
Convex Self-Attention Mechanisms

Marshall Fisher
Computer Science & Engineering
University of California San Diego

Zhaoyang Jia
Computer Science & Engineering
University of California San Diego

Matthew O'Malley-Nichols
Computer Science & Engineering
University of California San Diego

Richard Zhang
Computer Science & Engineering
University of California San Diego

Abstract

Following the publication of "Attention is All You Need" [1], transformer models and the self-attention mechanism have become enormously popular. They dominate the current generation of deep learning, especially in natural language processing. However, self-attention is not convex, and theoretical analysis of the attention mechanism's efficacy has proven difficult. To address this gap in theoretical understanding, several works [2][3][4] have investigated convex alternatives to self-attention and the softmax function. Building on "Convexifying Transformers" [2], in this work we investigate a convex approximation of the self-attention mechanism. We discuss the primal convex optimization problem introduced by [2] and provide a novel derivation of the dual problem from [2]. We extend this formulation by preconditioning the input of the attention mechanism with feature maps induced by a convex approximation of the Gaussian kernel to afford frequency learning. Then, we examine how these convex approximations affect generalization past training convergence in the phenomenon called grokking. For completeness, we provide new write-ups of the core proofs from [2] in the Appendices.

Contents

1	Introduction	3
1.1	Motivation	3
1.2	Contributions	3
1.3	Task Assignment	3
2	Related Works	3
2.1	Transformers	3
2.2	Grokking	4
3	Problem Statement and approach	4
3.1	Self-Attention Mechanism	4
3.2	Primal statement	4
3.3	Dual derivation	5
3.4	KKT conditions	7

4	Convex Kernel Preconditioning	8
5	Experimental approach	8
5.1	Datasets	8
5.2	Multihead Convex Transformer	9
6	Experimental results	10
7	Spectral Analysis of the Convex Models	10
8	Conclusion and potential future work	12
9	Acknowledgments	12
A	Lemma 1	13
B	Theorem 1	14
C	Theorem 2	16

1 Introduction

1.1 Motivation

Self-attention is considered vital to the extraordinary performance of recent Transformer models and LLMs. The goal of this work is to study the self-attention mechanism through the lens of convex optimization. We begin with discussion and analysis of the self-attention convex approximation as presented in [2]. We then discuss a convex approximation of the Gaussian kernel to precondition the input of the attention mechanism.

To produce experimental results, we build on the code from [2] and implement the convex self-attention surrogate into a multi-head attention network, where the attention layer is a string of convex blocks with the last block outputting class logits. We also run experiments with the convex kernel approximation applied as preconditioning prior to the attention mechanism. We analyze the performance of the convex, multi-layer network on several tasks, as compared to task performance when using the original self-attention layer in the same network architecture.

1.2 Contributions

A summary of our intended contribution are the following: (1) a detailed review “Convexifying Transformers”[2] and a new presentation of its proofs, (2) an independent derivation of the convexified dual problem, (3) a summary of the problem’s KKT conditions, (4) a log-linear time convex approximation of the Gaussian kernel and its effects when used preconditioning the inputs of the attention mechanism, (5) a spectral analysis of weight matrices challenging a strict interpretation of generalization through the rank-minimization hypothesis, (6) experimental comparisons between the original nonconvex self-attention and the two convex self-attention on the dataset from [2] and more complicated synthetic datasets.

1.3 Task Assignment

All authors contributed to this project equally. M. Fisher wrote the primal statement, the dual derivation, and the KKT conditions. Z. Jia wrote this paper’s versions of the proofs for the theorems from [2], and he contributed to verifying the primal/dual and experimental results. M. O’Malley-Nichols identified, derived, and wrote the convex kernel approximations, wrote the experimental code for the kernel section, and performed spectral analysis for the grokking phenomenon. R. Zhang wrote the experimental code for the multi-layer convex attention network, and generated the experimental results comparing against non-convex performance.

2 Related Works

2.1 Transformers

Recent works have explored different reformulations of the attention mechanism in the Transformer model in order to improve interpretability, computational efficiency, and numerical stability. "Input Convex Neural Networks" [5] (ICNN) introduces constraints for neural network parameters so that the output is a convex function of a subset of the inputs. The ICNN formulation of neural networks allows for convex optimization approaches during model training, and affords convex analysis techniques to understand neural networks. Building on this, [2] reformulates the attention mechanism in transformer models as a convex optimization problem, leading to a reduction in computational overhead and improving generalization speeds.

In addition to convex formulations of attention, the attention mechanism has been viewed as an implicit similarity kernel, allowing for the reduced computational complexity of the attention mechanism. [6] approximates attention matrices using positive orthogonal random features, reducing the time complexity of attention from $O(n^2d)$ to $O(n \log d)$. In [7], Chowdhury et al. propose a framework for learning the kernel function by learning the spectral distribution of the attention kernel. They compare kernel approximations based on random Fourier features [8], structured random projections [9], and positive random features [6]. They note that a Monte Carlo approximation of the positive random features leads to a low-variance estimate of the softmax kernel compared to the unbiased estimate provided by random Fourier features. In empirical tests using long input sequences, the learned transformer kernel outperforms the regular attention mechanism in text retrieval accuracy.

2.2 Grokking

Grokking refers to a two-stage phenomenon that occurs in model training, identified by memorization and generalization phases. In the memorization phase, a model achieves high training performance with low validation performance. After a large amount of iterations, the model’s validation performance sharply increases marking the generalization phase. First discovered by [10], grokking has been tied to weight decay’s penalization on parameter norm promoting compressed representations of the training data [11], displayed by rank minimization in model weights as generalization occurs. This proposal coincides with the Lottery Ticket Hypothesis [12], which suggests over-parameterized feed forward networks contain sub-networks attaining the test performance of the full network with a similar amount of training iterations. Building on this, [13] proposes a spectral dynamics framework to distinguish memorizing and generalizing networks. The spectral dynamics allow grokking to be quantified through the effective rank of the weight matrices.

3 Problem Statement and approach

3.1 Self-Attention Mechanism

We first present a brief definition of self-attention, following the same terminology and definitions as in [2]. Given $\mathbf{X} \in \mathbb{R}^{n \times d}$, a sequence of n tokens with embedding dimension d , define the query, key, and value matrices as

$$\begin{aligned}\mathbf{Q} &= \mathbf{X}\mathbf{W}_q, \mathbf{W}_q \in \mathbb{R}^{d \times d} \\ \mathbf{K} &= \mathbf{X}\mathbf{W}_k, \mathbf{W}_k \in \mathbb{R}^{d \times d} \\ \mathbf{V} &= \mathbf{X}\mathbf{W}_v, \mathbf{W}_v \in \mathbb{R}^{d \times d}\end{aligned}$$

and define a single self-attention head as:

$$\mathbf{A} = \text{softmax}(\mathbf{Q}\mathbf{K}^\top)\mathbf{V}\mathbf{W}_o, \mathbf{W}_o \in \mathbb{R}^{d \times d}$$

where all \mathbf{W}_i are learnable weights matrices. Fully expanded, the prediction from this attention head can be expressed as:

$$\hat{\mathbf{Y}} = \text{softmax}(\mathbf{X}\mathbf{W}_q\mathbf{W}_k^\top\mathbf{X}^\top)\mathbf{X}\mathbf{W}_v\mathbf{W}_o$$

To extend this to the more popular ‘multi-head attention’ formulation, consider the prediction as the sum over h such attention heads:

$$\hat{\mathbf{Y}} = \sum_{j=1}^h \text{softmax}(\mathbf{X}\mathbf{W}_{qj}\mathbf{W}_{kj}^\top\mathbf{X}^\top)\mathbf{X}\mathbf{W}_{vj}\mathbf{W}_{oj}$$

