf{W}_{2j} \in \mathbb{R}^{c \times a}}} \sum_{i=1}^n F\left(\sum_{j=1}^m \mathbf{W}_{2j}\mathbf{P}_a(\mathbf{X}_i\mathbf{w}_{1j})_+\right) + \frac{\beta}{2}\sum_{j=1}^m \|\mathbf{w}_{1j}\|_2^2 + \|\mathbf{W}_{2j}\|_F^2. \tag{82}$$

We can re-write this as (Bach et al., 2008; Pilanci & Ergen, 2020)

$$p^*_{CNN} = \min_{\substack{\mathbf{w}_{1j} \in \mathcal{B}_2 \\ \mathbf{W}_{2j} \in \mathbb{R}^{c \times a}}} \sum_{i=1}^n F\left(\sum_{j=1}^m \mathbf{W}_{2j}\mathbf{P}_a(\mathbf{X}_i\mathbf{w}_{1j})_+\right) + \beta\sum_{j=1}^m \|\mathbf{W}_{2j}\|_F. \tag{83}$$

We can also re-write this as

$$p^*_{CNN} = \min_{\substack{\mathbf{w}_{1j} \in \mathcal{B}_2 \\ \mathbf{W}_{2j} \in \mathbb{R}^{c \times a} \\ \mathbf{r}_i}} \sum_{i=1}^n F(\mathbf{r}_i) + \beta\sum_{j=1}^m \|\mathbf{W}_{2j}\|_F \text{ s.t. } \sum_{j=1}^m \mathbf{W}_{2j}\mathbf{P}_a(\mathbf{X}_i\mathbf{w}_{1j})_+ = \mathbf{r}_i \tag{84}$$

Forming the Lagrangian, we have

$$p^*_{CNN} = \min_{\substack{\mathbf{w}_{1j} \in \mathcal{B}_2 \\ \mathbf{W}_{2j} \in \mathbb{R}^{c \times a} \\ \mathbf{r}_i}} \max_{\mathbf{v}_i} \sum_{i=1}^n F(\mathbf{r}_i) + \beta\sum_{j=1}^m \|\mathbf{W}_{2j}\|_F + \sum_{i=1}^n \mathbf{v}_i^\top\left(\sum_{j=1}^m \mathbf{W}_{2j}\mathbf{P}_a(\mathbf{X}_i\mathbf{w}_{1j})_+ - \mathbf{r}_i\right) \tag{85}$$

By Sion's minimax theorem, we can swap the minimum over $\mathbf{W}_{2j}, \mathbf{r}_i$ and maximum over $\mathbf{v}_i$, and minimize over $\mathbf{W}_{2j}, \mathbf{r}_i$ to obtain

$$p^*_{CNN} = \min_{\mathbf{u} \in \mathcal{B}_2} \max_{\mathbf{v}_i} -\sum_{i=1}^n -F^*(\mathbf{v}_i)$$

$$\text{s.t. } \|\sum_{i=1}^n \mathbf{P}_a(\mathbf{X}_i\mathbf{u})_+\mathbf{v}_i^\top\|_F \leq \beta \tag{86}$$

where F^* is the Fenchel conjugate of F. Now, as long as $\beta > 0$ and $m \geq m^*$ where $m^* \leq nac$, we can switch the order of max and min by Slater's condition (Shapiro, 2009; Sahiner et al., 2021b) to obtain

$$\begin{aligned} p^*_{CNN} = \max_{\mathbf{v}_i} & -\sum_{i=1}^{n} -F^*(\mathbf{v}_i) \\ \text{s.t.} & \max_{\mathbf{u}\in\mathcal{B}_2} \| \sum_{i=1}^{n} \mathbf{P}_a(\mathbf{X}_i\mathbf{u})_+ \mathbf{v}_i^\top \|_F \leq \beta. \end{aligned} \tag{87}$$

Enumerating over the hyperplane arrangements $\{\mathbf{D}_k\}_{k=1}^P$, we can further write this as

$$\begin{aligned} p^*_{CNN} = \max_{\mathbf{v}_i} & -\sum_{i=1}^{n} -F^*(\mathbf{v}_i) \\ \text{s.t.} & \max_{\substack{k\in[P] \\ \mathbf{u}\in\mathcal{B}_2 \\ (2\mathbf{D}_k^{(i)}-\mathbf{I})\mathbf{X}_i\mathbf{u}\geq 0}} \| \sum_{i=1}^{n} \mathbf{P}_a \mathbf{D}_k^{(i)} \mathbf{X}_i \mathbf{u} \mathbf{v}_i^\top \|_F \leq \beta. \end{aligned} \tag{88}$$

Now, noting that $\text{vec}(\mathbf{ABC}) = (\mathbf{C}^\top \otimes \mathbf{A})\text{vec}(\mathbf{B})$ (Magnus & Neudecker, 2019), this is equivalent to

$$\begin{aligned} p^*_{CNN} = \max_{\mathbf{v}_i} & -\sum_{i=1}^{n} F^*(\mathbf{v}_i) \\ \text{s.t.} & \max_{\substack{k\in[P] \\ \mathbf{u}\in\mathcal{B}_2 \\ (2\mathbf{D}_k^{(i)}-\mathbf{I})\mathbf{X}_i\mathbf{u}\geq 0}} \| \sum_{i=1}^{n} (\mathbf{v}_i \otimes \mathbf{P}_a \mathbf{D}_k^{(i)} \mathbf{X}_i)\mathbf{u} \|_2 \leq \beta \end{aligned} \tag{89}$$

This may also be written further as

$$\begin{aligned} p^*_{CNN} = \max_{\mathbf{v}_i} & -\sum_{i=1}^{n} F^*(\mathbf{v}_i) \\ \text{s.t.} & \max_{\substack{k\in[P] \\ \mathbf{u}\in\mathcal{B}_2 \\ (2\mathbf{D}_k^{(i)}-\mathbf{I})\mathbf{X}_i\mathbf{u}\geq 0 \\ \mathbf{g}\in\mathcal{B}_2}} \mathbf{g}^\top \sum_{i=1}^{n} (\mathbf{v}_i \otimes \mathbf{P}_a \mathbf{D}_k^{(i)} \mathbf{X}_i)\mathbf{u} \leq \beta, \end{aligned} \tag{90}$$

and thereby as

$$\begin{aligned} p^*_{CNN} = \max_{\mathbf{v}_i} & -\sum_{i=1}^{n} F^*(\mathbf{v}_i) \\ \text{s.t.} & \max_{\substack{k\in[P] \\ \mathbf{u}\in\mathcal{B}_2 \\ (2\mathbf{D}_k^{(i)}-\mathbf{I})\mathbf{X}_i\mathbf{u}\geq 0 \\ \mathbf{g}\in\mathcal{B}_2}} \text{trace}\left(\sum_{i=1}^{n} (\mathbf{v}_i \otimes \mathbf{P}_a \mathbf{D}_k^{(i)} \mathbf{X}_i)\mathbf{u}\mathbf{g}^\top \right) \leq \beta. \end{aligned} \tag{91}$$

