Characterizing the Loss Landscape in Non-Negative Matrix Factorization

Johan Bjorck, Anmol Kabra, Kilian Q. Weinberger, Carla P. Gomes
Cornell University
{njb225,ak2426,kqw4,gomes}@cornell.edu

Abstract

Non-negative matrix factorization (NMF) is a highly celebrated algorithm for matrix decomposition that guarantees non-negative factors. The underlying optimization problem is computationally intractable, yet in practice, gradient-descent-based methods often find good solutions. In this paper, we revisit the NMF optimization problem and analyze its loss landscape in non-worst-case settings. We specifically study star-convexity, which implies that the gradients point towards the final minimizer. We show that such a property holds with high probability for NMF, provably in a non-worst case model with a planted solution, and empirically across an extensive suite of real-world NMF problems spanning collaborative filtering, scientific analysis, and image analysis. Our analysis predicts that this property becomes more likely with a growing number of parameters, and experiments suggest that a similar trend might also hold for deep neural networks—turning increasing dataset sizes and model sizes into a blessing from an optimization perspective.

Introduction

Non-negative matrix factorization (NMF) is a ubiquitous technique for data analysis, where one attempts to factorize a measurement matrix $\mathbf{X}$ into the product of non-negative matrices $\mathbf{U}, \mathbf{V}$ (Lee and Seung 1999). This simple problem has applications in recommender systems (Luo et al. 2014), scientific analysis (Berne et al. 2007; Trindade, Abel, and Watts 2017), computer vision (Gillis 2012), internet distance prediction (Mao, Saul, and Smith 2006), audio processing (Schmidt, Larsen, and Hsiao 2007), and many more domains. The non-negativity is often crucial for interpretability; in the context of crystallography for example, the light sources—represented as matrix factors—have non-negative intensity (Suram et al. 2016).

Like many other non-convex optimization problems, e.g. optimizing neural networks (Blum and Rivest 1989), finding the exact solution to NMF is NP-hard (Pardalos and Vavasis 1991; Vavasis 2009). NMF's tremendous practical success is at odds with such worst-case analysis, and simple algorithms based on gradient descent are known to find good solutions in real-world settings (Lee and Seung 2001). At the time when NMF was proposed, most analyses of optimization problems in machine learning focused on convex formulations such as SVMs (Cortes and Vapnik 1995). However, owing to the success of neural networks, non-convex optimization has experienced a resurgence in interest. While non-convex problems that can be optimized efficiently via saddle point characterization have been studied extensively (Ge et al. 2015), NMF has seen less theoretical progress. While the NMF problem is NP-hard, its empirical experience and widespread usage suggests that the problem might be tractable in the average case, albeit not in the worst case.

In this paper, we prove theoretically and empirically that a benign convexity property called *star-convexity* typically holds in NMF. From a theoretical perspective, we consider NMF instances with planted randomized solutions, inspired by the stochastic block model for social networks (Holland, Laskey, and Leinhardt 1983; Decelle et al. 2011) and the planted clique problem studied in sum-of-squares literature (Barak et al. 2016). We prove that between two random points, the loss is convex with high probability, and conclude that the loss surface is star-convex in the typical case—even if the loss is computed over unobserved data. From an empirical perspective, we verify that our theoretical results hold for an extensive collection of real-world datasets spanning collaborative filtering (Zhou et al. 2008; Kula 2017; Harper and Konstan 2016), signal decomposition (Zhu 2016; Li and Ngom 2013; Li et al. 2001; Erichson et al. 2018), and audio processing (Flenner and Hunter 2017). Finally, we show that star-convex behavior becomes more likely with a growing number of parameters, suggesting that a similar result may hold in neural networks as they become wider. We provide supporting empirical evidence for this hypothesis on modern network architectures. We summarize the contributions of this paper as follows:

- We prove that the NMF loss surface has benign convexity properties in the average case, which might explain why NMF typically performs well despite being NP-hard in the worst case.
- We verify that our theoretical predictions hold in an extensive suite of real-world datasets.
- Based on our theoretical results, we hypothesize that increasing width in neural networks should improve convexity. We also provide supporting experimental evidence.

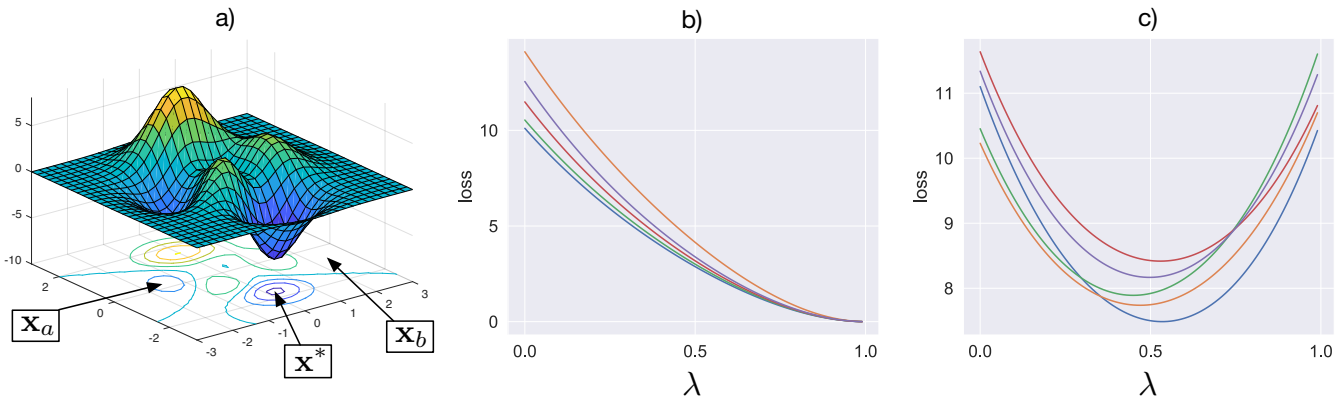

Figure 1: A non-convex loss surface is illustrated in a). In general, the loss will be non-convex on straight paths connecting random points $\mathbf{x}_a, \mathbf{x}_b$ and the global minimizer $\mathbf{x}^*$. We consider a model of NMF with a randomized planted solution; as shown in b), the loss is typically convex on straight paths between points $\mathbf{x}_a$ and a planted solution $\mathbf{x}^*$. Additionally, as illustrated in c), the loss is typically convex on straight paths between points $\mathbf{x}_a$ and $\mathbf{x}_b$.

NMF and Star-Convexity

NMF aims to decompose some large measurement matrix $\mathbf{X} \in \mathcal{R}^{n \times m}$ into two *non-negative* matrices $\mathbf{U} \in \mathcal{R}_+^{n \times r}$ and $\mathbf{V} \in \mathcal{R}_+^{r \times m}$ such that $\mathbf{X} \approx \mathbf{UV}$. The canonical formulation of NMF is

$$\min_{\mathbf{U},\mathbf{V} \geqslant 0} \quad \ell(\mathbf{U}, \mathbf{V}), \text{ where } \ell(\mathbf{U}, \mathbf{V}) = \frac{1}{2}\|\mathbf{UV} - \mathbf{X}\|_F^2. \quad (1)$$

Practitioners commonly use NMF in recommender systems, where an entry (i, j) of $\mathbf{X}$, for example, corresponds to the rating user i gave to movie j (Luo et al. 2014). In such settings, data might be missing if all users did not rate all movies. In those cases, it is common to only consider the loss over observed data (Zhang et al. 2006; Candès and Recht 2009). We let $\hat{1}_{(i,j)}$ be an indicator variable that is 1 if entry (i, j) is "observed" and 0 otherwise. The loss function is then

$$\min_{\mathbf{U},\mathbf{V} \geqslant 0} \quad \ell(\mathbf{U}, \mathbf{V}) = \frac{1}{2}\sum_{i,j} \hat{1}_{(i,j)} \left([\mathbf{UV}]_{ij} - \mathbf{X}_{ij} \right)^2. \quad (2)$$

NMF's non-negative constraints prevent practitioners from applying spectral strategies, which can be otherwise used in, e.g., PCA. This restriction results in NMF's NP-hardness (Vavasis 2009). Even so, previous work on the computational complexity of NMF has shown that the problem is tractable for small constant dimensions r via algebraic methods (Arora et al. 2012). However, practitioners use simple variants of gradient descent, which are known to work reliably, rather than these algorithms (Koren, Bell, and Volinsky 2009; Lee and Seung 2001). This gap between theoretical hardness and practical performance is also found in deep learning. Optimizing neural networks is generally NP-hard (Blum and Rivest 1989), but in practice, they can be optimized with simple stochastic gradient descent algorithms to outperform humans in tasks such as verifying faces (Lu and Tang 2015) and playing Atari-games (Mnih et al. 2015). Recent work on understanding the geometry of neural network loss surfaces has promoted the idea of convexity properties. Izmailov et al. (2018) show that the network's loss surface is convex around the local optimum, while Zhou et al. (2019) and Kleinberg, Li, and Yuan (2018) empirically show that the gradients during optimization typically point towards the local minima to which the network eventually converges. Of central importance in this line of work is **star-convexity**, which is a property of a function f that guarantees that f is convex along straight paths towards its optima x^*. See Figure 2 for an example. Formally, it is defined as

Definition 1. *A function $f : \mathcal{R}^n \to \mathcal{R}$ is **star-convex** towards $\boldsymbol{x}^*$ if for all $\lambda \in [0, 1]$ and $\boldsymbol{x} \in \mathcal{R}^n$, we have $f(\lambda \boldsymbol{x} + (1 - \lambda)\boldsymbol{x}^*) \leqslant \lambda f(\boldsymbol{x}) + (1 - \lambda) f(\boldsymbol{x}^*)$.*

Star-convex functions can be optimized in polynomial time (Lee and Valiant 2016). Moreover, the function only needs to be star-convex under a natural noise model (Kleinberg, