Supervised Matrix Factorization: Local Landscape Analysis and Applications

Joowon Lee [*1] **Hanbaek Lyu** [*2] **Weixin Yao** [*3]

Abstract

Supervised matrix factorization (SMF) is a classical machine learning method that seeks low-dimensional feature extraction and classification tasks at the same time. Training an SMF model involves solving a non-convex and factor-wise constrained optimization problem with at least three blocks of parameters. Due to the high non-convexity and constraints, theoretical understanding of the optimization landscape of SMF has been limited. In this paper, we provide an extensive local landscape analysis for SMF and derive several theoretical and practical applications. Analyzing diagonal blocks of the Hessian naturally leads to a block coordinate descent (BCD) algorithm with adaptive step sizes. We provide global convergence and iteration complexity guarantees for this algorithm. Full Hessian analysis gives minimum L_2-regularization to guarantee local strong convexity and robustness of parameters. We establish a local estimation guarantee under a statistical SMF model. We also propose a novel GPU-friendly neural implementation of the BCD algorithm and validate our theoretical findings through numerical experiments. Our work contributes to a deeper understanding of SMF optimization, offering insights into the optimization landscape and providing practical solutions to enhance its performance.

1. Introduction

In classical classification models, the standard approach uses observed high-dimensional raw features as the input. In many cases, these features may include vast amounts of irrelevant or redundant information, posing challenges for generalization and interpretability. To address this, the integration of interpretable dimension reduction techniques prior to classification becomes important.

Matrix factorization (MF) is a classical unsupervised feature extraction framework that learns latent structures in complex datasets. It is regularly applied in the analysis of text and images (Elad & Aharon, 2006; Mairal et al., 2007; Peyré, 2009). In particular, *nonnegative matrix factorization* (NMF) (Lee & Seung, 2000) stands out as one of the most widely used modern MF tools, aiming to approximately factorize a data matrix into the product of two *nonnegative* matrices. Nonnegativity is crucial for enabling interpretable ”parts-based learning” (Lee & Seung, 1999) of high-dimensional objects. This feature has led NMF finding applications in various domains, including text analysis for topic modeling, image reconstruction, bioinformatics, and the extraction of latent motifs from networks (Sitek et al., 2002; Berry & Browne, 2005; Berry et al., 2007; Chen et al., 2011; Taslaman & Nilsson, 2012; Boutchko et al., 2015; Ren et al., 2018; Lyu et al., 2024).

Supervised matrix factorization (SMF) is a popular classical machine learning method that aims to perform low-dimensional feature extraction and classification tasks simultaneously. Given that matrix factorization and classification are not inherently aligned objectives, SMF involves a necessary trade-off when aiming to achieve both goals simultaneously. As its name implies, SMF integrates a classification model and MF into a single optimization problem. While it has been applied to various problem domains (Zhao et al., 2015; Yankelevsky & Elad, 2017; Li et al., 2019), our current understanding of its optimization landscape and the behavior of widely used iterative optimization algorithms remains limited.

At its core, training SMF requires solving a non-convex constrained optimization problem involving three or four blocks of parameters. Even the optimization landscape of NMF, a two-block constrained bi-convex problem, is not completely understood (Panageas et al., 2020; Bjorck et al., 2021) to date. This lack of thorough understanding makes the optimization landscape of SMF challenging to

[*] Equal contribution [1] Department of Mathematics, University of Wisconsin - Madison, WI, USA [2] Department of Statistics, University of Wisconsin - Madison, WI, USA [3] Department of Statistics, University of California, Riverside, CA, USA. Correspondence to: Hanbaek Lyu <hlyu@math.wisc.edu>.

Proceedings of the 41^{st} International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

unravel. The goal of this paper is to establish a theoretical and algorithmic foundation for SMF, providing researchers with a reliable and rigorous background.

1.1. Contributions

We establish the following novel contributions in this work:

- **Local Landscape Analysis**: We provide a local landscape analysis of the general SMF optimization problem. We explicitly compute the (4×4) block structure of the corresponding Hessian matrix and determine the minimum L_2-regularization on each parameter for local strong convexity. (Theorems 4.3 and C.6)
- **BCD Algorithm and Convergence Guarantee**: We derive a *block coordinate descent* (BCD) algorithm for SMF and establish its convergence guarantees by obtaining bounds on the eigenvalues of the diagonal blocks in the Hessian matrix. Additionally, we demonstrate that the algorithm achieves an ε-stationary point of the objective within $O(\varepsilon^{-1}(\log \varepsilon^{-1})^2)$ iterations (Theorem 4.4).
- **Local Consistency and Estimation Guarantee**: We show the existence of a local minimizer of an L_2-regularized landscape near a stationary point of SMF. Under a statistical SMF model, we demonstrate that at least one matrix factor can be locally consistently estimated with high probability (Theorem 4.5).
- **Neural Network Implementation**: We provide a compact neural network implementation of the proposed BCD algorithm for SMF that enables GPU acceleration. (Figure 2).

1.2. Related works

Recently, Lee et al. (Lee et al., 2023) found a method to reformulate SMF problems as low-rank matrix estimation by employing a 'double-lifting' idea in the parameter space. When the lifted problem is well-conditioned, they demonstrated that *low-rank projected gradient descent* (LPGD) can find a global optimum for the original problem at an exponential rate. However, their approach faces limitations in handling constraints on individual factor matrices, such as enforcing the nonnegativity of factors. It is because one cannot find an optimal nonnegative matrix decomposition from *singular value decomposition* (SVD). To address this limitation, we take a different approach by directly analyzing the local (constrained) landscape of SMF and investigating the robustness of local optima under L_2-regularization.

The SMF training problem in (3) is a non-convex and potentially constrained optimization problem, often featuring non-unique minimizers. Since it is difficult to solve exactly, approximate procedures such as BCD (see, e.g., (Wright, 2015)) are often used. These approaches utilize the fact that the objective function in (3) is convex in each of the four (matrix) variables. Such an algorithm iteratively optimizes one block while fixing the others (see (Mairal et al., 2008; Austin et al., 2018; Leuschner et al., 2019; Ritchie et al., 2020)). However, existing literature on the convergence analysis or statistical estimation bounds for such algorithms remains somewhat limited. Referring to established convergence results for BCD methods (Grippo & Sciandrone, 2000; Xu & Yin, 2013), one can, at best, guarantee asymptotic convergence to the stationary points. Alternatively, polynomial convergence toward Nash equilibria or the objective (3) is achievable, contingent upon careful verification of the assumptions underpinning these general findings. Our derivation and analysis of Algorithm 1 and 2 are based on the framework of block projected gradient descent viewed as block majorization-minimization (Lyu & Li, 2023).

One of our main results of non-asymptotic consistency for constrained and regularized *maximum likelihood estimation* (MLE) (Theorem D.1) plays a crucial role in establishing the local consistency of SMF in the general case (Theorem 4.5). This result draws inspiration from the work on local consistency guarantees for non-concave penalized MLE in (Fan & Li, 2001).

Various SMF-type models have been proposed in the past two decades. Following (Lee et al., 2023), we divide them into two categories depending on whether the extracted low-dimensional feature or the feature extraction mechanism itself is supervised. We refer to them as feature-based and filter-based SMF, respectively. Feature-based SMF models include the one by Mairal et al. (Mairal et al., 2008; 2011) as well as the more recent model of convolutional matrix factorization by (Kim et al., 2016). Filter-based SMF models have been studied more recently in the literature on SMF, particularly in studies on supervised nonnegative matrix factorization (Austin et al., 2018; Leuschner et al., 2019) and supervised *principal component analysis* (PCA) (Ritchie et al., 2020).

2. Preliminaries

2.1. Notations

In this paper, we use the notation $\mathbb{R}^p$ to represent the ambient space for data, equipped with standard inner project $\langle \cdot, \cdot \rangle$, inducing the Euclidean norm $\|\cdot\|$. We refer to the set $\{0, 1, \ldots, \kappa\}$ as the space of class labels, containing $\kappa + 1$ classes. For a convex subset $\boldsymbol{\Theta}$ in an Euclidean space, we denote $\Pi_{\boldsymbol{\Theta}}$ the projection operator onto $\boldsymbol{\Theta}$.

For a matrix $\mathbf{A} = (a_{ij}) \in \mathbb{R}^{m \times n}$, the expressions $\mathbf{A}[i, :]$ and $\mathbf{A}[:, j]$ refer to the ith row and the jth column of $\mathbf{A}$ for each $1 \leq i \leq m$ and $1 \leq j \leq n$, respectively. For each integer $n \geq 1$, $\mathbf{I}_n$ denotes the $n \times n$ identity matrix. We denote its Frobenius, operator (2-),

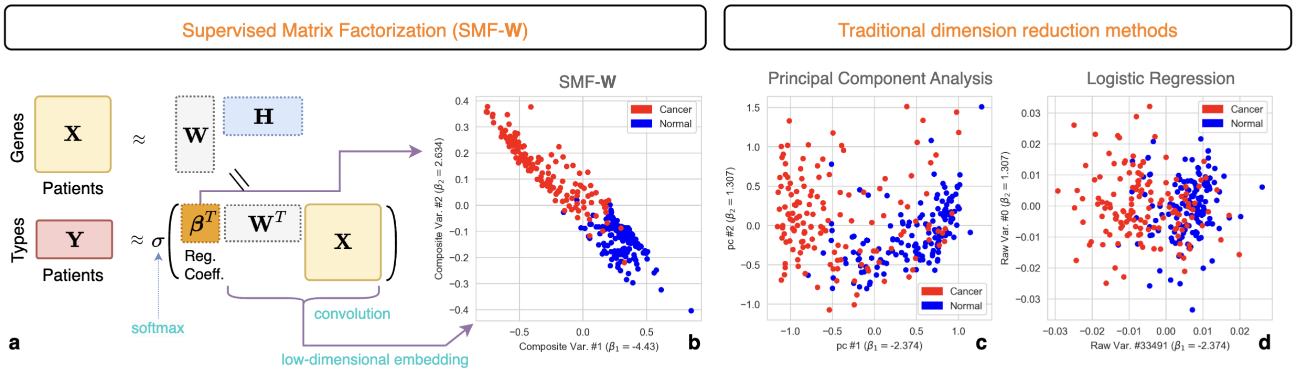


Figure 1. (**a**) Overall scheme of Supervised Matrix Factorization (specifically, SMF-$\mathbf{W}$ with rank $r = 2$). The columns of $\mathbf{W}$ serve as 'composite variables' or 'filters', whose association with the labels is given by the regression coefficients in $\boldsymbol{\beta}$. Taking convolution of the raw data matrix $\mathbf{W}$ with $\mathbf{W}$ gives a supervised dimension reduction, as illustrated in **b** for a $35,982$-dimensional gene microarray sequence data for breast cancer patients. Similar dimension reduction results obtained by (**c**) principal component analysis along with logistic regression and (**d**) logistic regression to select the two most highly associated raw variables show less clear separation.

and supremum norm by $\|\mathbf{A}\|_F^2 := \sum_{i,j} a_{ij}^2$, $\|\mathbf{A}\|_2 := \sup_{\mathbf{x}\in\mathbb{R}^n,\, \|\mathbf{x}\|=1} \|\mathbf{A}\mathbf{x}\|$, $\|\mathbf{A}\|_\infty := \max_{i,j} |a_{ij}|$, respectively. For square symmetric matrices $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{n\times n}$, $\mathbf{A} \preceq \mathbf{B}$ indicates that $\mathbf{v}^T\mathbf{A}\mathbf{v} \le \mathbf{v}^T\mathbf{B}\mathbf{v}$ holds for all unit vectors $\mathbf{v} \in \mathbb{R}^n$. If $0 < \alpha^- < \alpha^+$, then we write $A \asymp \alpha^\pm B$ to denote $\alpha^- B \preceq A \preceq \alpha^+ B$. The horizontal concatenation of two matrices $\mathbf{A}$ and $\mathbf{B}$ is denoted by $[\mathbf{A}, \mathbf{B}]$ when their dimensions match.

2.2. Model formulation

Here we give a mathematical formulation of the SMF problem. For the simplicity of presentation, here we focus on the case of binary labels. We provide full details on general multi-label cases and score functions for the classifier in Appendix B. Consider the following problem setting: we have a set of n observations $(y_i, \mathbf{x}_i, \mathbf{x}_i')$ for $i = 1, \dots, n$ where $y_i \in \{0, 1\}$ represents an observed binary label, $\mathbf{x}_i \in \mathbb{R}^p$ denotes a high-dimensional feature, and $\mathbf{x}_i' \in \mathbb{R}^q$ is a low-dimensional auxiliary feature for the i-th individual ($p \gg q$). To predict y_i, a low-dimensional representation of $\mathbf{x}_i$ in dimension $r \ll p$ for some suitable r may be utilized, combined with $\mathbf{x}_i'$. This implies that the observed $\mathbf{x}_i$ is approximated by a linear transformation of the *basis* vectors $\mathbf{w}_1, \dots, \mathbf{w}_r \in \mathbb{R}^p$ using a suitable code $\mathbf{h}_i$. Let $\mathbf{W} = [\mathbf{w}_1, \dots, \mathbf{w}_r] \in \mathbb{R}^{p\times r}$ be referred to as the *(latent) factor matrix*, and $\mathbf{H} = [\mathbf{h}_1, \dots, \mathbf{h}_n] \in \mathbb{R}^{r\times n}$ as its *code matrix*. In a more compact form, $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_n] \approx \mathbf{W}\mathbf{H}$, known as *reconstruction*. In practical terms, we can determine r as the approximate rank of the data matrix $\mathbf{X}$.

Now, we present our probabilistic modeling assumption. Consider fixed parameters $\mathbf{W} \in \mathbb{R}^{p\times r}$, $\mathbf{h}_i \in \mathbb{R}^r$, $\boldsymbol{\beta} \in \mathbb{R}^r$, and $\boldsymbol{\gamma} \in \mathbb{R}^q$. Suppose y_i is a realization of a random variable whose conditional distribution is defined as

$$\mathbb{P}\left(y_i = 1 \,|\, \mathbf{x}_i, \mathbf{x}_i'\right) = \frac{\exp(a_i)}{1 + \exp(a_i)}, \tag{1}$$

where $a_i \in \mathbb{R}$ is the *activation* for y_i. The activation is defined in two ways, depending on whether we use a 'feature-based' model (SMF-$\mathbf{H}$) or a 'filter-based' model (SMF-$\mathbf{W}$):

$$a_i = \begin{cases} \boldsymbol{\beta}^T\mathbf{W}^T\mathbf{x}_i + \boldsymbol{\gamma}^T\mathbf{x}_i' & \text{for SMF-}\mathbf{W} \\ \boldsymbol{\beta}^T\mathbf{h}_i + \boldsymbol{\gamma}^T\mathbf{x}_i' & \text{for SMF-}\mathbf{H}. \end{cases} \tag{2}$$

Here, $(\boldsymbol{\beta}, \boldsymbol{\gamma})$ are logistic regression coefficients associated with input features $(\mathbf{h}_i, \mathbf{x}_i')$ or $(\mathbf{W}^T\mathbf{x}_i, \mathbf{x}_i')$, respectively. In equation (2), the code $\mathbf{h}_i$ or the 'filtered feature' $\mathbf{W}^T\mathbf{x}_i$ is the low-dimensional representation of $\mathbf{x}_i$. Notable differences between SMF-$\mathbf{H}$ and SMF-$\mathbf{W}$ arise when predicting the unknown label of a test point (Lee et al., 2023).

Let $\mathbf{Z} := (\mathbf{W}, \mathbf{H}, \boldsymbol{\beta}, \boldsymbol{\gamma})$ be our block parameters of interest. In order to estimate $\mathbf{Z}$ from observed data $(\mathbf{x}_i, \mathbf{x}_i', y_i)$ for $i = 1, \dots, n$, we consider the following multi-objective non-convex *constrained* optimization problem:

$$\min_{\substack{\mathbf{W}\in\mathcal{C}_1, \mathbf{H}\in\mathcal{C}_2 \\ \boldsymbol{\beta}\in\mathcal{C}_3, \boldsymbol{\Gamma}\in\mathcal{C}_4}} f(\mathbf{Z}) := \xi\|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2 + \sum_{i=1}^n \ell(y_i, a_i) \tag{3}$$

$$\text{where } \ell(y_i, a_i) = \log(1 + \exp(a_i)) - y_i a_i.$$

Here $\mathcal{C}_j$ for $j = 1, \dots, 4$ represent convex constraint sets of each block parameter, $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_n] \in \mathbb{R}^{p\times n}$, a_i is as in (2), and the last term in (3) is the classification loss defined as the negative log-likelihood. Note that the four block parameters are *individually* assumed to be constrained in (3). A tuning parameter ξ controls the trade-off between the dual objectives of classification and matrix factorization. The stated problem is inherently non-convex, involving four block parameters that may come with additional constraints such as bounded norm. This formulation encompasses several classical models as special cases. Specifically, when $\xi \gg 1$, it transforms into the classical matrix factorization with constraints (Lee & Seung, 1999; 2000).

3. Methods

3.1. Sketch of idea for Constrained Matrix Factorization

We illustrate our approach to analyzing SMF by demonstrating it for the simpler setting of constrained MF without supervision, which amounts to minimizing the bi-convex objective $(\mathbf{W}, \mathbf{H}) \mapsto \|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2$ under factor-wise constraints on $\mathbf{W}$ and $\mathbf{H}$. Its Hessian is given by

$$\begin{array}{c} \\ \mathrm{vec}(\mathbf{W}) \\ \mathrm{vec}(\mathbf{H}) \end{array} \begin{array}{c} \begin{array}{cc} \quad \mathrm{vec}(\mathbf{W})^T & \qquad \mathrm{vec}(\mathbf{H})^T \end{array} \\ \begin{bmatrix} \mathbf{H}\mathbf{H}^T \otimes \mathbf{I}_p & A_{12} \\ A_{12}^T & \mathbf{I}_n \otimes \mathbf{W}^T\mathbf{W} \end{bmatrix} \end{array}, \tag{4}$$

where $A_{12} = [(\mathbf{H} \otimes \mathbf{W}) + \mathbf{I}_r \otimes (\mathbf{W}\mathbf{H} - \mathbf{X})]\, \mathbf{C}^{(r,n)}$ with commutation matrix $\mathbf{C}^{(r,n)} \in \{0,1\}^{rn \times rn}$ (See Appendix A for a formal definition). Denoting the diagonal blocks as A_{11} and A_{22}, we have

$$\begin{aligned} \lambda_{\min}(\mathbf{H}\mathbf{H}^T)\mathbf{I}_{pr} \preceq A_{11} \preceq \lambda_{\max}(\mathbf{H}\mathbf{H}^T)\mathbf{I}_{pr} \\ \lambda_{\min}(\mathbf{W}^T\mathbf{W})\mathbf{I}_{nr} \preceq A_{22} \preceq \lambda_{\max}(\mathbf{W}^T\mathbf{W})\mathbf{I}_{nr}. \end{aligned} \tag{5}$$

We first leverage the upper bounds in (5) to derive a BCD algorithm with adaptive step size as well as its iteration complexity for achieving an ε-stationary point. Namely, from (5), it follows that the marginal loss restricted to $\mathbf{W}$ or $\mathbf{H}$ has Lipschitz continuous gradients with parameters $\lambda_{\max}(\mathbf{H}\mathbf{H}^T)$ and $\lambda_{\max}(\mathbf{W}^T\mathbf{W})$, respectively. So we can naturally derive the following BCD algorithm ($\varepsilon > 0$ fixed)

$$\begin{aligned} \mathbf{W} &\leftarrow \Pi \left(\mathbf{W} - \frac{1}{\lambda_{\max}(\mathbf{H}\mathbf{H}^T) + \varepsilon} (\mathbf{W}\mathbf{H} - \mathbf{X})\mathbf{H}^T \right), \\ \mathbf{H} &\leftarrow \Pi' \left(\mathbf{H} - \frac{1}{\lambda_{\max}(\mathbf{W}\mathbf{W}^T) + \varepsilon} \mathbf{W}^T(\mathbf{W}\mathbf{H} - \mathbf{X}) \right) \end{aligned} \tag{6}$$

with Π, Π' being suitable projection operators. Using the recent complexity analysis of block majorization-minimization algorithms in (Lyu & Li, 2023), we can obtain iteration complexity of the BCD algorithm (6) for MF.

Next, when $\mathbf{X}$ can be approximated by a low-rank factorization $\mathbf{X} \approx \mathbf{W}_\star\mathbf{H}_\star$ with the true factors $\mathbf{W}_\star$ and $\mathbf{H}_\star$, it is desirable to introduce regularization to the objective to ensure that the new objective is locally strongly convex and can be minimized near $(\mathbf{W}_\star, \mathbf{H}_\star)$ for efficient and robust parameter estimation. While L_2-regularization naturally improves local convexity, it may significantly perturb the local landscape. Therefore, applying the least amount of L_2-regularization is ideal to minimize this perturbation. While it may be challenging to 'curve-up' the landscape to maintain minimization at $(\mathbf{W}_\star, \mathbf{H}_\star)$, we can *preserve at least one of the factors*, either $\mathbf{W}_\star$ or $\mathbf{H}_\star$, at the new minimizer.

We establish these claims by a local landscape analysis. In the 'large-sample regime' ($n \gg p$), we find that regularization is required only for $\mathbf{H}$. This results in a new local landscape that is strongly convex near $(\mathbf{W}_\star, \mathbf{H}_\star)$ and is minimized at $(\mathbf{W}_\star, \mathbf{H}')$ for some $\mathbf{H}'$. The distance between $\mathbf{H}'$ and $\mathbf{H}_\star$ is minimized when the added L_2-regularization term for $\mathbf{H}$ is the smallest. Similarly, in the 'high-dimensional regime' ($p \gg n$), regularization is only necessary for $\mathbf{W}$ and obtain a new local landscape that is strongly convex near $(\mathbf{W}_\star, \mathbf{H}_\star)$ and minized at $(\mathbf{W}', \mathbf{H}_\star)$ for some $\mathbf{W}'$.

To illustrate the key idea, first recall that block-diagonal dominance is a well-established sufficient condition to ensure that a block matrix is positive definite, as outlined in (Feingold & Varga, 1962). Let λ_1 and λ_2 denote the L_2-regularization parameters for $\mathbf{W}$ and $\mathbf{H}$ respectively. In our context, this condition can be expressed as follows:

$$\lambda_{\min}(\mathbf{H}_\star\mathbf{H}_\star^T) + \lambda_1 - \|A_{12}\|_2 > 0, \tag{7}$$

$$\lambda_{\min}(\mathbf{W}_\star^T\mathbf{W}_\star) + \lambda_2 - \|A_{12}\|_2 > 0. \tag{8}$$

For simplicity, assume typical orders for the eigenvalues of the matrices in the Hessian (4):

$$\lambda_{\min}(\mathbf{H}_\star\mathbf{H}_\star^T) = \Theta(rn), \quad \lambda_{\min}(\mathbf{W}_\star^T\mathbf{W}_\star) = \Theta(rp), \\ \|A_{12}\|_2 = \Theta(r\sqrt{pn}).$$

Now consider the large-sample setting ($n \gg p$). The $\mathbf{W}$-block already has block-diagonal dominance $\lambda_{\min}(A_{11}) - \|A_{12}\|_2 = \Theta(rn) - \Theta(r\sqrt{pn}) > 0$ but the $\mathbf{H}$-block does not: $\lambda_{\min}(A_{22}) - \|A_{21}\|_2 = \Theta(rp) - \Theta(r\sqrt{pn}) < 0$. This allows us to set $\lambda_1 = 0$ (i.e., no L_2-regularization for $\mathbf{W}$ needed), while we may use $\lambda_2 = \Theta(r\sqrt{pn})$. Consequently, the L_2-regularized objective $\|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2 + \frac{\lambda_2}{2}\|\mathbf{H}\|_F^2$ is ρ-strongly convex at $(\mathbf{W}_\star, \mathbf{H}_\star)$ with $\rho = \lambda_2 - \Theta(r\sqrt{pn})$. By Taylor expansion, one can show that it is locally minimized at $(\mathbf{W}_\star, \mathbf{H}')$, where $\|\mathbf{H}' - \mathbf{H}_\star\|_F \leq \frac{3\lambda_2\|\mathbf{H}_\star\|_F}{\lambda_2 - \Theta(r\sqrt{pn})}$ when $\|\mathbf{H}_\star\|_F$ is sufficiently small.

Conversely, in the high-dimensional setting ($p \gg n$), we can set $\lambda_2 = 0$ and $\lambda_1 = \Theta(r\sqrt{pn})$. Then the L_2-regularized objective $\|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2 + \frac{\lambda_1}{2}\|\mathbf{W}\|_F^2$ is ρ-strongly convex near $(\mathbf{W}_\star, \mathbf{H}_\star)$ with $\rho = \lambda_1 - \Theta(r\sqrt{pn})$. It is locally minimized at $(\mathbf{W}', \mathbf{H}_\star)$, where $\|\mathbf{W}' - \mathbf{W}_\star\|_F \leq \frac{3\lambda_1\|\mathbf{W}_\star\|_F}{\lambda_1 - \Theta(r\sqrt{pn})}$.

While our analysis for SMF follows a similar logical framework as illustrated here for MF, the full analysis is substantially more challenging due to the Hessian's representation as a 4×4 block matrix, involving intricate interactions among the four block parameters $\mathbf{W}, \mathbf{H}, \boldsymbol{\beta}$ and $\boldsymbol{\Gamma}$.

3.2. BCD algorithm for SMF

We consider both filter- and feature-based SMF models in (3), allowing for convex constraints on each of the variables $\mathbf{W}, \mathbf{H}, \boldsymbol{\beta}$, and $\boldsymbol{\Gamma}$. A key scenario of interest involves incorporating nonnegativity constraints on both $\mathbf{W}$ and $\mathbf{H}$, resulting in the SMF model (3) that combines NMF with logistic regression in two different ways. For simplicity, we only give a full statement of the BCD algorithm for SMF-$\mathbf{W}$. The corresponding algorithm for SMF-$\mathbf{H}$ is given in Algorithm 2 in Appendix.

Algorithm 1 BCD algorithm for SMF-$\mathbf{W}$

```
Input: $\mathbf{X} \in \mathbb{R}^{p\times n}$ (Data); $\mathbf{X}_{\text{aux}} \in \mathbb{R}^{q\times n}$ (Auxiliary covariate); $\mathbf{Y}_{\text{label}} \in \{0,\dots,\kappa\}^{1\times n}$ (Label);
Constraints: Convex subsets $\mathcal{C}_1 \subseteq \mathbb{R}^{p\times r}$, $\mathcal{C}_2 \subseteq \mathbb{R}^{r\times n}$, $\mathcal{C}_3 \subseteq \mathbb{R}^{r\times\kappa}$, $\mathcal{C}_4 \subseteq \mathbb{R}^{q\times\kappa}$
Parameters: $\xi \geq 0$ (Tuning parameter); $T \in \mathbb{N}$ (number of iterations); $(\eta_{k;i})_{k\geq 1, 1\leq i\leq 4}$ (step-sizes)
Initialize $\mathbf{W} \in \mathcal{C}_1$, $\mathbf{H} \in \mathcal{C}_2$, $\boldsymbol{\beta} \in \mathcal{C}_3$, $\boldsymbol{\Gamma} \in \mathcal{C}_4$
For $k = 1, 2, \dots, T$ do:    (▷ For $\alpha^+$ see B.1 and B.1)
    (Update $\mathbf{W}$)
    Update activation $a_1, \dots, a_n$ and $\mathbf{K}$
    $\nabla_{\mathbf{W}} f(\mathbf{Z}) \leftarrow \mathbf{X}\mathbf{K}^T\boldsymbol{\beta}^T + 2\xi(\mathbf{W}\mathbf{H} - \mathbf{X})\mathbf{H}^T$
    Choose $\eta_{k,1}^{-1} > L_1 := \alpha^+\|\boldsymbol{\beta}\|_2^2 \cdot \|\mathbf{X}\|_2^2 + 2\xi\|\mathbf{H}\|_2^2$
    $\mathbf{W} \leftarrow \Pi_{\mathcal{C}_1}(\mathbf{W} - \eta_{k;1}\nabla_{\mathbf{W}} f(\mathbf{Z}))$
    (Update $\mathbf{H}$)
    $\nabla_{\mathbf{H}} f(\mathbf{Z}) \leftarrow 2\xi\mathbf{W}^T(\mathbf{W}\mathbf{H} - \mathbf{X})$
    Choose $\eta_{k,2}^{-1} > L_2 := 2\xi\|\mathbf{W}\|_2^2$
    $\mathbf{H} \leftarrow \Pi_{\mathcal{C}_2}(\mathbf{H} - \eta_{k;2}\nabla_{\mathbf{H}} f(\mathbf{Z}))$
    (Update $\boldsymbol{\beta}$)
    Update activation $a_1, \dots, a_n$ and $\mathbf{K}$
    $\nabla_{\boldsymbol{\beta}} f(\mathbf{Z}) \leftarrow \mathbf{W}^T\mathbf{X}\mathbf{K}^T$
    Choose $\eta_{k,3}^{-1} > L_3 := \alpha^+\|\mathbf{W}\|_2^2 \cdot \|\mathbf{X}\|_2^2$
    $\boldsymbol{\beta} \leftarrow \Pi_{\mathcal{C}_3}(\boldsymbol{\beta} - \eta_{k;3}\nabla_{\boldsymbol{\beta}} f(\mathbf{Z}))$
    (Update $\boldsymbol{\Gamma}$)
    Update activation $a_1, \dots, a_n$ and $\mathbf{K}$
    $\nabla_{\boldsymbol{\Gamma}} f(\mathbf{Z}) \leftarrow \mathbf{X}_{\text{aux}}\mathbf{K}^T$
    Choose $\eta_{k,4}^{-1} > L_4 := \alpha^+\|\mathbf{X}_{\text{aux}}\|_2^2$
    $\boldsymbol{\Gamma} \leftarrow \Pi_{\mathcal{C}_4}(\boldsymbol{\Gamma} - \eta_{k;4}\nabla_{\boldsymbol{\Gamma}} f(\mathbf{Z}))$
End for
Output: $\mathbf{Z} = (\mathbf{W}, \mathbf{H}, \boldsymbol{\beta}, \boldsymbol{\Gamma})$
```

Our algorithm, outlined in Algorithm 1, iteratively performs BCD on the four blocks with an adaptively chosen step-size. For its statement, note that κ takes any integer value above 1, with $\kappa = 1$ for binary labels. Denote $\mathbf{K} := [\dot{h}(y_1, a_1), \dots, \dot{h}(y_n, a_n)] \in \mathbb{R}^{1\times n}$ where

$$\nabla_a \ell(y, a) =: \dot{h}(y, a) = \frac{\exp(a)}{(1 + \exp(a))^2} \in \mathbb{R}.$$

This matrix appears in the gradient of the SMF objective f. In most of the experiments in this paper, we choose the convex constraint sets to be $\mathcal{C}_1 = \{\mathbf{W} \in \mathbb{R}_{\geq 0}^{p\times r} \,|\, \|\mathbf{W}\|_F \leq 1\}$, $\mathcal{C}_2 = \{\mathbf{H} \in \mathbb{R}_{\geq 0}^{r\times n} \,|\, \|\mathbf{H}\|_F \leq C_1\}$, $\mathcal{C}_3 = \{\boldsymbol{\beta} \in \mathbb{R}^{r\times\kappa} \,|\, \|\boldsymbol{\beta}\|_F \leq C_2\}$, and $\mathcal{C}_4 = \{\boldsymbol{\Gamma} \in \mathbb{R}^{q\times\kappa} \,|\, \|\boldsymbol{\Gamma}\|_F \leq C_3\}$, where $C_1, C_2, C_3 > 0$ are fixed constants.

Here are some remarks on the computational complexity of the algorithms. In Algorithm 1, the per-iteration cost is proportional to the cost of computing gradients for each block variable in the objective (e.g., $\mathbf{W}, \mathbf{H}, \boldsymbol{\beta}, \boldsymbol{\Gamma}$), which is $O((pr + q)n)$ for both SMF-$\mathbf{W}$ and SMF-$\mathbf{H}$. While they have the same asymptotic order, computing gradients for SMF-$\mathbf{W}$ are constant factors more expensive than that for SMF-$\mathbf{H}$, which can be seen by comparing the gradient formulas. Namely, SMF-$\mathbf{W}$ computes the additional $\mathbf{X}\mathbf{K}^T\boldsymbol{\beta}^T$ for the gradient of $\mathbf{W}$, and the gradient of $\boldsymbol{\beta}$ uses more expensive matrix multiplication $\mathbf{W}^T\mathbf{X}\mathbf{K}^T$ of complexity $O(rpn\kappa)$. In contrast, SMF-$\mathbf{H}$ employs $\mathbf{H}\mathbf{K}^T$ for its gradient or smaller order $O(rn\kappa)$, independent of p.

Using BCD instead of full *gradient descent* (GD) allows for larger step sizes, which has the potential for fast convergence. Namely, the allowed step size for each block in Algorithm 1 is determined by the reciprocal of the largest eigenvalue of the diagonal blocks of the Hessian (59) (see Theorem 4.3). In contrast, with GD, the step size is limited to the reciprocal of the largest eigenvalue of the entire Hessian, which may be considerably smaller.

3.3. Neural implementation of SMF-W for GPU acceleration

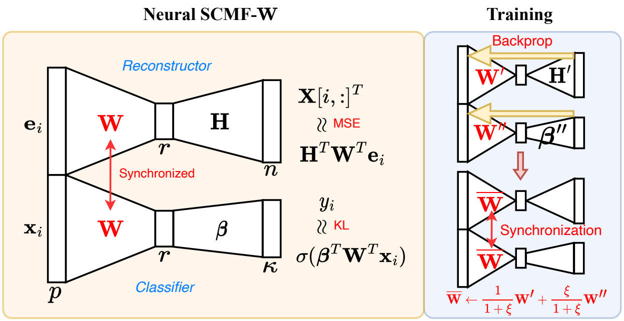


Figure 2. The SMF-$\mathbf{W}$ implementation involves two coupled two-layer neural networks: reconstructor and classifier. These networks share the first layer weight $\mathbf{W}$. The training process consists of repeating backpropagation in each network and subsequently synchronizing their first-layer weights through their convex combination. This configuration allows for extremely fast training on GPU. n data points are the columns of $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_n] \in \mathbb{R}^{p\times n}$ and $\mathbf{e}_i$ is the ith standard basis vector in $\mathbb{R}^p$.

While our BCD algorithm for SMF is derived from a careful local landscape analysis with rigorous theoretical guarantee, we provide a neural network architecture (see Figure. 2) that approximately implements our BCD algorithm in order to bring the advantage of a modern GPU computation to the practitioners in the ML community.

Reconstructor network: The reconstructor network operates as a two-layer neural network with weights $\mathbf{W} \in \mathbb{R}^{p\times r}$ and $\mathbf{H} \in \mathbb{R}^{r\times n}$ with identity activation to expedite matrix factorization. Each input vector $\mathbf{e}_i$ for this network is the ith standard basis vector in $\mathbb{R}^p$. Each p-dimensional input is transformed into an r-dimensional vector $\mathbf{W}^T\mathbf{e}_i$, which is then transformed to an n-dimensional vector $\mathbf{H}^T\mathbf{W}^T\mathbf{e}_i$. The target output is the ith row of the data matrix, $\mathbf{X}[i,:]^T \in \mathbb{R}^n$. Using *mean-squared error* (MSE) loss for this network, the overall loss is for this network is exactly $\frac{1}{n}\|\mathbf{X}-\mathbf{W}\mathbf{H}\|_F^2$.

Classifier network: The classifier network serves for both dimension reduction and classification within a neural network framework. Each input vector $\mathbf{x}_i = \mathbf{X}[i,:]^T$ has p dimensions, where i ranges from 1 to n. The network uses weight matrices $\mathbf{W} \in \mathbb{R}^{p\times r}$ for dimension reduction and $\boldsymbol{\beta} \in \mathbb{R}^r$ to compress each p-dimensional input $\mathbf{x}_i$ to an r-dimensional vector $\mathbf{W}^T\mathbf{x}_i$. The second layer with weight $\boldsymbol{\beta} \in \mathbb{R}^{r\times\kappa}$ and sigmoid activation σ yields the predicted probability distribution $\sigma(\boldsymbol{\beta}^T\mathbf{W}^T x_i)$ for the output $y_i \in \{0, 1, \dots, \kappa\}$. For this layer we use the cross-entropy loss for back-propagation.

Synchronizing the first-layer weight: The novel feature of our neural implementation of SMF is that we synchronize the the first-layer weight $\mathbf{W}$ after every step of back-propagation. Note that given the current first-layer weight $\mathbf{W}$, back-propagation within the reconstructor and the classifier networks updates $\mathbf{W}$ separately to two versions $\mathbf{W}'$ and $\mathbf{W}''$, respectively. The synchronization step takes a convex combination of these two versions as $\overline{\mathbf{W}} \leftarrow \frac{1}{1+\xi}\mathbf{W}' + \frac{\xi}{1+\xi}\mathbf{W}''$, which agrees with updating $\mathbf{W}$ by a gradient descent with $\nabla_{\mathbf{W}} f(\mathbf{W}, \mathbf{H}, \boldsymbol{\beta})$ for f the SMF-$\mathbf{W}$ loss in (3). We can then replace $\overline{W}$ with $\max\{O, \overline{\mathbf{W}}\}$ to ensure nonnegativity.

4. Statement of results

4.1. Assumptions

We introduce two minor assumptions below.

Assumption 4.1. (Constraint sets) The constraint sets $\mathcal{C}_1, \dots, \mathcal{C}_4$ in (3) are closed, convex, and compact.

Assumption 4.2. (Bounded activation) The activation $a \in \mathbb{R}^\kappa$ defined in (2) assumes bounded norm, i.e., $\|a\| \le M$ for some constant $M \in (0, \infty)$. (*c.f. Note that $\kappa = 1$ in the main text but we discuss the multi-label case $\kappa \ge 1$ in the appendix, see Sec. B.*)

Assumption 4.1 allows one to constrain each factor within a compact and convex set. A typical choice would be bounded nonnegative orthant, which entails supervised nonnegative matrix factorization models (Austin et al., 2018; Leuschner et al., 2019). It does not, however, entail supervised PCA models (Ritchie et al., 2020) or low-rank matrix constraints as the Grassmannian constraint is non-convex.

Assumption 4.2 imposes a constraint on the norm of the activation $\mathbf{a}$, as the input for the classification model in (3) is bounded. This is standard in the literature (see, e.g., (Negahban & Wainwright, 2011; Yaskov, 2016; Lecué & Mendelson, 2017; Lee et al., 2023)) to uniformly bound the eigenvalues of the Hessian of the multinomial logistic regression model.

Under Assumption 4.2, we introduce the following constants:

$$\begin{aligned}
\gamma_{\max} &:= 1 + \frac{e^M}{1 + e^M + (\kappa-1)e^{-M}} \le 2 && (9)\\
\alpha^- &:= \frac{e^{-M}}{1 + e^{-M} + (\kappa-1)e^{M}} \\
\alpha^+ &:= \frac{e^M\left(1 + 2(\kappa-1)e^M\right)}{\left(1 + e^M + (\kappa-1)e^{-M}\right)^2} \le 1/4.
\end{aligned}$$

These constants will appear in uniform bounds on the first and the second derivatives of the log likelihood $\ell(y, a)$ and the first derivative of the predictive probability distribution (see (Böhning, 1992)).

4.2. How does the local landscape look like?

In Theorem 4.3, we provide a local landscape result for SMF-$\mathbf{W}$. A key step is to compute the Hessian of the objective f in (3), which turns out to take the following 4×4 block form:

$$\begin{array}{c} \\ \mathrm{vec}(\mathbf{W}) \\ \mathrm{vec}(\mathbf{H}) \\ \mathrm{vec}(\boldsymbol{\beta}) \\ \mathrm{vec}(\boldsymbol{\Gamma}) \end{array}
\begin{array}{c} \begin{array}{cccc} \mathrm{vec}(\mathbf{W})^T & \mathrm{vec}(\mathbf{H})^T & \mathrm{vec}(\boldsymbol{\beta})^T & \mathrm{vec}(\boldsymbol{\Gamma})^T \end{array} \\
\left[\begin{array}{cccc}
A_{11} & A_{12} & A_{13} & \mathbf{O} \\
A_{21} & A_{22} & \mathbf{O} & \mathbf{O} \\
A_{31} & \mathbf{O} & A_{33} & A_{34} \\
\mathbf{O} & \mathbf{O} & A_{43} & A_{44}
\end{array}\right] \end{array} \qquad (10)$$

The exact formulas for each block entry are given in Lemma C.2. For our analysis, we consider the following L_2-regularized objective $F(\mathbf{Z})$ defined by

$$f(\mathbf{Z}) + \frac{\lambda_1}{2}\|\mathbf{W}\|_F^2 + \frac{\lambda_2}{2}\|\mathbf{H}\|_F^2 + \frac{\lambda_3}{2}\|\boldsymbol{\beta}\|_F^2 + \frac{\lambda_4}{2}\|\boldsymbol{\Gamma}\|_F^2 \qquad (11)$$

Also denote

$$\begin{aligned}
\Lambda_1 &:= \lambda_{\min}(\mathbf{H}\mathbf{H}^T) - \|\mathbf{W}\|_2\|\mathbf{H}\|_2 - \|\mathbf{W}\mathbf{H} - \mathbf{X}\|_2, && (12)\\
\Lambda_2 &:= \lambda_{\min}(\mathbf{W}^T\mathbf{W}) - \|\mathbf{W}\|_2\|\mathbf{H}\|_2 - \|\mathbf{W}\mathbf{H} - \mathbf{X}\|_2.
\end{aligned}$$

Theorem 4.3 (Local landscape of SMF-$\mathbf{W}$). *Let $f(\mathbf{Z})$ denote the objective of SMF-$\mathbf{W}$ in (3). Suppose Assumptions 4.1 and 4.2 hold. Then the followings hold:*

(i)

$$\begin{aligned}
A_{11} &\asymp \alpha^{\pm}(\boldsymbol{\beta}\boldsymbol{\beta}^T \otimes \mathbf{X}\mathbf{X}^T) + 2\xi(\mathbf{H}\mathbf{H}^T \otimes \mathbf{I}_p),\\
A_{22} &= 2\xi(\mathbf{I}_n \otimes \mathbf{W}^T\mathbf{W}),\\
A_{33} &\asymp \alpha^{\pm}(\mathbf{I}_\kappa \otimes \mathbf{W}^T\mathbf{X}\mathbf{X}^T\mathbf{W}),\\
A_{44} &\asymp \alpha^{\pm}(\mathbf{I}_\kappa \otimes \mathbf{X}_{\mathrm{aux}}\mathbf{X}_{\mathrm{aux}}^T).
\end{aligned}$$

(ii) *F is ρ-strongly convex at $\mathbf{Z} = (\mathbf{W}, \mathbf{H}, \boldsymbol{\beta}, \boldsymbol{\Gamma})$ for $\rho = \min_{1\le i\le 4}(\lambda_i - \lambda_i^*)$ where*

$$\begin{aligned}\lambda_1^* &:= \gamma_{\max}\sqrt{\kappa n}\|\mathbf{X}\|_2 + \alpha^+\|\boldsymbol{\beta}\|_2\|\mathbf{W}\|_2\|\mathbf{X}\|_2^2\\ &\qquad - 2\xi\Lambda_1 - \alpha^-\lambda_{\min}(\boldsymbol{\beta}\boldsymbol{\beta}^T)\lambda_{\min}(\mathbf{X}\mathbf{X}^T),\\ \lambda_2^* &:= -2\xi\Lambda_2,\\ \lambda_3^* &:= \gamma_{\max}\sqrt{\kappa n}\|\mathbf{X}\|_2 + +\alpha^+\|\boldsymbol{\beta}\|_2\|\mathbf{W}\|_2\|\mathbf{X}\|_2^2\\ &\quad + \alpha^+\|\mathbf{X}_{\text{aux}}\|_2\|\mathbf{W}^T\mathbf{X}\|_2 - \alpha^-\lambda_{\min}(\mathbf{W}^T\mathbf{X}\mathbf{X}^T\mathbf{W}),\\ \lambda_4^* &:= \alpha^+\|\mathbf{X}_{\text{aux}}\|_2\|\mathbf{W}^T\mathbf{X}\|_2 - \alpha^-\lambda_{\min}(\mathbf{X}_{\text{aux}}\mathbf{X}_{\text{aux}}^T).\end{aligned}$$

(iii) *Suppose $\mathbf{Z}_\star = [\mathbf{W}_\star, \mathbf{H}_\star, \boldsymbol{\beta}_\star, \boldsymbol{\Gamma}_\star]$ is a stationary point of f over $\boldsymbol{\Theta}$. If $\Lambda_1 > 0$, $\xi \gg 1$, and $\lambda_1 = 0$, then F is locally minimized at $(\mathbf{W}_\star, \theta')$ with the following perturbation bound:*

$$\|\theta' - \theta_\star\|_F \le \frac{3\max_{1\le i\le 4}(\lambda_i)}{\min_{1\le i\le 4}(\lambda_i - \lambda_{i\star})}\|\theta_\star\|_F, \qquad (13)$$

where $\theta' := (\mathbf{H}', \boldsymbol{\beta}', \boldsymbol{\Gamma}')$, $\theta_\star := (\mathbf{H}_\star, \boldsymbol{\beta}_\star, \boldsymbol{\Gamma}_\star)$ and $\|\theta_\star\|_F$ is assumed to be sufficiently small.

If $\Lambda_2 > 0$, then by taking $\lambda_2 = 0$ and denoting $\theta' := (\mathbf{W}', \boldsymbol{\beta}', \boldsymbol{\Gamma}')$ and $\theta_\star := (\mathbf{W}_\star, \boldsymbol{\beta}_\star, \boldsymbol{\Gamma}_\star)$, whenever $\|\theta_\star\|_F$ is sufficiently small, F is locally minimized at $(\mathbf{H}_\star, \theta')$ with the same perturbation bound in (13).

The interpretation of Theorem 4.3 **(iii)** aligns with our earlier discussion on the simpler MF case. Specifically, in the high-dimensional regime ($p \gg n$), it is likely that $\Lambda_2 = \Omega(rp) - O(r\sqrt{pn}) = \Omega(rp) > 0$. Consequently, we can introduce suitable L_2-regularization only to $\mathbf{W}, \boldsymbol{\beta}, \boldsymbol{\Gamma}$ so that the regularized landscape attains local minimization at the stationary point $\mathbf{H}$ with the other stationary factors perturbed. This implies that $\mathbf{H}_\star$ can be locally robustly estimated in this scenario. In the large-sample regime ($n \gg p$), it is likely that $\Lambda_1 = \Omega(rn) - O(r\sqrt{pn}) = \Omega(rn) > 0$. By choosing a sufficiently large tuning parameter ξ such that $\lambda_{1\star} \le 0$, we can use suitable L_2-regularization to $\mathbf{H}, \boldsymbol{\beta}, \boldsymbol{\Gamma}$. It ensures that the regularized landscape is locally minimized at the stationary point $\mathbf{W}_\star$ with the other stationary factors perturbed. Consequently, $\mathbf{W}_\star$ can be locally and robustly estimated in this scenario.

In Theorem C.6, we provide a similar local landscape result for SMF-$\mathbf{H}$. One notable difference is that, for SMF-$\mathbf{W}$, we require a large weight ξ on the matrix factorization loss in the large-sample regime, whereas, it should be used in the high-dimensional regime for SMF-$\mathbf{H}$.

Next, in Theorem 4.4 below, we establish the convergence of Algorithm 1 and 2 to the stationary points of the SMF objective f in (3). Furthermore, these algorithms converge to an 'ε-stationary point' solution within $\tilde{O}(\varepsilon^{-1})$ iterations. More precisely, consider the problem of minimizing a function $f : \mathbb{R}^p \to \mathbb{R}$ over a convex set $\boldsymbol{\Theta} \subset \mathbb{R}^p$. A $\boldsymbol{\theta}^* \in \boldsymbol{\Theta}$ is a *stationary point* of f over $\boldsymbol{\Theta}$ if $\inf_{\boldsymbol{\theta}\in\boldsymbol{\Theta}} \langle \nabla f(\boldsymbol{\theta}^*),\, \boldsymbol{\theta} - \boldsymbol{\theta}^*\rangle \ge 0$. This is equivalent to stating that $-\nabla f(\boldsymbol{\theta}^*)$ is in the normal cone of $\boldsymbol{\Theta}$ at $\boldsymbol{\theta}^*$. Every local minimum of f over $\boldsymbol{\Theta}$ is a stationary point. Relaxing this notion, for each $\varepsilon \ge 0$, we define $\boldsymbol{\theta}^* \in \boldsymbol{\Theta}$ to be an *ε-stationary point* of f over $\boldsymbol{\Theta}$ if

$$\text{Gap}(\boldsymbol{\theta}_\star) := \sup_{\boldsymbol{\theta}\in\boldsymbol{\Theta},\, \|\boldsymbol{\theta}-\boldsymbol{\theta}^*\|\le 1} \langle -\nabla f(\boldsymbol{\theta}^*),\, \boldsymbol{\theta} - \boldsymbol{\theta}^*\rangle \le \varepsilon. \quad (14)$$

Theorem 4.4 (Convergence rate of BCD)**.** *Suppose Assumptions 4.1 and 4.2 hold. Let $\mathbf{Z}_t = (\mathbf{W}_t, \mathbf{H}_t, \boldsymbol{\beta}_t, \boldsymbol{\Gamma}_t)$, $t \ge 1$ denote the sequence of estimated parameters from Algorithm 1 or 2. Then for every initial estimate $\mathbf{Z}_0$ and choice of parameters ξ, the followings hold:*

(i) $$\min_{1\le t\le T} \text{Gap}(\mathbf{Z}_t) = O(T^{-1/2}\log T). \qquad (15)$$

(ii) *For each $\varepsilon > 0$, an ε-stationary point is achieved within iteration $O(\varepsilon^{-1}(\log\varepsilon^{-1})^2)$.*

(iii) *Further assume that the step sizes $\eta_{k,i}$ are uniformly upper bounded. Then $\mathbf{Z}_t$ converges to the set of stationary points of f over $\boldsymbol{\Theta}$.*

Proofs of Theorems 4.3 and 4.4 are in Appendices C.

4.3. How close is an MLE to the true parameter?

We can extend Theorem 4.3 to provide a local estimation guarantee for generative SMF models that we introduce below. Fix parameters $\mathbf{W}_\star \in \mathbb{R}^{p\times r}$, $\mathbf{H}_\star \in \mathbb{R}^{r\times n}$, $\boldsymbol{\beta}_\star \in \mathbb{R}^{r\times\kappa}$, $\boldsymbol{\Gamma}_\star \in \mathbb{R}^{q\times\kappa}$, and $\boldsymbol{\lambda}_\star \in \mathbb{R}^{q\times 1}$. Suppose the data, auxiliary covariate, and label triples $(\mathbf{x}_i, \mathbf{x}'_i, y_i)$ are drawn independently (not necessarily identically distributed) according to the following generative model:

$$\begin{aligned}&\mathbf{x}_i \sim \text{N}\left(\mathbf{W}_\star\mathbf{H}_\star[:, i], \sigma^2\mathbf{I}_p\right),\ \ \mathbf{x}'_i \sim \text{N}(\boldsymbol{\lambda}_\star, (\sigma')^2\mathbf{I}_q),\\ &y_i \,|\, \mathbf{x}_i, \mathbf{x}'_i \sim \text{Bernoulli}\left(\frac{\exp(\mathbf{a}_i)}{1+\exp(\mathbf{a}_i)}\right) \qquad (16)\\ &\text{where} \quad \mathbf{a}_i := (\boldsymbol{\beta}_\star)^T(\mathbf{W}_\star)^T\mathbf{x}_i + (\boldsymbol{\Gamma}_\star)^T\mathbf{x}_i.\end{aligned}$$

For consistent estimation, we further assume that the mean r-dimensional representation $\mathbf{H}_\star[:, i]$ of the ith data column $\mathbf{x}_i$ is an $1/\sqrt{n}$-perturbation of a 'true mean vector' $\mathbf{h}_\star \in \mathbb{R}^r$: $\|\mathbf{H}_\star[:, i] - \mathbf{h}_\star\|_F \le c/\sqrt{n}$ for some constant $c > 0$. (c.f. When $\kappa \ge 1$, the conditional distribution of y_i in (16) is taken to be the multinomial distribution with probability of label c being proportional to $h(\mathbf{a}_i[c])$ with h general score function. See Appendix B.)

We assume $(\mathbf{x}_i, \mathbf{x}'_i, y_i)$ for $i = 1, \dots, n$ are independent, and also $\mathbf{x}_i$ and $\mathbf{x}'_i$ are independent for each $1 \le i \le n$. We refer to the above as the *generative SMF-$\mathbf{W}$ model*. Assuming that σ and σ' are known, our goal is to estimate the true factors $\mathbf{W}_\star$, $\mathbf{h}_\star$, $\boldsymbol{\beta}_\star, \boldsymbol{\Gamma}_\star$, and $\boldsymbol{\lambda}_\star$ from an observed sample $(\mathbf{x}_i, \mathbf{x}'_i, y_i)$, $i = 1, \dots, n$ of size n, where n is large and fixed. We consider the maximum likelihood estimation framework with L_2-regularization of the parameters. Namely, denote $\mathbf{Z} := (\mathbf{W}, \mathbf{h}, \boldsymbol{\beta}, \boldsymbol{\Gamma})$, $\mathbf{X} := [\mathbf{x}_1, \dots, \mathbf{x}_n]$, and $\mathbf{X}_{\text{aux}} := [\mathbf{x}'_1, \dots, \mathbf{x}'_n]$. Then the negative log-likelihood of observing the given data with an additional L_2-regularization is (up to a constant), letting F is as in (11),

$$\mathcal{L}(\mathbf{Z}, \boldsymbol{\lambda}) := F(\mathbf{Z}) + \frac{1}{2(\sigma')^2} \sum_{i=1}^{n} \|\mathbf{x}'_i - \boldsymbol{\lambda}\|^2. \tag{17}$$

The added L_2-regularizer in F can be understood by using a Gaussian prior for the parameters and interpreting the above as the negative logarithm of the posterior distribution.

Let $\bar{\mathcal{L}}(\mathbf{Z}, \boldsymbol{\lambda}) := \mathbb{E}\left[\mathcal{L}(\mathbf{Z}, \boldsymbol{\lambda})\right]$ denote the expected regularized negative log-likelihood function. In classical local consistency theory of MLE (e.g., (Fan & Li, 2001)), it is crucial that $\bar{\mathcal{L}}$ with zero L_2-regularization is strongly convex at the true parameter. Equivalently, this means that *Fisher information*, which is the Hessian $\nabla^2 \bar{\mathcal{L}}$ of the expected negative log-likelihood function (with no L_2-regularizer) evaluated at the true parameter, is positive definite. However, this is not the case for the generative SMF-$\mathbf{W}$ model in (16) (e.g., the model parameter in (16) is not identifiable), unless we add suitable L_2 regularization. Our key observation in Theorem 4.3 was that, in the large-sample or high-dimensional setting, such L_2-regularization is unnecessary for $\mathbf{W}$ or $\mathbf{H}$, respectively. We extend this to the statistical setting to obtain local consistency of the MLEs. The following result can be regarded as a high-probability $(1/\sqrt{n})$-perturbation of the local landscape result in Theorem 4.3.

Theorem 4.5. *(Regularized local consistency) Consider the generative SMF-$\mathbf{W}$ model (16). Assume that Assumptions 4.1 and 4.2 hold. Suppose $\rho := \min_{1 \le i \le 4}(\lambda_i - \lambda_{i\star}) > 0$.*

Suppose $\Lambda_1 > 0$, $\lambda_1 = 0$, and $\sigma \ll 1$ (resp., $\Lambda_2 > 0$ and λ_2=0). Fix $\varepsilon > 0$. Then there exists a constant $C > 0$ such that with probability at least $1 - \varepsilon$, $\mathcal{L}$ in (17) is minimized locally at some $(\hat{\mathbf{W}}, \hat{\theta}, \hat{\boldsymbol{\lambda}})$ (resp., $(\hat{\mathbf{H}}, \hat{\theta}, \hat{\boldsymbol{\lambda}})$) with

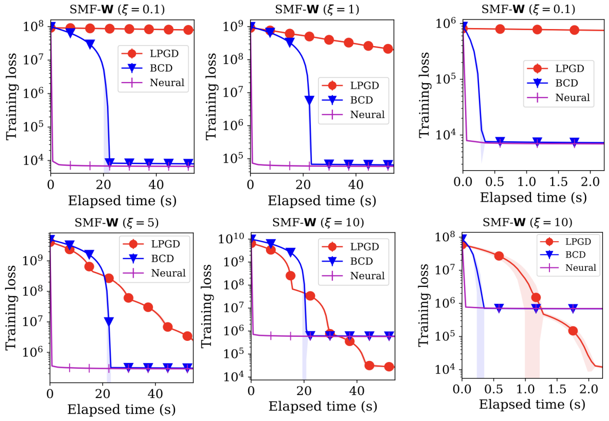
$$\|\hat{\mathbf{W}} - \mathbf{W}_\star\| \le C/\sqrt{n} \text{ (resp., } \|\hat{\mathbf{H}} - \mathbf{H}_\star\| \le C/\sqrt{n}) \tag{18}$$
$$\|\hat{\boldsymbol{\lambda}} - \boldsymbol{\lambda}_\star\| \le C/\sqrt{n}$$
$$\|\hat{\theta} - \theta_\star\|_F \le C n^{-1/2} \left(1 + \frac{3\max\{\lambda_2, \lambda_3, \lambda_4\}}{\rho} \|\theta_\star\|_F \right),$$

where $\theta' := (\mathbf{H}', \boldsymbol{\beta}', \boldsymbol{\Gamma}')$, $\theta_\star := (\mathbf{H}_\star, \boldsymbol{\beta}_\star, \boldsymbol{\Gamma}_\star)$ (resp., $\theta' := (\mathbf{W}', \boldsymbol{\beta}', \boldsymbol{\Gamma}')$, $\theta_\star := (\mathbf{W}_\star, \boldsymbol{\beta}_\star, \boldsymbol{\Gamma}_\star)$) and $\|\theta_\star\|_F$ is assumed to be sufficiently small.

Recall that in the generative SMF-$\mathbf{W}$ model (16), the Fisher information is a 5×5 block matrix with the first 4×4 block sub-matrix being the Hessian of the SMF objective f in (3) which is *not* positive definite. Hence the classical local consistency theory of MLE is not applicable. Our proof of Theorem 4.5 relies on Theorem 4.3, along with a substantial non-asymptotic generalization of such theory, which we establish Theorem D.1 in Section D. To prove this result, we use uniform McDirmid's inequality (Lemma D.2) and Berry-Esseen theorem for independent but non-identically distributed random variables (Thoerem D.3). See Appendix D for details.

5. Simulation and Applications

In Figure 3, we provide numerical verification of Theorem 4.4. The first dataset is generated from the MNIST database (LeCun & Cortes, 2010) ($p = 28^2 = 784$, $q = 0$, $n = 500$, $\kappa = 1$) for digit detection, and the second dataset is a text dataset named 'Employment Scam Aegean Dataset' (Laboratory of Information and Communication Systems, 2016) ($p = 2840, q = 72, n = 17880, \kappa = 1$) for fake job posting prediction. Details about these datasets are in Section G. We used Algorithms 1 and 2 with $r = 20$ for both datasets. We see sublinear convergence of both algorithms for various instances as stated in Theorem 4.4. Notably, algorithms for SMF-$\mathbf{H}$ (resp., SMF-$\mathbf{W}$) converge faster for large (resp., small) ξ. This is consistent with the implications of Theorems 4.3 and C.6. Also, our neural implementation (Figure 2) enjoys significant GPU acceleration, especially for large datasets.



a Job postings

b Semi-synthetic MNIST data

Figure 3. Plots of training loss vs. elapsed time at different ξ values for fitting SMF-$\mathbf{W}$ using Algorithm 1 (BCD), the neural implementation in Figure 2 (Neural), and low-rank projected gradient descent (LPGD) in (Lee et al., 2023). Shaded regions indicate one standard deviation across 10 runs.

In Figure 4, we evaluate the performance of different methods on the bi-objective tasks of SMF through a Pareto plot of F-score/Accuracy vs. relative reconstruction error. The baseline methods include logistic regression (LR) on raw data and NMF followed by logistic regression (MF-LR). Additionally, low-rank projected gradient descent algorithms for SMF (LPGD) in (Lee et al., 2023) are used. Increasing the tuning parameter ξ in the various SMF models seems to interpolate between two extremes of LR and MF-LR. Notably, SMF-$\mathbf{W}$ shows the best overall performance achieving both objectives.

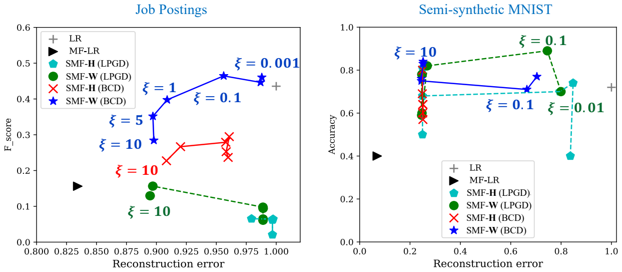


Figure 4. Pareto plots of relative reconstruction error vs. classification accuracy/F-score for different models.

Methods	Pancreatic	Breast
SMF-**W** (BCD)	0.869 (0.02)	**0.924 (0.01)**
SMF-**H** (BCD)	0.823 (0.06)	0.880 (0.02)
SMF-**W** (Neural)	0.854 (0.04)	0.881 (0.02)
SMF-**W** (LPGD)	0.869 (0.02)	0.894 (0.02)
SMF-**H** (LPGD)	**0.885 (0.07)**	0.875 (0.01)
PCA-LR	0.747 (0.13)	0.454 (0.27)
CNN	0.769 (0.07)	0.854 (0.06)
FFNN	0.816 (0.04)	0.890 (0.02)
Naive Bayes	0.815 (0.07)	0.810 (0.02)
SVM	0.746 (0.09)	0.866 (0.02)
Random Forest	0.815 (0.06)	0.844 (0.02)

Table 1. Cancer classification results using microarray data.

Lastly in Figure 5, we demonstrate supervised topic modeling with auxiliary covariates using SMF-**W** under nonnegative constraints. We compare SMF with the classic topic modeling approaches *Latent Dirichlet allocation* (LDA), NMF, and a recent deep learning-based approach, neural topic model with Gaussian Softmax distribution (GSM) (Miao et al., 2017).

Unsupervised topics mostly related to the true job postings, representing 95% of the dataset. Hence, applying unsupervised topic modeling methods is expected to learn topics that are mostly describing the true jobs postings, neglecting possible topics related to the scarce fake job postings. Indeed, our experiment shows that while traditional topic modeling methods successfully identify topics that describe the majority of job posting data, these topics may not be effective for classifying fake and true job postings. In contrast, our SMF with nonnegative constraints successfully ”tilts” the topics to faithfully represent the 5% fake job postings. This is why our method achieves the best classification performance in terms of the F-score.

Next, we apply the proposed methods to two datasets from the Curated Microarray Database (CuMiDa) (Feltes et al., 2019). CuMiDa provides well-preprocessed microarray data for various cancer types for various machine-learning approaches. One consists of 54,676 gene expressions from 51 subjects with binary labels indicating pancreatic cancer; Another we use has 35,982 gene expressions from 289 subjects with binary labels indicating breast cancer. The primary purpose of the analysis is to classify cancer patients solely based on their gene expression.

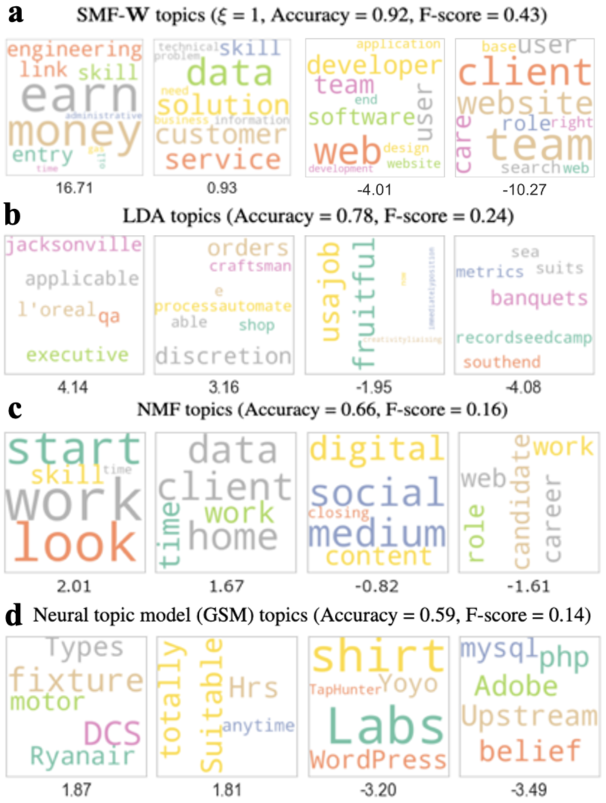


Figure 5. Topics in the job postings data learned by (**a**) SMF-**W** with $\xi = 1$, (**b**) latent Dirichlet allocation, (**c**) NMF, and (**d**) a neural topic model (GSM) in Miao et al. (2017). Without supervision, the learned topics are highly skewed toward the true job postings consisting of 95% of the data and lead to poor classification.

6. Conclusion and Limitations

This study contributes to the advancement of SMF, a classical machine learning method designed for simultaneous low-dimensional feature extraction and classification. Despite facing non-convex optimization challenges, we propose a BCD algorithm with adaptive step size, ensuring global convergence and providing iteration complexity guarantees. Minimum L_2-regularization enhances local strong convexity, and we explore parameter robustness within a statistical SMF model. Our GPU-friendly neural BCD implementation bridges theoretical insights with practical applicability, validated through numerical experiments for effectiveness. Our contributions enhance the understanding and application of SMF, addressing non-convexity and constraints in machine learning optimization.

Acknowledgements

HL is partially supported by NSF grants DMS-2206296 and DMS-2010035. WY is partially supported by NSF grant DMS-2210272.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none of which we feel must be specifically highlighted here.

References

Austin, W., Anderson, D., and Ghosh, J. Fully supervised non-negative matrix factorization for feature extraction. In *IGARSS 2018-2018 IEEE International Geoscience and Remote Sensing Symposium*, pp. 5772–5775. IEEE, 2018.

Berry, M. W. and Browne, M. Email surveillance using non-negative matrix factorization. *Computational & Mathematical Organization Theory*, 11(3):249–264, 2005.

Berry, M. W., Browne, M., Langville, A. N., Pauca, V. P., and Plemmons, R. J. Algorithms and applications for approximate nonnegative matrix factorization. *Computational statistics & data analysis*, 52(1):155–173, 2007.

Bjorck, J., Kabra, A., Weinberger, K. Q., and Gomes, C. Characterizing the loss landscape in non-negative matrix factorization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 6768–6776, 2021.

Böhning, D. Multinomial logistic regression algorithm. *Annals of the institute of Statistical Mathematics*, 44(1): 197–200, 1992.

Boutchko, R., Mitra, D., Baker, S. L., Jagust, W. J., and Gullberg, G. T. Clustering-initiated factor analysis application for tissue classification in dynamic brain positron emission tomography. *Journal of Cerebral Blood Flow & Metabolism*, 35(7):1104–1111, 2015.

Chen, Y., Wang, X., Shi, C., Lua, E. K., Fu, X., Deng, B., and Li, X. Phoenix: A weight-based network coordinate system using matrix factorization. *IEEE Transactions on Network and Service Management*, 8(4):334–347, 2011.

Elad, M. and Aharon, M. Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Transactions on Image processing*, 15(12):3736–3745, 2006.

Fan, J. and Li, R. Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American statistical Association*, 96(456):1348–1360, 2001.

Feingold, D. G. and Varga, R. S. Block diagonally dominant matrices and generalizations of the gerschgorin circle theorem. *Pacific Journal of Mathematics*, 12(4):1241–1250, 1962.

Feller, W. On the berry-esseen theorem. *Zeitschrift für Wahrscheinlichkeitstheorie und Verwandte Gebiete*, 10 (3):261–268, 1968.

Feltes, B. C., Chandelier, E. B., Grisci, B. I., and Dorn, M. Cumida: An extensively curated microarray database for benchmarking and testing of machine learning approaches in cancer research. *Journal of Computational Biology*, 26(4):376–386, 2019. doi: 10.1089/cmb.2018.0238. URL https://doi.org/10.1089/cmb.2018.0238. PMID: 30789283.

Grippo, L. and Sciandrone, M. On the convergence of the block nonlinear gauss–seidel method under convex constraints. *Operations research letters*, 26(3):127–136, 2000.

Kim, D., Park, C., Oh, J., Lee, S., and Yu, H. Convolutional matrix factorization for document context-aware recommendation. In *Proceedings of the 10th ACM conference on recommender systems*, pp. 233–240, 2016.

Laboratory of Information and Communication Systems. Emscad employment scam aegean dataset. http://icsdweb.aegean.gr/emscad, 2016. Accessed: 2023-05-16.

Lecué, G. and Mendelson, S. Sparse recovery under weak moment assumptions. *Journal of the European Mathematical Society*, 19(3):881–904, 2017.

LeCun, Y. and Cortes, C. MNIST handwritten digit database. 2010. URL http://yann.lecun.com/exdb/mnist/.

Lee, D. and Seung, H. S. Algorithms for non-negative matrix factorization. *Advances in neural information processing systems*, 13:556–562, 2000.

Lee, D. D. and Seung, H. S. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755): 788, 1999.

Lee, J., Lyu, H., and Yao, W. Exponentially convergent algorithms for supervised matrix factorization. *Neural Information Processing Systems*, 2023.

Leuschner, J., Schmidt, M., Fernsel, P., Lachmund, D., Boskamp, T., and Maass, P. Supervised non-negative matrix factorization methods for maldi imaging applications. *Bioinformatics*, 35(11):1940–1947, 2019.

Li, Z., Zhang, Z., Qin, J., Zhang, Z., and Shao, L. Discriminative fisher embedding dictionary learning algorithm for object recognition. *IEEE transactions on neural networks and learning systems*, 31(3):786–800, 2019.

Lyu, H. and Li, Y. Block majorization-minimization with diminishing radius for constrained nonconvex optimization. *arXiv preprint arXiv:2012.03503*, 2023.

Lyu, H., Kureh, Y. H., Vendrow, J., and Porter, M. A. Learning low-rank latent mesoscale structures in networks. *Nature Communications*, 15(1):224, 2024.

Mairal, J., Elad, M., and Sapiro, G. Sparse representation for color image restoration. *IEEE Transactions on Image Processing*, 17(1):53–69, 2007.

Mairal, J., Ponce, J., Sapiro, G., Zisserman, A., and Bach, F. Supervised dictionary learning. *Advances in Neural Information Processing Systems*, 21:1033–1040, 2008.

Mairal, J., Bach, F., and Ponce, J. Task-driven dictionary learning. *IEEE transactions on pattern analysis and machine intelligence*, 34(4):791–804, 2011.

Miao, Y., Grefenstette, E., and Blunsom, P. Discovering discrete latent topics with neural variational inference. In *International conference on machine learning*, pp. 2410–2419. PMLR, 2017.

Negahban, S. and Wainwright, M. J. Estimation of (near) low-rank matrices with noise and high-dimensional scaling. *The Annals of Statistics*, 39(2):1069–1097, 2011.

Nesterov, Y. Introductory lectures on convex programming volume i: Basic course. *Lecture notes*, 3(4):5, 1998.

Panageas, I., Skoulakis, S., Varvitsiotis, A., and Wang, X. Convergence to second-order stationarity for nonnegative matrix factorization: Provably and concurrently. *arXiv preprint arXiv:2002.11323*, 2020.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830, 2011.

Peyré, G. Sparse modeling of textures. *Journal of Mathematical Imaging and Vision*, 34(1):17–31, 2009.

Ren, B., Pueyo, L., Zhu, G. B., Debes, J., and Duchêne, G. Non-negative matrix factorization: robust extraction of extended structures. *The Astrophysical Journal*, 852(2): 104, 2018.

Ritchie, A., Balzano, L., Kessler, D., Sripada, C. S., and Scott, C. Supervised pca: A multiobjective approach. *arXiv preprint arXiv:2011.05309*, 2020.

Sitek, A., Gullberg, G. T., and Huesman, R. H. Correction for ambiguous solutions in factor analysis using a penalized least squares objective. *IEEE transactions on medical imaging*, 21(3):216–225, 2002.

Taslaman, L. and Nilsson, B. A framework for regularized non-negative matrix factorization, with application to the analysis of gene expression data. *PloS one*, 7(11):e46331, 2012.

Vershynin, R. *High-dimensional probability: An introduction with applications in data science*, volume 47. Cambridge university press, 2018.

Wright, S. J. Coordinate descent algorithms. *Mathematical Programming*, 151(1):3–34, 2015.

Xu, Y. and Yin, W. A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion. *SIAM Journal on imaging sciences*, 6(3):1758–1789, 2013.

Yankelevsky, Y. and Elad, M. Structure-aware classification using supervised dictionary learning. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4421–4425. IEEE, 2017.

Yaskov, P. Controlling the least eigenvalue of a random gram matrix. *Linear Algebra and its Applications*, 504: 108–123, 2016.

Zhao, S., Han, J., Lv, J., Jiang, X., Hu, X., Zhao, Y., Ge, B., Guo, L., and Liu, T. Supervised dictionary learning for inferring concurrent brain networks. *IEEE transactions on medical imaging*, 34(10):2036–2045, 2015.

Supervised Matrix Factorization: Local Landscape Analysis and Applications
Supplementary Material

A. Preliminaries

This section covers key notations and fundamental concepts of linear algebra and matrix calculus.

If $A = (a_{ij})$ is an $m \times n$ matrix and B is a $p \times q$ matrix, then the Kronecker product $A \otimes B$ is the $mp \times nq$ matrix such that

$$A \otimes B = \begin{bmatrix} a_{11}B & \cdots & a_{1n}B \\ \vdots & \ddots & \vdots \\ a_{m1}B & \cdots & a_{mn}B \end{bmatrix}.$$

Recall that we have

$$(A \otimes B)^T = A^T \otimes B^T.$$

If A' is an $m \times n'$ matrix and B' is a $p' \times q$ matrix, we define the $m \times (n + n')$ horizontally stacked matrix as $[A, A']$ and the the $(p + p') \times n$ vertically stacked matrix as $[B \| B'] := [B^T, (B')^T]^T$. Then by properties of the Kronecker product, we have

$$[A, A'] \otimes B = [A \otimes B, A' \otimes B], \qquad A \otimes [B \| B'] = [A \otimes B \parallel A \otimes B']. \tag{19}$$

For each $m \times n$ matrix $A = [a_1, \dots, a_n]$, we define its vectorization as $\operatorname{vec}(A) = [a_1^T, \dots, a_n^T]^T \in \mathbb{R}^{mn}$.

The *commutation matrix* $\mathbf{C}^{(a,b)}$ is the $ab \times ab$ matrix such that

$$\mathbf{C}^{(a,b)} \operatorname{vec}(A) = \operatorname{vec}(A^T),$$

for any $a \times b$ matrix A. For each pair of integers $a, b \geq 1$, there is a unique matrix $\mathbf{C}^{(a,b)} \in \{0,1\}^{ab \times ab}$. Recall the following properties of the commutation matrix:

- $(\mathbf{C}^{(a,b)})^T = \mathbf{C}^{(b,a)}$.
- $(\mathbf{C}^{(a,b)})^T \mathbf{C}^{(a,b)} = \mathbf{I}_{ab},$ that is, $\mathbf{C}^{(a,b)}$ is positive semi-definite.
- $\mathbf{C}^{(a,1)} = \mathbf{I}_a = \mathbf{C}^{1,a}$.
- $\mathbf{C}^{(p,m)}(A \otimes B) = (B \otimes A)\mathbf{C}^{(q,n)}$ for every $m \times n$ matrix A and $p \times q$ matrix B (20)

. • $(A \otimes B)(C \otimes D) = (AC) \otimes (BD)$ for any matrices with compatible sizes for the products AC and BD. (21)

Furthermore, for any matrices $A \in \mathbb{R}^{a \times b}$, $B \in \mathbb{R}^{b \times c}$, and $C \in \mathbb{R}^{c \times d}$, the vectorizing product of matrices is given by

$$\operatorname{vec}(AB) = (\mathbf{I}_c \otimes A) \operatorname{vec}(B) = (B^T \otimes \mathbf{I}_a) \operatorname{vec}(A), \tag{22}$$

$$\operatorname{vec}(ABC) = (C^T \otimes A) \operatorname{vec}(B) = (\mathbf{I}_d \otimes AB) \operatorname{vec}(C) = (C^T B^T \otimes \mathbf{I}_a) \operatorname{vec}(A). \tag{23}$$

Next, for differentiable functions $f : \mathbb{R}^{a \times 1} \to \mathbb{R}^{b \times 1}$ and $g : \mathbb{R}^{b \times 1} \to \mathbb{R}^{c \times 1}$, the Jacobian J_f can be represented as

$$\begin{aligned} J_f(x) &= \left(\nabla_x f(x)^T\right)^T \\ \nabla_x \left(g(f(x))^T\right) &= \nabla_x \left(f(x)^T\right) \nabla_{f(x)} \left(g(f(x))^T\right), \end{aligned} \tag{24}$$

where the second equality holds by chain rule $J_{g \circ f}(x) = J_g(f(x)) J_f(x)$. And for any $a \times b$ matrix A, we have

$$\begin{aligned} \nabla_{\operatorname{vec}(A)} \operatorname{vec}(A)^T &= \mathbf{I}_{ab}, \\ \nabla_{\operatorname{vec}(A)} \operatorname{vec}(A^T)^T &= \nabla_{\operatorname{vec}(A)} \operatorname{vec}(A)^T \mathbf{C}^{(b,a)} = \mathbf{C}^{(b,a)}. \end{aligned}$$

B. Model formulation for general multi-label setting

Consider the following problem setting: we have a set of n observations $(y_i, \mathbf{x}_i, \mathbf{x}'_i)$ for $i = 1, \dots, n$ where $y_i \in \{0, 1, \dots, \kappa\}$ represents an observed label, $\mathbf{x}_i \in \mathbb{R}^p$ denotes a high-dimensional feature, and $\mathbf{x}'_i \in \mathbb{R}^q$ is a low-dimensional auxiliary feature for the i-th individual ($p \gg q$). To predict y_i, a low-dimensional representation of $\mathbf{x}_i$ in dimension $r \ll p$ for some suitable r may be utilized, combined with $\mathbf{x}'_i$. This implies that the observed $\mathbf{x}_i$ is approximated by a linear transformation of the *basis* vectors $\mathbf{w}_1, \dots, \mathbf{w}_r \in \mathbb{R}^p$ using a suitable code $\mathbf{h}_i$. Let $\mathbf{W} = [\mathbf{w}_1, \dots, \mathbf{w}_r] \in \mathbb{R}^{p \times r}$ be referred to as the *(latent) factor matrix*, and $\mathbf{H} = [\mathbf{h}_1, \dots, \mathbf{h}_n] \in \mathbb{R}^{r \times n}$ as its *code matrix*. In a more compact form, $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_n] \approx \mathbf{W}\mathbf{H}$, known as *reconstruction*. In practical terms, we can determine r as the approximate rank of the data matrix $\mathbf{X}$.

Now, we present our probabilistic modeling assumption. Consider fixed parameters $\mathbf{W} \in \mathbb{R}^{p \times r}$, $\mathbf{h}_i \in \mathbb{R}^r$, $\boldsymbol{\beta} \in \mathbb{R}^{r \times \kappa}$, and $\boldsymbol{\gamma} \in \mathbb{R}^{q \times \kappa}$. Let $h : \mathbb{R} \to [0, \infty)$ be a score function. Suppose y_i is a realization of a random variable whose conditional distribution is defined as

$$[\mathbb{P}(y_i = 0 \,|\, \mathbf{x}_i, \mathbf{x}'_i), \dots, \mathbb{P}(y_i = \kappa \,|\, \mathbf{x}_i, \mathbf{x}'_i)] := C[1, h(a_{i,1}), \dots, h(a_{i,\kappa})], \tag{25}$$

where C is the normalization constant and $\mathbf{a}_i = (a_{i,1}, \dots, a_{i,\kappa}) \in \mathbb{R}^\kappa$ is the activation for y_i. For multinomial logistic regression, we have

$$[\mathbb{P}(y_i = 0 \,|\, \mathbf{x}_i, \mathbf{x}'_i), \dots, \mathbb{P}(y_i = \kappa \,|\, \mathbf{x}_i, \mathbf{x}'_i)] = \frac{1}{1 + \sum_{c=1}^{\kappa} \exp(a_{i,c})}[1, \exp(a_{i,1}), \dots, \exp(a_{i,\kappa})],$$

where the score function $h(\cdot) = \exp(\cdot)$.

The activation is defined in two ways, depending on whether we use a 'feature-based' model (SMF-$\mathbf{H}$) or a 'filter-based' model (SMF-$\mathbf{W}$):

$$\mathbf{a}_i = \begin{cases} \boldsymbol{\beta}^T \mathbf{W}^T \mathbf{x}_i + \boldsymbol{\gamma}^T \mathbf{x}'_i & \text{for SMF-}\mathbf{W} \\ \boldsymbol{\beta}^T \mathbf{h}_i + \boldsymbol{\gamma}^T \mathbf{x}'_i & \text{for SMF-}\mathbf{H} \end{cases} \in \mathbb{R}^\kappa. \tag{26}$$

Here, $(\boldsymbol{\beta}, \boldsymbol{\gamma})$ are multinomial logistic regression coefficients associated with input features $(\mathbf{h}_i, \mathbf{x}'_i)$ or $(\mathbf{W}^T\mathbf{x}_i, \mathbf{x}'_i)$, respectively. In equation (26), the code $\mathbf{h}_i$ or the 'filtered feature' $\mathbf{W}^T\mathbf{x}_i$ is the low-dimensional representation of $\mathbf{x}_i$.

Let $\mathbf{Z} := (\mathbf{W}, \mathbf{H}, \boldsymbol{\beta}, \boldsymbol{\gamma})$ be our block parameters of interest. In order to estimate $\mathbf{Z}$ from observed data $(\mathbf{x}_i, \mathbf{x}'_i, y_i)$ for $i = 1, \dots, n$, we consider the following multi-objective non-convex *constrained* optimization problem:

$$\min_{\substack{\mathbf{W} \in \mathcal{C}_1, \mathbf{H} \in \mathcal{C}_2 \\ \boldsymbol{\beta} \in \mathcal{C}_3, \boldsymbol{\Gamma} \in \mathcal{C}_4}} f(\mathbf{Z}) := \sum_{i=1}^{n} \ell(y_i, \mathbf{a}_i) + \xi \|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2, \tag{27}$$

where $\mathcal{C}_j$ for $j = 1, \dots, 4$ represents convex constraint sets of each block parameter, $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_n] \in \mathbb{R}^{p \times n}$, $\mathbf{a}_i$ is as in (26), and $\ell(\cdot)$ is the classification loss defined as the negative log-likelihood:

$$\ell(y, \mathbf{a}) = \log\left(1 + \sum_{c=1}^{\kappa} h(a_c)\right) - \sum_{c=1}^{\kappa} \mathbf{1}_{\{y=c\}} \log h(a_c).$$

Note that the four block parameters are *individually* assumed to be constrained in (27). A tuning parameter ξ controls the trade-off between the dual objectives of classification and matrix factorization.

With the choice of general score function h in (25), we impose the following assumption on uniform bounds on the first and the second derivatives observed information and the first derivative of the predictive probability distribution (see (Böhning, 1992)).

Assumption B.1. (Bounded stiffness and eigenvalues of observed information) The score function $h : \mathbb{R} \to [0, \infty)$ is twice continuously differentiable. Further, let observed information $\ddot{\mathbf{H}}(y, \mathbf{a}) := \nabla_{\mathbf{a}} \nabla_{\mathbf{a}^T} \ell(y, \mathbf{a})$ for y and $\mathbf{a}$. Then, for the constant $M > 0$ in Assumption 4.2, there are constants $\gamma_{\max}, \alpha^-, \alpha^+ > 0$ s.t. $\gamma_{\max} := \sup_{\|\mathbf{a}\| \le M} \max_{1 \le s \le n} \|\nabla_{\mathbf{a}} \ell(y_s, \mathbf{a})\|_\infty$

$$\alpha^- := \inf_{\|\mathbf{a}\| \le M} \min_{1 \le s \le n} \lambda_{\min}(\ddot{\mathbf{H}}(y_s, \mathbf{a})), \tag{28}$$

$$\alpha^+ := \sup_{\|\mathbf{a}\| \le M} \max_{1 \le s \le n} \lambda_{\max}(\ddot{\mathbf{H}}(y_s, \mathbf{a})). \tag{29}$$

Under Assumption 4.2 and the multinomial logistic regression model $h(\cdot) = \exp(\cdot)$, the quantities $\gamma_{\max}$ and $\alpha^{\pm}$ in B.1 can be bounded as in (9) in the main text.

Remark B.2 (Multinomial Logistic Classifier). Let ℓ denote the negative log-likelihood function in (3), where we take the multinomial logistic model with the score function $h(\cdot) = \exp(\cdot)$. In this case Assumption B.1 is easily satisfied. To see this, denote $(\dot{h}_1, \dots, \dot{h}_\kappa) := \nabla_{\mathbf{a}} \ell(y, \mathbf{a})$ and $\ddot{\mathbf{H}}(y, \mathbf{a}) := \nabla_{\mathbf{a}} \nabla_{\mathbf{a}^T} \ell(y, \mathbf{a})$. Then in this special case, we have $\dot{h}_j(y, \mathbf{a}) = g_j(\mathbf{a}) - \mathbf{1}(y = j)$ and $\ddot{H}(y, \mathbf{a})_{i,j} = g_i(\mathbf{a})\,(\mathbf{1}(i = j) - g_j(\mathbf{a}))$ (See (167) and (169) in Appendix). Under Assumption 4.2, according to Lemma F.1, we can take $\gamma_{\max} = 1 + \frac{e^M}{1+e^M+(\kappa-1)e^{-M}} \le 2$, $\alpha^- = \frac{e^{-M}}{1+e^{-M}+(\kappa-1)e^M}$, and $\alpha^+ = \frac{e^M\left(1+2(\kappa-1)e^M\right)}{(1+e^M+(\kappa-1)e^{-M})^2}$. For binary classification, $\alpha^+ \le 1/4$.

C. Local landscape analysis for SMF

In this section, we prove Theorem 4.3 as well as Theorem 4.4 for the general multi-label setting we introduced in Section B.

Throughout this section, we denote $\mathbf{Z} = [\mathbf{W}, \mathbf{H}, \boldsymbol{\beta}, \boldsymbol{\Gamma}]$ for the combined SMF parameters. The activation $\mathbf{a}_s$ for the sth sample (see (2)) is given by

$$\mathbf{a}_s := \begin{cases} \boldsymbol{\beta}^T \mathbf{W}^T \mathbf{x}_s + \boldsymbol{\Gamma}^T \mathbf{x}'_s & \text{for SMF-}\mathbf{W} \\ \boldsymbol{\beta}^T \mathbf{h}_s + \boldsymbol{\Gamma}^T \mathbf{x}'_s & \text{for SMF-}\mathbf{H} \end{cases}.$$

Then the objective function in (3) in the general setting then can be written as

$$\begin{aligned} f(\mathbf{Z}) &= \sum_{s=1}^{n} \ell(y_s, \mathbf{a}_s) + \xi \|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2 && (30) \\ &= \sum_{s=1}^{n} \left(\log \sum_{c=0}^{\kappa} h(\mathbf{a}_s[c]) - \sum_{c=0}^{\kappa} \mathbf{1}_{\{y_s = c\}} \log h(\mathbf{a}_s[c]) \right) + \xi \|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2, && (31) \end{aligned}$$

where $\mathbf{a}_s[i] \in \mathbb{R}$ denotes the ith component of $\mathbf{a}_s \in \mathbb{R}^\kappa$ and $h(\mathbf{a}[0]) = 1$. Recall the functions $\dot{\mathbf{h}}$ and $\ddot{\mathbf{H}}$ introduced in Assumption B.1. An easy computation shows

$$\nabla_{\mathbf{a}} \ell(y, \mathbf{a}) =: \dot{\mathbf{h}}(y, \mathbf{a}) = (\dot{h}_1, \dots, \dot{h}_\kappa) \in \mathbb{R}^\kappa, \quad \nabla_{\mathbf{a}} \nabla_{\mathbf{a}^T} \ell(y, \mathbf{a}) = \ddot{\mathbf{H}}(y, \mathbf{a}) =: (\ddot{h}_{ij}) \in \mathbb{R}^{\kappa \times \kappa}, \tag{32}$$

where

$$\begin{aligned} \dot{h}_j = \dot{h}_j(y, \mathbf{a}) &:= \left(\frac{h'(a_j)}{1 + \sum_{c=1}^{\kappa} h(a_c)} - \mathbf{1}(y = j) \frac{h'(a_j)}{h(a_j)} \right), && (33) \\ \ddot{h}_{ij} &:= \left(\frac{h''(a_j) \mathbf{1}(i = j)}{1 + \sum_{c=1}^{\kappa} h(a_c)} - \frac{h'(a_i) h'(a_j)}{(1 + \sum_{c=1}^{\kappa} h(a_c))^2} \right) - \mathbf{1}(y = i = j) \left(\frac{h''(a_j)}{h(a_j)} - \frac{(h'(a_j))^2}{(h(a_j))^2} \right). && (34) \end{aligned}$$

For the forthcoming computations, define matrices

$$\mathbf{K} := [\dot{\mathbf{h}}(y_1, \mathbf{a}_1), \dots, \dot{\mathbf{h}}(y_n, \mathbf{a}_n)] \in \mathbb{R}^{\kappa \times n}, \qquad \mathbf{M} := \operatorname{diag}\left(\ddot{\mathbf{H}}(y_1, \mathbf{a}_1), \dots, \ddot{\mathbf{H}}(y_n, \mathbf{a}_n) \right) \in \mathbb{R}^{\kappa n \times \kappa n}. \tag{35}$$

C.1. Proof for SMF-W

In this section, we prove Theorem 4.4 for SMF-$\mathbf{W}$. An analogous argument for SMF-$\mathbf{H}$ will be provided in the next section.