For a training set $\{\mathbf{X}_i, \mathbf{Y}_i\}_{i=1}^N$ with $\mathbf{X} \in \mathbb{R}^{n \times d}$ and $\mathbf{Y} \in \mathbb{R}^{n \times c}$, we can define the regularized training optimization problem as:

$$\min_{\mathbf{W}_{qj}, \mathbf{W}_{kj}, \mathbf{W}_{vj}, \mathbf{W}_{oj}} \sum_{i=1}^N \mathcal{L}(\hat{\mathbf{Y}}, Y_i) + \frac{\beta}{2} \sum_{i \in \{q, k, v, o\}, j=1}^h \|\mathbf{W}_{ij}\|_F^2$$

The full regularized attention training problem is then defined as:

$$\min_{\mathbf{W}_{qj}, \mathbf{W}_{kj}, \mathbf{W}_{vj}, \mathbf{W}_{oj}} \sum_{i=1}^N \mathcal{L} \left(\sum_{j=1}^h \text{softmax}(\mathbf{X}\mathbf{W}_{qj}\mathbf{W}_{kj}^\top\mathbf{X}^\top)\mathbf{X}\mathbf{W}_{vj}\mathbf{W}_{oj}, Y_i \right) + \frac{\beta}{2} \sum_{i \in \{q, k, v, o\}, j=1}^h \|\mathbf{W}_{ij}\|_F^2 \quad (1)$$

Throughout this work, we assume the loss function \mathcal{L} to be convex. In the next section, we discuss the relaxation of the attention training problem (1) into a convex optimization problem.

3.2 Primal statement

Following equation (1), the paper [2] makes the first key relaxation towards convexifying the problem. Observe that the softmax function is row-unit-normalized, so the row sum must be 1. Since the softmax is also a ratio of exponentials, each value must be non-negative. Therefore, the rows of the softmax output are each constrained by a unit simplex constraint. Consider the unit simplex constraints $\Delta := \{\mathbf{W} \in \mathbb{R}^{n \times n} : \mathbf{w}_i \geq 0, \mathbf{1}^\top \mathbf{w}_i = 1, \forall i \in [n]\}$; these constraints are convex, since they form an intersection of half-spaces and linear constraints.

As discussed, all softmax output must follow these set of unit-simplex constraints, so for any $\mathbf{U} \in \mathbb{R}^{n \times n}$, there exists $\mathbf{W} \in \Delta$ such that $\text{softmax}(\mathbf{U})\mathbf{X} = \mathbf{W}\mathbf{X}$.

The regularized attention training, after relaxing softmax to a unit-simplex linear operator, is reformulated to:

$$\min_{\mathbf{w}_{1j} \in \Delta, \mathbf{w}_{2j} \in \mathbb{R}^{d \times d}, \mathbf{w}_{3j} \in \mathbb{R}^{d \times c}} \sum_{i=1}^N \mathcal{L} \left(\sum_{j=1}^h \mathbf{w}_{1j} \mathbf{X}_i \mathbf{w}_{2j} \mathbf{w}_{3j}, \mathbf{Y}_i \right) + \frac{\beta}{2} \left(\sum_{j=1}^h \|\mathbf{w}_{2j}\|_F^2 + \|\mathbf{w}_{3j}\|_F^2 \right).$$

The paper [2] then makes a second key relaxation, simplifying the problem from learning weight matrices \mathbf{W} to learning weight vectors \mathbf{w} . As a proof-of-concept study by the author, the derivations were limited to learning vector weights. We acknowledge that the bi-dual derivations will be much more challenging with matrix weights, which is worth future investigation.

$$\min_{\mathbf{w}_{1j} \in \Delta, \mathbf{w}_{2j} \in \mathbb{R}^d, \mathbf{w}_{3j} \in \mathbb{R}^c} \sum_{i=1}^N \mathcal{L} \left(\sum_{j=1}^h \mathbf{w}_{1j} \mathbf{X}_i \mathbf{w}_{2j} \mathbf{w}_{3j}, \mathbf{y}_i \right) + \frac{\beta}{2} \left(\sum_{j=1}^h \|\mathbf{w}_{2j}\|_2^2 + \|\mathbf{w}_{3j}\|_1^2 \right). \quad (2)$$

An equivalent l_1 regularization term using only \mathbf{w}_3 was derived in the original paper (Lemma 1), which we have reviewed and explained in Appendix A:

$$\min_{\mathbf{w}_{1j} \in \Delta, \mathbf{w}_{2j}: \|\mathbf{w}_{2j}\|_2 \leq 1, \mathbf{w}_{3j} \in \mathbb{R}^c} \sum_{i=1}^N \mathcal{L} \left(\sum_{j=1}^h \mathbf{w}_{1j}^T \mathbf{X}_i \mathbf{w}_{2j} \mathbf{w}_{3j}, \mathbf{y}_i \right) + \beta \|\mathbf{w}_{3j}\|_1 \quad (3)$$

The primal Equation 3 is nonconvex, so the authors take its dual, then convexify the dual constraints to the equivalent convex dual formulation below. The derivation for scalar output was noted in Theorem 1 of the original paper and expanded to vector output in Theorem 2. We have reviewed and produced a detailed expansion in Appendix B, C. The convex dual for the vector output is:

$$\max_{\mathbf{v} \in \mathbb{R}^N} -\mathcal{L}^* \left(\{\mathbf{v}_i\}_{i=1}^N, \{\mathbf{y}_i\}_{i=1}^N \right) \quad \text{s.t.} \quad \left\| \sum_{i=1}^N \mathbf{v}_i \mathbf{x}_{ik} \right\|_2 \leq \beta, \quad \forall k \in [n], \forall l \in [c]. \quad (4)$$

Then the authors of [2] take the dual again (the bidual of the original problem), and since the convexified dual is equivalent to the initial nonconvex dual, the following convex primal is equivalent to the initial nonconvex primal. We conducted an independent derivation in the next section to further verify the relationship by going in the opposite direction, from this convex primal to its convex dual.

$$\min_{\mathbf{z}_l \in \mathbb{R}^{n \times d}} \sum_{i=1}^N \sum_{l=1}^c \mathcal{L} \left(\text{trace}(\mathbf{z}_l^T \mathbf{X}_i), y_{il} \right) + \beta \sum_{l=1}^c \sum_{k=1}^n \|\mathbf{z}_{lk}\|_2. \quad (5)$$

This is a convex primal optimization problem equivalent to Equation 3. We use equation (5) as our primal for analysis, and for the experiments in Section 5 and Section 6.

3.3 Dual derivation

Based on the convexified primal problem, we conduct an independent derivation of its dual, and the result is equivalent to the convexified dual from the paper [2]. Consultation with gpt-4o [14] was used in the construction of this derivation.

First, we restate the primal problem:

$$\min_{\mathbf{z}_l \in \mathbb{R}^{n \times d}} \sum_{i=1}^N \sum_{l=1}^c \mathcal{L} \left(\text{trace}(\mathbf{z}_l^T \mathbf{X}_i), y_{il} \right) + \beta \sum_{l=1}^c \sum_{k=1}^n \|\mathbf{z}_{lk}\|_2.$$

where $\mathbf{X}_i \in \mathbb{R}^{n \times d}$ and $y_{il} \in \mathbb{R}$. To convert the primal into a constrained optimization problem, we introduce the dummy variable $\mathbf{U}_l = \mathbf{z}_l$ and rewrite the problem as:

$$\min_{\mathbf{z}_l, \mathbf{U}_l} \sum_{i=1}^N \sum_{l=1}^c \mathcal{L} \left(\text{trace}(\mathbf{z}_l^T \mathbf{X}_i), y_{il} \right) + \beta \sum_{l=1}^c \sum_{k=1}^n \|\mathbf{U}_{lk}\|_2, \quad (6)$$

$$\text{s.t. } \mathbf{Z}_l = \mathbf{U}_l, \quad \forall l = 1, \dots, c.$$

Next, we define the Lagrangian using multiplier variables $\mathbf{\Lambda}_l \in \mathbb{R}^{n \times d}$:

$$L(\mathbf{Z}_l, \mathbf{U}_l, \mathbf{\Lambda}_l) = \sum_{i=1}^N \sum_{l=1}^c \mathcal{L}(\text{trace}(\mathbf{Z}_l^\top \mathbf{X}_i), y_{il}) + \sum_{l=1}^c \sum_{k=1}^n \beta \|\mathbf{U}_{lk}\|_2 + \sum_{l=1}^c \text{trace}(\mathbf{\Lambda}_l^\top (\mathbf{Z}_l - \mathbf{U}_l)). \quad (7)$$

By the properties of the trace operator, this is equivalent to:

$$L(\mathbf{Z}_l, \mathbf{U}_l, \mathbf{\Lambda}_l) = \sum_{i=1}^N \sum_{l=1}^c \mathcal{L}(\text{trace}(\mathbf{Z}_l^\top \mathbf{X}_i), y_{il}) + \sum_{l=1}^c \sum_{k=1}^n (\beta \|\mathbf{U}_{lk}\|_2 - \mathbf{\Lambda}_{lk}^\top \mathbf{U}_{lk}) + \sum_{l=1}^c \text{trace}(\mathbf{\Lambda}_l^\top \mathbf{Z}_l)$$

Now we can define

$$g(\mathbf{\Lambda}_l) = \min_{\mathbf{U}_l, \mathbf{Z}_l} \left[\sum_{i=1}^N \sum_{l=1}^c \mathcal{L}(\text{trace}(\mathbf{Z}_l^\top \mathbf{X}_i), y_{il}) + \sum_{l=1}^c \sum_{k=1}^n (\beta \|\mathbf{U}_{lk}\|_2 - \mathbf{\Lambda}_{lk}^\top \mathbf{U}_{lk}) + \sum_{l=1}^c \text{trace}(\mathbf{\Lambda}_l^\top \mathbf{Z}_l) \right]$$

By rearranging the finite summation and separating the minimization terms, g equivalently becomes:

$$g(\mathbf{\Lambda}_l) = \sum_{l=1}^c \left(\min_{\mathbf{Z}_l} \left[\text{trace}(\mathbf{\Lambda}_l^\top \mathbf{Z}_l) + \sum_{i=1}^N \mathcal{L}(\text{trace}(\mathbf{Z}_l^\top \mathbf{X}_i), y_{il}) \right] + \min_{\mathbf{U}_l} \left[\sum_{k=1}^n (\beta \|\mathbf{U}_{lk}\|_2 - \mathbf{\Lambda}_{lk}^\top \mathbf{U}_{lk}) \right] \right)$$

First we aim to solve the minimization over \mathbf{Z}_l . Assuming \mathcal{L} to be a convex loss function, we can express it in Fenchel dual form as the conjugate of its own conjugate:

$$\mathcal{L}(\text{trace}(\mathbf{Z}_l^\top \mathbf{X}_i), y_{il}) = \sup_{\alpha_{il}} [\alpha_{il} \text{trace}(\mathbf{Z}_l^\top \mathbf{X}_i) - \mathcal{L}^*(\alpha_{il}, y_{il})]$$

And thus we have

$$\min_{\mathbf{Z}_l} \left[\text{trace}(\mathbf{\Lambda}_l^\top \mathbf{Z}_l) + \sum_{i=1}^N \sup_{\alpha_{il}} [\alpha_{il} \text{trace}(\mathbf{Z}_l^\top \mathbf{X}_i) - \mathcal{L}^*(\alpha_{il}, y_{il})] \right]$$

Rearranging terms and applying the linearity of trace, we have equivalently

$$\min_{\mathbf{Z}_l} \left[\sup_{\alpha_{il}} \left[-\mathcal{L}^*(\alpha_{il}, y_{il}) + \text{trace}(\mathbf{Z}_l^\top (\mathbf{\Lambda}_l + \sum_{i=1}^N \alpha_{il} \mathbf{X}_i)) \right] \right]$$

Next we claim that it is valid to swap the min and sup order via Von Neumann's minimax theorem [15]. Let

$$f(\mathbf{Z}_l, \alpha_{il}) = -\mathcal{L}^*(\alpha_{il}, y_{il}) + \text{trace}(\mathbf{Z}_l^\top (\mathbf{\Lambda}_l + \sum_{i=1}^N \alpha_{il} \mathbf{X}_i))$$

Note that $\text{trace}(\mathbf{Z}_l^\top \mathbf{M})$ is convex in \mathbf{Z}_l and therefore f is convex in \mathbf{Z}_l . \mathcal{L}^* is convex in α_{il} , so $-\mathcal{L}^*$ is concave. Lastly, if we assume bounded values for \mathbf{Z}_l and α_{il} , then the min/max occurs over a compact convex set. Thus we have:

$$\min_{\mathbf{Z}_l} \max_{\alpha_{il}} f(\mathbf{Z}_l, \alpha_{il}) = \max_{\alpha_{il}} \min_{\mathbf{Z}_l} f(\mathbf{Z}_l, \alpha_{il})$$

which allows us to rewrite the expression above as

$$\sup_{\alpha_{il}} \left[\min_{\mathbf{Z}_l} \left[-\mathcal{L}^*(\alpha_{il}, y_{il}) + \text{trace}(\mathbf{Z}_l^\top (\mathbf{\Lambda}_l + \sum_{i=1}^N \alpha_{il} \mathbf{X}_i)) \right] \right]$$

This expression is unbounded below for $\mathbf{\Lambda}_l + \sum_{i=1}^N \alpha_{il} \mathbf{X}_i \neq \mathbf{0}$, so we arrive at our first dual constraint:

$$\mathbf{\Lambda}_l = - \sum_{i=1}^N \alpha_{il} \mathbf{X}_i$$

Under which the expression above reduces to

$$\sup_{\alpha_{il}} -\mathcal{L}^*(\alpha_{il}, y_{il})$$

Now we aim to solve the second minimization term over \mathbf{U}_l :

$$\min_{\mathbf{U}_l} \left[\sum_{k=1}^n (\beta \|\mathbf{U}_{lk}\|_2 - \mathbf{\Lambda}_{lk}^\top \mathbf{U}_{lk}) \right]$$

where $\mathbf{U}_{lk} \in \mathbb{R}^d$ and $\mathbf{\Lambda}_{lk} \in \mathbb{R}^d$. Note that the summation has one term per row of \mathbf{U}_l , and each row \mathbf{U}_{lk} can be set independently, so the minimization expression can be separated over each term in the summation. This provides

$$\min_{\mathbf{U}_{lk}} (\beta \|\mathbf{U}_{lk}\|_2 - \mathbf{\Lambda}_{lk}^\top \mathbf{U}_{lk})$$

First suppose $\|\mathbf{\Lambda}_{lk}\| > \beta$; then $\mathbf{\Lambda}_{lk}^\top \mathbf{U}_{lk} > \beta \|\mathbf{U}_{lk}\|_2$ is achievable and the minimization is unbounded below. If $\|\mathbf{\Lambda}_{lk}\| \leq \beta$, then $\mathbf{\Lambda}_{lk}^\top \mathbf{U}_{lk} \leq \beta \|\mathbf{U}_{lk}\|_2$ is guaranteed and the minimization is bounded below at 0. This gives us our second dual constraint:

$$\|\mathbf{\Lambda}_{lk}\|_2 \leq \beta$$

Combining our two dual constraints gives

$$\left\| \sum_{i=1}^N \alpha_{il} \mathbf{X}_{ik} \right\|_2 \leq \beta$$

and we can write the dual problem as

$$\max_{\alpha_{il}} \sum_{i=1}^N \sum_{l=1}^c -\mathcal{L}^*(\alpha_{il}, y_{il}) \quad (8)$$

$$\text{subject to } \left\| \sum_{i=1}^N \alpha_{il} \mathbf{X}_{ik} \right\|_2 \leq \beta \quad \forall l \in \{1, \dots, c\}, k \in \{1, \dots, n\}$$

Note that with the transformation of variable $\alpha_{il} = \mathbf{v}_{il}$, this result (8) is equivalent to the original dual derived in [2], and written here in equation (4).

