Yokohama Publishers
ISSN 1349-8169 ONLINE JOURNAL

Pacific Journal of Optimization
Vol. 21, No. 2, pp. 307–325, 2025

IMPROVING GENERALIZATION VIA COUPLED TENSOR NORM REGULARIZATION*

Ying Gao, Yunfei Qu, Chunfeng Cui†

Abstract: Overfitting is a common phenomenon in machine learning, wherein models almost can fit the samples on the training set but have poor generalization ability on the test set. Regularization is devoted to tackling this problem by imposing a penalty on the complexity or smoothness of the model. However, the performance of regularization is usually circumscribed by the lack of correlation with data samples, which restricts its potential efficiency for many practical models. In this paper, pursuing the seminal work by Zhu et al. (LDMNet), we develop a coupled tensor norm regularization. It can be customized to the model with small-sized structural samples. The main idea of this regularization, which is built upon empirical manifold observation that input data and output features have a low-dimensional structure, is an alternative representation of low-dimensionality. Concretely, coupled tensor norm regularization is the low-rank approximation of the coupled tensor rank function. Related theoretical properties are presented and we further test this regularization for multinomial logistic regression and deep neural networks by theoretical algorithm analysis and numerical experiments. Numerical simulations on real datasets demonstrate the compelling performance of proposed regularization.

Key words: *generalization, coupled tensor norm, data-dependent, multinomial logistic regression, deep neural networks*

Mathematics Subject Classification: *00A71, 15A72, 90C26, 90C30*

1 Introduction

Overfitting is a fundamental challenge for supervised machine learning, owing to the existence of data noise, the complexity of classifiers, and limited training samples. Regularization is an overarching technique in modern machine learning, achieving good generalization even when training on finite samples. From the perspective of optimization, regularization involves the introduction of a penalty term (also called a regularizer) to the objective function, thereby representing the prior preference. Mathematically, the regularized learning model can be formulated as

$$\min_{\theta} \quad L(\theta) + \lambda R(\theta), \tag{1.1}$$

where θ denotes the parameters to be learned, $L(\cdot)$ represents the loss function of the learning model, $R(\cdot)$ is the associated regularizer, and $\lambda > 0$ is a trade-off parameter. Of particular interest to us is the regularizer $R(\cdot)$.

*This research is supported by the National Natural Science Foundation of China (Nos. 12471282, 12126608, 12131004), the R&D project of Pazhou Lab (Huangpu) (No. 2023K0603), and the Fundamental Research Funds for the Central Universities (Grant No. YWF-22-T-204).

†Corresponding author

DOI: https:// doi.org/10.61208/pjo-2024-010

1.1 Related work

Over the past decades, there has been a flurry of research activities focused on regularization. Regularization strategies typically include limiting the model's complexity, making objectives smooth, target-dependent, and simpler to solve (see, e.g., [11, 19, 25, 45] and references therein). In this section, a review is provided of diversified regularization techniques, which have been empirically categorized into data-independent and data-dependent types.

1.1.1 Data-independent regularization

In traditional machine learning, data-independent regularization primarily imposes penalties directly on the variables. Recently, sparse vector-based regularization has garnered significant attention in practical applications (see, e.g., [32, 5, 21]). Specifically, sparse regularization essentially entails an ℓ^0-norm penalty term, i.e., $R(\theta) := \|\theta\|_0$, to constrain the number of nonzero components. However, this represents an intrinsically NP-hard combinatorial optimization problem. To promote sparsity, ℓ^1-norm regularization is utilized to facilitate favorable convex optimization problem [34]. With narrowing the gap between ℓ^0-norm and ℓ^1-norm, ℓ^p-norm becomes a significant regularization [34]. The special case of $p = 2$ proves beneficial for certain learning models [8], which prevents excessive fluctuation of the objective function value. Furthermore, Tikhonov regularization, also known as ridge regression, has been extensively discussed in the context of ill-posed problems [11]. It is characterized by a Tikhonov matrix Γ with a square-norm penalty, i.e., $R(\theta) := \|\Gamma\theta\|_2^2$. When $\Gamma = \alpha I$, with $\alpha > 0$ and I being the identity matrix, Tikhonov regularization becomes equivalent to ℓ^2-norm regularization.

Additionally, motivated by advancements in low-rank matrix recovery, low-rank matrix regularization has captured increasing research interest (see, e.g., [9, 23, 46]). The original form of low-rank regularization is represented by $\text{rank}(\Theta)$, indicating the rank of the parameter matrix Θ in (1.1). It is a prohibitively challenging nonconvex problem. A popular convex relaxation method for approximating the rank function is the nuclear norm $\|\Theta\|_*$ (see (2.1) for definition), which is the sum of singular values of Θ. In contrast, the Schatten p-norm of a matrix (defined by the ℓ^p-norm of the singular value vector) offers a more accurate approximation of the original rank function, yielding improved practical results.

In the context of deep learning, data-independent regularization can improve the generalization performance by limiting the model's complexity. DropOut method [38] and DropConnect method [44] can be considered as computationally inexpensive ways to train an exponentially large ensemble of DNN. Besides, weight decay [16] and batch normalization technique [14] can alleviate overfitting by reducing the magnitude of the weights and the features.

1.1.2 Data-dependent regularization

Data-dependent regularization techniques typically incorporate the inherent data structure to facilitate penalty terms. Data compression [2], tensor dropout [17], and tensor decomposition [33] are classical techniques of structural regularization. Besides, manifold learning is currently gaining importance [47, 26] based on the assumption that data possesses some inherent structure. More concretely, the observed data lies on a low-dimensional manifold embedded in a higher-dimensional space, which states that the shape of data is relatively simple. Intuitively, manifold learning regularization primarily imposes penalties on functions related to the intrinsic geometry of the data manifold. For instance, smoothness on the manifold, and low-dimensionality of the manifold.

In image processing, the patch of an image represents a sub-image with a fixed size and the patch set is the collection of all patches with the same size of image. And the patch set can sample a low-dimensional smooth manifold, which is called the patch manifold of the image. In [30, 31], researchers have discovered the fact that patch manifolds of many natural images exhibit a low-dimensional structure. Subsequently, the dimension of the patch manifold was proposed as a regularizer in image reconstruction using the low-dimensional manifold model (LDMM) [29]. Recently, LDMNet [49] has extended the manifold learning model to study the geometry of both the input data and the output features. Benefitting from the observation that input data and output features may sample a collection of low-dimensional manifolds, LDMNet encourages the learning of geometrically meaningful features through a dimensional penalty.

As discussed in [29, 49], calculating the manifold's dimension entails discretization and the associated minimization requires solving a series of variational subproblems. It is extremely complicated for algorithmic solvers and theoretical analysis. Accordingly, this paper is dedicated to improving the regularization aspects of LDMNet.

1.2 Motivation and contribution

In this paper, pursuing the track of regularization modeling in [29, 49], we propose a coupled tensor norm regularizer, which is an alternative representation of characterizing low-dimensionality. Considering the tensor representation of datasets, input data and output features can be regarded as the coupled tensor, which is the concatenation of a third-order tensor and matrix in this paper. Differing from the dimensional penalty in [49], the main idea of our regularization is built upon the low-rankness of the above coupled tensor. Accordingly, the coupled tensor norm is the approximation of coupled tensor's rank function with elegant computable properties. The contributions of this paper are summarized as follows:

(i) We devise a coupled tensor norm regularization based on the fact that both the input tensor and the output feature matrix possess low-dimensional structure.

(ii) A related analysis of the convexity and smoothness properties of this regularization is presented.

(iii) We evaluate this regularization for multinomial logistic regression (MLR) and deep neural networks (DNN) via theoretical algorithm analysis and numerical experiments.