Proposition C.1. *Let $f(\mathbf{Z})$ denote the objective of SMF-$\mathbf{W}$ in (30). Suppose Assumption B.1 holds. Let $\mathbf{a}_s := \boldsymbol{\beta}^T \mathbf{W}^T \mathbf{x}_s + \boldsymbol{\Gamma}^T \mathbf{x}'_s$ for $s = 1, \dots, n$. Then*

$$\begin{cases} \nabla_{\operatorname{vec}(\mathbf{W})} \ell(y_s, \mathbf{a}_s) &= \mathbf{C}^{(r,p)} (\mathbf{x}_s \otimes \boldsymbol{\beta})\, \dot{\mathbf{h}}(y_s, \mathbf{a}_s), \\ \nabla_{\operatorname{vec}(\boldsymbol{\beta})} \ell(y_s, \mathbf{a}_s) &= \mathbf{C}^{(\kappa,r)} (\mathbf{W}^T \mathbf{x}_s \otimes \mathbf{I}_\kappa)\, \dot{\mathbf{h}}(y_s, \mathbf{a}_s), \\ \nabla_{\operatorname{vec}(\boldsymbol{\Gamma})} \ell(y_s, \mathbf{a}_s) &= \mathbf{C}^{(k,q)} (\mathbf{x}'_s \otimes \mathbf{I}_\kappa)\, \dot{\mathbf{h}}(y_s, \mathbf{a}_s), \end{cases} \qquad \begin{cases} \nabla_{\operatorname{vec}(\mathbf{W})} \operatorname{vec}(\mathbf{K})^T &= (\boldsymbol{\beta} \otimes \mathbf{X}) \mathbf{C}^{(\kappa,n)} \mathbf{M}, \\ \nabla_{\operatorname{vec}(\boldsymbol{\beta})} \operatorname{vec}(\mathbf{K})^T &= (\mathbf{I}_\kappa \otimes \mathbf{W}^T \mathbf{X}) \mathbf{C}^{(\kappa,n)}\, \mathbf{M}, \\ \nabla_{\operatorname{vec}(\boldsymbol{\Gamma})} \operatorname{vec}(\mathbf{K})^T &= (\mathbf{I}_\kappa \otimes \mathbf{X}_{\text{aux}}) \mathbf{C}^{(\kappa,n)}\, \mathbf{M}. \end{cases} \tag{36}$$

Proof. We first show

$$\begin{cases} \nabla_{\mathrm{vec}(\mathbf{W})}\mathbf{a}_s^T &= \mathbf{C}^{(r,p)}(\mathbf{x}_s \otimes \boldsymbol{\beta}), \\ \nabla_{\mathrm{vec}(\mathbf{H})}\mathbf{a}_s^T &= \mathbf{O}, \\ \nabla_{\mathrm{vec}(\boldsymbol{\beta})}\mathbf{a}_s^T &= \mathbf{C}^{(\kappa,r)}(\mathbf{W}^T\mathbf{x}_s \otimes \mathbf{I}_\kappa), \\ \nabla_{\mathrm{vec}(\boldsymbol{\Gamma})}\mathbf{a}_s^T &= \mathbf{C}^{(k,q)}(\mathbf{x}'_s \otimes \mathbf{I}_\kappa). \end{cases} \tag{37}$$

$\nabla_{\mathrm{vec}(\mathbf{H})}\mathbf{a}_s^T = \mathbf{O}$ is clear. For differentiating $\mathbf{a}_s$ by $\mathrm{vec}(\mathbf{W})$, observe that by using (22), we can write

$$\mathbf{a}_s = \mathrm{vec}(\mathbf{a}_s) = \mathrm{vec}\left(\boldsymbol{\beta}^T\mathbf{W}^T\mathbf{x}_s + \boldsymbol{\Gamma}^T\mathbf{x}'_s\right) = (\mathbf{x}_s^T \otimes \boldsymbol{\beta}^T)\,\mathrm{vec}(\mathbf{W}^T) + \mathrm{vec}(\boldsymbol{\Gamma}^T\mathbf{x}'_s).$$

Noting that $\mathrm{vec}(\mathbf{W}^T)^T = (\mathbf{C}^{(p,r)}\,\mathrm{vec}(\mathbf{W}))^T = \mathrm{vec}(\mathbf{W})^T\mathbf{C}^{(r,p)}$,

$$\begin{aligned} \nabla_{\mathrm{vec}(\mathbf{W})}\mathbf{a}_s^T &= \nabla_{\mathrm{vec}(\mathbf{W})}\,\mathrm{vec}(\mathbf{W}^T)^T(\mathbf{x}_s \otimes \boldsymbol{\beta}) \\ &= \nabla_{\mathrm{vec}(\mathbf{W})}\,\mathrm{vec}(\mathbf{W})^T\mathbf{C}^{(r,p)}(\mathbf{x}_s \otimes \boldsymbol{\beta}) \\ &= \mathbf{C}^{(r,p)}(\mathbf{x}_s \otimes \boldsymbol{\beta}). \end{aligned}$$

For differentiating $\mathbf{a}_s$ by $\mathrm{vec}(\boldsymbol{\beta})$, writing $\mathbf{a}_s = (\mathbf{x}_s^T\mathbf{W} \otimes \mathbf{I}_\kappa)\,\mathrm{vec}(\boldsymbol{\beta}^T) + \mathrm{vec}(\boldsymbol{\Gamma}^T\mathbf{x}'_s)$, we get

$$\begin{aligned} \nabla_{\mathrm{vec}(\boldsymbol{\beta})}\mathbf{a}_s^T &= \nabla_{\mathrm{vec}(\boldsymbol{\beta})}\,\mathrm{vec}(\boldsymbol{\beta}^T)^T(\mathbf{W}^T\mathbf{x}_s \otimes \mathbf{I}_\kappa) \\ &= \nabla_{\mathrm{vec}(\boldsymbol{\beta})}\,\mathrm{vec}(\boldsymbol{\beta})^T\mathbf{C}^{(\kappa,r)}(\mathbf{W}^T\mathbf{x}_s \otimes \mathbf{I}_\kappa) \\ &= \mathbf{C}^{(\kappa,r)}(\mathbf{W}^T\mathbf{x}_s \otimes \mathbf{I}_\kappa). \end{aligned}$$

For differentiating $\mathbf{a}_s$ by $\mathrm{vec}(\boldsymbol{\Gamma})$, writing $\mathbf{a}_s = \mathrm{vec}(\boldsymbol{\beta}^T\mathbf{W}^T\mathbf{x}_s) + ((\mathbf{x}'_s)^T \otimes \mathbf{I}_\kappa)\,\mathrm{vec}(\boldsymbol{\Gamma}^T)$, we get

$$\begin{aligned} \nabla_{\mathrm{vec}(\boldsymbol{\Gamma})}\mathbf{a}_s^T &= \nabla_{\mathrm{vec}(\boldsymbol{\Gamma})}\,\mathrm{vec}(\boldsymbol{\Gamma}^T)^T(\mathbf{x}'_s \otimes \mathbf{I}_n) \\ &= \nabla_{\mathrm{vec}(\boldsymbol{\Gamma})}\,\mathrm{vec}(\boldsymbol{\Gamma})^T\mathbf{C}^{(k,q)}(\mathbf{x}'_s \otimes \mathbf{I}_\kappa) \\ &= \mathbf{C}^{(k,q)}(\mathbf{x}'_s \otimes \mathbf{I}_\kappa). \end{aligned}$$

This verifies (37). Then by using the chain rule (24), we get

$$\nabla_{\mathrm{vec}(\mathbf{W})}\ell(y_s, \mathbf{a}_s) = \nabla_{\mathrm{vec}(\mathbf{W})}\mathbf{a}_s^T\;\nabla_{\mathbf{a}_s}\ell(y_s, \mathbf{a}_s) = \mathbf{C}^{(r,p)}(\mathbf{x}_s \otimes \boldsymbol{\beta})\,\dot{\mathbf{h}}(y_s, \mathbf{a}_s).$$

Other gradients $\nabla_{\mathrm{vec}(\boldsymbol{\beta})}\,\ell(y_s, \mathbf{a}_s)$ and $\nabla_{\mathrm{vec}(\boldsymbol{\Gamma})}\,\ell(y_s, \mathbf{a}_s)$ also follow from (37) and the chain rule.

Next, we compute the gradients of $\mathrm{vec}(\mathbf{K})^T$ in (36). First, using (37), the chain rule (24), and (32),

$$\begin{aligned} \nabla_{\mathrm{vec}(\mathbf{W})}\,\dot{\mathbf{h}}(y_s, \mathbf{a}_s)^T &= \nabla_{\mathrm{vec}(\mathbf{W})}\mathbf{a}_s^T\;\nabla_{\mathbf{a}_s}\dot{\mathbf{h}}(y_s, \mathbf{a}_s)^T = \mathbf{C}^{(r,p)}(\mathbf{x}_s \otimes \boldsymbol{\beta})\ddot{\mathbf{H}}(y_s, \mathbf{a}_s), \\ \nabla_{\mathrm{vec}(\boldsymbol{\beta})}\,\dot{\mathbf{h}}(y_s, \mathbf{a}_s)^T &= \nabla_{\mathrm{vec}(\boldsymbol{\beta})}\mathbf{a}_s^T\;\nabla_{\mathbf{a}_s}\dot{\mathbf{h}}(y_s, \mathbf{a}_s)^T = \mathbf{C}^{(\kappa,r)}(\mathbf{W}^T\mathbf{x}_s \otimes \mathbf{I}_\kappa)\ddot{\mathbf{H}}(y_s, \mathbf{a}_s), \\ \nabla_{\mathrm{vec}(\boldsymbol{\Gamma})}\,\dot{\mathbf{h}}(y_s, \mathbf{a}_s)^T &= \nabla_{\mathrm{vec}(\boldsymbol{\Gamma})}\mathbf{a}_s^T\;\nabla_{\mathbf{a}_s}\dot{\mathbf{h}}(y_s, \mathbf{a}_s)^T = \mathbf{C}^{(\kappa,q)}(\mathbf{x}'_s \otimes \mathbf{I}_\kappa)\ddot{\mathbf{H}}(y_s, \mathbf{a}_s). \end{aligned}$$

Now since $\mathrm{vec}(\mathbf{K})^T = [\dot{\mathbf{h}}(y_1, \mathbf{a}_1)^T, \dots, \dot{\mathbf{h}}(y_n, \mathbf{a}_n)^T]$ and $\mathrm{vec}(\mathbf{K}^T)^T = (\mathbf{C}^{(\kappa,n)}\,\mathrm{vec}(\mathbf{K}))^T = \mathrm{vec}(\mathbf{K})^T\mathbf{C}^{(n,\kappa)}$, it follows that

$$\begin{aligned} \nabla_{\mathrm{vec}(\mathbf{W})}\,\mathrm{vec}(\mathbf{K})^T &\overset{(a)}{=} \left[\mathbf{C}^{(r,p)}(\mathbf{x}_1 \otimes \boldsymbol{\beta})\ddot{\mathbf{H}}(y_1, \mathbf{a}_1), \dots, \mathbf{C}^{(r,p)}(\mathbf{x}_n \otimes \boldsymbol{\beta})\ddot{\mathbf{H}}(y_n, \mathbf{a}_n)\right] \\ &\overset{(b)}{=} \mathbf{C}^{(r,p)}\,[\mathbf{x}_1 \otimes \boldsymbol{\beta}, \dots, \mathbf{x}_n \otimes \boldsymbol{\beta}]\,\mathrm{diag}\left(\ddot{\mathbf{H}}(y_1, \mathbf{a}_1), \dots, \ddot{\mathbf{H}}(y_n, \mathbf{a}_n)\right) \\ &\overset{(c)}{=} \mathbf{C}^{(r,p)}(\mathbf{X} \otimes \boldsymbol{\beta})\mathbf{M} \\ &\overset{(d)}{=} (\boldsymbol{\beta} \otimes \mathbf{X})\mathbf{C}^{(\kappa,n)}\mathbf{M}, \end{aligned}$$

where (a) follows from (37) and the chain rule, (b) is an algebra, (c) follows from (19), and (d) follows from (20). The other gradients $\nabla_{\mathrm{vec}(\boldsymbol{\beta})}\,\mathrm{vec}(\mathbf{K})^T$ and $\nabla_{\mathrm{vec}(\boldsymbol{\Gamma})}\,\mathrm{vec}(\mathbf{K})^T$ follow from similar computations. □

Lemma C.2 (Derivatives of the SMF-$\mathbf{W}$ objective)**.** *Let $f(\mathbf{Z})$ denote the objective of SMF-$\mathbf{W}$ in (30). Suppose Assumption B.1 holds. Recall $\dot{\mathbf{h}}$ and $\ddot{\mathbf{H}}$ defined in (33). Then the gradients of $f(\mathbf{Z})$ are given by*

$$\nabla_{\mathbf{W}} f(\mathbf{Z}) = \mathbf{X}\mathbf{K}^T\boldsymbol{\beta}^T + 2\xi(\mathbf{W}\mathbf{H} - \mathbf{X})\mathbf{H}^T, \tag{38}$$

$$\nabla_{\mathbf{H}} f(\mathbf{Z}) = 2\xi\mathbf{W}^T(\mathbf{W}\mathbf{H} - \mathbf{X}), \tag{39}$$

$$\nabla_{\boldsymbol{\beta}} f(\mathbf{Z}) = \mathbf{W}^T\mathbf{X}\mathbf{K}^T, \tag{40}$$

$$\nabla_{\boldsymbol{\Gamma}} f(\mathbf{Z}) = \mathbf{X}_{\text{aux}}\mathbf{K}^T. \tag{41}$$

The block-diagonal terms in the Hessian are given by

$$\nabla_{\mathrm{vec}(\mathbf{W})}\nabla_{\mathrm{vec}(\mathbf{W})^T} f(\mathbf{Z}) = (\boldsymbol{\beta}\otimes\mathbf{X})\mathbf{C}^{(\kappa,n)}\mathbf{M}\mathbf{C}^{(n,\kappa)}(\boldsymbol{\beta}\otimes\mathbf{X})^T + 2\xi(\mathbf{H}\mathbf{H}^T\otimes\mathbf{I}_p), \tag{42}$$

$$\nabla_{\mathrm{vec}(\mathbf{H})}\nabla_{\mathrm{vec}(\mathbf{H})^T} f(\mathbf{Z}) = 2\xi(\mathbf{I}_n\otimes\mathbf{W}^T\mathbf{W}), \tag{43}$$

$$\nabla_{\mathrm{vec}(\boldsymbol{\beta})}\nabla_{\mathrm{vec}(\boldsymbol{\beta})^T} f(\mathbf{Z}) = (\mathbf{I}_\kappa\otimes\mathbf{W}^T\mathbf{X})\mathbf{C}^{(\kappa,n)}\mathbf{M}\mathbf{C}^{(n,\kappa)}(\mathbf{I}_\kappa\otimes\mathbf{W}^T\mathbf{X})^T, \tag{44}$$

$$\nabla_{\mathrm{vec}(\boldsymbol{\Gamma})}\nabla_{\mathrm{vec}(\boldsymbol{\Gamma})^T} f(\mathbf{Z}) = (\mathbf{I}_\kappa\otimes\mathbf{X}_{\text{aux}})\mathbf{C}^{(\kappa,n)}\mathbf{M}\mathbf{C}^{(n,\kappa)}(\mathbf{I}_\kappa\otimes\mathbf{X}_{\text{aux}})^T. \tag{45}$$

The block-off-diagonal terms in the Hessian are given by

$$\nabla_{\mathrm{vec}(\mathbf{H})}\nabla_{\mathrm{vec}(\mathbf{W})^T} f(\mathbf{Z}) = 2\xi\left[(\mathbf{H}^T\otimes\mathbf{W}^T) + \mathbf{C}^{(n,r)}(\mathbf{I}_r\otimes(\mathbf{W}\mathbf{H}-\mathbf{X}))^T\right], \tag{46}$$

$$\nabla_{\mathrm{vec}(\boldsymbol{\beta})}\nabla_{\mathrm{vec}(\mathbf{W})^T} f(\mathbf{Z}) = \mathbf{C}^{(\kappa,r)}(\mathbf{I}_r\otimes\mathbf{X}\mathbf{K}^T)^T + (\mathbf{I}_\kappa\otimes\mathbf{W}^T\mathbf{X})\mathbf{C}^{(\kappa,n)}\mathbf{M}\mathbf{C}^{(n,\kappa)}(\boldsymbol{\beta}\otimes\mathbf{X})^T, \tag{47}$$

$$\nabla_{\mathrm{vec}(\boldsymbol{\Gamma})}\nabla_{\mathrm{vec}(\mathbf{W})^T} f(\mathbf{Z}) = \nabla_{\mathrm{vec}(\boldsymbol{\beta})}\nabla_{\mathrm{vec}(\mathbf{H})^T} f(\mathbf{Z}) = \nabla_{\mathrm{vec}(\boldsymbol{\Gamma})}\nabla_{\mathrm{vec}(\mathbf{H})^T} f(\mathbf{Z}) = \mathbf{O}, \tag{48}$$

$$\nabla_{\mathrm{vec}(\boldsymbol{\Gamma})}\nabla_{\mathrm{vec}(\boldsymbol{\beta})^T} f(\mathbf{Z}) = (\mathbf{I}_r\otimes\mathbf{X}_{\text{aux}})\mathbf{C}^{(\kappa,n)}\mathbf{M}\mathbf{C}^{(n,\kappa)}(\mathbf{I}_\kappa\otimes\mathbf{W}^T\mathbf{X})^T. \tag{49}$$

Proof. For convenience, recall that $\mathbf{W}\in\mathbb{R}^{p\times r}$, $\boldsymbol{\beta}\in\mathbb{R}^{r\times\kappa}$, $\mathbf{H}\in\mathbb{R}^{r\times n}$, and $\boldsymbol{\Gamma}\in\mathbb{R}^{q\times\kappa}$.

Computation of the first-order derivatives.

We first compute the following gradient:

$$\nabla_{\mathrm{vec}(\mathbf{W})}\sum_{s=1}^{n}\ell(y_s,\mathbf{a}_s) \overset{(a)}{=} \mathbf{C}^{(r,p)}\sum_{s=1}^{n}(\mathbf{x}_s\otimes\boldsymbol{\beta})\,\dot{\mathbf{h}}(y_s,\mathbf{a}_s) \tag{50}$$

$$= \mathbf{C}^{(r,p)}\left[\mathbf{x}_1\otimes\boldsymbol{\beta},\dots,\mathbf{x}_n\otimes\boldsymbol{\beta}\right]\begin{bmatrix}\dot{\mathbf{h}}(y_1,\mathbf{a}_1)\\ \vdots\\ \dot{\mathbf{h}}(y_n,\mathbf{a}_n)\end{bmatrix} \tag{51}$$

$$\overset{(b)}{=} \mathbf{C}^{(r,p)}(\mathbf{X}\otimes\boldsymbol{\beta})\,\mathrm{vec}(\mathbf{K}) \tag{52}$$

$$\overset{(c)}{=} (\boldsymbol{\beta}\otimes\mathbf{X})\mathbf{C}^{(\kappa,n)}\,\mathrm{vec}(\mathbf{K}) \tag{53}$$

$$\overset{(d)}{=} (\boldsymbol{\beta}\otimes\mathbf{X})\,\mathrm{vec}(\mathbf{K}^T), \tag{54}$$

where (a) follows from Proposition C.1, (b) follows from (19), (c) follows from (20), and (d) uses the definition of the commutation matrices. Then by using (22), we deduce

$$\nabla_{\mathbf{W}} f(\mathbf{Z}) = \mathbf{X}\mathbf{K}^T\boldsymbol{\beta}^T + 2\xi(\mathbf{W}\mathbf{H}-\mathbf{X})\mathbf{H}^T. \tag{55}$$

Next, we compute $\nabla_{\mathrm{vec}(\boldsymbol{\beta})} f(\mathbf{Z})$. By using similar computations as before, we get

$$\begin{aligned}
\nabla_{\mathrm{vec}(\boldsymbol{\beta})} \sum_{s=1}^{n} \ell(y_s, \mathbf{a}_s) &= \mathbf{C}^{(\kappa,r)} \sum_{s=1}^{n} (\mathbf{W}^T \mathbf{x}_s \otimes \mathbf{I}_\kappa)\, \dot{\mathbf{h}}(y_s, \mathbf{a}_s) \\
&= \mathbf{C}^{(\kappa,r)} \left[\mathbf{W}^T \mathbf{x}_1 \otimes \mathbf{I}_\kappa, \dots, \mathbf{W}^T \mathbf{x}_n \otimes \mathbf{I}_\kappa\right] \begin{bmatrix} \dot{\mathbf{h}}(y_1, \mathbf{a}_1) \\ \vdots \\ \dot{\mathbf{h}}(y_n, \mathbf{a}_n) \end{bmatrix} \\
&= \mathbf{C}^{(\kappa,r)} (\mathbf{W}^T \mathbf{X} \otimes \mathbf{I}_\kappa) \,\mathrm{vec}(\mathbf{K}) \\
&= (\mathbf{I}_\kappa \otimes \mathbf{W}^T \mathbf{X}) \mathbf{C}^{(\kappa,n)} \,\mathrm{vec}(\mathbf{K}) \\
&= (\mathbf{I}_\kappa \otimes \mathbf{W}^T \mathbf{X}) \,\mathrm{vec}(\mathbf{K}^T).
\end{aligned}$$

From this and (22), we deduce

$$\nabla_{\boldsymbol{\beta}} f(\mathbf{Z}) = \mathbf{W}^T \mathbf{X} \mathbf{K}^T. \tag{56}$$

We move on to compute $\nabla_{\mathrm{vec}(\Gamma)} f(\mathbf{Z})$. This yields

$$\begin{aligned}
\nabla_{\mathrm{vec}(\boldsymbol{\Gamma})} \sum_{s=1}^{n} \ell(y_s, \mathbf{a}_s) &= \mathbf{C}^{(k,q)} \sum_{s=1}^{n} (\mathbf{x}'_s \otimes \mathbf{I}_\kappa)\, \dot{\mathbf{h}}(y_s, \mathbf{a}_s) \\
&= \mathbf{C}^{(\kappa,q)} (\mathbf{X}_{\text{aux}} \otimes \mathbf{I}_\kappa) \,\mathrm{vec}(\mathbf{K}) \\
&= (\mathbf{I}_\kappa \otimes \mathbf{X}_{\text{aux}}) \mathbf{C}^{(\kappa,n)} \,\mathrm{vec}(\mathbf{K}) \\
&= (\mathbf{I}_\kappa \otimes \mathbf{X}_{\text{aux}}) \,\mathrm{vec}(\mathbf{K}^T)
\end{aligned}$$

From this and (22), we deduce

$$\nabla_{\boldsymbol{\Gamma}} f(\mathbf{Z}) = \mathbf{X}_{\text{aux}} \mathbf{K}^T. \tag{57}$$

The last derivative $\nabla_{\mathbf{H}}\, f(\mathbf{Z}) = 2\xi \mathbf{W}^T (\mathbf{W}\mathbf{H} - \mathbf{X})$ is easy.

Computation of the second-order derivatives.

By vectorizing (55), we get

$$\begin{aligned}
\nabla_{\mathrm{vec}(\mathbf{W})} f(\mathbf{Z}) &= \mathrm{vec}\left(\mathbf{X}\mathbf{K}^T \boldsymbol{\beta}^T\right) + 2\xi \,\mathrm{vec}(\mathbf{W}\mathbf{H}\mathbf{H}^T) - 2\xi \,\mathrm{vec}(\mathbf{X}\mathbf{H}^T) \\
&= (\boldsymbol{\beta} \otimes \mathbf{X}) \,\mathrm{vec}(\mathbf{K}^T) + 2\xi (\mathbf{H}\mathbf{H}^T \otimes \mathbf{I}_p) \,\mathrm{vec}(\mathbf{W}) - 2\xi \,\mathrm{vec}(\mathbf{X}\mathbf{H}^T).
\end{aligned} \tag{58}$$

Then using Proposition C.1 with (58) and noting that $\mathrm{vec}(\mathbf{K}^T)^T = (\mathbf{C}^{(\kappa,n)} \,\mathrm{vec}(\mathbf{K}))^T = \mathrm{vec}(\mathbf{K})^T \mathbf{C}^{(n,\kappa)}$, we get

$$\begin{aligned}
&\nabla_{\mathrm{vec}(\mathbf{W})} \nabla_{\mathrm{vec}(\mathbf{W})^T} f(\mathbf{Z}) \\
&\quad = \nabla_{\mathrm{vec}(\mathbf{W})} \left(\mathrm{vec}(\mathbf{K})^T \mathbf{C}^{(n,\kappa)} (\boldsymbol{\beta} \otimes \mathbf{X})^T + 2\xi \,\mathrm{vec}(\mathbf{W})^T (\mathbf{H}\mathbf{H}^T \otimes \mathbf{I}_p)^T - 2\xi \,\mathrm{vec}(\mathbf{X}\mathbf{H}^T)^T\right) \\
&\quad = (\boldsymbol{\beta} \otimes \mathbf{X}) \mathbf{C}^{(\kappa,n)} \mathbf{M} \mathbf{C}^{(n,\kappa)} (\boldsymbol{\beta} \otimes \mathbf{X})^T + 2\xi (\mathbf{H}\mathbf{H}^T \otimes \mathbf{I}_p).
\end{aligned}$$

Similarly, we can compute

$$\begin{aligned}
&\nabla_{\mathrm{vec}(\boldsymbol{\beta})} \nabla_{\mathrm{vec}(\boldsymbol{\beta})^T} f(\mathbf{Z}) \\
&\quad = \nabla_{\mathrm{vec}(\boldsymbol{\beta})} \,\mathrm{vec}(\mathbf{W}^T \mathbf{X} \mathbf{K}^T)^T \\
&\quad = \nabla_{\mathrm{vec}(\boldsymbol{\beta})} \,\mathrm{vec}(\mathbf{K}^T)^T (\mathbf{I}_\kappa \otimes \mathbf{W}^T \mathbf{X})^T \\
&\quad = \nabla_{\mathrm{vec}(\boldsymbol{\beta})} \,\mathrm{vec}(\mathbf{K})^T \mathbf{C}^{(n,\kappa)} (\mathbf{I}_\kappa \otimes \mathbf{W}^T \mathbf{X})^T \\
&\quad = (\mathbf{I}_\kappa \otimes \mathbf{W}^T \mathbf{X}) \mathbf{C}^{(\kappa,n)} \mathbf{M} \mathbf{C}^{(n,\kappa)} (\mathbf{I}_\kappa \otimes \mathbf{W}^T \mathbf{X})^T.
\end{aligned}$$

Also, note that

$$
\begin{aligned}
\nabla_{\mathrm{vec}(\boldsymbol{\Gamma})}\nabla_{\mathrm{vec}(\boldsymbol{\Gamma})^T} f(\mathbf{Z}) &= \nabla_{\mathrm{vec}(\boldsymbol{\Gamma})}\,\mathrm{vec}(\mathbf{X}_{\text{aux}}\mathbf{K}^T)^T \\
&= \nabla_{\mathrm{vec}(\boldsymbol{\Gamma})}\,\mathrm{vec}(\mathbf{K})^T\mathbf{C}^{(n,\kappa)}(\mathbf{I}_\kappa\otimes\mathbf{X}_{\text{aux}})^T \\
&= (\mathbf{I}_\kappa\otimes\mathbf{X}_{\text{aux}})\mathbf{C}^{(\kappa,n)}\mathbf{M}\mathbf{C}^{(n,\kappa)}(\mathbf{I}_\kappa\otimes\mathbf{X}_{\text{aux}})^T.
\end{aligned}
$$

Similarly, we get

$$
\begin{aligned}
\nabla_{\mathrm{vec}(\mathbf{H})}\nabla_{\mathrm{vec}(\mathbf{H})^T} f(\mathbf{Z}) &= \nabla_{\mathrm{vec}(\mathbf{H})}\left(2\xi\,\mathrm{vec}(\mathbf{W}^T\mathbf{W}\mathbf{H})^T - 2\xi\,\mathrm{vec}(\mathbf{W}^T\mathbf{X})^T\right) \\
&= 2\xi\nabla_{\mathrm{vec}(\mathbf{H})}\,\mathrm{vec}(\mathbf{H})^T(\mathbf{I}_n\otimes\mathbf{W}^T\mathbf{W}) \\
&= 2\xi(\mathbf{I}_n\otimes\mathbf{W}^T\mathbf{W}).
\end{aligned}
$$

Next, we compute the off-diagonal block terms in the Hessian of f. Recall that from (22), we have

$$
\mathrm{vec}(\mathbf{X}\mathbf{K}^T\boldsymbol{\beta}^T) = (\mathbf{I}_r\otimes\mathbf{X}\mathbf{K}^T)\,\mathrm{vec}(\boldsymbol{\beta}^T) = (\boldsymbol{\beta}\otimes\mathbf{X})\,\mathrm{vec}(\mathbf{K}^T).
$$

Then using the product rule, we get

$$
\begin{aligned}
\nabla_{\mathrm{vec}(\boldsymbol{\beta})}\nabla_{\mathrm{vec}(\mathbf{W})^T} f(\mathbf{Z}) &= \nabla_{\mathrm{vec}(\boldsymbol{\beta})}\,\mathrm{vec}(\mathbf{X}\mathbf{K}^T\boldsymbol{\beta}^T)^T \\
&= \left(\nabla_{\mathrm{vec}(\boldsymbol{\beta})}\,\mathrm{vec}(\boldsymbol{\beta}^T)^T\right)(\mathbf{I}_r\otimes\mathbf{X}\mathbf{K}^T)^T + \left(\nabla_{\mathrm{vec}(\boldsymbol{\beta})}\,\mathrm{vec}(\mathbf{K}^T)^T\right)(\boldsymbol{\beta}\otimes\mathbf{X})^T \\
&= \mathbf{C}^{(\kappa,r)}(\mathbf{I}_r\otimes\mathbf{X}\mathbf{K}^T)^T + (\mathbf{I}_\kappa\otimes\mathbf{W}^T\mathbf{X})\mathbf{C}^{(\kappa,n)}\mathbf{M}\mathbf{C}^{(n,\kappa)}(\boldsymbol{\beta}\otimes\mathbf{X})^T.
\end{aligned}
$$

Second, note that $\nabla_{\mathrm{vec}(\boldsymbol{\Gamma})}\nabla_{\mathrm{vec}(\mathbf{W})^T} f(\mathbf{Z}) = \mathbf{O}$. Third, for the forthcoming computation, note that from (22),

$$
\mathrm{vec}(\mathbf{H}\mathbf{H}^T) = (\mathbf{I}_r\otimes\mathbf{H})\,\mathrm{vec}(\mathbf{H}^T) = (\mathbf{H}\otimes\mathbf{I}_r)\,\mathrm{vec}(\mathbf{H}).
$$

So by the product rule,

$$
\begin{aligned}
\nabla_{\mathrm{vec}(\mathbf{H})}\,\mathrm{vec}(\mathbf{H}\mathbf{H}^T)^T &= \left(\nabla_{\mathrm{vec}(\mathbf{H})}\,\mathrm{vec}(\mathbf{H}^T)^T\right)(\mathbf{I}_r\otimes\mathbf{H})^T + \left(\nabla_{\mathrm{vec}(\mathbf{H})}\,\mathrm{vec}(\mathbf{H})^T\right)(\mathbf{H}\otimes\mathbf{I}_r)^T \\
&= \mathbf{C}^{(n,r)}(\mathbf{I}_r\otimes\mathbf{H}^T) + (\mathbf{H}^T\otimes\mathbf{I}_r).
\end{aligned}
$$

Now observe that

$$
\begin{aligned}
\nabla_{\mathrm{vec}(\mathbf{H})}\nabla_{\mathrm{vec}(\mathbf{W})^T} f(\mathbf{Z}) &= 2\xi\nabla_{\mathrm{vec}(\mathbf{H})}\left[\mathrm{vec}(\mathbf{W}\mathbf{H}\mathbf{H}^T) - \mathrm{vec}(\mathbf{X}\mathbf{H}^T)\right]^T \\
&= 2\xi\nabla_{\mathrm{vec}(\mathbf{H})}\left[\mathrm{vec}(\mathbf{H}\mathbf{H}^T)^T(\mathbf{I}_r\otimes\mathbf{W})^T - \mathrm{vec}(\mathbf{H}^T)^T(\mathbf{I}_r\otimes\mathbf{X})^T\right] \\
&= 2\xi\left(\nabla_{\mathrm{vec}(\mathbf{H})}\,\mathrm{vec}(\mathbf{H}\mathbf{H}^T)^T\right)(\mathbf{I}_r\otimes\mathbf{W})^T - \left(\nabla_{\mathrm{vec}(\mathbf{H})}\,\mathrm{vec}(\mathbf{H}^T)^T\right)(\mathbf{I}_r\otimes\mathbf{X})^T \\
&= 2\xi\left[\left(\mathbf{C}^{(n,r)}(\mathbf{I}_r\otimes\mathbf{H}^T) + (\mathbf{H}^T\otimes\mathbf{I}_r)\right)(\mathbf{I}_r\otimes\mathbf{W})^T - \mathbf{C}^{(n,r)}(\mathbf{I}_r\otimes\mathbf{X})^T\right] \\
&= 2\xi\left[\mathbf{C}^{(n,r)}(\mathbf{I}_r\otimes\mathbf{H}^T\mathbf{W}^T) + (\mathbf{H}^T\otimes\mathbf{W}^T) - \mathbf{C}^{(n,r)}(\mathbf{I}_r\otimes\mathbf{X})^T\right] \\
&= 2\xi\left[\mathbf{C}^{(n,r)}(\mathbf{I}_r\otimes(\mathbf{W}\mathbf{H}-\mathbf{X}))^T + (\mathbf{H}^T\otimes\mathbf{W}^T)\right].
\end{aligned}
$$

Fourth, observe that

$$
\begin{aligned}
\nabla_{\mathrm{vec}(\boldsymbol{\Gamma})}\nabla_{\mathrm{vec}(\boldsymbol{\beta})^T} f(\mathbf{Z}) &= \nabla_{\mathrm{vec}(\boldsymbol{\Gamma})}\,\mathrm{vec}(\mathbf{W}^T\mathbf{X}\mathbf{K}^T)^T \\
&= \nabla_{\mathrm{vec}(\boldsymbol{\Gamma})}\,\mathrm{vec}(\mathbf{K})^T\mathbf{C}^{(n,\kappa)}(\mathbf{I}_\kappa\otimes\mathbf{W}^T\mathbf{X})^T \\
&= (\mathbf{I}_\kappa\otimes\mathbf{X}_{\text{aux}})\mathbf{C}^{(\kappa,n)}\,\mathbf{M}\mathbf{C}^{(n,\kappa)}(\mathbf{I}_\kappa\otimes\mathbf{W}^T\mathbf{X})^T.
\end{aligned}
$$

The remaining zero-second derivatives are easy to see. □

For two matrices A, B of the same size, we write $A\succeq B$ if $A-B$ is positive semi-definite. The partial ordering $\succeq$ is called the Loewner ordering.

Lemma C.3. *Let $f(\mathbf{Z})$ denote the objective of SMF-$\mathbf{W}$ in (30). Suppose Assumption B.1 holds. Recall $\dot{\mathbf{h}}$ and $\ddot{\mathbf{H}}$ defined in (33). Then the following hold:*

(i) *Write the Hessian $\nabla^2 f(\mathbf{Z})$ as the 4×4 block matrix $(A_{ij})_{1 \le i,j \le 4}$. Then*

$$\begin{aligned} \alpha^-(\boldsymbol{\beta}\boldsymbol{\beta}^T \otimes \mathbf{X}\mathbf{X}^T) + 2\xi(\mathbf{H}\mathbf{H}^T \otimes \mathbf{I}_p) \preceq A_{11} &\preceq \alpha^+(\boldsymbol{\beta}\boldsymbol{\beta}^T \otimes \mathbf{X}\mathbf{X}^T) + 2\xi(\mathbf{H}\mathbf{H}^T \otimes \mathbf{I}_p) \\ A_{22} &= 2\xi(\mathbf{I}_n \otimes \mathbf{W}^T\mathbf{W}), \\ \alpha^-(\mathbf{I}_\kappa \otimes \mathbf{W}^T\mathbf{X}\mathbf{X}^T\mathbf{W}) \preceq A_{33} &\preceq \alpha^+(\mathbf{I}_\kappa \otimes \mathbf{W}^T\mathbf{X}\mathbf{X}^T\mathbf{W}), \\ \alpha^-(\mathbf{I}_\kappa \otimes \mathbf{X}_{\text{aux}}\mathbf{X}_{\text{aux}}^T) \preceq A_{44} &\preceq \alpha^+(\mathbf{I}_\kappa \otimes \mathbf{X}_{\text{aux}}\mathbf{X}_{\text{aux}}^T). \end{aligned}$$

(ii) *The function $f(\mathbf{Z}) = f(\mathbf{W}, \mathbf{H}, \boldsymbol{\beta}, \boldsymbol{\Gamma})$ restricted to each block coordinate has Lipschitz-continuous gradients with Lipschitz constants $L_{\mathbf{W}}, L_{\mathbf{H}}, L_{\boldsymbol{\beta}}, L_{\boldsymbol{\Gamma}}$ given by*

$$\begin{aligned} L_{\mathbf{W}} &:= \alpha^+\|\boldsymbol{\beta}\|_2^2 \cdot \|\mathbf{X}\|_2^2 + 2\xi\|\mathbf{H}\|_2^2, \\ L_{\mathbf{H}} &:= 2\xi\|\mathbf{W}\|_2^2, \\ L_{\boldsymbol{\beta}} &:= \alpha^+\|\mathbf{W}\|_2^2 \cdot \|\mathbf{X}\|_2^2, \\ L_{\boldsymbol{\Gamma}} &:= \alpha^+\|\mathbf{X}_{\text{aux}}\|_2^2. \end{aligned}$$

(iii) *The Hessian of the L_2-regularized objective $f(\mathbf{Z}) + \frac{\lambda_{\mathbf{W}}}{2}\|\mathbf{W}\|_F^2 + \frac{\lambda_{\mathbf{H}}}{2}\|\mathbf{H}\|_F^2 + \frac{\lambda_{\boldsymbol{\beta}}}{2}\|\boldsymbol{\beta}\|_F^2 + \frac{\lambda_{\boldsymbol{\Gamma}}}{2}\|\boldsymbol{\Gamma}\|_F^2$ is positive definite if*

$$\begin{aligned} \lambda_{\mathbf{W}} &> 2\xi\left(\|\mathbf{H}\|_2\,\|\mathbf{W}\|_2 + \|\mathbf{W}\mathbf{H} - \mathbf{X}\|_2 - \lambda_{\min}(\mathbf{H}\mathbf{H}^T)\right) + \gamma_{\max}\sqrt{\kappa n}\|\mathbf{X}\|_2 \\ &\quad + \alpha^+\|\boldsymbol{\beta}\|_2\|\mathbf{W}\|_2\|\mathbf{X}\|_2^2 - \alpha^-\lambda_{\min}(\boldsymbol{\beta}\boldsymbol{\beta}^T)\lambda_{\min}(\mathbf{X}\mathbf{X}^T) \\ \lambda_{\mathbf{H}} &> 2\xi\left(\|\mathbf{H}\|_2\,\|\mathbf{W}\|_2 + \|\mathbf{W}\mathbf{H} - \mathbf{X}\|_2 - \lambda_{\min}(\mathbf{W}^T\mathbf{W})\right), \\ \lambda_{\boldsymbol{\beta}} &> \gamma_{\max}\sqrt{\kappa n}\|\mathbf{X}\|_2 + \alpha^+\|\boldsymbol{\beta}\|_2\|\mathbf{W}\|_2\|\mathbf{X}\|_2^2 + \alpha^+\|\mathbf{X}_{\text{aux}}\|_2\|\mathbf{X}^T\mathbf{W}\|_2 - \alpha^-\lambda_{\min}(\mathbf{W}^T\mathbf{X}\mathbf{X}^T\mathbf{W}), \\ \lambda_{\boldsymbol{\Gamma}} &> \alpha^+\|\mathbf{X}_{\text{aux}}\|_2\|\mathbf{X}^T\mathbf{W}\|_2 - \alpha^-\lambda_{\min}(\mathbf{X}_{\text{aux}}\mathbf{X}_{\text{aux}}^T) \end{aligned}$$

Proof. Observe that the block-diagonal matrix $\mathbf{M}$ in (35) is symmetric by definition and is also positive definite by Assumption B.1:

$$0 < \alpha^- \le \lambda_{\min}(\mathbf{M}) \le \lambda_{\max}(\mathbf{M}) \le \alpha^+.$$

Since the commutation matrices are orthogonal and satisfies $\mathbf{C}^{(a,b)}\mathbf{C}^{(b,a)} = \mathbf{I}_{ab}$, it follows that

$$\alpha^-\mathbf{I}_{\kappa n} \preceq \mathbf{C}^{(\kappa,n)}\mathbf{M}\mathbf{C}^{(n,\kappa)} \preceq \alpha^+\mathbf{I}_{\kappa n}.$$

Then the first Loewner ordering for $A_{11} = \nabla_{\text{vec}(\mathbf{W})}\nabla_{\text{vec}(\mathbf{W})^T} f(\mathbf{Z})$ follows from Lemma C.2. The other Loewner orderings can be shown similarly. This shows **(i)**.

(ii) follows immediately from **(i)**, $\|A \otimes B\|_2 = \|A\|_2 \cdot \|B\|_2$, and the fact that the Lipschitz constant for the gradient is upper-bounded by the largest eigenvalue of the corresponding block Hessian, which are the diagonal blocks A_{ii} for $i = 1, \dots, 4$.

For **(iii)**, note that if L_2-regularization coefficients are large enough so that the following condition is satisfied

$$\lambda_{\min}(A_{ii}) + \lambda_i > \sum_{j \ne i}\|A_{ij}\|_2 \quad \forall 1 \le i \le 4,$$

where $\lambda_1 = \lambda_{\mathbf{W}}$, $\lambda_2 = \lambda_{\mathbf{H}}$, $\lambda_3 = \lambda_{\boldsymbol{\beta}}$, and $\lambda_4 = \lambda_{\boldsymbol{\Gamma}}$, then the L_2-regularized Hessian of the objective f is block diagonally dominant and is positive definite (see (Feingold & Varga, 1962)). The L_2-regularized Hessian takes the following 4×4 block form:

$$\begin{array}{c|cccc} & \text{vec}(\mathbf{W})^T & \text{vec}\,\mathbf{H}^T & \text{vec}(\boldsymbol{\beta})^T & \text{vec}(\boldsymbol{\Gamma})^T \\ \hline \text{vec}(\mathbf{W}) & A_{11} + \lambda_{\mathbf{W}}\mathbf{I}_{rp} & A_{12} & A_{13} & \mathbf{O} \\ \text{vec}(\mathbf{H}) & A_{21} & A_{22} + \lambda_{\mathbf{H}}\mathbf{I}_{rn} & \mathbf{O} & \mathbf{O} \\ \text{vec}(\boldsymbol{\beta}) & A_{31} & \mathbf{O} & A_{33} + \lambda_{\boldsymbol{\beta}}\mathbf{I}_{r\kappa} & A_{34} \\ \text{vec}(\boldsymbol{\Gamma}) & \mathbf{O} & \mathbf{O} & A_{43} & A_{44} + \lambda_{\boldsymbol{\Gamma}}\mathbf{I}_{q\kappa} \end{array} \tag{59}$$

Thus it suffices to take

$$\begin{aligned}\lambda_{\mathbf{W}} &> \|A_{12}\|_2 + \|A_{13}\|_2 - \lambda_{\min}(A_{11}),\\ \lambda_{\mathbf{H}} &> \|A_{12}\|_2 - \lambda_{\min}(A_{22}),\\ \lambda_{\boldsymbol{\beta}} &> \|A_{13}\|_2 + \|A_{34}\|_2 - \lambda_{\min}(A_{33}),\\ \lambda_{\boldsymbol{\Gamma}} &> \|A_{34}\|_2 - \lambda_{\min}(A_{44}).\end{aligned}$$

Using Lemma C.2 and Assumption B.1, we can upper bound the operator norm of the off-diagonal blocks as

$$\begin{aligned}\|A_{12}\|_2 &\le 2\xi\left(\|\mathbf{WH}\|_2 + \|\mathbf{WH}-\mathbf{X}\|_2\right),\\ \|A_{13}\|_2 &\le \|\mathbf{XK}^T\|_2 + \alpha^+\|\boldsymbol{\beta}\|_2\|\mathbf{W}\|_2\|\mathbf{X}\|_2^2\\ &\le \gamma_{\max}\sqrt{\kappa n}\|\mathbf{X}\|_2 + \alpha^+\|\boldsymbol{\beta}\|_2\|\mathbf{W}\|_2\|\mathbf{X}\|_2^2\\ A_{14} &= A_{23} = A_{24} = \mathbf{O},\\ \|A_{34}\|_2 &\le \alpha^+\|\mathbf{X}_{\text{aux}}\mathbf{X}^T\mathbf{W}^T\|_2,\end{aligned}$$

where we have used $\|\mathbf{1}_a\|_2 = \sqrt{a}$, $\|A\otimes B\|_2 = \|A\|_2\cdot\|B\|_2$, and $\|\mathbf{K}^T\|_2 \le \sqrt{\kappa n}\|\mathbf{K}\|_{\max} = \sqrt{\kappa n}\gamma_{\max}$. Furthermore, we can also get lower bounds on the eigenvalues of the diagonal blocks. Then the assertion in **(iii)** follows. □

Lemma C.4 (First-order approximation of functions with Lipschitz gradient)**.** *Let $f:\Omega(\subseteq\mathbb{R}^p)\to\mathbb{R}$ be differentiable and ∇f be L-Lipschitz continuous on Ω. Then for each $\theta,\theta'\in\Omega$,*

$$\left|f(\theta') - f(\theta) - \nabla f(\theta)^T(\theta'-\theta)\right| \le \frac{L}{2}\|\theta-\theta'\|^2.$$

Proof. This is a classical lemma. See Lemma 1.2.3 in (Nesterov, 1998). □

A simple but important lemma we use in our local landscape analysis is the following. It will be used in the proof of Theorems 4.3, C.6, and 4.5.

Lemma C.5 (L_2-perturbation of local landscape)**.** *Let $x\mapsto f(x)$ be three-times continuously differentiable function for $x\in\mathbb{R}^p$. Suppose $x_\star$ is a stationary point of f over a convex set $\boldsymbol{\Theta}\subseteq\mathbb{R}^p$. Suppose for constants $\lambda\ge 0$, $\rho>0$, $F(x) := f(x) + \frac{\lambda}{2}\|x\|^2$ is ρ-strongly convex at $x_\star$. Let $M = M(d)$ denote the supremum of the absolute values of all third-order partial derivatives of f over all x with $\|x-x_\star\|\le d$. Then as long as $\|x_\star\|$ and d are sufficiently small, there exists a local minimizer of F at some x' with $\|x'-x_\star\|\le d$.*

More precisely, we have

$$\inf_{x\in\boldsymbol{\Theta},\,\|x-x_\star\|=d} F(x) - F(x_\star) > 0 \tag{60}$$

provided d and $\|x_\star\|$ are sufficiently small so that

$$\frac{3\rho}{4} > M(d)d \quad\text{and}\quad \frac{\rho}{8}d \ge \lambda\|x_\star\|.$$

In particular, (60) holds if $d = \frac{4\lambda\|x_\star\|}{\rho}$ and $\|x_\star\|$ is sufficiently small so that $\frac{3\rho}{2} > M(d)d$ holds.

Proof. Since $x_\star$ is a stationary point of f over $\boldsymbol{\Theta}$, we have

$$\langle\nabla F(x_\star),\, x-x_\star\rangle \ge \langle\lambda x_\star,\, x-x_\star\rangle \ge -\lambda\|x_\star\|\cdot\|x-x_\star\|.$$

By Taylor's theorem, whenever $\|x-x_\star\| = d$,

$$\begin{aligned}F(x)-F(x_\star) &\ge -\lambda\|x_\star\|\cdot\|x-x_\star\| + \frac{1}{2}(x-x_\star)^T[\nabla_x\nabla_{x^T}F(x)]_{x=x_\star}(x-x_\star) - \frac{M(d)}{6}\|x-x_\star\|^3\\ &\ge d\underbrace{\left(-\lambda\|x_\star\| + \frac{\rho d}{4} - \frac{M(d)}{6}d^2\right)}_{=:\mathcal{I}} + \frac{\rho}{4}d^2.\end{aligned}$$

Note that $\mathcal{I} \geq 0$ if

$$\frac{\rho}{8} d \geq \lambda \|x_\star\| \quad \text{and} \quad \frac{\rho d}{8} > \frac{M(d)d^2}{3}.$$

The above condition is held by the hypothesis. This shows (60), as desired. □

Now we are ready to derive Theorem 4.3 as well as Theorem 4.4 for SMF-$\mathbf{W}$.

Proof of Theorem 4.3 *for SMF-*$\mathbf{W}$. Parts **(i)** and **(ii)** are re-statements Lemma C.3. Part **(iii)** follows from Lemmas C.3 and C.5. □

Proof of Theorem 4.4 *for SMF-*$\mathbf{W}$. Here we prove the statement for SMF-$\mathbf{W}$. Recall that Algorithm 1 is a block projected gradient descent with adaptive step sizes. This algorithm is well-known to be a special instance of a more general class of algorithms called block majorization-minimization (BMM) with prox-linear surrogates (Lyu & Li, 2023). For instance, for updating $\mathbf{W}_{k-1}$ to $\mathbf{W}_k$ given $\mathbf{S}_{k-1} := (\mathbf{H}_{k-1}, \boldsymbol{\beta}_{k-1}, \boldsymbol{\Gamma}_{k-1})$, we consider the following prox-linear surrogate

$$g_k^{(1)}(\mathbf{W}) := f(\mathbf{W}, \mathbf{S}_{k-1}) + \langle \nabla_{\mathbf{W}} f(\mathbf{W}_{k-1}, \mathbf{S}_{k-1}),\, \mathbf{W} - \mathbf{W}_{k-1} \rangle + \frac{1}{2\eta_{k;1}} \|\mathbf{W} - \mathbf{W}_{k-1}\|_F^2.$$

Note that $g_k^{(1)}(\mathbf{W}_{k-1}) = f(\mathbf{W}_{k-1}, \mathbf{S}_{k-1})$. By Lemma C.3, the marginal objective function $\mathbf{W} \mapsto f_k^{(1)}(\mathbf{W}) := f(\mathbf{W}, \mathbf{S}_{k-1})$ has $L_{\mathbf{W}}$-Lipschitz continuous gradient where $L_{\mathbf{W}} := \alpha^+ \|\boldsymbol{\beta}\|_2^2 \cdot \|\mathbf{X}\|_2^2 + 2\xi \|\mathbf{H}\|_2^2$. Hence by Lemma C.4, $g_k^{(1)}(\mathbf{W}) \geq f_k^{(1)}(\mathbf{W})$ for all $\mathbf{W} \in \mathcal{C}_1$ (i.e., $g_k^{(1)}$ is a majorizing surrogate of $f_k^{(1)}$ over $\mathcal{C}_1$) provided $\eta_{k;1}^{-1} > L_{\mathbf{W}}$. Indeed we choose $\eta_{k;1}^{-1} > L_{\mathbf{W}}$ in Algorithm 1. Furthermore, the marorization gap $g_k^{(1)} - f_k^{(1)}$ is quadratically lower-bounded:

$$g_k^{(1)}(\mathbf{W}) - f_k^{(1)}(\mathbf{W}) \geq \frac{\eta_{k;1}^{-1} - L_{\mathbf{W}}}{2} \|\mathbf{W} - \mathbf{W}_{k-1}\|_F^2 \quad \text{for all } \mathbf{W} \in \mathcal{C}_1. \tag{61}$$

Furthermore, one can easily verify that

$$\operatorname*{arg\,min}_{\mathbf{W} \in \mathcal{C}_1} g_k^{(1)}(\mathbf{W}) = \Pi_{\mathcal{C}_1}\left(\mathbf{W}_{k-1} - \frac{1}{\eta_{k;1}} \nabla_{\mathbf{W}} f(\mathbf{W}, \mathbf{S}_{k-1}) \right).$$

Hence we recover the projected gradient descent step for computing $\mathbf{W}_k$ in Algorithm 1 as minimizing the majorizing surrogate $g_k^{(1)}$ of $f_k^{(1)}$ over the constraint set $\mathcal{C}_1$. For other blocks, one can construct majorizing prox-linear surrogates $g_k^{(i)}$ of marginal loss functions $f_k^{(i)}$ for $i = 2, 3, 4$, defined similarly.

Asymptotic convergence to stationary points and iteration complexity of the BMM for smooth non-convex objective with convex constraints is recently established in Theorem 2.1 in (Lyu & Li, 2023). For the iteration complexity result, the hypotheses we need to verify are

(A1) The constraint sets $\mathcal{C}_1, \mathcal{C}_2, \mathcal{C}_3$, and $\mathcal{C}_4$ are closed and convex;

(A2) The objective $f : \boldsymbol{\Theta} \to \mathbb{R}$ is continuously differentiable, lower-bounded on $\boldsymbol{\Theta}$, and has L-Lipschitz continuous gradient over $\boldsymbol{\Theta}$ for some $L > 0$. Furthermore, the sub-level sets $\{\boldsymbol{\theta} \in \boldsymbol{\Theta} \,|\, f(\boldsymbol{\theta}) \leq a\}$ for $a \in \mathbb{R}$ are compact;

(A3) The majorizaiton gaps $h_k^{(i)} := g_k^{(i)} - f_k^{(i)}$ for $k \geq 1$ and $i = 1, 2, 3, 4$ are quadratically lower-bounded and has L_h-Lipscthiz continuous gradient over the constraint sets for some constant $L_h > 0$.

Indeed, (A1) and (A2) follow from 4.1 (especially with the compactness of the constraint sets) and Lemma C.2. The first part of (A3) follows from (61). For its second part, let L_g denote the supremum of the Lipschitz constants L_i for $i = 1, \dots, 4$ over all parameters in $\boldsymbol{\Theta}$. Since $\boldsymbol{\Theta}$ is compact by 4.1, $L_g < \infty$. Then $\eta_{k,i}^{-1} < L_g$ for all $k \geq 1$ and $i = 1, \dots, 4$, $\nabla g_k^{(i)}$s are L_g-Lipschtiz continuous. Recall that $\nabla f_k^{(i)}$ is L-Lipschitz continuous by (A2). Hence $\nabla h_k^{(i)}$s are $(L_g + L_f)$-Lipschitz continuous. Now the above three properties with Theorem 2.1 in (Lyu & Li, 2023) are enough to imply the iteration complexity results in Theorem 4.4 **(i)**-**(ii)**.

Lastly, asymptotic convergence of the iterates to the stationary points in (Lyu & Li, 2023) requires the following further assumption:

(A4) The majorizing surrogates $g_k^{(i)}$ for $k \geq 1$ and $i = 1, 2, 3, 4$ are ρ-strongly convex for some constant $\rho > 0$.

Since $g_k^{(i)}$ is $\eta_{k,i}^{-1}$-strongly convex and since we assume the step-sizes $\eta_{k,i}$ are uniformly upper bounded in Theorem 4.4**(iii)**, we can choose ρ to be the reciprocal of such uniform upper bound on $\eta_{k,i}$s. This finishes the proof. □

C.2. Proof for SMF-H

The following result stated in Theorem C.6 is the counterpart of the local landscape result (Theorem 4.3) for SMF-$\mathbf{H}$, which we prove in this section. We also establish Theorem 4.4 for SMF-$\mathbf{H}$. The structure of the argument is identical to that for SMF-$\mathbf{W}$ we provided in the previous section.

Theorem C.6 (Local landscape of SMF-$\mathbf{H}$)**.** *Let $f(\mathbf{Z})$ denote the objective of SMF-$\mathbf{H}$ in (30). Suppose Assumption B.1 holds. Then the following hold:*

(i)

$$\begin{aligned} A_{11} &= 2\xi(\mathbf{I}_p \otimes \mathbf{H}\mathbf{H}^T) \\ A_{22} &\asymp \alpha^{\pm}(\mathbf{I}_n \otimes \boldsymbol{\beta}\boldsymbol{\beta}^T) + 2\xi(\mathbf{I}_n \otimes \mathbf{W}^T\mathbf{W}), \\ A_{33} &\asymp \alpha^{\pm}(\mathbf{I}_\kappa \otimes \mathbf{H}\mathbf{H}^T), \\ A_{44} &\asymp \alpha^{\pm}(\mathbf{I}_\kappa \otimes \mathbf{X}_{\text{aux}}\mathbf{X}_{\text{aux}}^T). \end{aligned}$$

(ii) *F is ρ-strongly convex at $\mathbf{Z}_\star = (\mathbf{W}_\star, \mathbf{H}_\star, \boldsymbol{\beta}_\star, \boldsymbol{\Gamma}_\star)$ for $\rho = \min_{1\leq i\leq 4}(\lambda_i - \lambda_{i\star})$ where*

$$\begin{aligned} \lambda_{1\star} &:= -2\xi\Lambda_1, \\ \lambda_{2\star} &:= \gamma_{\max}\sqrt{\kappa n} + \alpha^{+}\|\boldsymbol{\beta}_\star\|_2\,(\|\mathbf{H}_\star\|_2 + \|\mathbf{X}_{\text{aux}}\|_2) \\ &\qquad - 2\xi\Lambda_2 - \alpha^{-}\lambda_{\min}(\boldsymbol{\beta}_\star\boldsymbol{\beta}_\star^T), \\ \lambda_{3\star} &:= \gamma_{\max}\sqrt{\kappa n} + \alpha^{+}\|\boldsymbol{\beta}_\star\|_2\,(\|\mathbf{H}_\star\|_2 + \|\mathbf{X}_{\text{aux}}\|_2) \\ &\qquad - \alpha^{-}\lambda_{\min}(\boldsymbol{\beta}_\star\boldsymbol{\beta}_\star^T) \\ \lambda_{4\star} &:= \alpha^{+}\|\mathbf{X}_{\text{aux}}\|_2\,(\|\boldsymbol{\beta}_\star\|_2 + \|\mathbf{H}_\star\|_2) - \alpha^{-}\lambda_{\min}(\mathbf{X}_{\text{aux}}\mathbf{X}_{\text{aux}}^T). \end{aligned}$$

(iii) *Suppose $\Lambda_1 > 0$. Denote $\theta' := (\mathbf{H}', \boldsymbol{\beta}', \boldsymbol{\Gamma}')$ and $\theta_\star := (\mathbf{H}_\star, \boldsymbol{\beta}_\star, \boldsymbol{\Gamma}_\star)$. If $\|\theta_\star\|_F$ is sufficiently small, then F is minimized locally at $(\mathbf{W}_\star, \theta')$ with*

$$\|\theta' - \theta_\star\|_F \leq \frac{3\max_{1\leq i\leq 4}(\lambda_i)}{\min_{1\leq i\leq 4}(\lambda_i - \lambda_{i\star})}\|\theta_\star\|_F. \tag{62}$$

If $\Lambda_2 > 0$ and $\xi \gg 1$, then for $\theta' := (\mathbf{W}', \boldsymbol{\beta}', \boldsymbol{\Gamma}')$ and $\theta_\star := (\mathbf{W}_\star, \boldsymbol{\beta}_\star, \boldsymbol{\Gamma}_\star)$, if $\|\theta_\star\|_F$ is sufficiently small, then F is minimized locally at $(\theta', \mathbf{H}_\star)$ with (62).

For each $s = 1, \dots, n$, let $\mathbf{e}_s$ denote the sth standard basis vector in $\mathbb{R}^n$.

Proposition C.7. *Let $f(\mathbf{Z})$ denote the objective of SMF-$\mathbf{H}$ in (30). Suppose Assumptions 4.1, 4.2, and B.1 hold. Let $\mathbf{a}_s := \boldsymbol{\beta}^T\mathbf{H}[:, s] + \boldsymbol{\Gamma}^T\mathbf{x}'_s$ for $s = 1, \dots, n$. Then*

$$\begin{cases} \nabla_{\text{vec}(\mathbf{H})}\,\ell(y_s, \mathbf{a}_s) & = (\mathbf{e}_s \otimes \boldsymbol{\beta})\,\dot{\mathbf{h}}(y_s, \mathbf{a}_s), \\ \nabla_{\text{vec}(\boldsymbol{\beta})}\,\ell(y_s, \mathbf{a}_s) & = \mathbf{C}^{(\kappa,r)}(\mathbf{H}[:, s] \otimes \mathbf{I}_\kappa)\,\dot{\mathbf{h}}(y_s, \mathbf{a}_s), \\ \nabla_{\text{vec}(\boldsymbol{\Gamma})}\,\ell(y_s, \mathbf{a}_s) & = \mathbf{C}^{(\kappa,q)}(\mathbf{x}'_s \otimes \mathbf{I}_\kappa)\,\dot{\mathbf{h}}(y_s, \mathbf{a}_s), \end{cases} \quad \begin{cases} \nabla_{\text{vec}(\mathbf{H})}\,\text{vec}(\mathbf{K})^T & = (\mathbf{I}_n \otimes \boldsymbol{\beta})\mathbf{M}, \\ \nabla_{\text{vec}(\boldsymbol{\beta})}\,\text{vec}(\mathbf{K})^T & = (\mathbf{I}_\kappa \otimes \mathbf{H})\mathbf{C}^{(\kappa,n)}\,\mathbf{M}, \\ \nabla_{\text{vec}(\boldsymbol{\Gamma})}\,\text{vec}(\mathbf{K})^T & = (\mathbf{I}_\kappa \otimes \mathbf{X}_{\text{aux}})\mathbf{C}^{(\kappa,n)}\,\mathbf{M}. \end{cases} \tag{63}$$

Proof. We first show

$$\begin{cases} \nabla_{\text{vec}(\mathbf{H})}\mathbf{a}_s^T & = \mathbf{e}_s \otimes \boldsymbol{\beta}, \\ \nabla_{\text{vec}(\boldsymbol{\beta})}\mathbf{a}_s^T & = \mathbf{C}^{(\kappa,r)}(\mathbf{H}[:, s] \otimes \mathbf{I}_\kappa), \\ \nabla_{\text{vec}(\boldsymbol{\Gamma})}\mathbf{a}_s^T & = \mathbf{C}^{(\kappa,q)}(\mathbf{x}'_s \otimes \mathbf{I}_\kappa). \end{cases} \tag{64}$$

For differentiating $\mathbf{a}_s$ by $\mathrm{vec}(\mathbf{H})$, observe that by using (22), we can write

$$\mathbf{a}_s = \mathrm{vec}(\mathbf{a}_s) = \boldsymbol{\beta}^T\mathbf{H}[:,s] + \mathrm{vec}\left(\boldsymbol{\Gamma}^T\mathbf{x}'_s\right) = (\mathbf{x}_s^T\otimes\boldsymbol{\beta}^T)\,\mathrm{vec}(\mathbf{W}^T) + \mathrm{vec}(\boldsymbol{\Gamma}^T\mathbf{x}'_s).$$

Noting that $\mathrm{vec}(\mathbf{W}^T)^T = (\mathbf{C}^{(p,r)}\,\mathrm{vec}(\mathbf{W}))^T = \mathrm{vec}(\mathbf{W})^T\mathbf{C}^{(r,p)}$,

$$\nabla_{\mathrm{vec}(\mathbf{H})}\mathbf{a}_s^T = \nabla_{\mathrm{vec}(\mathbf{H})}\mathbf{H}[:,s]^T\boldsymbol{\beta} = \mathbf{e}_s\otimes\boldsymbol{\beta}.$$

For differentiating $\mathbf{a}_s$ by $\mathrm{vec}(\boldsymbol{\beta})$, writing $\mathbf{a}_s = (\mathbf{H}[:,s]^T\otimes\mathbf{I}_\kappa)\,\mathrm{vec}(\boldsymbol{\beta}^T) + \mathrm{vec}(\boldsymbol{\Gamma}^T\mathbf{x}'_s)$, we get

$$\begin{aligned}\nabla_{\mathrm{vec}(\boldsymbol{\beta})}\mathbf{a}_s^T &= \nabla_{\mathrm{vec}(\boldsymbol{\beta})}\,\mathrm{vec}(\boldsymbol{\beta}^T)^T(\mathbf{H}[:,s]\otimes\mathbf{I}_\kappa)\\ &= \nabla_{\mathrm{vec}(\boldsymbol{\beta})}\,\mathrm{vec}(\boldsymbol{\beta}^T)\mathbf{C}^{(\kappa,r)}(\mathbf{H}[:,s]\otimes\mathbf{I}_\kappa)\\ &= \mathbf{C}^{(\kappa,r)}(\mathbf{H}[:,s]\otimes\mathbf{I}_\kappa).\end{aligned}$$

For differentiating $\mathbf{a}_s$ by $\mathrm{vec}(\boldsymbol{\Gamma})$, writing $\mathbf{a}_s = \mathrm{vec}(\boldsymbol{\beta}^T\mathbf{H}[:,s]) + ((\mathbf{x}'_s)^T\otimes\mathbf{I}_\kappa)\,\mathrm{vec}(\boldsymbol{\Gamma}^T)$, we get

$$\begin{aligned}\nabla_{\mathrm{vec}(\boldsymbol{\Gamma})}\mathbf{a}_s^T &= \nabla_{\mathrm{vec}(\boldsymbol{\Gamma})}\,\mathrm{vec}(\boldsymbol{\Gamma}^T)^T(\mathbf{x}'_s\otimes\mathbf{I}_n)\\ &= \nabla_{\mathrm{vec}(\boldsymbol{\Gamma})}\,\mathrm{vec}(\boldsymbol{\Gamma})^T\mathbf{C}^{(\kappa,q)}(\mathbf{x}'_s\otimes\mathbf{I}_\kappa)\\ &= \mathbf{C}^{(\kappa,q)}(\mathbf{x}'_s\otimes\mathbf{I}_\kappa).\end{aligned}$$

This verifies (64). Then by using the chain rule (24), we get

$$\nabla_{\mathrm{vec}(\mathbf{H})}\ell(y_s,\mathbf{a}_s) = \nabla_{\mathrm{vec}(\mathbf{H})}\mathbf{a}_s^T\ \nabla_{\mathbf{a}_s}\ell(y_s,\mathbf{a}_s) = (\mathbf{e}_s\otimes\boldsymbol{\beta})\,\dot{\mathbf{h}}(y_s,\mathbf{a}_s).$$

The other gradients $\nabla_{\mathrm{vec}(\boldsymbol{\beta})}\,\ell(y_s,\mathbf{a}_s)$ and $\nabla_{\mathrm{vec}(\boldsymbol{\Gamma})}\,\ell(y_s,\mathbf{a}_s)$ also follows from (64) and the chain rule.

Next, we compute the gradients of $\mathrm{vec}(\mathbf{K})^T$ in (63). First, using (64), the chain rule (24), and (32),

$$\nabla_{\mathrm{vec}(\mathbf{H})}\,\dot{\mathbf{h}}(y_s,\mathbf{a}_s)^T = \nabla_{\mathrm{vec}(\mathbf{W})}\mathbf{a}_s^T\ \nabla_{\mathbf{a}_s}\dot{\mathbf{h}}(y_s,\mathbf{a}_s)^T = (\mathbf{e}_s\otimes\boldsymbol{\beta})\ddot{\mathbf{H}}(y_s,\mathbf{a}_s), \tag{65}$$

$$\nabla_{\mathrm{vec}(\boldsymbol{\beta})}\,\dot{\mathbf{h}}(y_s,\mathbf{a}_s)^T = \nabla_{\mathrm{vec}(\boldsymbol{\beta})}\mathbf{a}_s^T\ \nabla_{\mathbf{a}_s}\dot{\mathbf{h}}(y_s,\mathbf{a}_s)^T = \mathbf{C}^{(\kappa,r)}(\mathbf{H}[:,s]\otimes\mathbf{I}_\kappa)\ddot{\mathbf{H}}(y_s,\mathbf{a}_s), \tag{66}$$

$$\nabla_{\mathrm{vec}(\boldsymbol{\Gamma})}\,\dot{\mathbf{h}}(y_s,\mathbf{a}_s)^T = \nabla_{\mathrm{vec}(\boldsymbol{\Gamma})}\mathbf{a}_s^T\ \nabla_{\mathbf{a}_s}\dot{\mathbf{h}}(y_s,\mathbf{a}_s)^T = \mathbf{C}^{(\kappa,q)}(\mathbf{x}'_s\otimes\mathbf{I}_\kappa)\ddot{\mathbf{H}}(y_s,\mathbf{a}_s). \tag{67}$$

Now since $\mathrm{vec}(\mathbf{K})^T = [\dot{\mathbf{h}}(y_1,\mathbf{a}_1)^T,\dots,\dot{\mathbf{h}}(y_n,\mathbf{a}_n)^T]$ and $\mathrm{vec}(\mathbf{K}^T)^T = (\mathbf{C}^{(\kappa,n)}\,\mathrm{vec}(\mathbf{K}))^T = \mathrm{vec}(\mathbf{K})^T\mathbf{C}^{(n,\kappa)}$, it follows that

$$\nabla_{\mathrm{vec}(\mathbf{H})}\,\mathrm{vec}(\mathbf{K})^T \overset{(a)}{=} \Big[(\mathbf{e}_1\otimes\boldsymbol{\beta})\ddot{\mathbf{H}}(y_1,\mathbf{a}_1),\dots,(\mathbf{e}_n\otimes\boldsymbol{\beta})\ddot{\mathbf{H}}(y_n,\mathbf{a}_n)\Big] \tag{68}$$

$$\overset{(b)}{=} [\mathbf{e}_1\otimes\boldsymbol{\beta},\dots,\mathbf{e}_n\otimes\boldsymbol{\beta}]\,\mathrm{diag}\left(\ddot{\mathbf{H}}(y_1,\mathbf{a}_1),\dots,\ddot{\mathbf{H}}(y_n,\mathbf{a}_n)\right) \tag{69}$$

$$\overset{(c)}{=} (\mathbf{I}_n\otimes\boldsymbol{\beta})\mathbf{M}, \tag{70}$$

where (a) follows from (64) and the chain rule, (b) is an algebra, (c) follows from (19), The other gradients $\nabla_{\mathrm{vec}(\boldsymbol{\beta})}\,\mathrm{vec}(\mathbf{K})^T$ and $\nabla_{\mathrm{vec}(\boldsymbol{\Gamma})}\,\mathrm{vec}(\mathbf{K})^T$ follow from similar computations. □

Lemma C.8 (Derivatives of the SMF-$\mathbf{H}$ objective)**.** *Let $f(\mathbf{Z})$ denote the objective of SMF-$\mathbf{H}$ in (30). Suppose Assumption B.1 holds. Recall $\dot{\mathbf{h}}$ and $\ddot{\mathbf{H}}$ defined in (33). Then the gradients of $f(\mathbf{Z})$ are given by*

$$\nabla_{\mathbf{W}}\,f(\mathbf{Z}) = 2\xi(\mathbf{W}\mathbf{H}-\mathbf{X})\mathbf{H}^T, \tag{71}$$

$$\nabla_{\mathbf{H}}\,f(\mathbf{Z}) = \boldsymbol{\beta}\mathbf{K} + 2\xi\mathbf{W}^T(\mathbf{W}\mathbf{H}-\mathbf{X}), \tag{72}$$

$$\nabla_{\boldsymbol{\beta}}\,f(\mathbf{Z}) = \mathbf{H}\mathbf{K}^T \tag{73}$$

$$\nabla_{\boldsymbol{\Gamma}}\,f(\mathbf{Z}) = \mathbf{X}_{\text{aux}}\mathbf{K}^T. \tag{74}$$

The block-diagonal terms in the Hessian are given by

$$\nabla_{\text{vec}(\mathbf{W})}\nabla_{\text{vec}(\mathbf{W})^T} f(\mathbf{Z}) = 2\xi(\mathbf{I}_p \otimes \mathbf{H}\mathbf{H}^T), \tag{75}$$

$$\nabla_{\text{vec}(\mathbf{H})}\nabla_{\text{vec}(\mathbf{H})^T} f(\mathbf{Z}) = (\mathbf{I}_n \otimes \boldsymbol{\beta})\mathbf{M}(\mathbf{I}_n \otimes \boldsymbol{\beta})^T + 2\xi(\mathbf{I}_n \otimes \mathbf{W}^T\mathbf{W}), \tag{76}$$

$$\nabla_{\text{vec}(\boldsymbol{\beta})}\nabla_{\text{vec}(\boldsymbol{\beta})^T} f(\mathbf{Z}) = (\mathbf{I}_\kappa \otimes \mathbf{H})\mathbf{C}^{(\kappa,n)}\mathbf{M}\mathbf{C}^{(n,\kappa)}(\mathbf{I}_\kappa \otimes \mathbf{H})^T, \tag{77}$$

$$\nabla_{\text{vec}(\boldsymbol{\Gamma})}\nabla_{\text{vec}(\boldsymbol{\Gamma})^T} f(\mathbf{Z}) = (\mathbf{I}_\kappa \otimes \mathbf{X}_{\text{aux}})\mathbf{C}^{(\kappa,n)}\mathbf{M}\mathbf{C}^{(n,\kappa)}(\mathbf{I}_\kappa \otimes \mathbf{X}_{\text{aux}})^T. \tag{78}$$

The block-off-diagonal terms in the Hessian are given by

$$\nabla_{\text{vec}(\mathbf{H})}\nabla_{\text{vec}(\mathbf{W})^T} f(\mathbf{Z}) = 2\xi\mathbf{C}^{(n,r)}\left[(\mathbf{H}^T \otimes \mathbf{W}^T) + \mathbf{I}_r \otimes (\mathbf{W}\mathbf{H} - \mathbf{X})^T\right] \tag{79}$$

$$\nabla_{\text{vec}(\boldsymbol{\beta})}\nabla_{\text{vec}(\mathbf{W})^T} f(\mathbf{Z}) = \nabla_{\text{vec}(\boldsymbol{\Gamma})}\nabla_{\text{vec}(\mathbf{W})^T} f(\mathbf{Z}) = \mathbf{O}, \tag{80}$$

$$\nabla_{\text{vec}(\boldsymbol{\beta})}\nabla_{\text{vec}(\mathbf{H})^T} f(\mathbf{Z}) = (\mathbf{K} \otimes \mathbf{I}_r) + (\mathbf{I}_\kappa \otimes \mathbf{H})\mathbf{C}^{(\kappa,n)}\,\mathbf{M}(\mathbf{I}_n \otimes \boldsymbol{\beta})^T, \tag{81}$$

$$\nabla_{\text{vec}(\boldsymbol{\Gamma})}\nabla_{\text{vec}(\mathbf{H})^T} f(\mathbf{Z}) = (\mathbf{I}_\kappa \otimes \mathbf{X}_{\text{aux}})\mathbf{C}^{(\kappa,n)}\,\mathbf{M}(\mathbf{I}_n \otimes \boldsymbol{\beta})^T, \tag{82}$$

$$\nabla_{\text{vec}(\boldsymbol{\Gamma})}\nabla_{\text{vec}(\boldsymbol{\beta})^T} f(\mathbf{Z}) = (\mathbf{I}_\kappa \otimes \mathbf{X}_{\text{aux}})\mathbf{C}^{(\kappa,n)}\,\mathbf{M}(\mathbf{I}_\kappa \otimes \mathbf{H}). \tag{83}$$

Proof. For convenience, recall that $\mathbf{W} \in \mathbb{R}^{p\times r}$, $\boldsymbol{\beta} \in \mathbb{R}^{r\times\kappa}$, $\mathbf{H} \in \mathbb{R}^{r\times n}$, and $\boldsymbol{\Gamma} \in \mathbb{R}^{q\times\kappa}$.

Computation of the first-order derivatives.

We first compute the following gradient:

$$\nabla_{\text{vec}(\mathbf{H})} \sum_{s=1}^{n} \ell(y_s, \mathbf{a}_s) \overset{(a)}{=} \sum_{s=1}^{n} (\mathbf{e}_s \otimes \boldsymbol{\beta})\, \dot{\mathbf{h}}(y_s, \mathbf{a}_s) \tag{84}$$

$$= [\mathbf{e}_1 \otimes \boldsymbol{\beta}, \dots, \mathbf{e}_n \otimes \boldsymbol{\beta}] \begin{bmatrix} \dot{\mathbf{h}}(y_1, \mathbf{a}_1) \\ \vdots \\ \dot{\mathbf{h}}(y_n, \mathbf{a}_n) \end{bmatrix} \tag{85}$$

$$\overset{(b)}{=} (\mathbf{I}_n \otimes \boldsymbol{\beta})\, \text{vec}(\mathbf{K}), \tag{86}$$

where (a) follows from Proposition C.1, (b) follows from (19), (c) follows from (20), and (d) uses the definition of the commutation matrices. Then by using (22), we deduce

$$\nabla_{\mathbf{H}} f(\mathbf{Z}) = \boldsymbol{\beta}\mathbf{K} + 2\xi\mathbf{W}^T(\mathbf{W}\mathbf{H} - \mathbf{X}). \tag{87}$$

Next, we compute $\nabla_{\text{vec}(\boldsymbol{\beta})} f(\mathbf{Z})$. By using similar computations as before, we get

$$\begin{aligned}
\nabla_{\text{vec}(\boldsymbol{\beta})} \sum_{s=1}^{n} \ell(y_s, \mathbf{a}_s) &= \mathbf{C}^{(\kappa,r)} \sum_{s=1}^{n} (\mathbf{H}[:, s] \otimes \mathbf{I}_\kappa)\, \dot{\mathbf{h}}(y_s, \mathbf{a}_s) \\
&= \mathbf{C}^{(r,p)}\, [\mathbf{H}[:, 1] \otimes \mathbf{I}_\kappa, \dots, \mathbf{H}[:, n] \otimes \mathbf{I}_\kappa] \begin{bmatrix} \dot{\mathbf{h}}(y_1, \mathbf{a}_1) \\ \vdots \\ \dot{\mathbf{h}}(y_n, \mathbf{a}_n) \end{bmatrix} \\
&= \mathbf{C}^{(r,p)}(\mathbf{H} \otimes \mathbf{I}_\kappa)\, \text{vec}(\mathbf{K}) \\
&= (\mathbf{I}_\kappa \otimes \mathbf{H})\mathbf{C}^{(\kappa,n)}\, \text{vec}(\mathbf{K}) \\
&= (\mathbf{I}_\kappa \otimes \mathbf{H})\, \text{vec}(\mathbf{K}^T).
\end{aligned}$$

From this and (22), we deduce

$$\nabla_{\boldsymbol{\beta}} f(\mathbf{Z}) = \mathbf{H}\mathbf{K}^T. \tag{88}$$

That $\nabla_{\text{vec}(\Gamma)} f(\mathbf{Z}) = \mathbf{X}_{\text{aux}}\mathbf{K}^T$ as in the proof of Lemma C.2. The last derivative $\nabla_{\mathbf{W}} f(\mathbf{Z}) = 2\xi(\mathbf{W}\mathbf{H} - \mathbf{X})\mathbf{H}^T$ is easy.

Computation of the second-order derivatives.

By vectorizing $\nabla_{\mathbf{H}} f(\mathbf{Z})$ in (71), we get

$$\begin{aligned}\nabla_{\mathrm{vec}(\mathbf{H})} f(\mathbf{Z}) &= \mathrm{vec}(\boldsymbol{\beta}\mathbf{K}) + 2\xi\, \mathrm{vec}(\mathbf{W}^T\mathbf{W}\mathbf{H}) - \mathrm{vec}(\mathbf{W}^T\mathbf{X}) \\ &= (\mathbf{I}_n \otimes \boldsymbol{\beta})\, \mathrm{vec}(\mathbf{K}) + 2\xi(\mathbf{I}_n \otimes \mathbf{W}^T\mathbf{W})\, \mathrm{vec}(\mathbf{H}) - \mathrm{vec}(\mathbf{W}^T\mathbf{X}).\end{aligned}$$

Then using Proposition C.1 with (89), we get

$$\begin{aligned}&\nabla_{\mathrm{vec}(\mathbf{H})} \nabla_{\mathrm{vec}(\mathbf{H})^T} f(\mathbf{Z}) \\ &\quad= \nabla_{\mathrm{vec}(\mathbf{H})} \left(\mathrm{vec}(\mathbf{K})^T (\mathbf{I}_n \otimes \boldsymbol{\beta})^T + 2\xi\, \mathrm{vec}(\mathbf{H})^T (\mathbf{I}_n \otimes \mathbf{W}^T\mathbf{W})^T \right) \\ &\quad= (\mathbf{I}_n \otimes \boldsymbol{\beta})\mathbf{M}(\mathbf{I}_n \otimes \boldsymbol{\beta})^T + 2\xi(\mathbf{I}_n \otimes \mathbf{W}^T\mathbf{W}).\end{aligned}$$

Similarly, we can compute

$$\begin{aligned}&\nabla_{\mathrm{vec}(\boldsymbol{\beta})} \nabla_{\mathrm{vec}(\boldsymbol{\beta})^T} f(\mathbf{Z}) \\ &\quad= \nabla_{\mathrm{vec}(\boldsymbol{\beta})}\, \mathrm{vec}(\mathbf{H}\mathbf{K})^T \\ &\quad= \nabla_{\mathrm{vec}(\boldsymbol{\beta})}\, \mathrm{vec}(\mathbf{K}^T)^T (\mathbf{I}_\kappa \otimes \mathbf{H})^T \\ &\quad= \nabla_{\mathrm{vec}(\boldsymbol{\beta})}\, \mathrm{vec}(\mathbf{K})^T \mathbf{C}^{(n,\kappa)} (\mathbf{I}_\kappa \otimes \mathbf{H})^T \\ &\quad= (\mathbf{I}_\kappa \otimes \mathbf{H})\mathbf{C}^{(\kappa,n)}\mathbf{M}\mathbf{C}^{(n,\kappa)}(\mathbf{I}_\kappa \otimes \mathbf{H})^T.\end{aligned}$$

Also, note that

$$\begin{aligned}\nabla_{\mathrm{vec}(\boldsymbol{\Gamma})} \nabla_{\mathrm{vec}(\boldsymbol{\Gamma})^T} f(\mathbf{Z}) &= \nabla_{\mathrm{vec}(\boldsymbol{\Gamma})}\, \mathrm{vec}(\mathbf{X}_{\text{aux}}\mathbf{K}^T)^T \\ &= \nabla_{\mathrm{vec}(\boldsymbol{\Gamma})}\, \mathrm{vec}(\mathbf{K})^T \mathbf{C}^{(n,\kappa)} (\mathbf{I}_\kappa \otimes \mathbf{X}_{\text{aux}})^T \\ &= (\mathbf{I}_\kappa \otimes \mathbf{X}_{\text{aux}})\mathbf{C}^{(\kappa,n)}\mathbf{M}\mathbf{C}^{(n,\kappa)}(\mathbf{I}_\kappa \otimes \mathbf{X}_{\text{aux}})^T.\end{aligned}$$

Similarly, we get

$$\begin{aligned}\nabla_{\mathrm{vec}(\mathbf{W})} \nabla_{\mathrm{vec}(\mathbf{W})^T} f(\mathbf{Z}) &= \nabla_{\mathrm{vec}(\mathbf{W})} \left(2\xi\, \mathrm{vec}(\mathbf{W}\mathbf{H}\mathbf{H}^T)^T - 2\xi\, \mathrm{vec}(\mathbf{X}\mathbf{H}^T)^T \right) \\ &= 2\xi \nabla_{\mathrm{vec}(\mathbf{H})}\, \mathrm{vec}(\mathbf{H})^T (\mathbf{I}_p \otimes \mathbf{H}\mathbf{H}^T) \\ &= 2\xi(\mathbf{I}_p \otimes \mathbf{H}\mathbf{H}^T).\end{aligned}$$

Next, we compute the off-diagonal block terms in the Hessian of f. Recall that from (84) and (22), we have

$$\mathrm{vec}(\boldsymbol{\beta}\mathbf{K}) = (\mathbf{I}_n \otimes \boldsymbol{\beta})\, \mathrm{vec}(\mathbf{K}) = (\mathbf{K}^T \otimes \mathbf{I}_r)\, \mathrm{vec}(\boldsymbol{\beta}).$$

Then using the product rule, we get

$$\begin{aligned}\nabla_{\mathrm{vec}(\boldsymbol{\beta})} \nabla_{\mathrm{vec}(\mathbf{H})^T} f(\mathbf{Z}) &= \nabla_{\mathrm{vec}(\boldsymbol{\beta})}\, \mathrm{vec}(\boldsymbol{\beta}\mathbf{K})^T \\ &= \left(\nabla_{\mathrm{vec}(\boldsymbol{\beta})}\, \mathrm{vec}(\boldsymbol{\beta})^T\right) (\mathbf{K}^T \otimes \mathbf{I}_r)^T + \left(\nabla_{\mathrm{vec}(\boldsymbol{\beta})}\, \mathrm{vec}(\mathbf{K})^T\right) (\mathbf{I}_n \otimes \boldsymbol{\beta})^T \\ &= (\mathbf{K} \otimes \mathbf{I}_r) + (\mathbf{I}_\kappa \otimes \mathbf{H})\mathbf{C}^{(\kappa,n)}\, \mathbf{M}(\mathbf{I}_n \otimes \boldsymbol{\beta})^T.\end{aligned}$$

Second, note that

$$\begin{aligned}\nabla_{\mathrm{vec}(\boldsymbol{\Gamma})} \nabla_{\mathrm{vec}(\mathbf{H})^T} f(\mathbf{Z}) &= \nabla_{\mathrm{vec}(\boldsymbol{\Gamma})}\, \mathrm{vec}(\boldsymbol{\beta}\mathbf{K})^T \\ &= \nabla_{\mathrm{vec}(\boldsymbol{\Gamma})}\, \mathrm{vec}(\mathbf{K})^T (\mathbf{I}_n \otimes \boldsymbol{\beta})^T \\ &= (\mathbf{I}_\kappa \otimes \mathbf{X}_{\text{aux}})\mathbf{C}^{(\kappa,n)}\, \mathbf{M}(\mathbf{I}_n \otimes \boldsymbol{\beta})^T.\end{aligned}$$

Second, note that $\nabla_{\mathrm{vec}(\boldsymbol{\Gamma})} \nabla_{\mathrm{vec}(\mathbf{H})^T} f(\mathbf{Z}) = O$. Third, by the same computation as in the proof of Lemma C.2,

$$\begin{aligned}\nabla_{\mathrm{vec}(\mathbf{H})} \nabla_{\mathrm{vec}(\mathbf{W})^T} f(\mathbf{Z}) &= 2\xi \nabla_{\mathrm{vec}(\mathbf{H})} \left[\mathrm{vec}(\mathbf{W}\mathbf{H}\mathbf{H}^T) - \mathrm{vec}(\mathbf{X}\mathbf{H}^T)\right]^T \\ &= 2\xi \mathbf{C}^{(n,r)} \left[\left((\mathbf{H}^T \otimes \mathbf{W}^T) + (\mathbf{I}_r \otimes \mathbf{H}^T\mathbf{W}^T)\right) - (\mathbf{I}_r \otimes \mathbf{X})^T\right].\end{aligned}$$

Fourth, observe that $(\mathbf{I}_n \otimes \boldsymbol{\beta})\,\mathrm{vec}(\mathbf{K})$

$$\begin{aligned}\nabla_{\mathrm{vec}(\boldsymbol{\Gamma})}\nabla_{\mathrm{vec}(\boldsymbol{\beta})^T} f(\mathbf{Z}) &= \nabla_{\mathrm{vec}(\boldsymbol{\Gamma})}\,\mathrm{vec}(\mathbf{HK})^T \\ &= \nabla_{\mathrm{vec}(\boldsymbol{\Gamma})}\,\mathrm{vec}(\mathbf{K})^T(\mathbf{I}_\kappa \otimes \mathbf{H})^T \\ &= (\mathbf{I}_\kappa \otimes \mathbf{X}_{\text{aux}})\mathbf{C}^{(\kappa,n)}\,\mathbf{M}(\mathbf{I}_\kappa \otimes \mathbf{H}).\end{aligned}$$

The remaining zero-second derivatives are easy to see. □

Lemma C.9. *Let $f(\mathbf{Z})$ denote the objective of SMF-$\mathbf{H}$ in (30). Suppose Assumption B.1 holds. Recall $\dot{\mathbf{h}}$ and $\ddot{\mathbf{H}}$ defined in (33). Then the following hold:*

(i) *Write the Hessian $\nabla^2 f(\mathbf{Z})$ as the 4×4 block matrix $(A_{ij})_{1\le i,j\le 4}$. Then*

$$A_{11} = 2\xi(\mathbf{I}_p \otimes \mathbf{HH}^T) \tag{89}$$
$$\alpha^-(\mathbf{I}_n \otimes \boldsymbol{\beta}\boldsymbol{\beta}^T) + 2\xi(\mathbf{I}_n \otimes \mathbf{W}^T\mathbf{W}) \preceq A_{22} \preceq \alpha^+(\mathbf{I}_n \otimes \boldsymbol{\beta}\boldsymbol{\beta}^T) + 2\xi(\mathbf{I}_n \otimes \mathbf{W}^T\mathbf{W}) \tag{90}$$
$$\alpha^-(\mathbf{I}_\kappa \otimes \mathbf{HH}^T) \preceq A_{33} \preceq \alpha^+(\mathbf{I}_\kappa \otimes \mathbf{HH}^T), \tag{91}$$
$$\alpha^-(\mathbf{I}_\kappa \otimes \mathbf{X}_{\text{aux}}\mathbf{X}_{\text{aux}}^T) \preceq A_{44} \preceq \alpha^+(\mathbf{I}_\kappa \otimes \mathbf{X}_{\text{aux}}\mathbf{X}_{\text{aux}}^T). \tag{92}$$

(ii) *The function $f(\mathbf{Z}) = f(\mathbf{W},\mathbf{H},\boldsymbol{\beta},\boldsymbol{\Gamma})$ restricted to each block coordinate has Lipschitz-continuous gradients with Lipschitz constants $L_{\mathbf{W}}, L_{\mathbf{H}}, L_{\boldsymbol{\beta}}, L_{\boldsymbol{\Gamma}}$ given by*

$$L_{\mathbf{W}} := 2\xi\|\mathbf{H}\|_2^2, \tag{93}$$
$$L_{\mathbf{H}} := \alpha^+\|\boldsymbol{\beta}\|_2^2 + 2\xi\|\mathbf{W}\|_2^2, \tag{94}$$
$$L_{\boldsymbol{\beta}} := \alpha^+\|\mathbf{H}\|_2^2, \tag{95}$$
$$L_{\boldsymbol{\Gamma}} := \alpha^+\|\mathbf{X}_{\text{aux}}\|_2^2. \tag{96}$$

(iii) *The Hessian of the L_2-regularized objective $f(\mathbf{Z}) + \frac{\lambda_{\mathbf{W}}}{2}\|\mathbf{W}\|_F^2 + \frac{\lambda_{\mathbf{H}}}{2}\|\mathbf{H}\|_F^2 + \frac{\lambda_{\boldsymbol{\beta}}}{2}\|\boldsymbol{\beta}\|_F^2 + \frac{\lambda_{\boldsymbol{\Gamma}}}{2}\|\boldsymbol{\Gamma}\|_F^2$ is positive definite if*

$$\lambda_{\mathbf{W}} > 2\xi\left(\|\mathbf{H}\|_2\cdot\|\mathbf{W}\|_2 + \|\mathbf{WH}-\mathbf{X}\|_2 - \lambda_{\min}(\mathbf{HH}^T)\right) \tag{97}$$
$$\lambda_{\mathbf{H}} > 2\xi\left(\|\mathbf{H}\|_2\cdot\|\mathbf{W}\|_2 + \|\mathbf{WH}-\mathbf{X}\|_2 - \lambda_{\min}(\mathbf{W}^T\mathbf{W})\right) + \gamma_{\max}\sqrt{\kappa n} \tag{98}$$
$$+\,\alpha^+\|\boldsymbol{\beta}\|_2\left(\|\mathbf{H}\|_2 + \|\mathbf{X}_{\text{aux}}\|_2\right) - \alpha^-\lambda_{\min}(\boldsymbol{\beta}\boldsymbol{\beta}^T), \tag{99}$$
$$\lambda_{\boldsymbol{\beta}} > \gamma_{\max}\sqrt{\kappa n} + \alpha^+\|\boldsymbol{\beta}\|_2\left(\|\mathbf{H}\|_2 + \|\mathbf{X}_{\text{aux}}\|_2\right) - \alpha^-\lambda_{\min}(\mathbf{HH}^T), \tag{100}$$
$$\lambda_{\boldsymbol{\Gamma}} > \alpha^+\|\mathbf{X}_{\text{aux}}\|_2\left(\|\boldsymbol{\beta}\|_2 + \|\mathbf{H}\|_2\right) - \alpha^-\lambda_{\min}(\mathbf{X}_{\text{aux}}\mathbf{X}_{\text{aux}}^T). \tag{101}$$

Proof. Observe that the block-diagonal matrix $\mathbf{M}$ in (35) is symmetric by definition and is also positive definite by Assumption B.1:

$$0 < \alpha^- \le \lambda_{\min}(\mathbf{M}) \le \lambda_{\max}(\mathbf{M}) \le \alpha^+. \tag{102}$$

Since the commutation matrices are orthogonal and satisfies $\mathbf{C}^{(a,b)}\mathbf{C}^{(b,a)} = \mathbf{I}_{ab}$, it follows that

$$\alpha^-\mathbf{I}_{\kappa n} \preceq \mathbf{C}^{(\kappa,n)}\mathbf{M}\mathbf{C}^{(n,\kappa)} \preceq \alpha^+\mathbf{I}_{\kappa n}. \tag{103}$$

Then the first Loewner ordering for $A_{11} = \nabla_{\mathrm{vec}(\mathbf{W})}\nabla_{\mathrm{vec}(\mathbf{W})^T} f(\mathbf{Z})$ follows from Lemma C.2. The other Loewner orderings can be shown similarly. This shows **(i)**.

(ii) follows immediately from **(i)** and the fact that the Lipschitz constant for the gradient is upper-bounded by the largest eigenvalue of the corresponding block Hessian, which are the diagonal blocks A_{ii} for $i = 1,\dots,4$.

For **(iii)**, note that if L_2-regularization coefficients are large enough so that the following condition is satisfied

$$\lambda_{\min}(A_{ii}) + \lambda_i > \sum_{j\ne i}\|A_{ij}\|_2 \quad \forall 1 \le i \le 4, \tag{104}$$

where $\lambda_1 = \lambda_{\mathbf{W}}$, $\lambda_2 = \lambda_{\mathbf{H}}$, $\lambda_3 = \lambda_{\boldsymbol{\beta}}$, and $\lambda_4 = \lambda_{\boldsymbol{\Gamma}}$, then the L_2-regularized Hessian of the objective f is block diagonally dominant and is positive definite (see (Feingold & Varga, 1962)). The L_2-regularized Hessian takes the following 4×4 block form:

$$\begin{array}{c|cccc} & \mathrm{vec}(\mathbf{W})^T & \mathrm{vec}\,\mathbf{H}^T & \mathrm{vec}(\boldsymbol{\beta})^T & \mathrm{vec}(\boldsymbol{\Gamma})^T \\ \hline \mathrm{vec}(\mathbf{W}) & A_{11} + \lambda_{\mathbf{W}} \mathbf{I}_{rp} & A_{12} & \mathbf{O} & \mathbf{O} \\ \mathrm{vec}(\mathbf{H}) & A_{21} & A_{22} + \lambda_{\mathbf{H}} \mathbf{I}_{rn} & A_{23} & A_{24} \\ \mathrm{vec}(\boldsymbol{\beta}) & \mathbf{O} & A_{32} & A_{33} + \lambda_{\boldsymbol{\beta}} \mathbf{I}_{r\kappa} & A_{34} \\ \mathrm{vec}(\boldsymbol{\Gamma}) & \mathbf{O} & A_{42} & A_{43} & A_{44} + \lambda_{\boldsymbol{\Gamma}} \mathbf{I}_{q\kappa} \end{array} \tag{105}$$

Thus it suffices to take

$$\lambda_{\mathbf{W}} > \|A_{12}\|_2 - \lambda_{\min}(A_{11}), \tag{106}$$
$$\lambda_{\mathbf{H}} > \|A_{12}\|_2 + \|A_{23}\|_2 + \|A_{24}\|_2 - \lambda_{\min}(A_{22}), \tag{107}$$
$$\lambda_{\boldsymbol{\beta}} > \|A_{23}\|_2 + \|A_{34}\|_2 - \lambda_{\min}(A_{33}), \tag{108}$$
$$\lambda_{\boldsymbol{\Gamma}} > \|A_{34}\|_2 + \|A_{24}\|_2 - \lambda_{\min}(A_{44}). \tag{109}$$

Using Assumption B.1, we can upper bound the operator norm of the off-diagonal blocks as

$$\|A_{12}\|_2 \le 2\xi \left(\|\mathbf{H}\|_2 \cdot \|\mathbf{W}\|_2 + \|\mathbf{WH} - \mathbf{X}\|_2 \right) \tag{110}$$
$$\|A_{23}\|_2 \le \gamma_{\max} \sqrt{\kappa n} + \alpha^+ \|\mathbf{H}\|_2 \cdot \|\boldsymbol{\beta}\|_2, \tag{111}$$
$$\|A_{24}\|_2 \le \alpha^+ \|\mathbf{X}_{\text{aux}}\|_2 \cdot \|\boldsymbol{\beta}\|_2, \tag{112}$$
$$\|A_{34}\|_2 \le \alpha^+ \|\mathbf{X}_{\text{aux}}\|_2 \cdot \|\mathbf{H}\|_2. \tag{113}$$

We can also obtain lower bounds on the eigenvalues of the diagonal blocks. Then the assertion in **(iii)** follows. □

We now prove Theorems C.6 and 4.4 for SMF-$\mathbf{H}$.

***Proof of Theorems C.6 and 4.4** for SMF-$\mathbf{H}$.* Theorem C.6 follows from Lemmas C.9 and C.5.

The proof of Theorem 4.4 for SMF-$\mathbf{H}$ again amounts to verify the hypothesis of Theorem 2.1 in (Lyu & Li, 2023) for the block projected gradient descent algorithm in Algorithm 1 as a BMM with suitable prox-linear surrogates. The argument is identical to that for SMF-$\mathbf{W}$ we provided in the previous section, together with the corresponding lemmas establishing gradient and Hessian computations for SMF-$\mathbf{H}$ (Lemmas C.8 and C.9). □

D. Proof of Theorem 4.5: A non-asymptotic local consistency of MLE

In this section, we provide a general result on the non-asymptotic local consistency of MLE in a general setting, where the data samples are assumed to be independent but may not be identically distributed, and the unknown true parameter used for a generative model may lie on the boundary of the parameter space and the Fisher information at the true parameter is not necessarily positive definite. The result we present (Theorem D.1) in this section is general and could be of independent interest. From this general result and Theorem 4.3 we can deduce Theorem 4.5.

Fix a sample size $n \ge 1$. Suppose π_θ is a probability distribution on $\mathbb{R}^d$ parameterized by $\theta \in \Theta \subseteq \mathbb{R}^p$. If an n-tuple $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_n]$ of vectors in $\mathbb{R}^d$ is observed under the product distribution $\pi_{\boldsymbol{\theta}} := \pi_{\theta_1} \otimes \cdots \otimes \pi_{\theta_n}$, $\boldsymbol{\theta} = (\theta_1, \dots, \theta_n)$, then the regularized negative log-likelihood of observing $\mathbf{X}$ under $\pi_{\boldsymbol{\theta}}$ is

$$\mathcal{L}(\mathbf{X}\,;\,\boldsymbol{\theta}) := \sum_{i=1}^{n} \left(\mathcal{L}_0(\mathbf{x}_i; \theta_i) + R(\theta_i) \right), \quad \mathcal{L}_0(\mathbf{x}; \theta) := -\log \pi_\theta(\mathbf{x}), \tag{114}$$

where $R(\theta_i)$ is a suitable choice of regularizer for parameter θ_i. Denote $R(\boldsymbol{\theta}) := \sum_{i=1}^{n} R(\theta_i)$. We denote

$$\mathcal{L}_0(\mathbf{X}\,;\,\boldsymbol{\theta}) := \sum_{i=1}^{n} \mathcal{L}_0(\mathbf{x}; \theta_i). \tag{115}$$

Now suppose there is true and unknown parameter $\boldsymbol{\theta}_\star = (\theta_{1\star}, \dots, \theta_{n\star})$ such that we have independent samples $\mathbf{x}_1, \dots, \mathbf{x}_n$ jointly from $\pi_{\boldsymbol{\theta}_\star}$. Let $\hat{\boldsymbol{\theta}}_n$ denote a (possibly non-unique) minimizer of the above function over the n-fold product parameter space Θ^n. This is a minimizer of the random loss function $\mathcal{L}$ over the product constraint set Θ^n, which we call the *constrained and regularized maximum likelihood estimator* (MLE) of $\boldsymbol{\theta}_\star$. Note that here we consider a general constrained MLE problem in three aspects: (1) The distribution of n data samples are parameterized separately by $\theta_1, \dots, \theta_n$; (2) The constraint set Θ^n may be a proper convex subset of $\mathbb{R}^{p\times n}$ and $\boldsymbol{\theta}_\star$ could be at the boundary of Θ^n; (3) The loss function $\mathcal{L}$ in (114) may be non-convex and may have multiple local minima.

In this general setting, we would like to provide a high-probability guarantee that there exists a local minimizer of (114) that is close to the true parameter $\boldsymbol{\theta}_\star$. In the special case where we impose $\theta_1 = \cdots = \theta_n$ and $\boldsymbol{\theta}_\star$ is assumed to be in the interior of Θ^n, this type of result is provided by the classical local consistency theory of MLE (Fan & Li, 2001) in an asymptotic setting where the sample size n tends to infinity. Below in Theorem D.1, we generalize such a classical result in the non-asymptotic, constrained, and regularized setting. For its proof, we combine a classical approach in (Fan & Li, 2001) with concentration inequalities, namely, a classical Berry-Esseen bound for deviations from standard normal distribution for independent but non-identically distributed random variables and a uniform McDirmid bound (Lemma D.2). The former is used to control the linear term in the second-order Taylor expansion of the log-likelihood function, and the latter is used to control the second-order term. By using an ε-net argument, the latter concentration inequality can be extended to a setting where the random variables are parameterized within a compact set.

Theorem D.1 (Non-asymptotic local consistency of constrained and regularized MLE)**.** *Consider the constrained and regularized MLE problem* (114) *with unknown parameters $\theta_{1\star}, \dots, \theta_{n\star}$ from a convex subset $\Theta \subseteq \mathbb{R}^p$. Fix a convex set $\boldsymbol{\Theta} \subseteq \Theta^n$. Assume the following holds:*

(a0) *(Parameter consistency) Suppose that there exists $\theta_\star \in \Theta$ and a constant $c > 0$ such that*

$$\max_{1\le i\le n} \|\theta_\star - \theta_{i\star}\| \le c/\sqrt{n}. \tag{116}$$

(a1) *(Smoothness) For each realization of the data $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_n] \in \mathbb{R}^{p\times n}$, the function $\boldsymbol{\theta} \mapsto \mathcal{L}(\mathbf{X}\,;\boldsymbol{\theta})$ is three-times continuously differentiable and $R(\boldsymbol{\theta})$ is differentiable. Furthermore, denote $Y_i := \nabla_\theta \mathcal{L}_0(\mathbf{x}_i; \theta_{i\star}) \in \mathbb{R}^p$, $\overline{Y}_i := Y_i - \mathbb{E}[Y_i]$, and $W_i := \left\langle \overline{Y}_i,\, \frac{\theta_\star - \theta_{i\star}}{c/\sqrt{n}} \right\rangle$. Suppose there are constants $D_1, d_1 \in (0, \infty)$ such that*

$$\max_{1\le i\le n} \mathbb{E}[\|\overline{Y}_i\|^3] < D_1, \quad \max_{1\le i\le n} \mathbb{E}\left[W_i^3\right] < D_1, \quad \min_{1\le i\le n} \min_{1\le k\le p} \mathrm{Var}(Y_i(k)) > d_1, \quad \min_{1\le i\le n} \mathrm{Var}(W_i) > d_1. \tag{117}$$

(a2) *(First-order optimality) The true parameter $\boldsymbol{\theta}_\star := (\theta_{1\star}, \dots, \theta_{n\star})$ is a stationary point of the expected negative log-likelihood function $\overline{\mathcal{L}}_0(\boldsymbol{\theta}) := \mathbb{E}_{\boldsymbol{\theta}_\star}\left[\mathcal{L}_0(\mathbf{X}; \boldsymbol{\theta})\right]$ over $\boldsymbol{\Theta}$:*

$$\langle \nabla_{\boldsymbol{\theta}} \overline{\mathcal{L}}_0(\boldsymbol{\theta}),\, \boldsymbol{\theta} - \boldsymbol{\theta}_\star \rangle \ge 0 \quad \forall \boldsymbol{\theta} \in \boldsymbol{\Theta}. \tag{118}$$

(a3) *(Approximate second-order optimality) Let $\bar{\mathcal{L}}(\boldsymbol{\theta}) := \overline{\mathcal{L}}_0(\boldsymbol{\theta}) + R(\boldsymbol{\theta})$ denote the expected regularized negative log likelihood function. Then the regularized 'joint Fisher information' $\nabla^2 \bar{\mathcal{L}}(\boldsymbol{\theta})$ is positive definite at $\boldsymbol{\theta} = \boldsymbol{\theta}_\star$ with minimum eigenvalue $\rho > 0$.*

Fix a constant $C > 0$ and let $D = n^{-1/2}\left(C + \frac{4\|\nabla R(\boldsymbol{\theta}_\star)\|}{\rho}\right)$. Let $M = M(D) > 0$ denote the supremum of the absolute values of all third-order partial derivatives of $\mathcal{L}$ over all $\boldsymbol{\theta}$ with $\|\boldsymbol{\theta} - \boldsymbol{\theta}_\star\| \le D$. Suppose $\|\nabla R(\boldsymbol{\theta}_\star)\|$ is small enough so that

$$\sqrt{n} D \le \frac{3\rho}{4M(D)}. \tag{119}$$

Then there are constants $c_1, c_2, c_3 > 0$ such that

$$\mathbb{P}\left(\inf_{\substack{\theta\in\Theta \\ \|\theta - \theta_\star\| = D}} \mathcal{L}(\mathbf{X}\,; \theta, \dots, \theta) - \mathcal{L}(\mathbf{X}\,; \boldsymbol{\theta}_\star) > 0 \right) \ge 1 - c_1 \exp\left(-\frac{C^2\rho^2}{64}\right) - \frac{c_2}{\sqrt{n}} - O(\exp(-c_3 n)). \tag{120}$$

That is, with high probability explicitly depending on C, ρ, p, and n, there exists a local maximizer of $\theta \mapsto \mathcal{L}(\mathbf{X}\,; \theta, \dots, \theta)$ in Θ within distance D from $\theta_\star$.

We can easily deduce Theorem 4.5 from Theorem D.1.

Proof of Theorem 4.5. This is a straightforward application of the general result we just established in Theorem D.1 and the local landscape result in Theorem 4.3. Details are omitted. □

We devote the rest of this section to proving Theorem D.1.

Lemma D.2 (A uniform McDirmid's inequality). *Let $X_1, \dots, X_n$ be independent random vectors in $\mathbb{R}^d$ from a joint distribution π. Fix a compact parameter space $\mathbf{\Theta} \subseteq \mathbb{R}^p$ and $f_{\boldsymbol{\theta}} : \mathbb{R}^d \to [-M, M]$ is a bounded functional for each $\boldsymbol{\theta} \in \mathbf{\Theta}$ such that*

$$\|f_{\boldsymbol{\theta}} - f_{\boldsymbol{\theta}'}\|_\infty \le L\|\boldsymbol{\theta} - \boldsymbol{\theta}'\|, \qquad \forall \boldsymbol{\theta}, \boldsymbol{\theta}' \in \mathbf{\Theta} \tag{121}$$

for some constant $L > 0$. Further assume that $\mathbb{E}[f_{\boldsymbol{\theta}}(X_k)] = 0$ for all $\boldsymbol{\theta} \in \mathbf{\Theta}$ and $k = 1, \dots, n$. Then there exists constants $K, M > 0$ such that for each $n \ge 0$, and $\eta > 0$,

$$\mathbb{P}\left(\sup_{\boldsymbol{\theta} \in \mathbf{\Theta}} \left| \frac{1}{n} \sum_{k=1}^n f_{\boldsymbol{\theta}}(X_k) \right| \ge \eta \right) \le K \left(\frac{2L\,\mathrm{diam}(\mathbf{\Theta})}{\eta} \right)^p \exp\left(-\frac{\eta^2 n}{2M^2} \right). \tag{122}$$

Proof. Recall that $\mathbf{\Theta} \subseteq \mathbb{R}^p$ is compact, so it can be covered by a finite number of L^2-balls of any given radius $\varepsilon > 0$. Denote by $\mathcal{U}_\varepsilon$ such an open cover using the least number of balls of radius $\varepsilon > 0$. Let $N(\varepsilon) = |\mathcal{U}_\varepsilon|$ denote the smallest number of such balls to cover $\mathbf{\Theta}$. Moreover, let $\mathrm{diam}(\mathbf{\Theta})$ be the diameter of $\mathbf{\Theta}$, which is finite since $\mathbf{\Theta}$ is compact. Then $\mathbf{\Theta}$ is contained in a p-dimensional box of side length $\mathrm{diam}(\mathbf{\Theta})$. Thus there exists a constant $K > 0$, depending only on $\mathrm{diam}(\mathbf{\Theta})$ and d, for which

$$N(\varepsilon) \le K \left(\frac{\mathrm{diam}(\mathbf{\Theta})}{\varepsilon} \right)^p. \tag{123}$$

Next, fix $\eta > 0$, $\boldsymbol{\theta} \in \mathbf{\Theta}$, and $\varepsilon > 0$. Let $\boldsymbol{\theta}_1, \cdots, \boldsymbol{\theta}_{N(\varepsilon)}$ be the centers of balls in the open cover $\mathcal{U}_\varepsilon$. Then there exists $1 \le j \le N(\varepsilon)$ such that $\|\boldsymbol{\theta} - \boldsymbol{\theta}_j\| < \varepsilon$. By the hypothesis, $f_{\boldsymbol{\theta}}$ depends on $\boldsymbol{\theta}$ uniformly continuously with respect to the supremum norm. Hence there exists $\delta = \delta(\varepsilon) > 0$ such that

$$\|f_{\boldsymbol{\theta}} - f_{\boldsymbol{\theta}_j}\|_\infty \le L\varepsilon. \tag{124}$$

Denote $H_n(\boldsymbol{\theta}) := n^{-1} \sum_{k=1}^n f_{\boldsymbol{\theta}}(X_k)$. Then it follows that, almost surely,

$$|H_n(\boldsymbol{\theta}) - H_n(\boldsymbol{\theta}_j)| \le L\varepsilon. \tag{125}$$

Furthermore, by the hypothesis, $\|f_{\boldsymbol{\theta}}\|_\infty$ is uniformly bounded by $M > 0$. It follows that for each $\boldsymbol{\theta} \in \mathbf{\Theta}$, $H_n(\boldsymbol{\theta})$ changes its value at most by M when one of $X_1, \dots, X_n$ is replaced arbitrarily. Therefore by the standard McDirmid's inequality (see, Theorem 2.9.1. in (Vershynin, 2018)) and a union bound, with choosing $\varepsilon = \eta/(2L)$, we have

$$\mathbb{P}\left(|H_n(\boldsymbol{\theta})| \ge \eta\right) \le \sum_{j=1}^{N(\eta/2L)} \mathbb{P}\left(|H_n(\boldsymbol{\theta}_j)| \ge \eta/2\right) \le K \left(\frac{2L\,\mathrm{diam}(\mathbf{\Theta})}{\eta} \right)^p \exp\left(-\frac{n\eta^2}{2M^2} \right). \tag{126}$$

The above holds for all $n \ge 1$, $\eta > 0$, and $\boldsymbol{\theta} \in \mathbf{\Theta}$. □

Next, we recall the classical Berry-Esseen theorem for the rate of convergence of normal approximation for the sum of independent but not necessarily identically distributed random variables due to Feller.

Theorem D.3 (Berry-Esseen, Feller '68 (Feller, 1968)). *Let $X_1, X_2, \dots, X_n$ be independent and not necessarily identically distributed random variables with zero means and finite variances. Define $W = \sum_{i=1}^n X_i$ and assume that* $\mathrm{Var}(W) = 1$. *Let F be the distribution function of W and Φ be the standard normal distribution function. Then*

$$\|F - \Phi\|_\infty \le 6 \left(\sum_{i=1}^n \mathbb{E}[X_i^2 \mathbf{1}(|X_i| > 1)] + \sum_{i=1}^n \mathbb{E}[X_i^3 \mathbf{1}(|X_i| \le 1)] \right). \tag{127}$$

Now we prove Theorem D.1. The proof is essentially handling an additional probabilistic perturbation in the proof of Lemma C.5.

Proof of Theorem D.1. Suppose an n-tuple $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_n]$ of vectors $\mathbf{x}_s$ in $\mathbb{R}^d$ is observed under the product distribution $\pi_{\theta_{1\star}} \otimes \cdots \otimes \pi_{\theta_{n\star}}$. Denote $\boldsymbol{\theta}_\star = (\theta_{1\star}, \dots, \theta_{n\star})$. Also by the hypothesis, $\mathcal{L}_0$ is twice continuously differentiable, so $\mathbb{E}[\nabla \mathcal{L}_0] = \nabla \mathbb{E}[\mathcal{L}_0]$ and $\mathbb{E}[\nabla^2 \mathcal{L}_0] = \nabla^2 \mathbb{E}[\mathcal{L}_0]$ by the dominated convergence theorem.

Fix $\boldsymbol{\theta} = (\theta_1, \dots, \theta_n) \in \boldsymbol{\Theta}$ such that $\|\theta_i - \theta_\star\| = D$ for all $i = 1, \dots, n$. Then from (116), for all $i = 1, \dots, n$,

$$D - \frac{c}{\sqrt{n}} \le \|\theta_i - \theta_{i\star}\| \le D + \frac{c}{\sqrt{n}}. \tag{128}$$

We introduce two random variables that we will bound to be small by using some concentration inequalities:

$$T_n(\boldsymbol{\theta}) := \frac{1}{\sqrt{n}D} \left\langle \nabla_{\boldsymbol{\theta}} \mathcal{L}_0(\mathbf{X}; \boldsymbol{\theta}_\star) - \mathbb{E}\left[\nabla_{\boldsymbol{\theta}} \mathcal{L}_0(\mathbf{X}; \boldsymbol{\theta}_\star)\right], \boldsymbol{\theta} - \boldsymbol{\theta}_\star \right\rangle, \tag{129}$$

$$S_n(\boldsymbol{\theta}) := \frac{1}{n\|\boldsymbol{\theta} - \boldsymbol{\theta}_\star\|^2} (\boldsymbol{\theta} - \boldsymbol{\theta}_\star)^T \left(\nabla_{\boldsymbol{\theta}} \nabla_{\boldsymbol{\theta}^T} \mathcal{L}(\mathbf{X}\,;\, \boldsymbol{\theta}_\star) - \nabla_{\boldsymbol{\theta}} \nabla_{\boldsymbol{\theta}^T} \left(\mathbb{E}\left[\mathcal{L}(\mathbf{X}\,;\, \boldsymbol{\theta}_\star)\right]\right)\right) (\boldsymbol{\theta} - \boldsymbol{\theta}_0). \tag{130}$$

Since $\boldsymbol{\theta} \mapsto \mathcal{L}(\mathbf{X};\, \boldsymbol{\theta})$ is assumed to be three times continuously differentiable, the quantity M in the assertion is well-defined and is finite. Then using a Taylor expansion, we may write

$$\mathcal{L}(\mathbf{X}\,;\, \boldsymbol{\theta}) - \mathcal{L}(\mathbf{X}\,;\, \boldsymbol{\theta}_\star) \ge \langle \nabla_{\boldsymbol{\theta}} \mathcal{L}(\mathbf{X}; \boldsymbol{\theta}_\star),\, \boldsymbol{\theta} - \boldsymbol{\theta}_\star \rangle + \frac{1}{2} (\boldsymbol{\theta} - \boldsymbol{\theta}_\star)^T \nabla_{\boldsymbol{\theta}} \nabla_{\boldsymbol{\theta}^T} \mathcal{L}(\mathbf{X}\,;\, \boldsymbol{\theta}_\star)(\boldsymbol{\theta} - \boldsymbol{\theta}_\star) \tag{131}$$

$$- \sum_{i=1}^n \frac{M(\|\theta - \theta_{i\star}\|)}{6} \|\theta - \theta_{i\star}\|^3. \tag{132}$$

We will lower bound the first two terms on the right-hand side above. Note that

$$\langle \nabla_{\boldsymbol{\theta}} \mathcal{L}(\mathbf{X}; \boldsymbol{\theta}_\star),\, \boldsymbol{\theta} - \boldsymbol{\theta}_\star \rangle = \left[\langle \nabla_{\boldsymbol{\theta}} \mathcal{L}_0(\mathbf{X}; \boldsymbol{\theta}_\star),\, \boldsymbol{\theta} - \boldsymbol{\theta}_\star \rangle - \mathbb{E}\left[\langle \nabla_{\boldsymbol{\theta}} \mathcal{L}_0(\mathbf{X}; \boldsymbol{\theta}_\star),\, \boldsymbol{\theta} - \boldsymbol{\theta}_\star \rangle\right]\right] \tag{133}$$

$$+ \langle \nabla_{\boldsymbol{\theta}} \mathbb{E}[\mathcal{L}_0(\mathbf{X}; \boldsymbol{\theta}_\star)],\, \boldsymbol{\theta} - \boldsymbol{\theta}_\star \rangle + \langle \nabla R(\boldsymbol{\theta}_\star),\, \boldsymbol{\theta} - \boldsymbol{\theta}_\star \rangle \tag{134}$$

$$\overset{(a)}{\ge} \langle \nabla_{\boldsymbol{\theta}} \mathcal{L}_0(\mathbf{X}; \boldsymbol{\theta}_\star),\, \boldsymbol{\theta} - \boldsymbol{\theta}_\star \rangle - \mathbb{E}\left[\langle \nabla_{\boldsymbol{\theta}} \mathcal{L}_0(\mathbf{X}; \boldsymbol{\theta}_\star),\, \boldsymbol{\theta} - \boldsymbol{\theta}_\star \rangle\right] - \|\nabla R(\boldsymbol{\theta}_\star)\|\, \|\boldsymbol{\theta} - \boldsymbol{\theta}_\star\| \tag{135}$$

$$\overset{(b)}{=} -\sqrt{n}D\, T_n(\boldsymbol{\theta}) - \|\nabla R(\boldsymbol{\theta}_\star)\|\, \|\boldsymbol{\theta} - \boldsymbol{\theta}_\star\|, \tag{136}$$

where for (a) we use the fact that $\boldsymbol{\theta}_\star$ is a stationary point of $\mathbb{E}[\mathcal{L}_0(\mathbf{X}; \boldsymbol{\theta})]$ over $\boldsymbol{\Theta}$ and Cauchy-Schwarz inequality; for (b) we used the definition of $T_n(\boldsymbol{\theta})$.

Next, we turn our attention to the second-order term in the Taylor expansion (131). Recall that from the hypothesis,

$$\mathbb{E}\left[\nabla_{\boldsymbol{\theta}} \nabla_{\boldsymbol{\theta}^T} \mathcal{L}(\mathbf{X}\,;\, \boldsymbol{\theta}_\star)\right] = \nabla_{\boldsymbol{\theta}} \nabla_{\boldsymbol{\theta}^T} \left(\mathbb{E}\left[\mathcal{L}(\mathbf{X}\,;\, \boldsymbol{\theta}_\star)\right]\right) \succeq \rho \mathbf{I}_{pn}, \tag{137}$$

where $\rho > 0$ is a constant. It follows that

$$(\boldsymbol{\theta} - \boldsymbol{\theta}_\star)^T \nabla_{\boldsymbol{\theta}} \nabla_{\boldsymbol{\theta}^T} \mathcal{L}(\mathbf{X}\,;\, \boldsymbol{\theta}_\star)(\boldsymbol{\theta} - \boldsymbol{\theta}_\star) \tag{138}$$

$$\ge (\boldsymbol{\theta} - \boldsymbol{\theta}_\star)^T \left[\nabla_{\boldsymbol{\theta}} \nabla_{\boldsymbol{\theta}^T} \mathcal{L}(\mathbf{X}\,;\, \boldsymbol{\theta}_\star) - \nabla_{\boldsymbol{\theta}} \nabla_{\boldsymbol{\theta}^T} \left(\mathbb{E}\left[\mathcal{L}(\mathbf{X}\,;\, \boldsymbol{\theta}_\star)\right]\right)\right] (\boldsymbol{\theta} - \boldsymbol{\theta}_\star) + \rho \|\boldsymbol{\theta} - \boldsymbol{\theta}_\star\|^2 \tag{139}$$

$$\ge \|\boldsymbol{\theta} - \boldsymbol{\theta}_\star\|^2 \left(S_n(\boldsymbol{\theta}) + \rho\right). \tag{140}$$

Combining the above inequalities with noting that $\|\boldsymbol{\theta} - \boldsymbol{\theta}_\star\|^2 = nD^2$, we obtain

$$\frac{\mathcal{L}(\mathbf{X}\,;\, \boldsymbol{\theta}) - \mathcal{L}(\mathbf{X}\,;\, \boldsymbol{\theta}_\star)}{\|\boldsymbol{\theta} - \boldsymbol{\theta}_\star\|^2} \ge \frac{1}{\|\boldsymbol{\theta} - \boldsymbol{\theta}_\star\|} \underbrace{\left(-\|\nabla R(\boldsymbol{\theta}_\star)\| + \frac{\rho}{4}\sqrt{n}D - \frac{M(D)}{6} nD^2\right)}_{=:I_1} \tag{141}$$

$$+ \underbrace{\left(\frac{1}{2}(S_n(\boldsymbol{\theta}) + \frac{\rho}{2}) - \frac{1}{\sqrt{n}D} T_n(\boldsymbol{\theta})\right)}_{=:I_2}. \tag{142}$$

Note that $I_1 \geq 0$ if

$$\frac{\rho}{8}\sqrt{n}D \geq \|\nabla R(\boldsymbol{\theta}_\star)\| \quad \text{and} \quad \frac{\rho}{8}\sqrt{n}D \geq \frac{M(D)}{6}nD^2. \tag{143}$$

The former condition holds by the choice of D, and the latter condition holds by the hypothesis. Thus $I_1 \geq 0$.

We now take infimum over all $\boldsymbol{\theta} = (\theta, \dots, \theta) \in \Theta^n$ such that $\|\theta - \theta_\star\| = D$. In the proof of Lemma C.5, we have seen that the infimum of I_1 defined above is positive under the hypothesis. Hence it suffices to show that the random variable I_2 defined above is positive with high probability. To this end, write

$$\inf_{\substack{\boldsymbol{\theta}=(\theta,\dots,\theta)\in\Theta^n \\ \|\theta-\theta_\star\|=D}} I_2 \geq \underbrace{\left(\inf_{\substack{\boldsymbol{\theta}=(\theta,\dots,\theta)\in\Theta^n \\ \|\theta-\theta_\star\|=D}} \frac{-T_n(\boldsymbol{\theta})}{\sqrt{n}D} \right)}_{=:A} + \underbrace{\left(\inf_{\substack{\theta\in\Theta \\ \|\theta-\theta_\star\|=D}} \left(S_n(\theta,\dots,\theta) + \frac{\rho}{2} \right) \right)}_{=:B}. \tag{144}$$

Then the last expression in (144) is at least $\rho/8$ if $A \geq -\rho/8$ and $B \geq \rho/4$. Thus

$$\mathbb{P}\left(\inf_{\substack{\boldsymbol{\theta}=(\theta,\dots,\theta)\in\Theta^n \\ \|\theta-\theta_\star\|=D}} \frac{\mathcal{L}(\mathbf{X}\,;\boldsymbol{\theta}) - \mathcal{L}(\mathbf{X}\,;\boldsymbol{\theta}_\star)}{\|\boldsymbol{\theta}-\boldsymbol{\theta}_\star\|^2} \geq \rho/8 \right) \geq \mathbb{P}(A \geq -\rho/8) + \mathbb{P}(B \geq \rho/4) - 1. \tag{145}$$

By the hypothesis, $D = O(1)$ so it is uniformly bounded. Then by the uniform McDirmid's inequality in Lemma D.2, there exists constants $C', C'' > 0$ such that

$$\mathbb{P}(B < \rho/4) \leq \mathbb{P}\left(\inf_{\substack{\theta\in\Theta \\ \|\theta-\theta_\star\|=D}} S_n(\theta,\dots,\theta) < -\rho/4 \right) \leq D^p C' \exp(-C''n). \tag{146}$$

Next, we will show the following inequalities: For $K = 6D_1/d_1^{3/2}$,

$$\mathbb{P}(A < -\rho/8) \overset{(c)}{\leq} \mathbb{P}\left(\inf_{\substack{\boldsymbol{\theta}=(\theta,\dots,\theta)\in\Theta^n \\ \|\theta-\theta_\star\|=D}} T_n(\boldsymbol{\theta}) \geq \frac{\sqrt{n}D\rho}{8} \right) \overset{(d)}{\leq} (p+1)\left(\mathbb{P}\left(Z \geq p^{-1/2}\left(\frac{\sqrt{n}D\rho}{8} \right) \right) + \frac{K}{\sqrt{n}} \right) \tag{147}$$

$$\overset{(e)}{\leq} (p+1)\left(\exp\left(-\frac{nD^2\rho^2}{64} \right) + \frac{K}{\sqrt{n}} \right) \tag{148}$$

$$\overset{(f)}{\leq} (p+1)\left(\exp\left(-\frac{C^2\rho^2}{64} \right) + \frac{K}{\sqrt{n}} \right), \tag{149}$$

where $Z \sim N(0,1)$ is an independent standard normal random variable. Then the assertion will follow by combining (144), (145), (146), and (149). Note that (c) in (149) follows from the definition of A in (144). Also, note that (e) is a simple consequence of the standard Gaussian tail bound $\mathbb{P}(N(0,1) > x) \leq \frac{e^{-x^2/2}}{x\sqrt{2\pi}}$ and that $\sqrt{n}D \geq C$. (f) follows from the choice of D which yields $nD^2 \geq C^2$.