3.4 KKT conditions

Using the primal from (6), with the \mathbf{U}_{lk} substitution performed to help derive the Dual, its corresponding Lagrangian from (7), and the dual problem from (8), we have the following KKT conditions:

Primal feasibility

$$\mathbf{Z}_l = \mathbf{U}_l \quad \forall l \in \{1, \dots, l\}$$

Dual feasibility

$$\left\| \sum_{i=1}^N \alpha_{il} \mathbf{X}_{ik} \right\|_2 \leq \beta \quad \forall l \in \{1, \dots, l\}, k \in \{1, \dots, n\}$$

Complementary slackness

$$\begin{aligned} \mathbf{\Lambda}_l^\top (\mathbf{Z}_l - \mathbf{U}_l) &= \mathbf{0}_{n \times d} \\ \lambda_{lk} (\left\| \sum_{i=1}^N \alpha_{il} \mathbf{X}_{ik} \right\|_2 - \beta) &= 0, \quad \lambda_{lk} \geq 0 \end{aligned}$$

Stationarity

$$\begin{aligned} \frac{\partial L}{\partial \mathbf{Z}_l} &= \sum_{i=1}^N \sum_{l=1}^c \frac{\partial \mathcal{L}}{\partial \mathbf{Z}_l} (\text{trace}(\mathbf{Z}_l^\top \mathbf{X}_i), y_{il}) \mathbf{X}_i + \sum_{l=1}^c \mathbf{\Lambda}_l = \mathbf{0}_{n \times d} \\ \frac{\partial L}{\partial \mathbf{U}_l} &= \beta G(\mathbf{U}_l) - \sum_{l=1}^c \mathbf{\Lambda}_l = \mathbf{0}_{n \times d} \end{aligned}$$

where

$$G(\mathbf{U}_l) = \begin{bmatrix} g(\mathbf{U}_{l1}) \\ \vdots \\ g(\mathbf{U}_{ln}) \end{bmatrix}$$

and

$$g(\mathbf{U}_{lk}) = \begin{cases} \frac{\mathbf{U}_{lk}}{\|\mathbf{U}_{lk}\|_2} & \text{if } \mathbf{U}_{lk} \neq \mathbf{0}_d \\ \mathbf{0}_d & \text{otherwise} \end{cases}$$

i.e. $g(\mathbf{U}_{lk})$ is a subgradient of $\|\mathbf{U}_{lk}\|_2$.

4 Convex Kernel Preconditioning

The previously derived convex formulation of self-attention increases interpretability through the convex analysis framework but may limit model expressivity. In order to increase pattern discovery we precondition the input x to each attention mechanism by applying an approximate Gaussian kernel feature map.

We start by considering the approximate Gaussian feature mapping with positive random features provided by [7]:

$$\phi(x) = \exp\left(\frac{1}{\sigma\sqrt{d}}SHG\Pi HB - (|x^2| + \frac{1}{2}\log d)\right)$$

Where d is the feature mapping dimensions, S is a diagonal scaling matrix with elements pulled from the Chi distribution and scaled by the norm of G , H is the Walsh-Hadamard matrix, G is a matrix with diagonal values pulled from the normalized Gaussian distribution, $\Pi \in \{0, 1\}$ is a permutation matrix, and B is a diagonal matrix with values pulled from the Bernoulli distribution. This formulation gives us $O(nd \log d)$ time complexity compared to $O(ndD)$ with Random Kitchen Sinks [8].

We note that Gaussian scaling can produce values on the entire real number line, leading to non-monotonic transformations caused by sign-flipping. As such, we replace the Gaussian matrix G with a matrix G_χ with elements pulled from the Chi-distribution, as it approximates the Gaussian distribution as the amount of random variables (random features) increase. This allows us to remove the exponential function and therefore the offset terms $-(|x^2| + \frac{1}{2}\log d)$, simplifying our calculations. We now have the convex approximations of the feature maps given by $\phi_{cvx}(x)$:

$$\phi_{cvx}(x) = \frac{1}{\sigma\sqrt{d}}SHG_\chi\Pi HB$$

With these convex feature mappings, we define the primal kernel-preconditioned optimization problem as follows:

$$\min_{\mathbf{Z}_l \in \mathbb{R}^{n \times d}} \sum_{i=1}^N \sum_{l=1}^c \mathcal{L}(\text{trace}(\mathbf{Z}_l^\top \phi_{cvx}(\mathbf{X}_i)), y_{il}) + \beta \sum_{l=1}^c \sum_{k=1}^n \|\mathbf{Z}_{lk}\|_2. \quad (9)$$

5 Experimental approach

5.1 Datasets

We generate two synthetic datasets based on mathematical formulas to test the models' performance and ability to generalize. We selected a synthetic Modular Division dataset to selectively assess the model, based on what was used in [2]. Each sample contains a modular division operation, and the model is trained to predict the resulting number c :

$$(a/b) \bmod p \equiv c$$

The modulus p is unknown to the network during training and evaluation, and must be inferred. Gromov [16] demonstrated that a minimal neural network architecture is capable of learning modular arithmetic

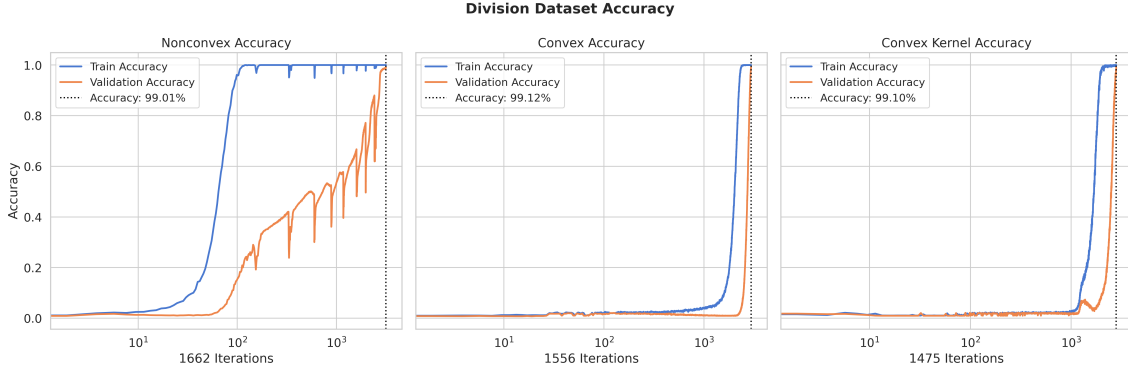


Figure 1: Accuracy on the Division Dataset

tasks exhibiting grokking—a phenomenon where models abruptly achieve perfect generalization after initially overfitting. The division dataset has minimal complexity but is sufficiently rich for exploring this phenomenon. Then we chose another dataset called the Periodicity dataset, which generates a discrete, periodic pattern:

$$z = [(x^{-1} + y^2) \times (2x \bmod [\log_2 97])] \bmod 97$$

This dataset was chosen for its higher complexity than basic modulo arithmetic. This tests the network’s capacity to learn and represent more intricate algebraic structures, and whether increased complexity impedes or delays generalization.

We also tried to use the MNIST dataset for image classification, but the pixel dimensions required a larger model size which could not fit onto our GPU memory. An additional synthetic dataset with nonlinear operations was tested, but none of the three models were able to improve their validation performance after 30,000 epochs.

5.2 Multihead Convex Transformer

To evaluate the efficacy of the proposed convex self-attention mechanism, we implemented a two-layer architecture and compared its performance to the standard nonconvex self-attention mechanism. The goal was to assess generalization, computational efficiency, and learning dynamics across multiple tasks, including modular arithmetic.

For the convex model, the embedding dimension was set to 128, with one attention head per layer and a batch size of 32. Training employed the Adam optimizer with a learning rate of 1e-3, a regularization parameter (beta) of 1e-4, weight decay set at 1e-6, and a fixed training budget of 300,000 iterations.

We used cross entropy and regularization as our loss function to classify the performance of the model’s outputs.

$$\mathcal{L} = \frac{1}{N} \text{CrossEntropy}(y_{\text{hat}}, y) + \frac{\beta}{N} \left(\|Z\|_2^2 + \|Z2\|_2^2 + \sum_{\text{MLP params}} \|p\|_2^2 \right)$$

where y_{hat} are the predicted logits, y are the true labels, β is the regularization parameter, and N is the batch size.

Experiments were executed on 1 T4 GPU on Google Colab. Training continued until convergence criteria were met or until validation accuracy surpassed 99%, triggering early stopping.

The nonconvex transformer contains identical hyper-parameters except for the weight decay, which is 1 compared to $1e - 6$ in the convex transformer.

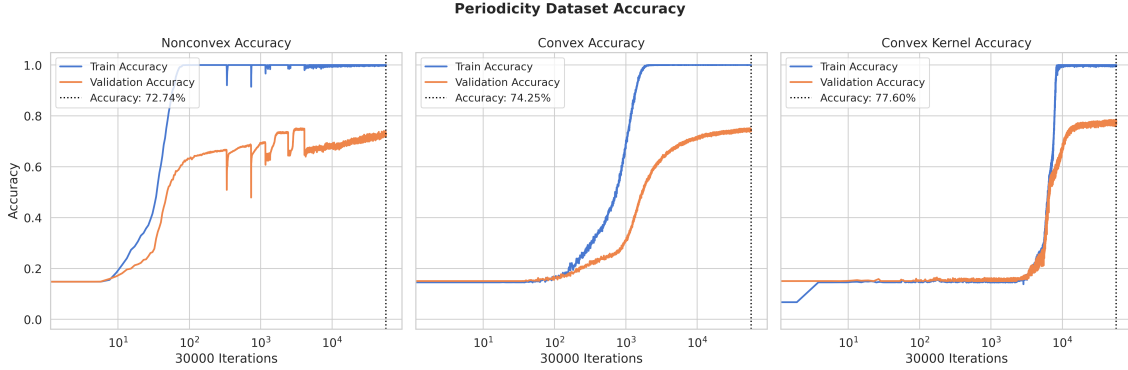


Figure 2: Accuracy on the Periodicity Dataset

6 Experimental results

Figure 1 and Figure 2 show the experimental results comparing the convex and non-convex model performance for the two datasets. The convex models display smoother accuracy convergence than in the non-convex case. This is a reassuring result, as smooth convergence is one of the main advantages of convex optimization approaches. In Figure 1 we observe that the grokking phenomenon for modular division is diminished, in that there are fewer iterations between train and validation convergence in the convex case, but the grokking phenomenon still occurs to some extent: perfect train accuracy is first achieved, followed by a sudden increase in validation accuracy.

For the periodicity dataset in Figure 2, we note that the increased problem difficulty prevented convergence to perfect validation accuracy in both the convex and non-convex case; this is perhaps reasonable given the relatively small size of the experimental models. Still, the convex models both demonstrate desirably smooth convergence to higher validation accuracy than the non-convex model. On this dataset, the simultaneous convergence of train and validation accuracy for the convex kernel method suggests that this method may help alleviate grokking.

For both datasets, our convex kernel model outperformed the convex model which outperforms the nonconvex model. Figure 1 shows the performance by the number of iterations required until convergence for each model, and Figure 2 shows the performance by the final validation accuracy attained.

The computational efficiency of each model configuration on the division dataset is documented in Table 1, summarizing average training times per epoch and total training durations.

Model Configuration	Seconds/Epoch	Total Training Time (s)	Total Epochs to Convergence
Convex	0.088	136.26	1556
Convex Kernel	0.38	567.70	1475
Nonconvex	0.23	390.51	1662

Table 1: Computational efficiency of different model configurations.

7 Spectral Analysis of the Convex Models

For each dataset, we track the distribution of singular values in the first and second layer \mathbf{Z} weight matrix for our convex models.

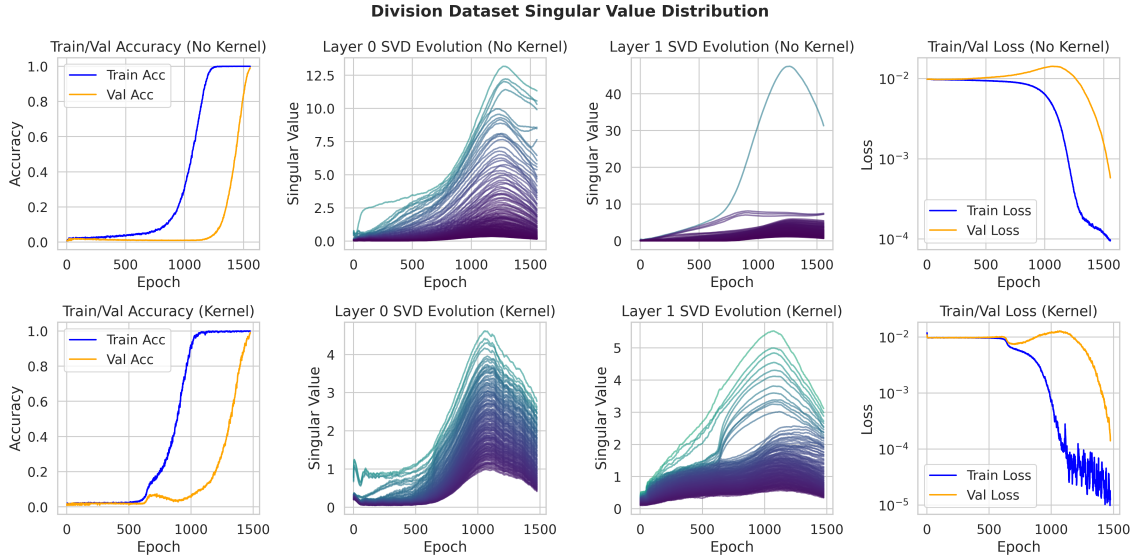


Figure 3: Division Dataset Singular Value Distribution

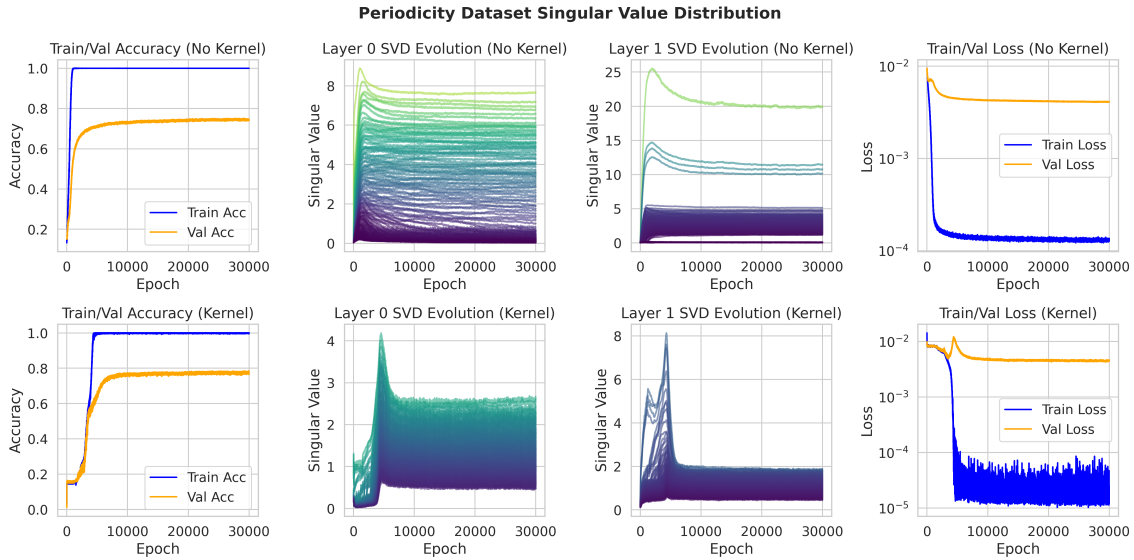


Figure 4: Periodicity Dataset Singular Value Distribution

For the standard convex model, early training increases the magnitude and spread of the singular values in the first layer, suggesting the weight matrices are beginning to encode a high effective rank representation of the dataset. Inversely, the second layer’s distribution is highly anisotropic, implying the weight matrices’ effective rank has collapsed as it finds a low dimensional representation of the problem. Notably, the spectral norms’ increase in magnitude relative to other singular values coincides with a sharp increase in training accuracy. As the validation accuracy begins to increase and the model begins to generalize, the magnitudes of the singular values decrease, implying the second layer has found a compressed representation of the problem. These results follow the rank minimization theory of grokking, which posits that grokking occurs as the model learns a representation of the memorized problem using only a few dominant directions.

The kernel preconditioned model exhibits the same trend of increasing singular value magnitude during memorization, and decreasing magnitude during generalization. However, the effective rank of the weight matrices does not collapse, and the magnitudes increase isotropically in the first layer. The second layer singular

value distribution initially exhibits an anisotropic distribution, though the effective rank increases as training continues. Interestingly, as training continues past initial generalization, the effective rank of the matrices increase as the singular values become more uniformly distributed. These results imply that random projections implicitly acts as a spectral regularizer, and generalization can occur without rank collapse in contradiction to the rank minimization hypothesis.

We therefore lead to the hypothesis that rank minimization is an artifact of standard training dynamics, rather than a strict cause of generalization capabilities. Additionally, our results support the idea that kernel preconditioning can improve generalization without rank collapse.

8 Conclusion and potential future work

In this paper, we have explored primal and dual problems derived from augmenting the self-attention architecture to be convex. Experimentally, we evaluated how this convexification affects convergence during training. We discover that kernel preconditioning leads to alternate routes to generalization beyond the rank minimization hypothesis of grokking, implying that generalization occurs through a separate complexity reduction process.

In the process of convexifying the attention formulation, two key relaxations were applied: (1) relaxing the rows of the softmax as a vector of simplex constraints, and (2) limiting the scope of learning to just vector weights. Both are worth investigating further. For relaxation (1), though the relaxed variable share the same probability space as the softmax function, there is no guarantee of a similar learning rate and convergence behavior. Relaxation (2) is more of a mathematical challenge; while there likely exists a similar convexification to derive an equivalent, convex bi-dual problem, the derivation will be much more challenging. It will be interesting to see this convexification applied to the general self-attention mechanism with full matrix weights and applied to similar experiments as those we have done in this paper.

9 Acknowledgments

We would like to thank the authors of [2], [7], and [13] for publishing code alongside the papers. Our experimental code was adapted from these releases.

References

- [1] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, 2017.
- [2] Tolga Ergen, Behnam Neyshabur, and Harsh Mehta. Convexifying transformers: Improving optimization and understanding of transformer networks. *arXiv preprint arXiv:2211.11052*, 2022.
- [3] Arda Sahiner, Tolga Ergen, Batu Ozturkler, John Pauly, Morteza Mardani, and Mert Pilanci. Unraveling attention via convex duality: Analysis and interpretations of vision transformers. *arXiv preprint arXiv:2205.08078*, 2022.
- [4] André F. T. Martins and Ramón Fernandez Astudillo. From softmax to sparsemax: A sparse model of attention and multi-label classification. *arXiv preprint arXiv:1602.02068*, 2016.
- [5] Brandon Amos, Lei Xu, and J. Zico Kolter. Input convex neural networks, 2017.
- [6] Krzysztof Choromanski, Valerii Likhoshesterov, David Dohan, Xingyou Song, Andreea Gane, Tamas Sarlos, Peter Hawkins, Jared Davis, Afroz Mohiuddin, Lukasz Kaiser, David Belanger, Lucy Colwell, and Adrian Weller. Rethinking attention with performers, 2022.
- [7] Sankalan Pal Chowdhury, Adamos Solomou, Avinava Dubey, and Mrinmaya Sachan. On learning the transformer kernel, 2022.
- [8] Ali Rahimi and Benjamin Recht. Random features for large-scale kernel machines. In *Neural Information Processing Systems*, 2007.

- [9] Quoc Viet Le, Tamas Sarlos, and Alexander Johannes Smola. Fastfood: Approximate kernel expansions in loglinear time, 2014.
- [10] Alethea Power, Yuri Burda, Harri Edwards, Igor Babuschkin, and Vedant Misra. Grokking: Generalization beyond overfitting on small algorithmic datasets, 2022.
- [11] Kaifeng Lyu, Jikai Jin, Zhiyuan Li, Simon S. Du, Jason D. Lee, and Wei Hu. Dichotomy of early and late phase implicit biases can provably induce grokking, 2024.
- [12] Jonathan Frankle and Michael Carbin. The lottery ticket hypothesis: Finding sparse, trainable neural networks, 2019.
- [13] David Yunis, Kumar Kshitij Patel, Samuel Wheeler, Pedro Savarese, Gal Vardi, Karen Livescu, Michael Maire, and Matthew R. Walter. Approaching deep learning through the spectral dynamics of weights, 2024.
- [14] OpenAI. Gpt-4o: Openai’s multimodal large language model. Online, 2024. Accessed: 2024-03-18.
- [15] J. von Neumann. Zur theorie der gesellschaftsspiele. *Mathematische Annalen*, 100:295–320, 1928.
- [16] Andrey Gromov. Grokking modular arithmetic, 2023.

A Lemma 1

The proofs for Lemma 1, Theorem 1, and Theorem 2 originally appear in [2]. Our proofs follow similar arguments, rewritten by us to include the mathematical language we find most direct and legible.

First, we prove that when the output is a scalar (i.e. $y_i \in \mathbb{R}, w_{3j} \in \mathbb{R}$), the claimed minimization of the l_1 term is equivalent to the minimization of the original regularization term. Then, observe that when the output is a vector, this result holds.

We begin with the minimization of the original regularization term

$$\min_{w_{2j} \in \mathbb{R}^d, w_{3j} \in \mathbb{R}} \frac{\beta}{2} \sum_{j=1}^h (\|w_{2j}\|_2^2 + (w_{3j})^2)$$

and we do not attempt to alter the minimization of the loss function term

$$\min_{w_{1j} \in \Delta, w_{2j} \in \mathbb{R}^d, w_{3j} \in \mathbb{R}} \sum_{i=1}^N \mathcal{L} \left(\sum_{j=1}^h w_{1j}^T X_i w_{2j} w_{3j}, y_i \right)$$

The proof is as follows:

We perform a scaling, $w_{2j} = \alpha_j w_{2j}$, $w_{3j} = \frac{w_{3j}}{\alpha_j}$ s.t. $\alpha_j > 0$. By AM-GM inequality ($x + y \geq 2\sqrt{xy}$), let $x = \alpha_j^2 \|w_{2j}\|_2^2$ and $y = \frac{(w_{3j})^2}{\alpha_j^2}$, we have

$$\sum_{j=1}^h (\|w_{2j}\|_2^2 + (w_{3j})^2) \geq 2 \sum_{j=1}^h (\|w_{2j}\|_2 |w_{3j}|)$$

In particular, the equality is achieved when $\alpha_j = \sqrt{\frac{|w_{3j}|}{\|w_{2j}\|_2}}$. Observe that with $\alpha_j > 0$, the product $w_{2j} w_{3j}$ remains unchanged in the minimization of the loss function term. Therefore, we can freely pick α_j so these two terms are equal. We then get the equivalent minimization of regularization term

$$\min_{w_{2j} \in \mathbb{R}^d, w_{3j} \in \mathbb{R}} \beta \sum_{j=1}^h (\|w_{2j}\|_2 |w_{3j}|)$$

Perform a variable change, $w'_{2j} = \frac{w_{2j}}{\|w_{2j}\|_2}$, $w'_{3j} = w_{3j} \|w_{2j}\|_2$ (since only the magnitude influences the regularization term, we separate the unit-direction w'_{2j} and integrate its magnitude into w'_{3j}). We get

$\|w_{2j}\|_2|w_{3j}| = |w'_{3j}|$. Observe in the loss function term, when $\|w'_{2j}\|_2 = 1$, the product $w_{2j}w_{3j} = w'_{2j}w'_{3j}$. We can further relax this to be $\|w'_{2j}\|_2 \leq 1$ as the minimization of $|w'_{3j}|$ is independent of length of the direction vector, $\|w'_{2j}\|_2$. As a result, since substitution of variables is the only modification on the minimization of the loss function term, the minimization of whole expression (loss + regularization) is equivalent when we swap-in the equivalent minimization of the regularization term.

$$\min_{w_{1j} \in \Delta, w'_{2j}: \|w'_{2j}\|_2 \leq 1, w'_{3j} \in \mathbb{R}} \sum_{i=1}^N \mathcal{L} \left(\sum_{j=1}^h w_{1j}^T X_i w'_{2j} w'_{3j}, y_i \right) + \beta \|w'_{3j}\|_1$$

For vector output, $y_i \in \mathbb{R}^c$, we have $w_{3j} \in \mathbb{R}^c$. The two transformations done to w_{3j} are 1) scaling by $\alpha_j > 0$ and 2) a change of variable, redefining w_{3j} to be $w_{3j} * \|w_{2j}\|_2$, effectively scaling by the magnitude of w_{2j} . Both operations will hold when w_{3j} is now a vector. The AM-GM inequality in step 1 also holds because it is establishing the inequality between norms. Thus, this lemma can be generalized to vector output.[2]

$$\min_{w_{1j} \in \Delta, w'_{2j}: \|w'_{2j}\|_2 \leq 1, w'_{3j} \in \mathbb{R}^c} \sum_{i=1}^N \mathcal{L} \left(\sum_{j=1}^h w_{1j}^T X_i w'_{2j} w'_{3j}, y_i \right) + \beta \|w'_{3j}\|_1$$

B Theorem 1

We aim to find an equivalent convex primal problem to the original, nonconvex primal problem. In summary, we find the dual of the optimization and reformulate a set of convex dual constraints, arriving at a convex dual problem. Then we take the dual of the dual (the bidual), which returns us an equivalent, convex primal problem. We first prove the case for a scalar prediction $y_i \in \mathbb{R}$, and then prove an analogous result for multi-dimensional predictions $y_i \in \mathbb{R}^d$ in Theorem 2 below. The original, scalar, nonconvex primal problem is as follows

$$\min_{w_{ij} \in \Delta, \|w_{2j}\|_2 \leq 1, w_{3j} \in \mathbb{R}} \sum_{i=1}^N \mathcal{L} \left(\sum_{j=1}^h w_{1j}^T X_i w_{2j} w_{3j}, y_i \right) + \beta \|w_3\|_1$$

Let $\hat{y}_i = \sum_{j=1}^h w_{1j}^T X_i w_{2j} w_{3j}$, $\forall i \in \{1, \dots, n\}$. Take this as a set of equality constraints and formulate the Lagrangian (L denotes the Lagrangian, while \mathcal{L} denotes the loss function):

$$\begin{aligned} L(v, y, w_3) &= \sum_{i=1}^N \mathcal{L}(\hat{y}_i, y_i) + \beta \|w_3\|_1 + \sum_{i=1}^N v_i \left(\hat{y}_i - \sum_{j=1}^h w_{1j}^T X_i w_{2j} w_{3j} \right) \\ &= \sum_{i=1}^N \mathcal{L}(\hat{y}_i, y_i) + \sum_{i=1}^N v_i \hat{y}_i + \beta \|w_3\|_1 - \sum_{j=1}^h \sum_{i=1}^N v_i w_{1j}^T X_i w_{2j} w_{3j} \end{aligned}$$

Notice that the only terms that depend on w_3 are the last two terms, $\beta \|w_3\|_1 - \sum_{i=1}^N \sum_{j=1}^h v_i w_{1j}^T X_i w_{2j} w_{3j}$. We can perform a partial minimization with respect to w_3 first.

Let $A_j = \sum_{i=1}^N v_i (w_{1j}^T X_i w_{2j})$ for $j \in \{1, \dots, h\}$, then

$$\gamma = \min_{w_3} \beta \|w_3\|_1 - \langle A, w_3 \rangle$$

Notice that if there exists a single $|A_j| > \beta$, we can set $w_{3j} \rightarrow +\infty$ and all other $w_{3k} = 0$, and we get $\gamma = -\infty$. When all $|A_j| \leq \beta$, the best minimization is at 0 when all $w_{3j} = 0$. This puts an upper bound of all $|A_j| \leq \beta$.

$$\min_{w_3} \beta \|w_3\|_1 - \langle A, w_3 \rangle = \begin{cases} 0, & \text{if } \max_{w_1, w_2} |A_j| \leq \beta \\ -\infty, & \text{otherwise} \end{cases}$$

Applying this partial minimization will remove the last two terms in the Lagrangian, leaving the constraints $\max_{w_1, w_2} |A_j| \leq \beta$, where $w_1 \in \Delta, \|w_2\|_2 \leq 1$. We convexify this constraint by upper-bounding it with convex constraints:

$$\begin{aligned}
\max_{w_1, w_2} |A_j| &= \max_{w_1, w_2} \left| \sum_{i=1}^N v_i w_1^T X_i w_2 \right| \\
&= \max_{w_1} \left\| \sum_{i=1}^N v_i w_1^T X_i \right\|_2 && \text{We can take } \|w_2\|_2 = 1 \text{ to maximize the abs value} \\
&= \max_{w_1} \left\| \sum_{k=1}^n \sum_{i=1}^N v_i w_{1k} X_{ik} \right\|_2 && \text{separate the } k \text{ terms in } w_1 \\
&\leq \max_{w_1} \sum_{k=1}^n \left\| \sum_{i=1}^N v_i w_{1k} X_{ik} \right\|_2 && \text{triangle inequality} \\
&= \max_{w_1} \sum_{k=1}^n w_{1k} \left\| \sum_{i=1}^N v_i X_{ik} \right\|_2 && w_1 \in \Delta \text{ is defined as non-negative}
\end{aligned}$$

Since $w_1 \in \Delta$ is simplex, w_1 sums to 1. To maximize the above sum, we set $w_{1k} = 1$ for the largest $\left\| \sum_{i=1}^N v_i X_{ik} \right\|_2$ term, and the rest $w_{1j} = 0$. Therefore, we have the new constraint bounding the initial non-convex constraint, so bounding it by β will also bound the initial non-convex constraint.

$$\max_{k \in [n]} \left\| \sum_{i=1}^N v_i x_{ik} \right\|_2 \leq \beta$$

Relaxing it gives us the finalized convex Dual constraint:

$$\left\| \sum_{i=1}^N v_i x_{ik} \right\|_2 \leq \beta, \forall k \in [n]$$

Lastly, the first two terms appears in the Lagrangian form the convex conjugate: given $\hat{y}_i, y_i \in \mathbb{R}$, $\min_{\hat{y}_i} [\mathcal{L}(\hat{y}_i, y_i) + v_i \hat{y}_i] = -\mathcal{L}^*(v_i, y_i)$. Combining the two summed term, we have the finalized Dual problem:

$$\sum_{i=1}^N \min_{\hat{y}_i} [\mathcal{L}(\hat{y}_i, y_i) + v_i \hat{y}_i] = \sum_{i=1}^N [-\mathcal{L}^*(v_i, y_i)] = -\mathcal{L}^*(v, y)$$