Now, we let $\mathbf{Z} = \mathbf{u}\mathbf{g}^\top$ to obtain

$$\begin{aligned} p^*_{CNN} = \max_{\mathbf{v}_i} & -\sum_{i=1}^{n} F^*(\mathbf{v}_i) \\ \text{s.t.} & \max_{\substack{k\in[P] \\ \mathbf{Z}=\mathbf{u}\mathbf{g}^\top \\ \mathbf{u}\in\mathcal{B}_2 \\ (2\mathbf{D}_k^{(i)}-\mathbf{I})\mathbf{X}_i\mathbf{u}\geq 0 \\ \mathbf{g}\in\mathcal{B}_2}} \text{trace}\left(\sum_{i=1}^{n} (\mathbf{v}_i \otimes \mathbf{P}_a \mathbf{D}_k^{(i)} \mathbf{X}_i)\mathbf{Z} \right) \leq \beta. \end{aligned} \tag{92}$$

We let $\mathcal{C}_k := \text{conv}\left\{\mathbf{u}\mathbf{g}^\top : (2\mathbf{D}_k^{(i)} - \mathbf{I})\mathbf{X}_i\mathbf{u} \geq 0,\ \mathbf{u} \in \mathcal{B}_2,\ \mathbf{g} \in \mathcal{B}_2\right\}$ and note that since our objective is linear we can take the convex hull of the constraints without changing the objective, to obtain

$$\begin{aligned} p^*_{CNN} = \max_{\mathbf{v}_i} & -\sum_{i=1}^n F^*(\mathbf{v}_i) \\ \text{s.t.} & \max_{\substack{k\in[P] \\ \mathbf{Z}\in\mathcal{C}_k}} \text{trace}\left(\sum_{i=1}^n (\mathbf{v}_i \otimes \mathbf{P}_a\mathbf{D}_k^{(i)}\mathbf{X}_i)\mathbf{Z}\right) \leq \beta. \end{aligned} \tag{93}$$

Note that the constraint $\mathbf{Z} \in \mathcal{C}_k$ is equivalent to stating that $\|\mathbf{Z}\|_{*,\mathrm{K}_k} \leq 1$ for the constrained nuclear norm definition with $\mathbf{K}_k = (2\mathbf{D}_k - \mathbf{I})\mathbf{X}$. Then, we have

$$\begin{aligned} p^*_{CNN} = \max_{\mathbf{v}_i} & -\sum_{i=1}^n F^*(\mathbf{v}_i) \\ \text{s.t.} & \max_{\substack{k\in[P] \\ \|\mathbf{Z}\|_{*,\mathrm{K}_k}\leq 1}} \text{trace}\left(\sum_{i=1}^n (\mathbf{v}_i \otimes \mathbf{P}_a\mathbf{D}_k^{(i)}\mathbf{X}_i)\mathbf{Z}\right) \leq \beta. \end{aligned} \tag{94}$$

Now, we form the Lagrangian, given by

$$p^*_{CNN} = \max_{\mathbf{v}_i} \min_{\|\mathbf{Z}_k\|_{*,\mathrm{K}_k}\leq 1} \min_{\lambda_k \geq 0} -\sum_{i=1}^n F^*(\mathbf{v}_i) + \sum_{k=1}^P \lambda_k \left(\beta - \sum_{i=1}^n \text{vec}(\mathbf{Z}_k)^\top \mathbf{vec}\left(\mathbf{v}_i^\top \otimes (\mathbf{P}_a\mathbf{D}_k^{(i)}\mathbf{X}_i)^\top\right)\right). \tag{95}$$

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

$$p^*_{CNN} = \min_{\lambda_k \geq 0} \min_{\|\mathbf{Z}_k\|_{*,\mathrm{K}_k}\leq 1} \max_{\mathbf{v}_i} -\sum_{i=1}^n F^*(\mathbf{v}_i) + \sum_{k=1}^P \lambda_k \left(\beta - \sum_{i=1}^n \text{vec}(\mathbf{Z}_k)^\top \mathbf{vec}\left(\mathbf{v}_i^\top \otimes (\mathbf{P}_a\mathbf{D}_k^{(i)}\mathbf{X}_i)^\top\right)\right). \tag{96}$$

Now, defining $\mathbf{K}_{a,1}$ as the $(a,1)$ commutation matrix we have the following identity from (Magnus & Neudecker, 2019):

$$\mathbf{vec}\left(\mathbf{v}_i^\top \otimes (\mathbf{P}_a\mathbf{D}_k^{(i)}\mathbf{X}_i)^\top\right) = \left(\mathbf{I}_c \otimes \left((\mathbf{K}_{a,1} \otimes \mathbf{I}_h)(\mathbf{vec}(\mathbf{X}_i^\top\mathbf{D}_k^{(i)}\mathbf{P}_a^\top)\right)\right)\mathbf{vec}(\mathbf{v}_i).$$

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

$$p^*_{CNN} = \min_{\|\mathbf{Z}_k\|_{*,\mathrm{K}_k}\leq 1} \min_{\lambda_k \geq 0} \sum_{i=1}^n F\left(\sum_{k=1}^P \left(\mathbf{I}_c \otimes \left(\mathbf{vec}(\mathbf{X}_i^\top\mathbf{D}_k^{(i)}\mathbf{P}_a^\top)^\top(\mathbf{K}_{1,a} \otimes \mathbf{I}_h)\right)\right)\mathbf{vec}(\mathbf{Z}_k)\right) + \beta\sum_{k=1}^P \lambda_k. \tag{97}$$

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

$$p^*_{CNN} = \min_{\mathbf{Z}_k\in\mathbb{R}^{h\times ac}} \sum_{i=1}^n F\left(\sum_{k=1}^P \left(\mathbf{I}_c \otimes \left(\mathbf{vec}(\mathbf{X}_i^\top\mathbf{D}_k^{(i)}\mathbf{P}_a^\top)^\top(\mathbf{K}_{1,a} \otimes \mathbf{I}_h)\right)\right)\mathbf{vec}(\mathbf{Z}_k)\right) + \beta\sum_{k=1}^P \|\mathbf{Z}_k\|_{*,\mathrm{K}_k}. \tag{98}$$

Simplifying further, we can write this as

$$p^*_{CNN} = \min_{\mathbf{Z}_k\in\mathbb{R}^{h\times ac}} \sum_{i=1}^n F\left(\sum_{k=1}^P \begin{bmatrix} \text{trace}(\mathbf{P}_a\mathbf{D}_k^{(i)}\mathbf{X}_i\mathbf{Z}_k^{(1)}) \\ \vdots \\ \text{trace}(\mathbf{P}_a\mathbf{D}_k^{(i)}\mathbf{X}_i\mathbf{Z}_k^{(c)}) \end{bmatrix}\right) + \beta\sum_{k=1}^P \|\mathbf{Z}_k\|_{*,\mathrm{K}_k}, \tag{99}$$

where $\mathbf{Z}_k^{(c')} \in \mathbb{R}^{h\times a}$. □

A.10 Proof of Lemma 3.7

Proof. We start with the convex formulation

$$p^*_{RCNN} = \min_{\mathbf{Z}_k\in\mathbb{R}^{h\times ac}} \sum_{i=1}^n F\left(\sum_{k=1}^P \begin{bmatrix} \text{trace}(\mathbf{P}_a\mathbf{D}_k^{(i)}\mathbf{X}_i\mathbf{Z}_k^{(1)}) \\ \vdots \\ \text{trace}(\mathbf{P}_a\mathbf{D}_k^{(i)}\mathbf{X}_i\mathbf{Z}_k^{(c)}) \end{bmatrix}\right) + \beta\sum_{k=1}^P \|\mathbf{Z}_k\|_{*,\mathrm{K}_k}. \tag{100}$$