It remains to verify (d) in (149). To this end, define $p \times n$ matrix Q by letting its ith column $Q[:,i]$ be

$$Q[:,i] := \sum_{i=1}^n \nabla_\theta \mathcal{L}_0(\mathbf{x}_i;\theta_{i\star}) - \mathbb{E}\left[\nabla_\theta \mathcal{L}_0(\mathbf{x}_i;\theta_{i\star})\right] = Y_i, \tag{150}$$

where Y_i is defined in the assertion. Note that Q has independent mean zero columns and they do not depend on any specific choice of the running parameter θ. Then we can write

$$T_n(\theta,\dots,\theta) = \frac{1}{\sqrt{n}} \sum_{i=1}^n \left\langle Q[:,i], \frac{\theta - \theta_{i\star}}{D} \right\rangle \tag{151}$$

$$= \left\langle \frac{1}{\sqrt{n}} \sum_{i=1}^n Q[:,i], \frac{\theta - \theta_\star}{D} \right\rangle + \frac{1}{\sqrt{n}} \frac{c/\sqrt{n}}{D} \sum_{i=1}^n \underbrace{\left\langle Q[:,i], \frac{\theta_\star - \theta_{i\star}}{c/\sqrt{n}} \right\rangle}_{=W_i}. \tag{152}$$

It is important to note that the random variables W_i do not depend on the specific parameter choice θ, while the first term in the last expression above does. Such dependence on θ can be removed by using Cauchy-Schwarz inequality. Namely, denote $Q_n^{k\bullet} := n^{-1/2}\sum_{i=1}^n Q[k,i]$ for $k=1,\dots,p$. Also noting that $c/\sqrt{n}\le D$, we deduce

$$T_n(\theta,\dots,\theta) \le \left\| \frac{1}{\sqrt{n}}\sum_{i=1}^n Q[:,i] \right\| + \underbrace{\frac{1}{\sqrt{n}}\sum_{i=1}^n W_i}_{=:Z_n} \tag{153}$$

$$= \sqrt{p}\sum_{k=1}^p \left|Q_n^{k\bullet}\right| + Z_n. \tag{154}$$

Note that for each $k=1,\dots,p$, $Q[k,i]$ for $i=1,\dots,n$ are independent and mean zero random variables with uniformly bounded variances. Likewise, X_i for $i=1,\dots,n$ are independent and mean zero random variables with uniformly bounded variances. Hence by union bound,

$$\mathbb{P}\left(\inf_{\substack{\boldsymbol{\theta}=(\theta,\dots,\theta)\in\Theta^n \\ \|\theta-\theta_\star\|=D}} T_n(\boldsymbol{\theta}) \ge t \right) \le \left[\sum_{k=1}^p \mathbb{P}\left(Q_n^{k\bullet} \ge \frac{t}{2\sqrt{p}} \right) \right] + \mathbb{P}\left(Z_n \ge \frac{t}{2\sqrt{p}} \right). \tag{155}$$

Then by the Berry-Esseen Theorem (Theorem D.3) and the hypothesis,

$$\sup_{z\in\mathbb{R}} \left| \mathbb{P}\left(Q_n^{k\bullet}\le z\right) - \mathbb{P}\left(Z\le z\right) \right| \le \frac{6\,\sum_{i=1}^n \mathbb{E}[\|Y_i\|^3]}{\left(\sum_{i=1}^n \mathrm{Var}(Y_i(k))\right)^{3/2}} \le \frac{6D_1}{d_1^{3/2}\sqrt{n}} \quad \text{for } k=1,\dots,p \tag{156}$$

$$\sup_{z\in\mathbb{R}} \left| \mathbb{P}\left(Z_n\le z\right) - \mathbb{P}\left(Z\le z\right) \right| \le \frac{6\,\sum_{i=1}^n \mathbb{E}[|W_i|^3]}{\left(\sum_{i=1}^n \mathrm{Var}(W_i)\right)^{3/2}} \le \frac{6D_1}{d_1^{3/2}\sqrt{n}}. \tag{157}$$

Combining with (155) and denoting $K = 6D_1/d_1^{3/2}$, we obtain

$$\mathbb{P}\left(\inf_{\substack{\boldsymbol{\theta}=(\theta,\dots,\theta)\in\Theta^n \\ \|\theta-\theta_\star\|=D}} T_n(\boldsymbol{\theta}) \ge t \right) \le (p+1)\left(\mathbb{P}\left(Z \ge \frac{t}{2\sqrt{p}} \right) + \frac{K}{\sqrt{n}} \right), \tag{158}$$

Thus (d) in (149) follows. □

E. BCD algorithm for SMF

In the main text, we introduce the BCD algorithm for SMF-$\mathbf{W}$. When $\kappa>1$, the algorithm can be easily extended to the multi-label setting. Here, $\mathbf{K}$ is defined as real-valued $\kappa\times n$ matrix such that

$$\mathbf{K} := [\dot{\mathbf{h}}(y_1,\mathbf{a}_1),\dots,\dot{\mathbf{h}}(y_n,\mathbf{a}_n)] \in \mathbb{R}^{\kappa\times n}$$

where each $\dot{\mathbf{h}}(y,\mathbf{a})$ denotes

$$\nabla_{\mathbf{a}}\ell(y,\mathbf{a}) =: \dot{\mathbf{h}}(y,\mathbf{a}) = (\dot{h}_1,\dots,\dot{h}_\kappa)\in\mathbb{R}^\kappa, \quad \dot{h}_j := \frac{h'(a_j)}{1+\sum_{c=1}^\kappa h(a_c)} - \frac{\mathbf{1}(y=j)h'(a_j)}{h(a_j)}, \tag{159}$$

with a proper score function $h(\cdot)$ in (25).

Below we state the BCD algorithm with adaptive step sizes for SMF-$\mathbf{H}$. The structure of the algorithm is identical to Algorithm 1 for SMF-$\mathbf{W}$.

Algorithm 2 BCD algorithm for SMF-$\mathbf{H}$

```
Input: $\mathbf{X} \in \mathbb{R}^{p\times n}$ (Data); $\mathbf{X}_{\text{aux}} \in \mathbb{R}^{q\times n}$ (Auxiliary covariate); $\mathbf{Y}_{\text{label}} \in \{0,\dots,\kappa\}^{1\times n}$ (Label);
Constraints: Convex subsets $\mathcal{C}_1 \subseteq \mathbb{R}^{p\times r}$, $\mathcal{C}_2 \subseteq \mathbb{R}^{r\times n}$, $\mathcal{C}_3 \subseteq \mathbb{R}^{r\times\kappa}$, $\mathcal{C}_4 \subseteq \mathbb{R}^{q\times\kappa}$
Parameters: $\xi \geq 0$ (Tuning parameter); $T \in \mathbb{N}$ (number of iterations); $(\eta_{k;i})_{k\geq 1, 1\leq i\leq 4}$ (step-sizes)
Initialize $\mathbf{W} \in \mathcal{C}_1$, $\mathbf{H} \in \mathcal{C}_2$, $\boldsymbol{\beta} \in \mathcal{C}_3$, $\boldsymbol{\Gamma} \in \mathcal{C}_4$
For $k = 1, 2, \dots, T$ do:        (▷ For $\alpha^+$ see Assumption B.1.)
    (Update $\mathbf{W}$)
    $\nabla_{\mathbf{W}} f(\mathbf{Z}) \leftarrow 2\xi(\mathbf{W}\mathbf{H} - \mathbf{X})\mathbf{H}^T$
    Choose $\eta_{k,1}^{-1} > L_1 := 2\xi\|\mathbf{H}\|_2^2$
    $\mathbf{W} \leftarrow \Pi_{\mathcal{C}_1}(\mathbf{W} - \eta_{k;1}\nabla_{\mathbf{W}} f(\mathbf{Z}))$

    (Update $\mathbf{H}$)
    Update activation $\mathbf{a}_1, \dots, \mathbf{a}_n$ and $\mathbf{K}$
    $\nabla_{\mathbf{H}} f(\mathbf{Z}) \leftarrow \boldsymbol{\beta}\mathbf{K} + 2\xi\mathbf{W}^T(\mathbf{W}\mathbf{H} - \mathbf{X})$
    Choose $\eta_{k,2}^{-1} > L_2 := \alpha^+\|\boldsymbol{\beta}\|_2 + 2\xi\|\mathbf{W}\|_2^2$
    $\mathbf{H} \leftarrow \Pi_{\mathcal{C}_2}(\mathbf{H} - \eta_{k;2}\nabla_{\mathbf{H}} f(\mathbf{Z}))$

    (Update $\boldsymbol{\beta}$)
    Update activation $\mathbf{a}_1, \dots, \mathbf{a}_n$ and $\mathbf{K}$
    $\nabla_{\boldsymbol{\beta}} f(\mathbf{Z}) \leftarrow \mathbf{X}\mathbf{K}^T$
    Choose $\eta_{k,3}^{-1} > L_3 := \alpha^+\|\mathbf{H}\|_2^2$
    $\mathbf{H} \leftarrow \Pi_{\mathcal{C}_3}(\boldsymbol{\beta} - \eta_{k;3}\nabla_{\boldsymbol{\beta}} f(\mathbf{Z}))$

    (Update $\boldsymbol{\Gamma}$)
    Update activation $\mathbf{a}_1, \dots, \mathbf{a}_n$ and $\mathbf{K}$
    $\nabla_{\boldsymbol{\Gamma}} f(\mathbf{Z}) \leftarrow \mathbf{X}_{\text{aux}}\mathbf{K}^T$
    Choose $\eta_{k,4}^{-1} > L_4 := \alpha^+\|\mathbf{X}_{\text{aux}}\|_2^2$
    $\boldsymbol{\Gamma} \leftarrow \Pi_{\mathcal{C}_4}(\boldsymbol{\Gamma} - \eta_{k;4}\nabla_{\boldsymbol{\Gamma}} f(\mathbf{Z}))$
End for
Output: $\mathbf{Z} = (\mathbf{W}, \mathbf{H}, \boldsymbol{\beta}, \boldsymbol{\Gamma})$
```

F. Generalized multinomial logistic regression

In this section, we provide some background on a generalized multinomial logistic regression and record some useful computations. (See (Böhning, 1992) for backgrounds on multinomial logistic regression.) Without loss of generality, we can assume that the $\kappa+1$ classes are the integers in $\{0, 1, \dots, \kappa\}$. Say we have training examples $(\boldsymbol{\phi}(\mathbf{x}_1), y_1), \dots, (\boldsymbol{\phi}(\mathbf{x}_n), y_n)$, where

- $\mathbf{x}_1, \dots, \mathbf{x}_n$: Input data (e.g., collection of all medical records of each patient)
- $\boldsymbol{\phi}_1 := \boldsymbol{\phi}(\mathbf{x}_1), \dots, \boldsymbol{\phi}_n := \boldsymbol{\phi}(\mathbf{x}_n) \in \mathbb{R}^p$: Features (e.g., some useful information for each patient)
- $y_1, \dots, y_n \in \{0, 1, \dots, \kappa\}$: $\kappa + 1$ class labels (e.g., digits from 0 to 9).

The basic idea of multinomial logistic regression is to model the output y as a discrete random variable Y with probability mass function $\mathbf{p} = [p_0, p_1, \dots, p_\kappa]$ that depends on the observed feature $\boldsymbol{\phi}(\mathbf{x})$, score function $h : \mathbb{R} \to \mathbb{R}$ (strictly increasing, twice differentiable, and $h(0) = 1$), and a matrix parameter $\mathbf{W} = [\mathbf{w}_1, \dots, \mathbf{w}_\kappa] \in \mathbb{R}^{p\times\kappa}$ through the following relation:

$$p_0 = \frac{1}{1 + \sum_{c=1}^{\kappa} h(\langle \boldsymbol{\phi}(\mathbf{x}), \mathbf{w}_c \rangle)}, \quad p_j = \frac{h(\langle \boldsymbol{\phi}(\mathbf{x}), \mathbf{w}_j \rangle)}{1 + \sum_{c=1}^{\kappa} h(\langle \boldsymbol{\phi}(\mathbf{x}), \mathbf{w}_c \rangle)}, \quad \text{for } j = 1, \dots, \kappa. \tag{160}$$

That is, given the feature vector $\boldsymbol{\phi}(\mathbf{x})$, the probability p_i of $\mathbf{x}$ having label i is proportional to h evaluated at the 'linear activation' $\langle \boldsymbol{\phi}(\mathbf{x}), \mathbf{w}_i \rangle$ with the base category of class 0. Note that using $h(x) = \exp(x)$, the above multiclass classification model reduces to the classical multinomial logistic regression. In this case, the corresponding predictive probability distribution $\mathbf{p}$ is called the *softmax distribution* with activation $\mathbf{a} = [a_1, \dots, a_\kappa]$ with $a_i = \langle \boldsymbol{\phi}(\mathbf{x}), \mathbf{w}_i \rangle$ for $i = 1, \dots, \kappa$. Notice that this model has parameter vectors $\mathbf{w}_1, \dots, \mathbf{w}_\kappa \in \mathbb{R}^p$, one for each of the κ nonzero class labels.

Next, we derive the maximum log-likelihood formulation for finding optimal parameter $\mathbf{W}$ for the given training set $(\boldsymbol{\phi}_i, y_i)_{i=1,\dots,n}$. For each $1 \le i \le n$, define the predictive probability mass function $\mathbf{p}_i = [p_{i0}, p_{i1}, \dots, p_{i\kappa}]$ using (160) with $\boldsymbol{\phi}(\mathbf{x})$ replaced by $\boldsymbol{\phi}_i$. We introduce the following matrix notations

$$\mathbf{Y} := \underbrace{\begin{bmatrix} \mathbf{1}(y_1 = 1) & \cdots & \mathbf{1}(y_1 = \kappa) \\ \vdots & & \vdots \\ \mathbf{1}(y_n = 1) & \cdots & \mathbf{1}(y_n = \kappa) \end{bmatrix}}_{\in \{0,1\}^{n \times \kappa}}, \quad \mathbf{P} := \underbrace{\begin{bmatrix} p_{11} & \cdots & p_{1\kappa} \\ \vdots & & \vdots \\ p_{n1} & \cdots & p_{n\kappa} \end{bmatrix}}_{\in [0,1]^{n \times \kappa}} \tag{161}$$

$$\boldsymbol{\Phi} := \underbrace{\begin{bmatrix} \uparrow & & \uparrow \\ \boldsymbol{\phi}(\mathbf{x}_1) & \cdots & \boldsymbol{\phi}(\mathbf{x}_n) \\ \downarrow & & \downarrow \end{bmatrix}}_{\in \mathbb{R}^{p \times n}}, \quad \mathbf{W} := \underbrace{\begin{bmatrix} \uparrow & & \uparrow \\ \mathbf{w}_1 & \cdots & \mathbf{w}_\kappa \\ \downarrow & & \downarrow \end{bmatrix}}_{\in \mathbb{R}^{p \times \kappa}}. \tag{162}$$

Note that the sth row of $\mathbf{Y}$ is a zero vector if and only if $y_s = 0$. Similarly, since $p_{s0} = 1 - (p_{s1} + \cdots + p_{s\kappa})$, the corresponding row of $\mathbf{P}$ determines its predictive probability distribution. Then the joint likelihood function of observing labels $(y_1, \dots, y_n)$ given input data $(\mathbf{x}_1, \dots, \mathbf{x}_n)$ under the above probabilistic model is

$$L(y_1, \dots, y_n \,;\, \mathbf{W}) = \mathbb{P}(Y_1 = y_1, \dots, Y_n = y_n \,;\, \mathbf{W}) = \prod_{s=1}^{n} \prod_{j=0}^{\kappa} (p_{sj})^{\mathbf{1}(y_s = j)}. \tag{163}$$

Denote $\mathbf{w}_0 = \mathbf{0}$. Then since $h(0) = 1$ by definition, we can conveniently write

$$p_{sj} = \frac{h(\langle \boldsymbol{\phi}_s, \mathbf{w}_j \rangle)}{\sum_{c=0}^{\kappa} h(\langle \boldsymbol{\phi}_s, \mathbf{w}_c \rangle)} \qquad \text{for } s = 1, \dots, n \text{ and } j = 0, 1, \dots, \kappa. \tag{164}$$

Now we can derive the negative log-likelihood $\ell(\boldsymbol{\Phi}, \mathbf{W}) := -\sum_{s=1}^{n} \sum_{j=0}^{\kappa} \mathbf{1}(y_s = j) \log p_{sj}$ in a matrix form as follows:

$$\begin{align} \ell(\boldsymbol{\Phi}, \mathbf{W}) &= \sum_{s=1}^{n} \log \left(1 + \sum_{c=1}^{\kappa} h(\langle \boldsymbol{\phi}(\mathbf{x}_s), \mathbf{w}_c \rangle) \right) - \sum_{s=1}^{n} \sum_{j=0}^{\kappa} \mathbf{1}(y_s = j) \log h\left(\langle \boldsymbol{\phi}(\mathbf{x}_s), \mathbf{w}_j \rangle \right) \tag{165} \\ &= \left(\sum_{s=1}^{n} \log \left(1 + \sum_{c=1}^{\kappa} h(\langle \boldsymbol{\phi}(\mathbf{x}_s), \mathbf{w}_c \rangle) \right) \right) - \operatorname{tr}\left(\mathbf{Y}^T h(\boldsymbol{\Phi}^T \mathbf{W}) \right), \tag{166} \end{align}$$

where $\operatorname{tr}(\cdot)$ denotes the trace operator. Then the maximum likelihood estimates $\hat{\mathbf{W}}$ is defined as the minimizer of the above loss function in $\mathbf{W}$ while fixing the feature matrix $\boldsymbol{\Phi}$.

Both the maps $\mathbf{W} \mapsto \ell(\boldsymbol{\Phi}, \mathbf{W})$ and $\boldsymbol{\Phi} \mapsto \ell(\boldsymbol{\Phi}, \mathbf{W})$ are convex and we can compute their gradients as well as the Hessian explicitly as follows. For each $y \in \{0, 1, \dots \kappa\}$, $\boldsymbol{\phi} \in \mathbb{R}^p$, and $\mathbf{W} \in \mathbb{R}^{p \times \kappa}$, define vector and matrix functions

$$\begin{align} \dot{\mathbf{h}}(y, \boldsymbol{\phi}, \mathbf{W}) &:= (\dot{h}_1, \dots, \dot{h}_\kappa)^T \in \mathbb{R}^{\kappa \times 1}, \ \dot{h}_j := \frac{h'(\langle \boldsymbol{\phi}, \mathbf{w}_j \rangle)}{1 + \sum_{c=1}^{\kappa} h(\langle \boldsymbol{\phi}, \mathbf{w}_c \rangle)} - \mathbf{1}(y = j) \frac{h'(\langle \boldsymbol{\phi}, \mathbf{w}_j \rangle)}{h(\langle \boldsymbol{\phi}, \mathbf{w}_j \rangle)} \tag{167} \\ \ddot{\mathbf{H}}(y, \boldsymbol{\phi}, \mathbf{W}) &:= \left(\ddot{\mathbf{H}}_{ij} \right)_{i,j} \in \mathbb{R}^{\kappa \times \kappa}, \tag{168} \\ \ddot{\mathbf{H}}_{ij} &= \tfrac{h''(\langle \boldsymbol{\phi}, \mathbf{w}_j \rangle) \mathbf{1}(i=j)}{1 + \sum_{c=1}^{\kappa} h(\langle \boldsymbol{\phi}, \mathbf{w}_c \rangle)} - \tfrac{h'(\langle \boldsymbol{\phi}, \mathbf{w}_i \rangle) h'(\langle \boldsymbol{\phi}, \mathbf{w}_j \rangle)}{\left(1 + \sum_{c=1}^{\kappa} h(\langle \boldsymbol{\phi}, \mathbf{w}_c \rangle)\right)^2} - \mathbf{1}(y = i = j) \left(\tfrac{h''(\langle \boldsymbol{\phi}, \mathbf{w}_j \rangle)}{h(\langle \boldsymbol{\phi}, \mathbf{w}_j \rangle)} - \tfrac{\left(h'(\langle \boldsymbol{\phi}, \mathbf{w}_j \rangle)\right)^2}{(h(\langle \boldsymbol{\phi}, \mathbf{w}_j \rangle))^2} \right). \tag{169} \end{align}$$

For each $\mathbf{W} = [\mathbf{w}_1, \dots, \mathbf{w}_\kappa] \in \mathbb{R}^{p \times \kappa}$, let $\mathbf{W}^{\text{vec}} := [\mathbf{w}_1^T, \dots, \mathbf{w}_\kappa^T]^T \in \mathbb{R}^{p\kappa}$ denote its vectorization. Note that

$$\mathbb{E}_y \left[\dot{\mathbf{h}}(y, \boldsymbol{\phi}, \mathbf{W}) \right] = \mathbf{0}. \tag{170}$$

A straightforward computation also shows

$$\nabla_{\mathrm{vec}(\mathbf{W})}\ell(\mathbf{\Phi},\mathbf{W}) = \sum_{s=1}^{n} \dot{\mathbf{h}}(y_s,\boldsymbol{\phi}_s,\mathbf{W}) \otimes \boldsymbol{\phi}_s, \tag{171}$$

$$\mathbf{H} := \nabla_{\mathrm{vec}(\mathbf{W})}\nabla_{\mathrm{vec}(\mathbf{W})^T}\ell(\mathbf{\Phi},\mathbf{W}) = \sum_{s=1}^{n} \ddot{\mathbf{H}}(y_s,\boldsymbol{\phi}_s,\mathbf{W}) \otimes \boldsymbol{\phi}_s\boldsymbol{\phi}_s^T, \tag{172}$$

where $\otimes$ above denotes the Kronecker product. Recall that the eigenvalues of $\mathbf{A}\otimes\mathbf{B}$, where $\mathbf{A}$ and $\mathbf{B}$ are two square matrices, are given by $\lambda_i\mu_j$, where λ_i and μ_j run over all eigenvalues of $\mathbf{A}$ and $\mathbf{B}$, respectively. Also, for two square matrices $\mathbf{A},\mathbf{B}$ of the same size, write $\mathbf{A}\preceq\mathbf{B}$ if $v^T\mathbf{A}v \le v^T\mathbf{B}v$ for all unit vectors v. Then denoting $\lambda^+ := \max_{1\le s\le n}\lambda_{\max}(\ddot{\mathbf{H}}(y_s,\boldsymbol{\phi}_s,\mathbf{W}))$,

$$\mathbf{H} \preceq \lambda^+ \sum_{s=1}^{n} \mathbf{I}\otimes\boldsymbol{\phi}_s\boldsymbol{\phi}_s^T = \lambda^+\mathbf{I}\otimes\mathbf{\Phi}\mathbf{\Phi}^T. \tag{173}$$

Similarly, $\lambda^-\mathbf{I}\otimes\mathbf{\Phi}\mathbf{\Phi}^T \preceq \mathbf{H}$, where λ^- denotes the minimum over all $\lambda_{\min}(\ddot{\mathbf{H}}(y_s,\boldsymbol{\phi}_s,\mathbf{W}))$. Hence we can deduce

$$\lambda^-\lambda_{\min}\left(\mathbf{\Phi}\mathbf{\Phi}^T\right) \le \lambda_{\min}(\mathbf{H}) \le \lambda_{\max}(\mathbf{H}) \le \lambda^+\lambda_{\max}\left(\mathbf{\Phi}\mathbf{\Phi}^T\right). \tag{174}$$

There are some particular cases worth noting. First, suppose binary classification case, $\kappa = 1$. Then the Hessian $\mathbf{H}$ above reduces to

$$\mathbf{H} = \sum_{s=1}^{n} \ddot{\mathbf{H}}_{11}(y_s,\boldsymbol{\phi}_s,\mathbf{W})\boldsymbol{\phi}_s\boldsymbol{\phi}_s^T. \tag{175}$$

Second, let $h(x) = \exp(x)$ and consider the multinomial logistic regression case. Then $h = h' = h''$ so the above yields the following concise matrix expression

$$\nabla_{\mathbf{W}}\,\ell(\mathbf{\Phi},\mathbf{W}) = \mathbf{\Phi}(\mathbf{P}-\mathbf{Y}) \in \mathbb{R}^{p\times\kappa}, \qquad \nabla_{\mathbf{\Phi}}\,\ell(\mathbf{\Phi},\mathbf{W}) = \mathbf{W}(\mathbf{P}-\mathbf{Y})^T \in \mathbb{R}^{p\times n}, \tag{176}$$

$$\mathbf{H} = \sum_{s=1}^{n} \begin{bmatrix} p_{s1}(1-p_{s1}) & -p_{s1}p_{s2} & \dots & -p_{s1}p_{s\kappa} \\ -p_{s2}p_{s1} & p_{s2}(1-p_{s2}) & \dots & -p_{s2}p_{s\kappa} \\ \vdots & \vdots & \ddots & \vdots \\ -p_{s\kappa}p_{s1} & -p_{s\kappa}p_{s2} & \dots & p_{s\kappa}(1-p_{s\kappa}) \end{bmatrix} \otimes \boldsymbol{\phi}_s\boldsymbol{\phi}_s^T. \tag{177}$$

Note that $\mathbf{H}$ in this case does not depend on y_s for $s = 1,\dots,n$. The bounds on the eigenvalues depend on the range of linear activation $\langle\boldsymbol{\phi}_i,\mathbf{w}_j\rangle$ may take. For instance, if we restrict the norms of the input feature vector ϕ_i and parameter $\mathbf{w}_j$, then we can find a suitable positive uniform lower bound on the eigenvalues of $\mathbf{H}$.

Lemma F.1 (Lemma B.1 in (Lee et al., 2023))**.** *Suppose* $h(\cdot) = \exp(\cdot)$. *Then*

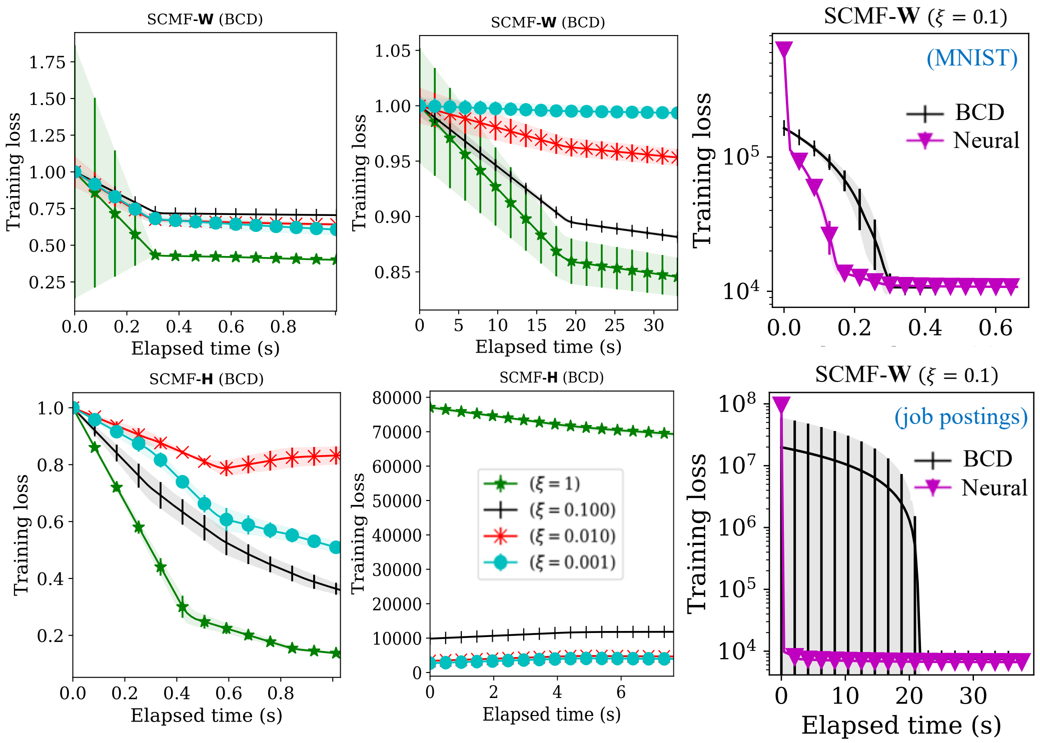
$$\lambda_{\min}\left(\ddot{\mathbf{H}}(\boldsymbol{\phi}_s,\mathbf{W})\right) \ge \min_{1\le i\le\kappa} \frac{\exp(\langle\boldsymbol{\phi}_s,\mathbf{w}_i\rangle)}{\left(1+\sum_{c=1}^{\kappa}\exp(\langle\boldsymbol{\phi}_s,\mathbf{w}_c\rangle)\right)^2}, \tag{178}$$

$$\lambda_{\max}\left(\ddot{\mathbf{H}}(\boldsymbol{\phi}_s,\mathbf{W})\right) \le \max_{1\le i\le\kappa} \frac{\exp(\langle\boldsymbol{\phi}_s,\mathbf{w}_i\rangle)}{\left(1+\sum_{c=1}^{\kappa}\exp(\langle\boldsymbol{\phi}_s,\mathbf{w}_c\rangle)\right)^2}\left(1+2\sum_{c=2}^{\kappa}\exp(\langle\boldsymbol{\phi}_s,\mathbf{w}_c\rangle)\right). \tag{179}$$

F.1. Additional Figures

G. Experimental details

All numerical experiments were performed on a workstation with Xeon Gold 6248R @ 3.00GHz CPU, 512GM of RAM, and two RTX A6000 GPUs.



a Semi-synthetic MNIST data **b** Job postings **c** BCD vs. Neural

Figure 6. (**a**,**b**) Plots of training loss for Algorithms 1 and 2 vs. elapsed time at different ξ values. (**c**) Comparison of convergence speed between BCD and its neural implementation ran on GPU at $\xi = 0.1$, with shaded regions indicating one standard deviation across 10 runs.

G.1. Experiments on semi-synthetic MNIST dataset

We follow the experimental setting in (Lee et al., 2023) for the semi-synthetic MNIST dataset. For the reader's convenience, we give details here. Denote $p = 28^2 = 784$, $n = 500$, $r = 2$, and $\kappa = 1$. First, we randomly select 10 images each from digits '2' and '5'. Vectorizing each image as a column in $p = 784$ dimension, we obtain a true factor matrix for features $\mathbf{W}_{\text{true},X} \in \mathbb{R}^{p\times r}$. Similarly, we randomly sample 10 images of each from digits '4' and '7' and obtain the true factor matrix of labels $\mathbf{W}_{\text{true},\mathbf{Y}} \in \mathbb{R}^{p\times r}$. Next, we sample a code matrix $\mathbf{H}_{\text{true}} \in \mathbb{R}^{r\times n}$ whose entries are i.i.d. with the uniform distribution $U([0,1])$. Then the 'pre-feature' matrix $\mathbf{X}_0 \in \mathbb{R}^{p\times n}$ of vectorized synthetic images is generated by $\mathbf{W}_{\text{true},X}\mathbf{H}_{\text{true}}$. The feature matrix $\mathbf{X}_{\text{data}} \in \mathbb{R}^{p\times n}$ is then generated by adding an independent Gaussian noise $\varepsilon_j \sim N(\mathbf{0}, \sigma^2 I_p)$ to the jth column of $\mathbf{X}_0$ for $j = 1,\dots,n$, with $\sigma = 0.5$. We generate the binary label matrix $\mathbf{Y} = [y_1,\dots,y_n] \in \{0,1\}^{1\times n}$ (recall $\kappa = 1$) as follows: Each entry y_i is an independent Bernoulli variable with probability $p_i = \left(1 + \exp\left(-\boldsymbol{\beta}_{\text{true},\mathbf{Y}}^T \mathbf{W}_{\text{true},\mathbf{Y}}^T \mathbf{X}_{\text{data}}[:,i]\right)\right)^{-1}$, where $\boldsymbol{\beta}_{\text{true},\mathbf{Y}} = [1,-1]$. No auxiliary features were used for the semi-synthetic dataset (i.e., $q = 0$).

G.2. Experiments on the Job postings dataset

Next, we provide detailed information about the dataset used in our study (Laboratory of Information and Communication Systems, 2016). The dataset consists of 17,880 job postings, encompassing 15 variables that include binary values, categorical variables, and textual information in the form of *job descriptions*. Within the dataset, 17,014 postings (95.1%) are classified as genuine job postings, while 866 postings (4.84%) are identified as fraudulent. This highlights a considerable class imbalance, with a significantly larger number of genuine postings compared to fraudulent ones. In our analysis, we designated fake job postings as positive examples and true job postings as negative examples.

In our experiments, we represented each job posting as a $p = 2480$ dimensional word frequency vector derived from its *job description*. This vector was augmented with $q = 72$ auxiliary features, encompassing binary and categorical variables. These features include indicators specifying whether a job posting includes a company logo or if the advertised job is located in the United States. To compute the word frequency vectors, we represented the job description variable as a term/document frequency matrix, applying Term Frequency-Inverse Document Frequency (TF-IDF) normalization (Pedregosa et al., 2011). This normalization method assigns lower importance to common words appearing in all documents and considers words specific to particular documents as more significant. In our analysis, we focused on the 2,480 most frequent words for further investigation. Due to the high imbalance, the accuracy of classification can be trivially high (e.g., by classifying everything to be negative), and hence achieving a high F-score is of importance.