Next, we take the dual of this equivalent dual to form our equivalent primal problem. Form the Lagrangian from the dual problem, with $\lambda_k \geq 0$ as the bidual variable for the inequality constraint.

$$L(v, y, \lambda) = -\mathcal{L}^*(v, y) + \sum_{k=1}^n \lambda_k (\beta - \left\| \sum_{i=1}^N v_i x_{ik} \right\|_2)$$

Given a vector a , and a constraint $\|a\|_2 \leq \beta$, we can introduce a variable r where $\|r\|_2 \leq 1$. By Cauchy-Schwarz inequality, $r^T a \leq \|r\|_2 \|a\|_2 \leq \|a\|_2$, so there exists such an r under the constraint $\|r\|_2 \leq 1$, where $r^T a = \|a\|_2$, which will also be the maximum of $r^T a$. So we have $\|a\|_2 = \max_{\|r\|_2 \leq 1} (r^T a)$. Switching the sign gives us $-\|a\|_2 = \min_{\|r\|_2 \leq 1} (-r^T a)$. Let $-\|a\|_2$ be the $-\left\| \sum_{i=1}^N v_i x_{ik} \right\|_2$ in the above Lagrangian formulation and form n separate r_k , we have

$$L(v, y, \lambda) = \min_{r_k: \|r_k\|_2 \leq 1} -\mathcal{L}^*(v, y) + \sum_{k=1}^n \lambda_k (\beta - r_k^T \sum_{i=1}^N v_i x_{ik})$$

Then the bi-dual optimization problem is following. The \mathcal{L}^* term is convex with respect to v by definition, so $-\mathcal{L}^*$ is concave with respect to v and independent of r_k . The remaining term is linear with respect to both r_k and v , so it is both convex and concave with respect to r_k and v . The domain of both v and r_k are convex. By Von Neumann's minimax conditions [15], when the function is convex in terms of the maximizing variable (v)

and concave in terms of the minimizing variable (r_k), and both domains are convex, the max and min can be exchanged.

$$\begin{aligned}
\min_{\lambda \geq 0} \max_{v \in \mathbb{R}^N} L(v, y, \lambda) &= \min_{\lambda \geq 0} \max_{v \in \mathbb{R}^N} \min_{r_k: \|r_k\|_2 \leq 1} -\mathcal{L}^*(v, y) + \sum_{k=1}^n \lambda_k (\beta - r_k^T \sum_{i=1}^N v_i x_{ik}) \\
&= \min_{\lambda \geq 0} \min_{r_k: \|r_k\|_2 \leq 1} \max_{v \in \mathbb{R}^N} -\mathcal{L}^*(v, y) + \sum_{k=1}^n \lambda_k (\beta - r_k^T \sum_{i=1}^N v_i x_{ik}) \\
&= \min_{\lambda \geq 0} \min_{r_k: \|r_k\|_2 \leq 1} \max_{v \in \mathbb{R}^N} -\mathcal{L}^*(v, y) - \sum_{i=1}^N \sum_{k=1}^n \lambda_k r_k^T x_{ik} (v_i) + \beta \sum_{k=1}^n \lambda_k
\end{aligned}$$

We then try to maximize the function by finding v^* , and then we combine the two minimizing terms by combining λ and r_k . The $\beta \sum_{k=1}^n \lambda_k$ is independent of v , so it is a constant. Assuming the loss function chosen is a convex function, then the conjugate of its conjugate will be original loss function. Define a dummy variable $\zeta_i = \sum_{k=1}^n \lambda_k r_k^T x_{ik}$, the first two terms become $\max_v [-\mathcal{L}^*(v, y) - \sum_{i=1}^N \zeta_i (v_i)] = \sum_{i=1}^N \mathcal{L}(\zeta_i, y_i)$ by the definition of conjugate. Substituting back ζ and combine with the constant, we are left with the following after maximizing over v :

$$\min_{\lambda \geq 0} \min_{r_k: \|r_k\|_2 \leq 1} \sum_{i=1}^N \mathcal{L}(\sum_{k=1}^n \lambda_k r_k^T x_{ik}, y_i) + \beta \sum_{k=1}^n \lambda_k$$

Define $z_k = \lambda_k r_k$ to merge the two minimization, then the constraint becomes the combination of the two $z_k: \|z_k\|_2 \leq \lambda_k$.

$$\min_{z_k: \|z_k\|_2 \leq \lambda_k} \sum_{i=1}^N \mathcal{L}(\sum_{k=1}^n z_k^T x_{ik}, y_i) + \beta \sum_{k=1}^n \lambda_k$$

Lastly, this constraint on z_k acts as the bi-dual constraint. By complementary slackness, $\exists \alpha_k$ s.t. $\alpha_k (\|z_k\|_2 - \lambda_k) = 0$. Either $\|z_k\|_2 = \lambda_k$ or $\alpha_k = 0$. However, when $\|z_k\|_2 < \lambda_k$ and $\alpha_k = 0$, notice we can keep decreasing λ_k to reduce the objective, as λ_k is in $\beta \sum_{k=1}^n \lambda_k$. This means at the optimal, all $\|z_k\|_2 = \lambda_k$. So we have the final equivalent, convex primal (bi-dual) problem, concluding the proof [2].

$$\begin{aligned}
&\min_{z_k} \sum_{i=1}^N \mathcal{L}(\sum_{k=1}^n z_k^T x_{ik}, y_i) + \beta \sum_{k=1}^n \|z_k\|_2 \\
&= \frac{1}{2} \sum_{i=1}^N \mathcal{L}(\text{trace}(Z^T X_i), y_i) + \beta \sum_{k=1}^n \|z_k\|_2
\end{aligned}$$

C Theorem 2

We generalize the result from Theorem 1 to multi-dimensional output; i.e. in Theorem 1, $y_i \in \mathbb{R}$, and here we expand it to $y_i \in \mathbb{R}^c$, where c is the dimension of the output. We want to show that the original, nonconvex primal problem of multi-dimensional output has an equivalent, convex bi-dual problem. The original, nonconvex primal problem is as follows

$$\min_{w_{ij} \in \Delta, w_{2j} \in \mathbb{R}^d, w_{3j} \in \mathbb{R}^c} \sum_{i=1}^N \mathcal{L}(\sum_{j=1}^h w_{1j}^T X_i w_{2j} w_{3j}, y_i) + \frac{\beta}{2} \sum_{j=1}^h \|w_{3j}\|_1$$

The proof is nearly identical to Theorem 1, except that the loss is taken as across the c dimensions, which is a direct sum over the loss at each dimension. The only point of significance is at the dual constraint. The initial, nonconvex dual constraint is $\max_{w_1, w_2} \|\sum_{i=1}^N v_i w_1^T X_i w_2\|_\infty \leq \beta$ instead of $\max_{w_1, w_2} \|\sum_{i=1}^N v_i w_1^T X_i w_2\| \leq \beta$. Effectively, we have to bound each of the c output dimension of dual variable by β , in addition to the n dual variables. This results in a set of similar convexified dual constraints:

$$\max_{l \in [c]} \max_{k \in [n]} \left\| \sum_{i=1}^N v_i l x_{ik} \right\|_2 \leq \beta$$

The equivalent, convexified dual problem is

$$\max_{v \in \mathbb{R}^N} -\mathcal{L}^*(\{v_i\}, \{y_i\})_{i \in [n]} \text{ s.t. } \left\| \sum_{i=1}^N v_{il} x_{ik} \right\|_2 \leq \beta; \forall k \in [n], l \in [c]$$

The equivalent, convexified bi-dual problem (primal) is

$$\min_{Z_l \in \mathbb{R}^{n \times d}} \sum_{i=1}^N \sum_{l=1}^c \mathcal{L}(\text{trace}(Z_l^T X_i), y_{il}) + \beta \sum_{l=1}^c \sum_{k=1}^n \|z_{lk}\|_2$$

[2]