The rest of this paper is organized as follows. In Section 2, we summarize some preliminaries, which will be useful for subsequent modeling and analysis. In Section 3, we present the proposed regularization and analyze its related properties. For multinomial logistic regression, a convergent gradient descent approach is adopted to solve the whole model. For deep neural networks, we devise an alternating minimization method and establish the theoretical convergence. In Section 4, we evaluate multinomial logistic regression and deep neural networks on a series of real datasets to demonstrate the numerical performance of the proposed regularization. Finally, conclusions are drawn in Section 5.

$\boxed{2}$ Preliminaries

For a vector, $x = (x_1, \cdots, x_n)^\top \in \mathbb{R}^n$, let $\|x\|_p := (\sum\limits_{i=1}^{n} |x_i|^p)^{1/p} (0 < p < \infty)$ denote the ℓ^p-norm of x. At the extreme, the ℓ^0-norm is defined by the number of non-zero elements. For brevity, we denote the ℓ^2-norm of x by $\|x\|$. Let $\mathbf{0}$ (resp., I) denote the zero (resp., identity) matrix whose dimension can be clear from the context. For a matrix $X \in \mathbb{R}^{m\times n}$ with $\mathrm{rank}(X) = r$, the singular value decomposition (SVD) of matrix X is $X = U\Sigma V^\top$, where $U \in \mathbb{R}^{n\times r}$ and $V \in \mathbb{R}^{n\times r}$ have orthonormal columns, and $\Sigma = diag(\sigma_1, \ldots, \sigma_r)$ has positive singular values. We denote the vectorized singular values of X in nonincreasing order by $\sigma(X) := (\sigma_1, \cdots, \sigma_r)^\top \in \mathbb{R}^r$. The Schatten norm of X, denoted by $\|X\|_{*,p}$, is the ℓ^p-norm of $\sigma(X)$, i.e.,

$$\|X\|_{*,p} := \|\sigma(X)\|_p, \quad \forall\, p \in [0, \infty]. \tag{2.1}$$

Particularly, $\|X\|_{*,0}$(resp., $\|X\|_{*,1}$,$\|X\|_{*,2}$, and $\|X\|_{*,\infty}$) corresponds to the rank (resp., nuclear, Frobenius, and spectral norm) of X. For brevity, the Frobenius norm of matrix X is denoted by $\|X\|_F$. For a tensor $\mathcal{X} \in \mathbb{R}^{I_1\times\cdots\times I_N}$, $X_{(n)} \in \mathbb{R}^{I_n\times J_n}$ with $J_n = \prod\limits_{i=1,i\neq n}^{N} I_i$ denotes the mode-n unfolding matrix. Tucker decomposition of $\mathcal{X}$ is defined by $\mathcal{X} = \mathcal{G} \times_1 U_1 \times_2 U_2 \times_3 \cdots \times_N U_N$, or equivalently,

$$X_{(n)} = U_n G_{(n)} \left(U_N \otimes \cdots \otimes U_{n+1} \otimes U_{n-1} \cdots \otimes U_1\right)^\top, \quad \forall\, n = 1, \ldots, N,$$

where $U_n \in \mathbb{R}^{I_n\times R_n}$ is the factor matrix (which is usually orthogonal) and $\mathcal{G} \in \mathbb{R}^{R_1\times\cdots\times R_N}$ is the core tensor, $G_{(n)}$ is the mode-n unfolding matrix of $\mathcal{G}$. Accordingly, the Tucker rank of tensor X (also called multilinear rank) is defined by $\mathrm{rank}_T(\mathcal{X}) := (\mathrm{rank}(X_{(1)}), \cdots, \mathrm{rank}(X_{(N)}))$.

The coupled tensor denotes the concatenation of tensors and the coupling between tensors occurs when they share a common mode, where one tensor can provide side information for others or they both mutually share information (see e.g., [43, 1, 35] and references therein). For instance, given an N-th order of tensor $\mathcal{X} \in \mathbb{R}^{I_1\times\cdots\times I_N}$ and an M-th order tensor $Y \in \mathbb{R}^{J_1\times\cdots\times J_M}$ with the dimension of the n-th mode of $\mathcal{X}$ equals the m-th mode of $\mathcal{Y}$, the coupled tensor of $\mathcal{X}$ and $\mathcal{Y}$, denoted by $[\mathcal{X}, \mathcal{Y}]_{(n,m)}$. For brevity, we simplify the notation by $[\mathcal{X}, \mathcal{Y}]$ whenever there is no confusion. Please see Figure 1 for an example of the coupling of a third-order tensor and a matrix at the first mode.

The coupled low-rank decomposition of $[\mathcal{X}, A]_{(n,1)}$ [35] factorizes a matrix $A \in \mathbb{R}^{I_n\times J}$ and tensor $\mathcal{X} \in \mathbb{R}^{I_1\times\cdots\times I_N}$ as the form of

$$A = U_n V^\top \text{ and } \mathcal{X} = \mathcal{G}{\times_1} U_1 \times_2 U_2 \times_3 \cdots \times_N U_N,$$

where $U_n \in \mathbb{R}^{I_n\times R_n}$ is shared between $\mathcal{X}$ and A with a coupled rank R_n. Therein, U_n can provide features for both $\mathcal{X}$ and A, which may increase the extracted information from both data sets. Naturally, the coupled tensor rank of $[\mathcal{X}, A]$ is defined as an extension of Tucker rank, i.e.,

$$\mathrm{rank}([\mathcal{X}, A]) := (\mathrm{rank}(X_{(1)}), \cdots, \mathrm{rank}([X_{(n)}, A]), \cdots, \mathrm{rank}(X_{(N)})).$$

The coupled tensor norm is a convex approximation of low-rankness for the coupled tensor, which may be used in the coupled tensor completion. Interested readers can refer to [43, 42] for more details. Concretely, the coupled tensor norm of $[\mathcal{X}, A]$ is defined by

$$\|[\mathcal{X}, A]\|_{\mathrm{c}} := \|[X_{(n)}, A]\|_* + \sum_{i=1, i\neq n}^{N} \|X_{(i)}\|_*, \tag{2.2}$$

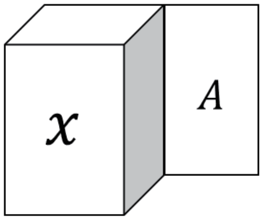

Figure 1: Illustration of the coupling between a tensor $\mathcal{X}$ and matrix A at the first mode.

where $X_{(n)}$ is the mode-n unfolding matrix.

Remark 2.1. In this paper, we only consider the case of $M = 2$, namely, the concatenation of a tensor and a matrix. However, our discussions may be extended to an arbitrary tensor coupling.

Let $f : \mathbb{R}^n \to \mathbb{R}_\infty := \mathbb{R} \cup \{+\infty\}$ be an extended value function. f is said to be proper (resp. closed) if $\mathrm{dom}(f) := \{x \in \mathbb{R}^n \mid f(x) < +\infty\}$ is nonempty (resp. $\mathrm{epi}(f) := \{(x, r) \in \mathbb{R}^n \times \mathbb{R} \mid f(x) \le r\}$ is closed). f is called regular at $\bar{x}$ if the set $\mathrm{epi}(f)$ is Clarke regular at $(\bar{x}, f(\bar{x}))$.

Some preliminaries on subdifferential definitions and calculus are stated as follows (see e.g., the monograph [40]).

Definition 2.2. Let $f : \mathbb{R}^n \to \mathbb{R}_\infty$ be a proper, closed function.

(i) The Fréchet subdifferential of f at $x \in \mathrm{dom}(f)$, denoted by $\hat{\partial} f(x)$, is the set of vectors $p \in \mathbb{R}^n$ which satisfy

$$f(y) \ge f(x) + \langle p, y - x \rangle + o(\|y - x\|).$$

(ii) The limiting subdifferential of f at $x \in \mathrm{dom}(f)$, is defined by

$$\partial f(x) := \{p \in \mathbb{R}^n \mid \exists\, x^k \to x,\ f(x^k) \to f(x),\ p^k \in \hat{\partial} f(x^k) \to p,\ \text{as } k \to \infty\}.$$

Notationally, $\hat{\partial} f(x) = \partial f(x) = \emptyset$ for all $x \notin \mathrm{dom}(f)$. It is follows from above definition that $\hat{\partial} f(x) \subset \partial f(x)$ for any $x \in \mathrm{dom}(f)$. The first set is closed and convex while the second one is closed. Moreover, if f is convex, then

$$\hat{\partial} f(x) = \partial f(x) = \{p \in \mathbb{R}^n \mid f(y) \ge f(x) + \langle p, y - x \rangle,\ \forall y \in \mathbb{R}^n\}.$$

If f is convex and differential, then $\hat{\partial} f(x) = \partial f(x) = \nabla f(x)$. The subdifferential of matrix nuclear norm can be referred to, e.g., [10, Theorem 2.1], which is useful for sequel analysis.

Lemma 2.3. *Let $X \in \mathbb{R}^{m \times n}$ be an arbitrary matrix and $USV^\top$ be the SVD of X, then the nuclear norm function of matrix $X \in \mathbb{R}^{m \times n}$ is convex and nonsmooth and its subdifferential is*

$$\partial \|X\|_* = \{UV^\top + Z \mid U^\top Z = 0, ZV = 0, \|Z\| \le 1\}.$$

The following subdifferential calculus rule for regular function can be referred to, e.g., [40, 10.9 Corollary].

Lemma 2.4. *If each f_i $(1 \leq i \leq n)$ is regular at $\bar{x}$, then $f = \sum\limits_{i=1}^{n} f_i$ is regular at $\bar{x}$ and*

$$\partial f(\bar{x}) = \partial f_1(\bar{x}) + \cdots + \partial f_n(\bar{x}). \tag{2.3}$$

The Kurdyka-Łojasiewicz (KŁ) property is a powerful tool in the field of nonconvex and nonsmooth optimization (see, e.g. [7]). For any $-\infty < c_1 < c_2 \leq \infty$, a sublevel set of f is defined by

$$[c_1 < f < c_2] := \{x \in \mathbb{R}^n \mid c_1 < f(x) < c_2\}.$$

For $\eta \in (0, \infty]$, let Φ_η denote the class of concave and continuous function $\varphi : [0, \eta) \to \mathbb{R}_+$ satisfying:

(i) φ is continuous at origin and $\varphi(0) = 0$;

(ii) φ is continuously differential on $(0, \eta)$;

(iii) $\varphi'(t) > 0$ for all $t \in (0, \eta)$.

Definition 2.5. A proper closed function $f : \mathbb{R}^n \to \mathbb{R}_\infty$ admits the KŁ property at $\bar{x} \in \mathrm{dom}(\partial f) := \{x \in \mathbb{R}^n \mid \partial f(x) \neq \emptyset\}$ if there exist $\eta \in (0, \infty]$, $\varphi \in \Phi_\eta$, and neighbourhood of $\bar{x}$ (denoted by $\mathcal{B}(\bar{x})$) such that

$$\varphi'(f(x) - f(\bar{x}))\mathrm{dist}(0, \partial f(x)) \geq 1, \quad \forall x \in \mathcal{B}(\bar{x}) \cap [f(\bar{x}) < f(x) < f(\bar{x}) + \eta]. \tag{2.4}$$

Moreover, f is called a KŁ function if it admits the KŁ property at any $\bar{x} \in \mathrm{dom}(\partial f)$.

$\boxed{3}$ Model and Algorithm

In this section, we present the coupled tensor norm regularizer and analyze its properties. Afterwards, this regularizer is applied to multinomial logistic regression (MLR) and deep neural networks (DNN), respectively.

Throughout this section, we focus on the multi-classification problem with n samples and c disjoint classes. Let $S = \{(\mathcal{X}_i, y_i)\}_{i=1}^{n}$ denote the independent and identically distributed training dataset, where $\mathcal{X}_i \in \mathbb{R}^{I_1 \times \cdots \times I_N}$ represents the input data information and $y_i \in \mathbb{R}^c$ the corresponding label. Specifically, $m = \prod_{i=1}^{N} I_i$ denotes the number of input features and $y_i = (y_i^{(1)}, y_i^{(2)}, \cdots, y_i^{(c)})^\top$ is defined as the one-bit vector with $y_i^{(k)} = 1$ if $\mathcal{X}_i$ belongs to the k-th class and $y_i^{(k)} = 0$ otherwise. For convenience, let $f_\theta(\mathcal{X}_i) \in \mathbb{R}^c$ represent the output feature of $\mathcal{X}_i$, where f_θ is the learning model with respect to parameter θ. Let $\mathcal{X} \in \mathbb{R}^{n \times I_1 \times \cdots \times I_N}$ (resp,. $Y \in \mathbb{R}^{n \times c}$, $f_\theta(\mathcal{X}) \in \mathbb{R}^{n \times c}$) correspond to the concatenation of $\mathcal{X}_i$ (resp., y_i, $f_\theta(\mathcal{X}_i)$), $i = 1, \cdots, n$ at the first mode.

Accordingly, the loss function of the classification model in (1.1) is

$$L(\theta) = \frac{1}{n} \sum_{i=1}^{n} l(f_\theta(\mathcal{X}_i), y_i), \tag{3.1}$$

where $l(\cdot)$ is the loss function, such as the cross entropy loss.

As discussed in [49], the concatenation $\mathcal{X}$ and $f_\theta(\mathcal{X})$ can sample a collection of low-dimensional manifolds, and the regularizer in [49] is the dimensionality of the manifold. Actually, the low-dimensional manifold model can be applied to surface patches in the point cloud, which uses the patch manifold prior to seeking self-similar patches and removing noise. Alternatively, for coupled tensor data $[\mathcal{X}, f_\theta(\mathcal{X})]$, the low-dimensional property can naturally be characterized by low-rankness. Hence, the new regularizer is defined as the rank of coupled tensor as follows,

$$R(\theta) = \operatorname{rank}([\mathcal{X}, f_\theta(\mathcal{X})]).$$

Furthermore, utilizing the classical convex approximation of the rank function for matrices, the coupled tensor norm based on the overlapped approach in (2.2) is adopted to characterize low-rankness, and we have

$$\|[\mathcal{X}, f_\theta(\mathcal{X})]\|_c = \|[X_{(1)}, f_\theta(\mathcal{X})]\|_* + \sum_{i=2}^{N} \|X_{(i)}\|_*.$$

By ignoring the terms that are not related to θ, we propose the following regularization function,

$$R(\theta) = \|[X_{(1)}, f_\theta(\mathcal{X})]\|_*. \tag{3.2}$$

The following theorem states the properties of the matrix concatenation function, which will be useful for subsequent analysis.

Theorem 3.1. *Let $X \in \mathbb{R}^{n\times m}$ and $\xi \in \mathbb{R}^{n\times c}$, the matrix row concatenation function, defined by $g(\xi) = \|[X, \xi]\|_*$, is not a norm. Further, it is a convex but nondifferentiable function in terms of ξ, and its subdifferential is*

$$\partial g(\xi) = \{UV_2^\top + Z_2 \,|\, U^\top Z = 0, ZV = 0, \|Z\| \le 1\},$$

where $U\Sigma V^\top$ is the SVD of $[\mathcal{X}, \xi]$, r is the rank of $[\mathcal{X}, \xi]$, $U \in \mathbb{R}^{n\times r}$, $\Sigma \in \mathbb{R}^{r\times r}$, $V \in \mathbb{R}^{(m+c)\times r}$, $V_2 \in \mathbb{R}^{c\times r}$ is the last c rows of V, and $Z_2 \in \mathbb{R}^{n\times c}$ is the last c columns of $Z \in \mathbb{R}^{n\times(m+c)}$.

Proof. Note that when ξ is equal to zero, $g(\xi)$ may not equal zero. Hence, $g(\xi)$ is not a norm.

Further, by rewriting the concatenation of X and ξ as a linear function of ξ,

$$[X, \xi] = \xi A + [X, \mathbf{0}], \tag{3.3}$$

where $A = [\mathbf{0}, I]$ with $\mathbf{0} \in \mathbb{R}^{c\times m}$, $I \in \mathbb{R}^{c\times c}$ and $[X, \mathbf{0}]$ with $\mathbf{0} \in \mathbb{R}^{n\times c}$. Obviously, $g(\xi)$ is convex provided that $g(\xi)$ is the composition of convex and linear functions. Suppose the SVD of matrix $[X, \xi]$ is $USV^\top$. Then the subdifferentiable of the nuclear norm at $[X, \xi]$ is

$$\partial\|[X, \xi]\|_* = \{UV^\top + Z \mid U^\top Z = 0, ZV = 0, \|Z\| \le 1\}.$$

Through the chain rule and equation (3.3), we have

$$\partial g(\xi) = \frac{\partial g(\xi)}{\partial [X, \xi]} \cdot \frac{\partial [X, \xi]}{\partial \xi} = (UV^\top + Z)A^\top = UV_2^\top + Z_2, \tag{3.4}$$

where V_2 is the last c rows of V and Z_2 is the last c columns of Z. This completes the proof. □

Based on Theorem 3.1, we can analyze the properties of our proposed regularizer, which is the composition of matrix nuclear norm, matrix concatenation, and the classification model.

Theorem 3.2. *Let $R(\cdot)$ be the regularizer defined by* (3.2)*. Then,*

(i) if the learning model f_θ is linear with respect to θ, then $R(\cdot)$ is convex for θ. Further, if $f_\theta(\mathcal{X}) = X_{(1)}\theta$ or $f_\theta(\mathcal{X}) = X_{(1)}\theta^\top$, then $R(\cdot)$ is also differential for θ;

(ii) if the learning model f_θ is nonconvex and nondifferentiable with respect to θ, then $R(\cdot)$ is nonconvex and nondifferentiable for θ.

Proof. If $\xi = f_\theta(\mathcal{X})$ in Theorem 3.1 is a linear model for θ, then $R(\cdot)$ is the composition of convex and linear functions, and is convex consequently. On the other hand, when the learning model $f_\theta(\cdot)$ is a nonconvex and nondifferentiable model for θ, $R(\cdot)$ is nonconvex and nondifferentiable. Furthermore, based on SVD of matrix $[X_{(1)}, f_\theta(\mathcal{X})]$, i.e., $[X_{(1)}, f_\theta(\mathcal{X})] = USV^\top$, we have

$$X_{(1)} = USV_1^\top, \;\; f_\theta(\mathcal{X}) = USV_2^\top,$$

where V_1, V_2 are the front m rows and the last c rows of V, respectively. From (3.4) in the proof of Theorem 3.1, we have that, if $f_\theta(\mathcal{X}) = X_{(1)}\theta$,

$$\nabla R(\theta) = V_1 S U^\top (UV_2^\top + Z_2) = V_1 S V_2^\top,$$

and if $f_\theta(\mathcal{X}) = X_{(1)}\theta^\top$,

$$\nabla R(\theta) = (UV_2^\top + Z_2)^\top USV_1^\top = V_2 S V_1^\top. \tag{3.5}$$

Hence, $R(\cdot)$ is differential in these two cases. The proof is thus completed. □

$\boxed{3.1}$ Multinomial logistic regression

Multinomial logistic regression (MLR) is a classical learning method for classification tasks. Let $x_i \in \mathbb{R}^m$ represent the vectorized form of the input tensor $\mathcal{X}_i$, and $X \in \mathbb{R}^{n\times m}$ denote the matrix of all input features from n training samples, which corresponds to the unfolding of $\mathcal{X}$ along the first mode. Suppose that the posterior probability that x_i belongs to the k-th class is modeled by the softmax function as follows,

$$P(y_i^{(k)} = 1 \mid x_i, W) := \frac{\exp(w_k^\top x_i)}{\sum\limits_{j=1}^{c} \exp(w_j^\top x_i)},$$

where $w_j \in \mathbb{R}^m$ is the j-th column of the weight matrix, and $W := (w_1, w_2, \cdots, w_c)^\top \in \mathbb{R}^{c\times m}$ is the weight matrix to be estimated from the training set. The MLR model estimates the parameter W using the maximum likelihood, or equivalently minimizing the negative log-likelihood,

$$\min_{W\in\mathbb{R}^{c\times m}} \; L(W) = -\frac{1}{n}\sum_{i=1}^{n}\big(\sum_{j=1}^{c} y_i^{(j)} w_j^\top x_i - \log\sum_{j=1}^{c}\exp(w_j^\top x_i)\big),$$

where $L(W)$ is called an average multinomial logistic loss function. The interested readers can refer to [18, 4] for more details on multinomial logistic regression. Obviously, it falls into a special case of (3.1) with $\theta = W$, $f_\theta(\mathcal{X}) := XW^\top$.

Previous studies, such as [48], have demonstrated that MLR may not generalize effectively on test datasets when the number of samples n is significantly smaller than the number of features m. To improve the generalization, the MLR model with the proposed regularizer (3.2) can be reformulated as the minimization problem

$$\min_{W\in\mathbb{R}^{c\times m}} \; G(W) := L(W) + \lambda\|[X, XW^\top]\|_*. \tag{3.6}$$

The nuclear norm function is typically nondifferentiable; however, the coupled nuclear norm regularizer introduced in this context is differentiable for W. More details are presented in the following lemma.

Lemma 3.3. *Given the regularizer term $R(W) = \|[X, XW^\top]\|_*$ in model* (3.6)*, and let $USV^\top$ be the SVD of matrix $[X, XW^\top]$. Suppose rank$([X, XW^\top]) = r$, by dividing V as $[V_1; V_2]$ with $V_1 \in \mathbb{R}^{m\times r}$ and $V_2 \in \mathbb{R}^{c\times r}$, it holds that*

- *(i) $R(\cdot)$ is differentiable with respect to W and its gradient is $V_2SV_1^\top$;*
- *(ii) $\nabla R(\cdot)$ is Lipschitz continuous if all singular values of matrix $XW^\top$ are nonzero.*

Proof. Actually, the regularizer $\|[X, XW^\top]\|_*$ in model (3.6) is a special case of (3.2). It follows from (3.5) that $\nabla R(W) = V_2SV_1^\top$. Hence, for any matrices $W, \hat{W} \in \mathbb{R}^{c\times m}$, we have

$$\|\nabla R(W) - \nabla R(\hat{W})\|_F = \|V_2SV_1^\top - \hat{V}_2^\top \hat{S}\hat{V}_1^\top\|_F = \|CB\|_F,$$

where $B = \begin{pmatrix} USV_1^\top \\ -\hat{U}\hat{S}\hat{V}_1^\top \end{pmatrix}$, $C = (V_2U^\top, \hat{V}_2\hat{U}^\top)$. Furthermore, by the fact that the largest singular value of C satisfies $\sigma_{\max}(C) \le 1$, we deduce

$$\|\nabla R(W) - \nabla R(\hat{W})\|_F \le \|B\|_F. \tag{3.7}$$

Let $E = \big(V_2SU^\top, \hat{V}_2\hat{S}\hat{U}^\top\big) = \big(WX^\top, \hat{W}X^\top\big)$, we obviously obtain

$$\begin{aligned}\|X^\top XW^\top - X^\top X\hat{W}^\top\|_F =& \|V_2S^2V_1^\top - \hat{V}_2\hat{S}^2\hat{V}_1^\top\|_F \\ =& \|V_2SU^\top USV_1^\top - \hat{V}_2\hat{S}\hat{U}^\top\hat{U}\hat{S}V_1^\top\|_F \\ =& \|EB\|_F \\ \ge& \sigma_{\min}(E)\|B\|_F, \end{aligned} \tag{3.8}$$

where $\sigma_{\min}(E)$ denotes the smallest singular value of E. It follows from the assumption that all singular values of $XW^\top$ are nonzero for all W and the fact that E is the concatenation of $WX^\top$ and $\hat{W}X^\top$ that $\sigma_{\min}(E) > 0$. Combining (3.7) and (3.8), we derive

$$\begin{aligned}\|\nabla R(W) - \nabla R(\hat{W})\|_F \le& \|B\|_F \\ \le& \frac{1}{\sigma_{\min}(E)}\|X^\top XW^\top - X^\top X\hat{W}^\top\|_F \\ \le& \frac{\lambda_{\max}(X^\top X)}{\sigma_{\min}(E)}\|W - \hat{W}\|_F. \end{aligned}$$