In order to compute the Burer-Monteiro factorization, we factor $\mathbf{Z}_k = \mathbf{U}_k \mathbf{V}_k^\top$, where $\mathbf{U}_k \in \mathbb{R}^{h \times m}$, $\mathbf{V}_k \in \mathbb{R}^{ac \times m}$, and $(2\mathbf{D}_k^{(i)} - \mathbf{I})\mathbf{X}_i \mathbf{U}_k \geq \mathbf{0}$. Then, with $\mathbf{V}_k^{(c')} \in \mathbb{R}^{a \times m}$. Then for each k,

$$\begin{aligned}
\begin{bmatrix} \text{trace}(\mathbf{P}_a \mathbf{D}_k^{(i)} \mathbf{X}_i \mathbf{Z}_k^{(1)}) \\ \vdots \\ \text{trace}(\mathbf{P}_a \mathbf{D}_k^{(i)} \mathbf{X}_i \mathbf{Z}_k^{(c)}) \end{bmatrix} &= \begin{bmatrix} \text{trace}(\mathbf{P}_a \mathbf{D}_k^{(i)} \mathbf{X}_i \mathbf{U}_k {\mathbf{V}_k^{(1)}}^\top) \\ \vdots \\ \text{trace}(\mathbf{P}_a \mathbf{D}_k^{(i)} \mathbf{X}_i \mathbf{U}_k {\mathbf{V}_k^{(c)}}^\top) \end{bmatrix} \\
&= \begin{bmatrix} \text{trace}({\mathbf{V}_k^{(1)}}^\top \mathbf{P}_a \mathbf{D}_k^{(i)} \mathbf{X}_i \mathbf{U}_k) \\ \vdots \\ \text{trace}({\mathbf{V}_k^{(c)}}^\top \mathbf{P}_a \mathbf{D}_k^{(i)} \mathbf{X}_i \mathbf{U}_k) \end{bmatrix} \\
&= \begin{bmatrix} \sum_{j=1}^m {\mathbf{v}_{jk}^{(1)}}^\top \mathbf{P}_a \mathbf{D}_k^{(i)} \mathbf{X}_i \mathbf{u}_{jk} \\ \vdots \\ \sum_{j=1}^m {\mathbf{v}_{jk}^{(c)}}^\top \mathbf{P}_a \mathbf{D}_k^{(i)} \mathbf{X}_i \mathbf{u}_{jk} \end{bmatrix} \\
&= \sum_{j=1}^m \mathbf{V}_{jk}^\top \mathbf{P}_a \mathbf{D}_k^{(i)} \mathbf{X}_i \mathbf{u}_{jk},
\end{aligned}$$

where

$$\mathbf{V}_{jk}^\top := \begin{bmatrix} {\mathbf{v}_{jk}^{(1)}}^\top \\ \cdots \\ {\mathbf{v}_{jk}^{(c)}}^\top \end{bmatrix} \in \mathbb{R}^{c \times a}. \tag{101}$$

The equivalent Burer-Monteiro formulation thus is given by

$$p^*_{RCNN} = \min_{\substack{\{\{\mathbf{u}_{jk} \in \mathbb{R}^h\}_{j=1}^m\}_{k=1}^P \\ \{\{\mathbf{V}_{jk} \in \mathbb{R}^{c \times a}\}_{j=1}^m\}_{k=1}^P \\ (2\mathbf{D}_k^{(i)} - \mathbf{I})\mathbf{X}_i \mathbf{u}_{jk} \geq 0}} \sum_{i=1}^n F\left(\sum_{k=1}^P \sum_{j=1}^m \mathbf{V}_{jk} \mathbf{P}_a \mathbf{D}_k^{(i)} \mathbf{X}_i \mathbf{u}_{jk}\right) + \frac{\beta}{2} \sum_{k=1}^P \sum_{j=1}^m \left(\|\mathbf{u}_{jk}\|_F^2 + \|\mathbf{V}_{jk}\|_F^2\right). \tag{102}$$

□

A.11 Proof of Corollary 3.8

Proof. We simply apply the result of Theorem 3.4, noting that stationary points correspond to global minima if the norm of the gradient is less than β. Thus, this condition is equivalent to

$$\|\sum_{i=1}^n \nabla_{\mathbf{Z}_k} F\left(\sum_{k'=1}^P \begin{bmatrix} \text{trace}(\mathbf{P}_a \mathbf{D}_{k'}^{(i)} \mathbf{X}_i \mathbf{Z}_{k'}^{(1)}) \\ \vdots \\ \text{trace}(\mathbf{P}_a \mathbf{D}_{k'}^{(i)} \mathbf{X}_i \mathbf{Z}_{k'}^{(c)}) \end{bmatrix}\right) \mathbf{u}\|_2 \leq \beta, \, \forall k \in [P], \, \forall \mathbf{u} \in \mathcal{B}_2 : \; (2\mathbf{D}_k^{(i)} - \mathbf{I})\mathbf{X}_i \mathbf{u} \geq 0. \tag{103}$$

□

A.12 Proof of Lemma 3.9

Proof. Courtesy of Theorem 3.1 of (Sahiner et al., 2022), we first present the equivalent convex models for linear activation self-attention. Specifically, as long as $m \geq m^*$, where $m^* \leq \min\{d^2, dc\}$, the objective in equation 8 is equivalent to

$$p^*_{LSA} = \min_{\mathbf{Z} \in \mathbb{R}^{d^2 \times dc}} \sum_{i=1}^n F\left(\sum_{k=1}^d \sum_{\ell=1}^d \mathbf{G}_i[k, \ell] \mathbf{X}_i \mathbf{Z}^{(k,\ell)}\right) + \beta \|\mathbf{Z}\|_*, \tag{104}$$

where $\mathbf{G}_i := \mathbf{X}_i^\top \mathbf{X}_i$, $\mathbf{G}_i[k, l] \in \mathbb{R}$, and $\{\mathbf{Z}^{(k,\ell)} \in \mathbb{R}^{d \times c}\}$ are block matrices which form $\mathbf{Z}$.

To prove Lemma 3.9, we begin from the convex formulation (104). Now, we seek to find the Burer-Monteiro factorization. We let $\mathbf{Z} = \mathbf{U}\mathbf{V}^\top$, where $\mathbf{U} \in \mathbb{R}^{d^2 \times m}$ and $\mathbf{V} \in \mathbb{R}^{dc \times m}$. Let $\text{vec}^{-1}(\mathbf{u_j}) \in \mathbb{R}^{d \times d}$ be the result of taking chunks of d-length vectors from $\mathbf{u}_j$ for $j \in [m]$ 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

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

Thus,

$$\begin{aligned}
\sum_{k=1}^{d}\sum_{\ell=1}^{d} \mathbf{G}_i[k,\ell]\mathbf{X}_i\mathbf{Z}^{(k,\ell)} &= \sum_{k=1}^{d}\sum_{\ell=1}^{d}\sum_{j=1}^{m} \mathbf{G}_i[k,\ell]\mathbf{X}_i \text{vec}^{-1}(\mathbf{u_j})_k \text{vec}^{-1}(\mathbf{v_j})_\ell^\top \\
&= \mathbf{X}_i \sum_{k=1}^{d}\sum_{\ell=1}^{d}\sum_{j=1}^{m} \text{vec}^{-1}(\mathbf{u_j})_k \mathbf{G}_i[k,\ell] \text{vec}^{-1}(\mathbf{v_j})_\ell^\top \\
&= \mathbf{X}_i \sum_{j=1}^{m} \text{vec}^{-1}(\mathbf{u_j}) \mathbf{G}_i \text{vec}^{-1}(\mathbf{v_j})^\top \\
&= \sum_{j=1}^{m} \mathbf{X}_i \text{vec}^{-1}(\mathbf{u_j}) \mathbf{X}_i^\top \mathbf{X}_i \text{vec}^{-1}(\mathbf{v_j})^\top.
\end{aligned}$$

Now, overloading notation, let $\text{vec}^{-1}(\mathbf{u_j}) = \mathbf{U}_j$ and $\text{vec}^{-1}(\mathbf{v_j})^\top = \mathbf{V}_j$. We have clearly that the Burer-Monteiro factorization of (104) is given by

$$p^*_{LSA} = \min_{\substack{\mathbf{U}_j \in \mathbb{R}^{d \times d} \\ \mathbf{V}_j \in \mathbb{R}^{d \times c}}} \sum_{i=1}^{n} F\left(\sum_{j=1}^{m} \mathbf{X}_i \mathbf{U}_j \mathbf{X}_i^\top \mathbf{X}_i \mathbf{V}_j\right) + \frac{\beta}{2}\sum_{j=1}^{m} \|\mathbf{U}_j\|_F^2 + \|\mathbf{V}_j\|_F^2. \tag{105}$$

□

A.13 Proof of Corollary 3.10

Proof. We simply need to apply the result of Theorem 3.3 to this setting. In this case, the non-convex linear self-attention network is equivalent to the Burer-Monteiro factorization of the convex form. To obtain this Burer-Monteiro factorization, we factorize convex weights $\mathbf{Z} \in \mathbb{R}^{d^2 \times dc}$, so letting $m^* \leq d^2 + dc$, we can observe that as long as the number of heads m exceeds m^*, from Lemma A.5, all local optima are global. Further, we can form $\hat{\mathbf{R}} = \begin{bmatrix} \hat{\mathbf{U}} \\ \hat{\mathbf{V}} \end{bmatrix}$, and as long as this is a rank-deficient local minimum, it also corresponds to a global minimum when F is twice-differentiable by Lemma A.7. □

B Additional Theoretical Results

B.1 MLPs

The following theorem demonstrates that we can extend the results of (Sahiner et al., 2021b) beyond simply weight-decay regularization, and to arbitrary regularization.

Theorem B.1. *The non-convex ReLU training objective*

$$p^* := \min_{\mathbf{w}_{1j}, \mathbf{w}_{2j}} F(\sum_{j=1}^{m} (\mathbf{X}\mathbf{w}_{1j})_+ \mathbf{w}_{2j}) + \frac{\beta}{2}\left(\sum_{j=1}^{m} \|\mathbf{w}_{1j}\|_C^2 + \|\mathbf{w}_{2j}\|_R^2\right) \tag{106}$$

is equivalent to the convex training objective

$$p^* = \min_{\mathbf{Z}_j} F(\sum_{j=1}^{P} \mathbf{D}_j \mathbf{X} \mathbf{Z}_j) + \beta \sum_{j=1}^{P} \|\mathbf{Z}_j\|_{D_j}, \tag{107}$$

as long as $\beta > 0$ *and* $m \geq m^*$ *where* $m^* \leq nc$*, where*

$$\|\mathbf{Z}\|_{D_j} := \max_{\mathbf{R}} \text{trace}(\mathbf{R}^\top \mathbf{Z}) \text{ s.t. } \mathbf{u}^\top \mathbf{R} \mathbf{v} \leq 1 \; \forall \mathbf{u} \in \mathcal{B}_C : (2\mathbf{D}_j - \mathbf{I}_n)\mathbf{X}\mathbf{u} \geq 0, \; \forall \mathbf{v} \in \mathcal{B}_R. \tag{108}$$

Proof. We start by re-stating the convex objective (Bach et al., 2008; Pilanci & Ergen, 2020)

$$p^* := \min_{\substack{\mathbf{w}_{1j} \in \mathcal{B}_C \\ \mathbf{w}_{2j}}} F(\sum_{j=1}^{m} (\mathbf{X}\mathbf{w}_{1j})_+ \mathbf{w}_{2j}) + \beta \sum_{j=1}^{m} \|\mathbf{w}_{2j}\|_R. \tag{109}$$

Then, we can re-write this in a constrained form

$$p^* = \min_{\substack{\mathbf{w}_{1j} \in \mathcal{B}_C \\ \mathbf{w}_{2j} \\ \mathbf{R}}} F(\mathbf{R}) + \beta \sum_{j=1}^{m} \|\mathbf{w}_{2j}\|_R \text{ s.t. } \sum_{j=1}^{m} (\mathbf{X}\mathbf{w}_{1j})_+ \mathbf{w}_{2j} = \mathbf{R}, \tag{110}$$

and then the Lagrangian

$$p^* = \min_{\substack{\mathbf{w}_{1j} \in \mathcal{B}_C \\ \mathbf{w}_{2j} \\ \mathbf{R}}} \max_{\mathbf{V}} F(\mathbf{R}) + \beta \sum_{j=1}^{m} \|\mathbf{w}_{2j}\|_R + \text{trace}\left(\mathbf{V}^\top \left(\sum_{j=1}^{m} (\mathbf{X}\mathbf{w}_{1j})_+ \mathbf{w}_{2j} - \mathbf{R}\right)\right). \tag{111}$$

By Sion's minimax theorem, we can swap the order of the maximization over $\mathbf{V}$ and minimization over $\mathbf{w}_{2j}$ and $\mathbf{R}$. Then, minimizing over these two, we have

$$p^* = \min_{\mathbf{u} \in \mathcal{B}_C} \max_{\mathbf{V}} -F^*(\mathbf{V}) \text{ s.t.} \|\mathbf{V}^\top (\mathbf{X}\mathbf{u})_+\|_R^* \leq \beta. \tag{112}$$

By Slater's condition, which holds when $\beta > 0$ and $m \leq m^*$ where $m^* \leq nc$ (Shapiro, 2009; Sahiner et al., 2021c), we can switch the order of minimum and maximum to obtain

$$p^* = \max_{\mathbf{V}} -F^*(\mathbf{V}) \text{ s.t. } \max_{\mathbf{u} \in \mathcal{B}_C} \|\mathbf{V}^\top (\mathbf{X}\mathbf{u})_+\|_R^* \leq \beta. \tag{113}$$

Introducing hyperplane arrangements, we have

$$p^* = \max_{\mathbf{V}} -F^*(\mathbf{V}) \text{ s.t. } \max_{\substack{j \in [P] \\ \mathbf{u} \in \mathcal{B}_C \\ (2\mathbf{D}_j - \mathbf{I})\mathbf{X}\mathbf{u} \geq 0}} \|\mathbf{V}^\top \mathbf{D}_j \mathbf{X}\mathbf{u}\|_R^* \leq \beta. \tag{114}$$

By the concept of dual norm, this is equivalent to

$$p^* = \max_{\mathbf{V}} -F^*(\mathbf{V}) \text{ s.t. } \max_{\substack{j \in [P] \\ \mathbf{u} \in \mathcal{B}_C \\ (2\mathbf{D}_j - \mathbf{I})\mathbf{X}\mathbf{u} \geq 0 \\ \mathbf{g} \in \mathcal{B}_r}} \text{trace}\left(\mathbf{V}^\top \mathbf{D}_j \mathbf{X}\mathbf{u}\mathbf{g}^\top\right) \leq \beta. \tag{115}$$

Define

$$\|\mathbf{Z}\|_j := \max_{t \geq 0} t \text{ s.t. } \mathbf{Z} \in t\text{conv}\{\mathbf{u}\mathbf{g}^\top : \; \mathbf{u} \in \mathcal{B}_C, \; (2\mathbf{D}_j - \mathbf{I})\mathbf{X}\mathbf{u} \geq 0, \; \mathbf{g} \in \mathcal{B}_r\}. \tag{116}$$

Then, we can write our problem as

$$p^* = \max_{\mathbf{V}} -F^*(\mathbf{V}) \text{ s.t. } \max_{\substack{j \in [P] \\ \|\mathbf{Z}\|_j \leq 1}} \text{trace}\left(\mathbf{V}^\top \mathbf{D}_j \mathbf{X}\mathbf{Z}\right) \leq \beta. \tag{117}$$

Now, observe that $\|\mathbf{Z}\|_j = \|\mathbf{Z}\|_{D_j}$. In particular, let us examine (108), which we can re-write as

$$\|\mathbf{Z}\|_{D_j} = \max_{\mathbf{R}} \text{trace}(\mathbf{R}^\top \mathbf{Z}) \text{ s.t. } \max_{\|\mathbf{Z}\|_j \leq 1} \text{trace}(\mathbf{Z}^\top \mathbf{R}) \leq 1 \tag{118}$$

$$= \max_{\mathbf{R}} \text{trace}(\mathbf{R}^\top \mathbf{Z}) \text{ s.t. } \|\mathbf{R}\|_j^* \leq 1 \tag{119}$$

$$= \|\mathbf{Z}\|_j, \tag{120}$$

where the simplifications are made noting the definition of the dual norm. Now, we can write our objective as

$$p^* = \max_{\mathbf{V}} -F^*(\mathbf{V}) \text{ s.t. } \max_{\substack{j\in[P] \\ \|\mathbf{Z}\|_{D_j}\leq 1}} \text{trace}\left(\mathbf{V}^\top \mathbf{D}_j \mathbf{X}\mathbf{Z}\right) \leq \beta. \tag{121}$$

Forming the Largrangian, we have

$$p^* = \max_{\mathbf{V}} \min_{\substack{\|\mathbf{Z}_j\|_{D_j}\leq 1 \\ \lambda_j \geq 0}} -F^*(\mathbf{V}) + \sum_{j=1}^{P} \lambda_j \left(\beta - \text{trace}\left(\mathbf{V}^\top \mathbf{D}_j \mathbf{X}\mathbf{Z}_j\right)\right). \tag{122}$$

By Sion's minimax theorem, we can switch max and min and solve over $\mathbf{V}$ to obtain

$$p^* = \min_{\substack{\|\mathbf{Z}_j\|_{D_j}\leq 1 \\ \lambda_j \geq 0}} F(\sum_{j=1}^{P} \lambda_j \mathbf{D}_j \mathbf{X}\mathbf{Z}_j) + \beta \sum_{j=1}^{P} \lambda_j. \tag{123}$$

Lastly, we can combine $\mathbf{Z}_j \lambda_j$ into one variable to obtain

$$p^* = \min_{\mathbf{Z}_j} F(\sum_{j=1}^{P} \mathbf{D}_j \mathbf{X}\mathbf{Z}_j) + \beta \sum_{j=1}^{P} \|\mathbf{Z}_j\|_{D_j}, \tag{124}$$

as desired. □

B.2 CNNs

Lemma B.2. *The Burer-Monteiro factorization of the convex CNN problem with linear and gated ReLU activation are given as follows.*

$$p^*_{LCNN} = \min_{\substack{\{\mathbf{u}_j \in \mathbb{R}^h\}_{j=1}^m \\ \{\mathbf{V}_j \in \mathbb{R}^{c\times a}\}_{j=1}^m}} \sum_{i=1}^{n} F\left(\sum_{j=1}^{m} \mathbf{V}_j \mathbf{P}_a \mathbf{X}_i \mathbf{u}_j\right) + \frac{\beta}{2} \sum_{j=1}^{m} \left(\|\mathbf{u}_j\|_2^2 + \|\mathbf{V}_j\|_F^2\right) \tag{125}$$

$$p^*_{GCNN} = \min_{\substack{\{\{\mathbf{u}_{jk} \in \mathbb{R}^h\}_{j=1}^m\}_{k=1}^P \\ \{\{\mathbf{V}_{jk} \in \mathbb{R}^{c\times a}\}_{j=1}^m\}_{k=1}^P}} \sum_{i=1}^{n} F\left(\sum_{k=1}^{P}\sum_{j=1}^{m} \mathbf{V}_{jk} \mathbf{P}_a \mathbf{D}_k^{(i)} \mathbf{X}_i \mathbf{u}_{jk}\right) + \frac{\beta}{2} \sum_{k=1}^{P}\sum_{j=1}^{m} \left(\|\mathbf{u}_{jk}\|_F^2 + \|\mathbf{V}_{jk}\|_F^2\right) \tag{126}$$

Proof. The proofs follow almost identically from the proof of Lemma 3.7. In the linear case, the convex objective is given by

$$p^*_{LCNN} = \min_{\mathbf{Z}\in\mathbb{R}^{h\times ac}} \sum_{i=1}^{n} F\left(\begin{bmatrix} \text{trace}(\mathbf{P}_a \mathbf{X}_i \mathbf{Z}^{(1)}) \\ \vdots \\ \text{trace}(\mathbf{P}_a \mathbf{X}_i \mathbf{Z}^{(c)}) \end{bmatrix}\right) + \beta \|\mathbf{Z}\|_*. \tag{127}$$

In order to compute the Burer-Monteiro factorization, we factor $\mathbf{Z} = \mathbf{U}\mathbf{V}^\top$, where $\mathbf{U} \in \mathbb{R}^{h\times m}$, $\mathbf{V} \in \mathbb{R}^{ac\times m}$. Then, with $\mathbf{V}^{(c')} \in \mathbb{R}^{a\times m}$. Then,

$$\begin{aligned}
\begin{bmatrix} \mathrm{trace}(\mathbf{P}_a\mathbf{X}_i\mathbf{Z}^{(1)}) \\ \vdots \\ \mathrm{trace}(\mathbf{P}_a\mathbf{X}_i\mathbf{Z}^{(c)}) \end{bmatrix} &= \begin{bmatrix} \mathrm{trace}(\mathbf{P}_a\mathbf{X}_i\mathbf{U}\mathbf{V}^{(1)\top}) \\ \vdots \\ \mathrm{trace}(\mathbf{P}_a\mathbf{X}_i\mathbf{U}\mathbf{V}^{(c)\top}) \end{bmatrix} \\
&= \begin{bmatrix} \mathrm{trace}(\mathbf{V}^{(1)\top}\mathbf{P}_a\mathbf{X}_i\mathbf{U}) \\ \vdots \\ \mathrm{trace}(\mathbf{V}^{(c)\top}\mathbf{P}_a\mathbf{X}_i\mathbf{U}) \end{bmatrix} \\
&= \begin{bmatrix} \sum_{j=1}^m \mathbf{v}_j^{(1)\top}\mathbf{P}_a\mathbf{X}_i\mathbf{u}_j \\ \vdots \\ \sum_{j=1}^m \mathbf{v}_j^{(c)\top}\mathbf{P}_a\mathbf{X}_i\mathbf{u}_j \end{bmatrix} \\
&= \sum_{j=1}^m \mathbf{V}_j^\top\mathbf{P}_a\mathbf{X}_i\mathbf{u}_j,
\end{aligned}$$

where

$$\mathbf{V}_j^\top := \begin{bmatrix} \mathbf{v}_j^{(1)\top} \\ \cdots \\ \mathbf{v}_j^{(c)\top} \end{bmatrix} \in \mathbb{R}^{c\times a}. \tag{128}$$

The equivalent Burer-Monteiro formulation thus is given by

$$p^*_{LCNN} = \min_{\substack{\{\mathbf{u}_j\in\mathbb{R}^h\}_{j=1}^m \\ \{\mathbf{V}_j\in\mathbb{R}^{c\times a}\}_{j=1}^m}} \sum_{i=1}^n F\left(\sum_{j=1}^m \mathbf{V}_j\mathbf{P}_a\mathbf{X}_i\mathbf{u}_j\right) + \frac{\beta}{2}\sum_{j=1}^m \left(\|\mathbf{u}_j\|_2^2 + \|\mathbf{V}_j\|_F^2\right). \tag{129}$$

In the gated ReLU case, the convex program is given by

$$p^*_{GCNN} = \min_{\mathbf{Z}_k\in\mathbb{R}^{h\times ac}} \sum_{i=1}^n F\left(\sum_{k=1}^P \begin{bmatrix} \mathrm{trace}(\mathbf{P}_a\mathbf{D}_k^{(i)}\mathbf{X}_i\mathbf{Z}_k^{(1)}) \\ \vdots \\ \mathrm{trace}(\mathbf{P}_a\mathbf{D}_k^{(i)}\mathbf{X}_i\mathbf{Z}_k^{(c)}) \end{bmatrix}\right) + \beta\sum_{k=1}^P \|\mathbf{Z}_k\|_*. \tag{130}$$

In order to compute the Burer-Monteiro factorization, we factor $\mathbf{Z}_k = \mathbf{U}_k\mathbf{V}_k^\top$, where $\mathbf{U}_k \in \mathbb{R}^{h\times m}$, $\mathbf{V}_k \in \mathbb{R}^{ac\times m}$. Then, with $\mathbf{V}_k^{(c')} \in \mathbb{R}^{a\times m}$. Then for each k,

$$\begin{aligned}
\begin{bmatrix} \mathrm{trace}(\mathbf{P}_a\mathbf{D}_k^{(i)}\mathbf{X}_i\mathbf{Z}_k^{(1)}) \\ \vdots \\ \mathrm{trace}(\mathbf{P}_a\mathbf{D}_k^{(i)}\mathbf{X}_i\mathbf{Z}_k^{(c)}) \end{bmatrix} &= \begin{bmatrix} \mathrm{trace}(\mathbf{P}_a\mathbf{D}_k^{(i)}\mathbf{X}_i\mathbf{U}_k\mathbf{V}_k^{(1)\top}) \\ \vdots \\ \mathrm{trace}(\mathbf{P}_a\mathbf{D}_k^{(i)}\mathbf{X}_i\mathbf{U}_k\mathbf{V}_k^{(c)\top}) \end{bmatrix} \\
&= \begin{bmatrix} \mathrm{trace}(\mathbf{V}_k^{(1)\top}\mathbf{P}_a\mathbf{D}_k^{(i)}\mathbf{X}_i\mathbf{U}_k) \\ \vdots \\ \mathrm{trace}(\mathbf{V}_k^{(c)\top}\mathbf{P}_a\mathbf{D}_k^{(i)}\mathbf{X}_i\mathbf{U}_k) \end{bmatrix} \\
&= \begin{bmatrix} \sum_{j=1}^m \mathbf{v}_{jk}^{(1)\top}\mathbf{P}_a\mathbf{D}_k^{(i)}\mathbf{X}_i\mathbf{u}_{jk} \\ \vdots \\ \sum_{j=1}^m \mathbf{v}_{jk}^{(c)\top}\mathbf{P}_a\mathbf{D}_k^{(i)}\mathbf{X}_i\mathbf{u}_{jk} \end{bmatrix} \\
&= \sum_{j=1}^m \mathbf{V}_{jk}^\top\mathbf{P}_a\mathbf{D}_k^{(i)}\mathbf{X}_i\mathbf{u}_{jk},
\end{aligned}$$

where

$$\mathbf{V}_{jk}^{\top} := \begin{bmatrix} \mathbf{v}_{jk}^{(1)\top} \\ \cdots \\ \mathbf{v}_{jk}^{(c)\top} \end{bmatrix} \in \mathbb{R}^{c \times a}. \tag{131}$$

The equivalent Burer-Monteiro formulation thus is given by

$$p_{GCNN}^{*} = \min_{\substack{\{\{\mathbf{u}_{jk} \in \mathbb{R}^{h}\}_{j=1}^{m}\}_{k=1}^{P} \\ \{\{\mathbf{V}_{jk} \in \mathbb{R}^{c \times a}\}_{j=1}^{m}\}_{k=1}^{P}}} \sum_{i=1}^{n} F\left(\sum_{k=1}^{P}\sum_{j=1}^{m} \mathbf{V}_{jk}\mathbf{P}_{a}\mathbf{D}_{k}^{(i)}\mathbf{X}_{i}\mathbf{u}_{jk}\right) + \frac{\beta}{2}\sum_{k=1}^{P}\sum_{j=1}^{m}\left(\|\mathbf{u}_{jk}\|_{F}^{2} + \|\mathbf{V}_{jk}\|_{F}^{2}\right). \tag{132}$$

□

B.3 Self-Attention

Lemma B.3. *The Burer-Monteiro factorization of the convex self-attention problem with gated ReLU and ReLU activations are given as follows.*

$$p_{GSA}^{*} = \min_{\substack{\mathbf{U}_{jk} \\ \mathbf{V}_{jk}}} \sum_{i=1}^{n} F\left(\sum_{k=1}^{P}\sum_{j=1}^{m}\left(\mathrm{diag}^{-1}(\mathbf{D}_{k}^{(i)}) \odot (\mathbf{X}_{i}\mathbf{U}_{jk}\mathbf{X}_{i}^{\top})\right)\mathbf{X}_{i}\mathbf{V}_{jk}\right) + \frac{\beta}{2}\sum_{k=1}^{P}\sum_{j=1}^{m}\|\mathbf{U}_{jk}\|_{F}^{2} + \|\mathbf{V}_{jk}\|_{F}^{2} \tag{133}$$

$$p_{RSA}^{*} = \min_{\substack{\mathbf{U}_{jk} \\ \mathbf{V}_{jk}}} \sum_{i=1}^{n} F\left(\sum_{k=1}^{P}\sum_{j=1}^{m}\left(\mathrm{diag}^{-1}(\mathbf{D}_{k}^{(i)}) \odot (\mathbf{X}_{i}\mathbf{U}_{jk}\mathbf{X}_{i}^{\top})\right)\mathbf{X}_{i}\mathbf{V}_{jk}\right) + \frac{\beta}{2}\sum_{k=1}^{P}\sum_{j=1}^{m}\|\mathbf{U}_{jk}\|_{F}^{2} + \|\mathbf{V}_{jk}\|_{F}^{2} \quad \text{s.t. } (2\mathrm{diag}^{-1}(\mathbf{D}_{k}^{(i)}) - \mathbf{1}\mathbf{1}^{\top}) \odot (\mathbf{X}_{i}\mathbf{U}_{jk}\mathbf{X}_{i}^{\top}) \geq \mathbf{0}, \tag{134}$$

where $\mathrm{diag}^{-1}(\mathbf{D}_{k}^{(i)}) \in \mathbb{R}^{s \times s}$ *takes elements along the diagonal of* $\mathbf{D}_{k}^{(i)}$ *and places them in matrix form.*

Proof. Courtesy of (Sahiner et al., 2022), we first present the equivalent convex models for gated ReLU and ReLU activation self attention. First, define

$$\mathbf{X} := \begin{bmatrix} \mathbf{X}_{1} \otimes \mathbf{X}_{1} \\ \cdots \\ \mathbf{X}_{n} \otimes \mathbf{X}_{n} \end{bmatrix}$$

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

then, we have

$$p_{GSA}^{*} = \min_{\mathbf{Z}_{j} \in \mathbb{R}^{d^{2} \times dc}} \sum_{i=1}^{n} \mathcal{L}\left(\sum_{j=1}^{P}\sum_{k=1}^{d}\sum_{\ell=1}^{d} \mathbf{G}_{i,j}^{(k,\ell)}\mathbf{X}_{i}\mathbf{Z}_{j}^{(k,\ell)}, \mathbf{Y}_{i}\right) + \beta\sum_{j=1}^{P}\|\mathbf{Z}_{j}\|_{*}, \tag{135}$$

$$p_{RSA}^{*} = \min_{\mathbf{Z}_{j} \in \mathbb{R}^{d^{2} \times dc}} \sum_{i=1}^{n} \mathcal{L}\left(\sum_{j=1}^{P}\sum_{k=1}^{d}\sum_{\ell=1}^{d} \mathbf{G}_{i,j}^{(k,\ell)}\mathbf{X}_{i}\mathbf{Z}_{j}^{(k,\ell)}, \mathbf{Y}_{i}\right) + \beta\sum_{j=1}^{P}\|\mathbf{Z}_{j}\|_{\mathrm{K}_{j},*}, \tag{136}$$

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 then proceed to take the Burer-Monteiro factorization of these models. Here, we will show the Burer-Monteiro factorization of the ReLU model, noting that the proof is the same for the Gated ReLU model sans the constraints.

We let $\mathbf{Z}_j = \mathbf{U}_j \mathbf{V}_j^\top$, where $\mathbf{U}_j \in \mathbb{R}^{d^2 \times m}$ and $\mathbf{V}_j \in \mathbb{R}^{dc \times m}$, where $\mathbf{K}_j \mathbf{U}_j \geq \mathbf{0}$. Let $\text{vec}^{-1}(\mathbf{u_{jx}}) \in \mathbb{R}^{d \times d}$ be the result of taking chunks of d-length vectors from $\mathbf{u}_{jx}$ for $j \in [m]$ 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 $\mathbf{v}_{jx}$ and stacking them in columns. Furthermore, we will let $\text{vec}^{-1}(\mathbf{u_{jx}})_k$ be the kth column of $\text{vec}^{-1}(\mathbf{u_{jx}})$. Then, recognize that

$$\mathbf{Z}_j{}^{(k,\ell)} = \sum_{x=1}^{m} \text{vec}^{-1}(\mathbf{u_{jx}})_k \text{vec}^{-1}(\mathbf{v_{jx}})_\ell^\top.$$

Thus, for each j,

$$\begin{aligned}
\sum_{k=1}^{d}\sum_{\ell=1}^{d} \mathbf{G}_{i,j}^{(k,\ell)} \mathbf{X}_i \mathbf{Z}_j^{(k,\ell)} &= \sum_{x=1}^{m}\sum_{k=1}^{d}\sum_{\ell=1}^{d} \mathbf{G}_{i,j}^{(k,\ell)} \mathbf{X}_i \text{vec}^{-1}(\mathbf{u_{jx}})_k \text{vec}^{-1}(\mathbf{v_{jx}})_\ell^\top \\
&= \sum_{x=1}^{m}\sum_{k=1}^{d}\sum_{\ell=1}^{d} \left[(\mathbf{X}_i \otimes \mathbf{I}_s)^\top \mathbf{D}_j^{(i)} (\mathbf{X}_i \otimes \mathbf{I}_s)\right]^{(k,\ell)} \mathbf{X}_i \text{vec}^{-1}(\mathbf{u_{jx}})_k \text{vec}^{-1}(\mathbf{v_{jx}})_\ell^\top \\
&= \sum_{x=1}^{m}\sum_{k=1}^{d}\sum_{\ell=1}^{d} (\mathbf{X}_i[\cdot,k] \otimes \mathbf{I}_s)^\top \mathbf{D}_j^{(i)} (\mathbf{X}_i[\cdot,\ell] \otimes \mathbf{I}_s) \mathbf{X}_i \text{vec}^{-1}(\mathbf{u_{jx}})_k \text{vec}^{-1}(\mathbf{v_{jx}})_\ell^\top \\
&= \sum_{y=1}^{s}\sum_{x=1}^{m}\sum_{k=1}^{d}\sum_{\ell=1}^{d} \mathbf{D}_j^{(i)^{(y,y)}} \mathbf{X}_i \text{vec}^{-1}(\mathbf{u_{jx}})_k \text{vec}^{-1}(\mathbf{v_{jx}})_\ell^\top \mathbf{X}_i[y,k] \mathbf{X}_i[y,\ell] \\
&= \sum_{x=1}^{m} \left(\text{diag}^{-1}(\mathbf{D}_j^{(i)}) \odot (\mathbf{X}_i \mathbf{U}_{jx} \mathbf{X}_i^\top)\right) \mathbf{X}_i \mathbf{V}_{jx},
\end{aligned}$$

where the constraint that $(2\mathbf{D}_j^{(i)} - \mathbf{I})\mathbf{X}\mathbf{U}_j \geq 0$ can also be re-written as $(2\text{diag}^{-1}(\mathbf{D}_k^{(i)}) - \mathbf{1}\mathbf{1}^\top) \odot (\mathbf{X}_i \mathbf{U}_{jx} \mathbf{X}_i^\top) \geq \mathbf{0}$ for all $x \in [m]$. Thus, we have proven the statement.

□

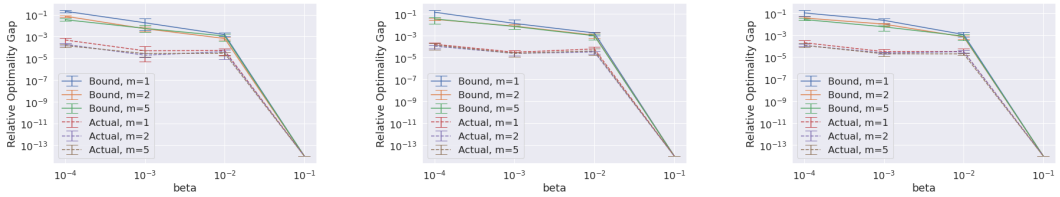
C Experimental Details

C.1 The Relative Optimality Bound

(a) $n = 45$ (b) $n = 150$

Figure 3: Example of three-class spiral dataset, with different numbers of samples n.

See Figure 3 for dataset evaluated. In all cases, we solve the BM factorization with GD using Pytorch (Paszke et al., 2019) on a CPU with a momentum parameter of 0.9, a learning rate of 1.0 which decays by a factor of 0.9 whenever the training loss plateaus, and train for 20000 epochs such that

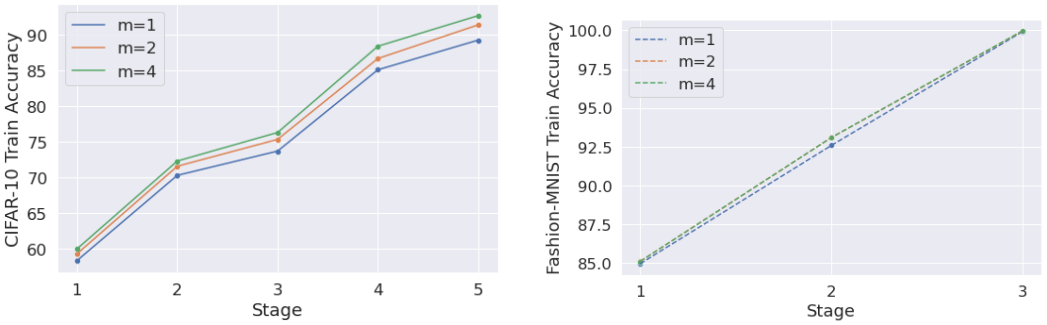


(a) $n = 3$ (b) $n = 45$ (c) $n = 75$

Figure 4: Supplement to Figure 2. Relative optimality gap of the non-convex BM factorization of a gated-ReLU two-layer MLP for three-class spiral data classification ($d = 2$, $c = 3$). For $n \in [3, 45, 75]$, we demonstrate how β and m affect relative optimality gap, both in in terms of the proposed bound and the actual gap, where the global minimum is determined by convex optimization.

GD always converges. For convex optimization, to determine the global optimum of each problem, we use the MOSEK interior point solver (Andersen & Andersen, 2000) with CVXPY (Diamond & Boyd, 2016). The parameters we evaluate are $n \in [3, 45, 75, 150]$, $\beta \in [10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}]$ and $m \in [1, 2, 5]$ (see Figure 4 for $n = 3, 45, 75$). We use a randomly subsampled set of $\hat{P} = 100$ hyperplane arrangements. We perform this experiment over three random seeds, which are used to generate the hyperplane arrangements as well as the random Gaussian initializations of the weights.

C.2 BM Enables Layerwise Training of Convex CNNs



(a) CIFAR-10 (b) Fashion-MNIST

Figure 5: Supplement to Figure 1. Training accuracies of layerwise trained BM factorized convex gated-ReLU CNNs.

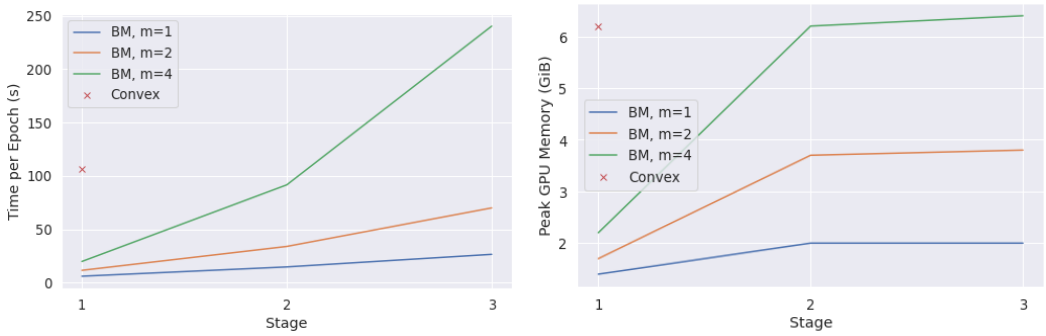
Our layerwise training procedure was trained on a single NVIDIA 1080 Ti GPU using the Pytorch deep learning library (Paszke et al., 2019). In particular, we follow the implementation of (Belilovsky et al., 2019), who proposed greedily, sequentially training two-layer CNNs. At each stage, a two-layer CNN (convolutional layer + average pooling + fully connected layer) is trained, and then the weights are frozen, the fully connected layer and average pooling are discarded, and the trained convolutional layer is used as a feature-generator for the following stage. At certain stages, before the CNN is applied, an invertible downsampling operation (Dinh et al., 2016) is used to reduce the spatial dimensions of the image. In (Belilovsky et al., 2019), ReLU activations are used as well as an average pooling operation to spatial dimensions of 2×2 (i.e. $a = 4$) is used, followed by a flattening operation and a fully connected layer. They also use a softmax cross-entropy loss, a batch size of 128, weight decay parameter of $\beta = 5e - 4$, along with stochastic gradient descent (SGD) with momentum fixed to 0.9, 50 epochs per stage, and learning rate decay by a factor of 0.2 every 15 epochs.

In our experiments, we keep all network and optimization parameters the same, aside from replacing the non-convex CNN at each stage with our convex CNN objective (23). We then apply the Burer-Monteiro factorization with $m \in [1, 2, 4]$ to this architecture to make it tractable for layerwise

learning as described in the main paper. At each stage, we randomly subsample $\hat{P} = 256$ hyperplane arrangements. We further use gated ReLU rather than ReLU activations for simplicity, which can work as well as ReLU in practice (Fiat et al., 2019). These techniques have been used effectively for convex learning to exceed the performance of two-layer non-convex neural networks (Pilanci & Ergen, 2020; Ergen & Pilanci, 2020; Ergen et al., 2021).

For the CIFAR-10 experiment, we use 5 stages (following (Belilovsky et al., 2019)), whereas for the Fashion-MNIST experiment, we use 3 stages, since the training accuracy saturates after 3 stages. We choose learning rates per stage from $\{10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}\}$ per stage based on training accuracy for CIFAR-10. The chosen learning rates were $[10^{-1}, 10^{-2}, 10^{-3}, 10^{-2}, 10^{-2}]$ for CIFAR-10. For Fashion-MNIST, we empirically observed the training loss was better optimized with slightly higher learning rates, so we used $[2 \times 10^{-1}, 5 \times 10^{-2}, 5 \times 10^{-3}]$. In Figure 5 we also provide the training accuracies at each stage for the results provided in Figure 1. Ultimately, our CIFAR-10 network with 5 stages took 9163 seconds to train, and the Fashion-MNIST network with 3 stages took 4931 seconds to train. In (Kiliçarslan & Celik, 2021), it is shown that an end-to-end 6-layer CNN with ReLU activations takes 640 seconds to train on CIFAR-10 and 285 seconds to train on Fasion-MNIST. We note that the purpose of this experiment is not to advocate for the use of layerwise BM networks over end-to-end trained networks, but simply to demonstrate the utility of the BM network in enabling convex neural networks to scale to the performance of end-to-end deep networks by using layerwise learning.

C.3 Additional Experimental Details



(a) Time per epoch.

(b) Peak GPU memory.

Figure 6: Supplement to Figure 1. Time per epoch and peak GPU memory consumption of different formulations of gated ReLU CNNs trained in a layerwise fashion on CIFAR-10. **For "Convex", training causes memory consumption to exceed NVIDIA 1080 Ti GPU's 12 GB memory capacity after the first stage, preventing ability to train convex CNN (23) in a layerwise fashion.**

In this subsection, we show additional details on the experiments in Figure 1 to show the scaling issues in using the convex formulation of the gated ReLU CNN described in (23) for layerwise training. We further compare the per-iteration complexity of the convex formulation to that of the BM factorization (126). In Figure 6, we measure the training time per epoch and GPU memory consumption across various stages of the training a BM-factorized CNN in a layerwise fashion on CIFAR-10, using the same experimental setup as described in Appendix C.2. We also compare training the BM formulation to training the convex formulation (23) in this setting. Due to memory constraints, we are not able to train more than a single stage of the convex formulation, illustrating that convex neural networks encounter major scaling issues that prevents their application to layerwise learning. We observe that the BM factorization leads to significantly reduced memory consumption and time per iteration compared to the original convex formulation, demonstrating the utility of the BM factorization for scaling convex neural networks to the layerwise training setting.