Hence, it means that $\nabla R(\cdot)$ is Lipschitz continuous and this completes the proof. □

The inherent convexity and differentiability of model (3.6) suggest the use of gradient-based optimization methods. Although the number of training samples may be small, the presence of numerous features can result in a substantial computational burden. The current work employs the classical gradient descent algorithm with line search to resolve this issue. The iterative process for updating W^{k+1} is based on following procedure,

$$W^{k+1} = W^k - \alpha_k \nabla G(W^k). \tag{3.9}$$

Here, α_k is determined by a line search algorithm that guarantees a sufficient decrease, as described in [22].

For a given threshold $\varepsilon \geq 0$, the termination criterion is $\|\nabla G(W)\|_2 \leq \varepsilon$. The convergence analysis is illuminated in the next theorem.

Theorem 3.4. *Suppose all singular values of output matrix $XW^\top$ are nonzero. For the convex differential minimization problem* (3.6)*, each accumulation point of the iterative sequence $\{W^k\}_{k=0}^{\infty}$ generated by the procedure* (3.9)*, is a global minimizer.*

Proof. The iterative procedure (3.9) is a gradient descent method with the stepsize α_k satisfying the Wolfe-Powell rules. The gradient $\nabla G(W)$ in model (3.6) is Lipschitz continuous, as established in Lemma 3.3. Hence, as introduced in [39, Theorem 2.5.7], for the sequence $\{W^k\}_{k=0}^{\infty}$ generated by the gradient descent method with Wolfe line search, either $\|\nabla G(W^k)\|_2 = 0$ for some k or $\|\nabla G(W^k)\|_2 \to 0$. It means that each accumulation point of the iterative sequence $\{W^k\}_{k=0}^{\infty}$ is a stationary point. Furthermore, the stationary point is also a global minimizer owing to the convexity of model (3.6). This completes the proof. □

3.2 Deep neural networks (DNN)

Let θ denote the set of weights and biases within a neural network. For each data $\mathcal{X}_i$, the classical DNN learns a feature $f_\theta(\mathcal{X}_i) \in \mathbb{R}^c$ by minimizing the empirical loss function (see (3.1)) over the training data. To reduce the risk of overfitting in DNN, as cautioned by [41], we incorporate a coupled tensor norm regularizer (3.2) into the loss function. This leads to the formulation of a regularized DNN model:

$$\min_{\theta} \ L(\theta) + \lambda \|[X_{(1)}, f_\theta(\mathcal{X})]\|_*, \tag{3.10}$$

where $f_\theta(\mathcal{X})$ is highly nonlinear and nondifferentiable with respect to θ owing to the deep network architecture. Therefore, from Theorem 3.2, the regularization term in model (3.10) is nonconvex, nondifferentiable, and nonseparable.

A fundamental assumption for solving DNN by stochastic gradient descent (SGD) method is that the objective function is separable. To circumvent the nonseparability in (3.10), we introduce an auxiliary variable ξ into (3.10) as follows,

$$\begin{aligned} &\min_{\theta,\xi} \ L(\theta) + \lambda \|[X_{(1)}, \xi]\|_*, \\ &\text{s.t.} \quad f_\theta(\mathcal{X}) = \xi. \end{aligned}$$

Then we penalize the constraint into the loss function using the quadratic penalty method and obtain the following unconstrained optimization model,

$$\min_{\theta,\xi} \ \mathcal{L}(\theta, \xi) := L(\theta) + \lambda \|[X_{(1)}, \xi]\|_* + \frac{\mu}{2} \|f_\theta(\mathcal{X}) - \xi\|_F^2, \tag{3.11}$$

where $\mu > 0$ is the penalty parameter. Problem (3.11) can be solved by alternating minimization method. Specifically, given (θ^k, ξ^k), we implement the following sub-steps:

(1) Update θ^{k+1} with the fixed ξ^k:

$$\min_{\theta} \; L(\theta) + \frac{\mu}{2}\|f_{\theta}(\mathcal{X}) - \xi^k\|_F^2. \tag{3.12}$$

(2) Update ξ^{k+1} with the fixed θ^{k+1}:

$$\min_{\xi} \; \lambda\|[X_{(1)}, \xi]\|_* + \frac{\mu}{2}\|f_{\theta^{k+1}}(\mathcal{X}) - \xi\|_F^2. \tag{3.13}$$

The θ-subproblem (3.12) is separable with respective to the samples $\mathcal{X}_i$ and can be solved using SGD. The ξ-subproblem (3.13) is strongly convex but nondifferentiable, as indicated by Theorem 3.1. Through the above analysis, we describe our algorithm framework for solving (3.11) in Algorithm 1.

Algorithm 1: The alternating minimization method for (3.11)

```
Require: Training data $\{(\mathcal{X}_i, y_i)\}_{i=1}^n$, hyperparameters $\lambda$ and $\mu$, and a neural network
  with initial weight $\theta^0$.
Ensure: Trained network weights $\theta^*$.
  Let $k = 0$. $\xi^0 \in \mathbb{R}^{n \times c}$ is initialized as zero.
  while not converge do
    1. Update $\theta^{k+1}$ in (3.12): solve the nonconvex problem by SGD.
    2. Update $\xi^{k+1}$ in (3.13): solve the convex problem by the subgradient method.
    3. $k \leftarrow k + 1$.
  end while
  $\theta^* = \theta^k$.
```

We will next discuss the global convergence of Algorithm 1 by invoking the KŁ property and regularity conditions. The assumptions for the convergence analysis of problem (3.11) are as follows:

Assumption 3.1.

(i) The loss function $\mathcal{L}(\theta, \xi)$ defined by (3.11) is regular and satisfies the KŁ property.

(ii) The subgradient of $f_{\theta}(\mathcal{X})$ is bounded above, i.e., there exists a positive constant ρ such that for all $\eta \in \partial f_{\theta}(\mathcal{X})$, it holds that $\|\eta\|_F \leq \rho$.

Remark 3.5. According to [15], the regularity assumption on $\mathcal{L}(\theta, \xi)$ can be met for certain DNN with the ReLU activation functions.

Theorem 3.6. *Under Assumption 3.1, let the sequence $\{(\theta^k, \xi^k)\}_{k\geq 0}$ be the sequence generated by Algorithm 1. Then the subsequence $\{\xi^k\}_{k\geq 0}$ has a finite length and converges globally to a point ξ^*. Moreover, if θ^* is any limit point of $\{\theta^k\}_{k\geq 0}$, then (θ^*, ξ^*) is a critical point of $\mathcal{L}$.*

Proof. Actually, it follows from [12, Definition 2] that $\{(\theta^k, \xi^k)\}_{k\geq 0}$ generated by Algorithm 1 is a bounded approximate gradient-like descent sequence. Specifically, there are four conditions for a bounded approximate gradient-like descent sequence. We shall present concrete proofs individually.

Firstly, for the θ subproblem in (3.12) and the ξ subproblem in (3.13), we have respectively,

$$\mathcal{L}(\theta^{k+1},\xi^k) \le \mathcal{L}(\theta^k,\xi^k), \tag{3.14}$$

$$0 \in \partial_\xi \mathcal{L}(\theta^{k+1},\xi^{k+1}). \tag{3.15}$$

By the strongly convexity of $\mathcal{L}(\theta,\xi)$ with respect to ξ, the following inequality holds,

$$\mathcal{L}(\theta^{k+1},\xi^{k+1}) + \langle g_\xi^{k+1}, \xi^k - \xi^{k+1}\rangle + \frac{\mu}{2}\|\xi^k - \xi^{k+1}\|_F^2 \le \mathcal{L}(\theta^{k+1},\xi^k),$$

for $g_\xi^{k+1} \in \partial_\xi \mathcal{L}(\theta^{k+1},\xi^{k+1})$. Furthermore, adding (3.14) to the above inequality and combining it with (3.15) yield

$$\frac{\mu}{2}\|\xi^k - \xi^{k+1}\|_F^2 \le \mathcal{L}(\theta^k,\xi^k) - \mathcal{L}(\theta^{k+1},\xi^{k+1}). \tag{3.16}$$

Clearly, condition C1 in [12, Definition 2] holds.

Secondly, it follows from the optimality condition of (3.12), (2.3), and Assumption 3.1 (i) that

$$0 \in \partial_\theta \mathcal{L}(\theta^{k+1},\xi^k) = \partial L(\theta^{k+1}) + \partial\|\mu f_{\theta^{k+1}}(\mathcal{X})\|_F^2 - \left\{\mu\eta_{k+1}\xi^k \mid \eta_{k+1} \in \partial f_{\theta^{k+1}}(\mathcal{X})\right\},$$

which means that $\mu\eta_{k+1}^\top(\xi^k - \xi^{k+1}) \in \partial\mathcal{L}_\theta(\theta^{k+1},\xi^{k+1})$. Then, combing it with (3.15), we get

$$W^{k+1} := \begin{pmatrix} \mu\eta_{k+1}^\top(\xi^k - \xi^{k+1}) \\ 0 \end{pmatrix} \in \partial\mathcal{L}(\theta^{k+1},\xi^{k+1}),$$

where $\eta_{k+1} \in \partial f_{\theta^{k+1}}(\mathcal{X})$. Hence, with triangle inequality and Assumption 3.1 (ii),

$$\|W^{k+1}\|_F \le \mu\|\eta_{k+1}\|_F\|\xi^k - \xi^{k+1}\|_F \le \mu\rho\|\xi^k - \xi^{k+1}\|_F. \tag{3.17}$$

At last, if $(\bar{\theta},\bar{\xi})$ is a limit point of some sub-sequence $\{(\theta^k,\xi^k)\}_{k\in\mathcal{K}\subseteq\mathbb{K}}$, based on the continuity of objective function $L(\theta,\xi)$, we can obtain

$$\limsup_{k\in\mathcal{K}\subseteq\mathbb{K}} \mathcal{L}(\theta^k,\xi^k) \le \mathcal{L}(\bar{\theta},\bar{\xi}). \tag{3.18}$$

Moreover, the condition C4 in [12, Definition 2] holds clearly when the subproblems (3.12) and (3.13) are solved exactly. Combining (3.16), (3.17), and (3.18), we have that the sequence $\{(\theta^k,\xi^k)\}_{k\ge 0}$ generated by Algorithm 1 is an approximate gradient-like descent sequence in [12].

Furthermore, let $u := (\theta,\xi)$ for convenience. The function $\mathcal{L}(\cdot)$ is proper, lower semicontinuous, which has the KŁ property directly. Hence, we can get the convergence with the KŁ property [6] and the approximate gradient-like descent sequence by Theorem 1 in [12]. □

Remark 3.7. Actually, the subproblems (3.12) and (3.13) can also be solved inexactly. We can find approximate solution for (3.12) and (3.13) until the following criteria satisfied,

$$\frac{\mu}{2}\|\xi^k - \xi^{k+1}\|_F^2 - (e_1^k)^2 \le \mathcal{L}(\theta^k,\xi^k) - \mathcal{L}(\theta^{k+1},\xi^{k+1}),$$
$$\|W^{k+1}\|_F - e_2^k \le \mu\rho\|\xi^k - \xi^{k+1}\|_F,$$

where $\{e_1^k\}_{k\ge 0}$ and $\{e_2^k\}_{k\ge 0}$ are required to be summable. Together with (3.18), the sequence $\{(\theta^k,\xi^k)\}_{k\ge 0}$ is also an approximate gradient-like descend sequence defined in [12].

4 Numerical Experiments

In this section, we verify the efficiency of our proposed coupled tensor norm regularization for both MLR and DNN on nine real datasets listed in Table 1. We first test the performance of MLR on three face image datasets (ORL, Yale, AR10P) and three biological datasets (lung, TOX-171, lymphoma) downloaded online[1]. Then we test the performance of DNN on Fashion-MNIST, CIFAR-10, and an MRI dataset (Brain Tumor)[2].

Table 1: Details of all datasets. n and n' are the number of training and testing samples, respectively. m denotes the number of features and c is the number of classes.

Dataset	n	n'	m	c
ORL	280	120	1024	40
Yale	100	65	1024	15
AR10p	90	40	2400	10
Lung	153	50	3312	5
TOX-171	100	71	5748	4
Lymphoma	56	40	4026	9
Fashion-MNIST	60000	10000	784	10
CIFAR-10	60000	10000	3072	10
Brain Tumor	2870	394	50176	4

4.1 Multinomial logistic regression

In this subsection, we compare the coupled tensor norm regularization model (3.6) with the ℓ^1-norm [37], ℓ^2-norm [27], Tikhonov regularization [3] models. For all regularized models, we traverse λ from $\{10^{-6}, 10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1\}$ and report the results corresponding to λ with the highest classification accuracy. The gradient or subgradient descent algorithm is adopted to solve the models. The corresponding stopping criteria is set as

$$\|W^{k+1} - W^k\|_F \leq 10^{-4} \text{ or } \|\nabla G(W^k)\|_F \leq 10^{-4},$$

and the maximum number of iterations is 2000. Moreover, we initialize $W^0 = 0$.

The training accuracy, testing accuracy, and the choices of the optimal parameters on the face and biological datasets are elaborated in Tables 2 and 3, respectively.

For all six datasets, our regularization guarantees the highest testing accuracy and lowest generalization error. We further compare the coupled tensor norm and Tikhonov regularizations for all λ in Table 4, which shows that our regularization is more robust.

4.2 Deep neural networks

We continue to compare the performance of DNN with the coupled tensor norm regularization with the ℓ_1 norm [24] and Tikhonov regularization [16]. The network structures we tested VGG-16 [36]. Also, we verify the efficiency of our proposed method by setting the number of training samples from small to large.

For all methods, the hyperparameters λ and μ are optimized from $\{10^{-i}, 5 \cdot 10^{-i}\}_{i=1}^{6}$ and we only report the best performance. The implementation details and the choices of hyperparameters are given in the appendix. For Fashion-MNIST, we show the performance of different regularizers with varying training sizes from 1000 to 60000. The detailed result is shown in Table 5.

[1] `https://jundongl.github.io/scikit-feature/datasets.html`

[2] `https://www.kaggle.com/competitions/machinelearninghackathon/data`

Table 2: Numerical results of multinomial logistic regression for three face datasets.

	ORL			Yale			AR10P		
Model	Training	Testing	λ	Training	Testing	λ	Training	Testing	λ
MLR	95.71%	90.83%	0	90.00%	75.38%	0	85.56%	90.00%	0
MLR-ℓ_1	96.07%	93.33%	10^{-4}	89.00%	75.38%	10^{-6}	85.56%	92.50%	10^{-3}
MLR-ℓ_2	96.07%	93.33%	10^{-6}	89.00%	75.38%	10^{-6}	85.56%	95.00%	10^{-4}
MLR-Tik	96.42%	93.33%	1	90.00%	78.46%	0.1	85.56%	97.50 %	10^{-2}
MLR-ours	96.07%	**95.00%**	10^{-4}	92.00%	**81.54%**	10^{-5}	85.56%	**100%**	10^{-5}

Table 3: Numerical results of multinomial logistic regression for three biological datasets.

	Lung			TOX-171			Lymphoma		
Model	Training	Testing	λ	Training	Testing	λ	Training	Testing	λ
MLR	95.43%	86.00%	0	75.00%	57.75%	0	98.21%	85.00%	0
MLR-ℓ_1	93.46%	92.00%	10^{-2}	66.00%	63.38%	10^{-2}	98.21%	90.00%	10^{-2}
MLR-ℓ_2	94.12%	94.00%	10^{-2}	66.00%	61.97%	10^{-4}	98.21%	87.50%	10^{-3}
MLR-Tik	96.08%	**96.00%**	10^{-1}	72.00%	67.61%	1	98.21%	**95.00%**	1
MLR-ours	96.08%	**96.00%**	10^{-2}	71.00%	**69.01%**	10^{-2}	98.21%	**95.00%**	1

Table 4: Comparisons of difference λ between ours and Tikhonov regularization for MLR on Lymphoma dataset.

λ	1	10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-6}
MLR-Tik	**95.00%**	82.50%	67.50%	42.50%	42.50%	45.00%	45.00%
MLR-ours	**95.00%**	92.50%	87.50%	82.50%	82.50%	82.50%	82.50%

Table 5: The testing accuracy of different regularizers for VGG-16 on Fashion-MNIST.

Model	VGG-16 on Fashion-MNIST			
Training per class	DNN	DNN-ℓ_1	DNN-Tik	DNN-ours
100	80.95%	82.40%	82.16%	**83.38%**
400	86.95%	87.78%	87.13%	**88.15%**
700	88.60%	90.03%	89.66%	**90.73%**
1000	90.67%	90.85%	90.76%	**91.28%**
3000	92.13%	92.70%	92.62%	**92.93%**
6000	93.88%	94.29%	94.30%	**94.73%**

At last, we present the results for CIFAR-10 and Brain Tumor in Tables 6 and 7, respectively.

Details of numerical implementation for DNN: Unless otherwise stated, all experiments use SGD with momentum fixed at 0.9 and mini-batch size fixed at 128. The networks are trained with a fixed learning rate $r_0 = 0.01$ on the first 50 epochs, and then $r_0/10$ for another

Table 6: The testing accuracy of different regularizers for VGG-16 on CIFAR-10.

Model	VGG-16 on CIFAR-10			
Training per class	DNN	DNN-ℓ_1	DNN-Tik	DNN-ours
100	48.57%	49.10%	48.98%	**50.26%**
400	72.90%	73.29%	73.32%	**74.15%**
700	78.97%	79.14%	79.31%	**80.29%**

Table 7: Numerical results of different regularizations for VGG-16 on the MRI dataset Brain Tumor.

	VGG-16 on Brain Tumor		
Model	Training	Testing	λ (μ)
DNN	99.50%	75.48%	0
DNN-ℓ_1	99.97%	76.40%	10^{-3}
DNN-Tik	99.83%	76.67%	10^{-3}
DNN-ours	99.97%	**77.41%**	$5 \cdot 10^{-4}$ (10^{-4})

50 epochs. At step 1 of Algorithm 1, θ is updated once every $M = 2$ epoch of SGD. And at step 2, the step size is set to $1/k$. The stopping criterion is $\|grad_\xi\|_F < tol$, where $grad_\xi$ is the subgradient of (3.13). Further, we set $tol = 10^{-2}$ and the maximum number of iterations as 50. Related hyperparameters are exhibited in the following tables.

Table 8: Hyperparameters of DNN for Fashion-MNIST dataset.

Model	VGG-16			
Training	DNN-ℓ_1	DNN-Tik	DNN-ours	
per class	λ	λ	λ	μ
100	10^{-5}	10^{-5}	$5 \cdot 10^{-4}$	$5 \cdot 10^{-3}$
400	10^{-5}	10^{-5}	$5 \cdot 10^{-4}$	10^{-3}
700	10^{-5}	10^{-5}	$5 \cdot 10^{-4}$	$5 \cdot 10^{-4}$
1000	10^{-4}	10^{-5}	$5 \cdot 10^{-4}$	$5 \cdot 10^{-4}$
3000	10^{-4}	10^{-5}	$5 \cdot 10^{-4}$	$5 \cdot 10^{-4}$
6000	10^{-4}	10^{-5}	$5 \cdot 10^{-4}$	$5 \cdot 10^{-4}$

Table 9: Hyperparameters used in CIFAR-10 dataset.

Model	VGG-16				
Training	DNN	DNN-ℓ_1	DNN-Tik	DNN-ours	
per class	λ	λ	λ	λ	μ
100	0	10^{-4}	10^{-5}	$5 \cdot 10^{-5}$	$5 \cdot 10^{-2}$
400	0	10^{-4}	10^{-3}	$5 \cdot 10^{-4}$	$5 \cdot 10^{-2}$
700	0	10^{-3}	10^{-5}	$5 \cdot 10^{-4}$	$5 \cdot 10^{-2}$

5 Conclusions

In this paper, we have introduced a novel regularization strategy that incorporates coupled tensor norms to improve the generalization capabilities of classification models. For MLR, we have established that the regularization exhibits convexity, differentiability, and Lipschitz continuous gradient. These properties have enabled us to prove the global convergence of the gradient descent algorithm when applied to MLR models introduced with our coupled tensor regularizer. In the field of DNN, different from MLR, the model is still a nonconvex and nondifferentiable optimization problem when coupled tensor norm regularizer is introduced. Despite these complexities, we have demonstrated the global convergence of an alternate minimization method tailored to such complicated models. Empirically, our regularization method has been rigorously evaluated against conventional regularization techniques, including ℓ_1, ℓ_2, and Tikhonov regularizations. Our experiments have substantiated the efficiency and robustness of our regularization, performing its superiority in terms of improving model performance and generalization.

Acknowledgement

We are grateful to the editor and anonymous reviewers for their insightful comments and helpful suggestions that have improved our manuscript.

References

[1] E. Acar, R. Bro and A.K. Smilde, Data fusion in metabolomics using coupled matrix and tensor factorizations, *Proc. IEEE* 103 (2015), 1602–1620 (2015).

[2] S. Arora, R. Ge, B. Neyshabur and Y. Zhang, Stronger generalization bounds for deep nets via a compression approach, in *International Conference on Machine Learning*, 2018, pp. 254–263.

[3] C.M. Bishop, Training with noise is equivalent to Tikhonov regularization, *Neural Comput.* 7 (1995) 108–116.

[4] C.M. Bishop, *Pattern recognition and machine learning*, Springer-Verlag, New York, 2006.

[5] C.R. Berger, Z. Wang, J. Huang and S. Zhou, Application of compressive sensing to sparse channel estimation, *IEEE Commun. Mag.* 48 (2010) 164–174.

[6] J. Bolte, A. Daniilidis and A. Lewis, The Łojasiewicz inequality for nonsmooth subanalytic functions with applications to subgradient dynamical systems, *SIAM J. Optim.* 17 (2007) 1205–1223.

[7] J. Bolte, S. Sabach and M. Teboulle, Proximal alternating linearized minimization for nonconvex and nonsmooth problems, *Math. Program.* 146 (2014) 459–494.

[8] C. Cortes, M. Mohri and A. Rostamizadeh, L2 regularization for learning kernels,*arXiv:1205.2653.*

[9] E. Candes and B. Recht, Exact matrix completion via convex optimization, *Commun. ACM* 55 (2012) 111–119.

[10] J.F. Cai, E.J. Candès and Z. Shen, A singular value thresholding algorithm for matrix completion, *SIAM J. Optim.* 20 (2010) 1956–1982.

[11] D. Gerth, A new interpretation of (Tikhonov) regularization, *Inverse Problem* 37 (2021) 064002.

[12] E. Gur, S. Sabach and S. Shtern, Convergent nested alternating minimization algorithms for nonconvex optimization problems, *Math. Oper. Res.* 48 (2023) 53–77.

[13] Z. Hu, F. Nie, R. Wang and X. Li, Low rank regularization: a review, *Neural Netw.* 136 (2021) 218–232.

[14] S. Ioffe and C. Szegedy, Batch normalization: Accelerating deep network training by reducing internal covariate shift, in *International conference on machine learning*, 2015, pp. 448–456.

[15] J. Jiang and X. Chen, Optimality conditions for nonsmooth nonconvex-nonconcave min-max problems and generative adversarial networks, *SIAM J. Math. Data Sci.* 5 (2023), 693-722.

[16] A. Krogh and J. Hertz, A simple weight decay can improve generalization, in *Advances in Neural Information Processing Systems*, 1991, pp. 950–957.

[17] A. Kolbeinsson, J. Kossaifi, Y. Panagakis, A. Bulat and A. Anandkumar, I. Tzoulaki, P. M. Matthews, Tensor dropout for robust learning, *IEEE J. Sel. Topics Signal Process.* 15 (2021) 630–640.

[18] B. Krishnapuram, L. Carin, M.A. Figueiredo and A.J. Hartemink, Sparse multinomial logistic regression: fast algorithms and generalization bounds, *IEEE Trans. Pattern Anal. Mach. Intell.* 27 (2005), 957–968.

[19] J. Kukačka, V. Golkov and D. Cremers, Regularization for deep learning: A taxonomy, *arXiv:1710.10686.*

[20] T.G. Kolda and B.W. Bader, Tensor decompositions and applications, *SIAM Rev.* 51 (2009) 455–500.

[21] M. Lustig, D. Donoho and J.M. Pauly, Sparse MRI: The application of compressed sensing for rapid MR imaging, *Magn. Reson. Med.* 58 (2007) 1182–1195.

[22] J.J. Moré and D.J. Thuente, Line search algorithms with guaranteed sufficient decrease, *ACM Trans. Math. Softw.* 20 (1994) 286–307.

[23] R. Mazumder, T. Hastie and R. Tibshirani, Spectral regularization algorithms for learning large incomplete matrices, *J. Mach. Learn. Res.* 11 (2010) 2287–2322.

[24] R. Ma, J. Miao, L. Niu and P. Zhang, Transformed ℓ_1 regularization for learning sparse deep neural networks, *Neural Netw.* 119 (2019) 286–298.

[25] R. Moradi, R. Berangi and B. Minaei, A survey of regularization strategies for deep models, *Artif. Intell. Rev.* 53 (2020) 3947–3986.

[26] X. Ma and W. Liu, Recent advances of manifold regularization, in *Manifolds II Theory and Applications*, Paul Bracken (ed.), IntechOpen London, UK, 2018.

[27] E. Ndiaye, O. Fercoq, A. Gramfort and J. Salmon, Gap safe screening rules for sparse multi-task and multi-class models, in *Advances in Neural Information Processing Systems*, 2015, pp. 811–819.

[28] F. Nie, H. Huang and C. Ding, Low-rank matrix recovery via efficient Schatten p-norm minimization, in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2012, pp. 655–661.

[29] S. Osher, Z. Shi and W. Zhu, Low dimensional manifold model for image processing, *SIAM J. Imaging Sci.* 10 (2017) 1669–1690.

[30] G. Peyré, Image processing with non-local spectral bases, *Multiscale Model. Simul.* 7 (2008) 703–730.

[31] G. Peyré, Manifold models for signals and images, *Comput. Vis. Image Understanding* 113 (2009) 248–260.

[32] L.C. Potter, E. Ertin, J.T. Parker and M. Cetin, Sparsity and compressed sensing in radar imaging, *Proc. IEEE* 98 (2010) 1006–1020.

[33] Y. Panagakis, J. Kossaifi, G.G. Chrysos, J. Oldfield and M.A. Nicolaou, A. Anandkumar, S. Zafeiriou, Tensor methods in computer vision and deep learning, *Proc. IEEE* 109 (2021) 863–890.

[34] B. Shekar and G. Dagnew, L1-regulated feature selection and classification of microarray cancer data using deep learning, in *Proceedings of 3rd International Conference on Computer Vision and Image Processing*, 2020, pp. 227–242.

[35] C. Schenker, J.E. Cohen and E. Acar, A flexible optimization framework for regularized matrix-tensor factorizations with linear couplings. *IEEE J. Sel. Topics Signal Process.* 15 (2020), 506–521.

[36] K. Simonyan and A. Zisserman, Very deep convolutional networks for large-scale image recognition, *arXiv:1409.1556*

[37] M. Schmidt, G. Fung and R. Rosales, Fast optimization methods for ℓ_1 regularization: A comparative study and two new approaches, in *18th European Conference on Machine Learning*, 2007, pp. 286–297.

[38] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever and R. Salakhutdinov, Dropout: a simple way to prevent neural networks from overfitting, *J. Mach. Learn. Res.* 15 (2014) 1929–1958.

[39] W. Sun and Y.-X. Yuan, *Optimization Theory and Methods: Nonlinear Programming*, Springer, New York, 2006.

[40] R.T. Rockafellar and R.J-B. Wets, *Variational analysis*, Springer-Verlag, Berlin Heidelberg, 2009.

[41] V.N. Vapnik, An overview of statistical learning theory, *IEEE Trans. Neural Netw.* 10 (1999) 988–999.

[42] K. Wimalawarne and H. Mamitsuka, Efficient convex completion of coupled tensors using coupled nuclear norms, in *Advances in Neural Information Processing Systems*, 2018, pp.6902–6910.

[43] K. Wimalawarne, M. Yamada and H. Mamitsuka, Convex coupled matrix and tensor completion. *Neural Comput.* 30 (2018), 3095–3127.

[44] L. Wan, M. Zeiler, S. Zhang, Y.L. Cun and R. Fergus. Regularization of neural networks using dropconnect. in *Proceedings of the 30th International Conference on Machine Learning*, 2013, pp. 1058–1066.

[45] C. Zhang, S. Bengio, M. Hardt, B. Recht and O. Vinyals, Understanding deep learning (still) requires rethinking generalization, *Commun. ACM* 64 (2021) 107–115.

[46] H. Zhang, J. Qian, B. Zhang, J. Yang, C. Gong and Y. Wei. Low-Rank matrix recovery via modified Schatten-p norm minimization with convergence guarantees. *IEEE Trans. Signal Process.* 29 (2019) 3132–3142.

[47] J. Zeng, G. Cheung, M. Ng, J. Pang and C. Yang, 3D point cloud denoising using graph Laplacian regularization of a low dimensional manifold model, *IEEE Trans. Image Process.* 29 (2019) 3474–3489.

[48] P. Zhang, R. Wang and N. Xiu, Multinomial logistic regression classifier via $\ell_{q,0}$-proximal newton algorithm, *Neurocomputing* 468 (2022) 148–164.

[49] W. Zhu, Q. Qiu, J. Huang, R. Calderbank, G. Sapiro and I. Daubechies, LDMNet: Low dimensional manifold regularized neural networks, in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 2743–2751.

Manuscript received 17 November 2022
revised 25 January 2024
accepted for publication 1 March 2024

Ying Gao
LMIB of the Ministry of Education
School of Mathematical Sciences
Beihang University
Beijing, 100191, People's Republic of China
E-mail address: gaoYn7115@163.com

Yunfei Qu
LMIB of the Ministry of Education
School of Mathematical Sciences
Beihang University
Beijing, 100191, People's Republic of China
E-mail address: yunfei_math@hotmail.com

Chunfeng Cui
LMIB of the Ministry of Education
School of Mathematical Sciences
Beihang University
Beijing, 100191, People's Republic of China
E-mail address: chunfengcui@buaa.edu.cn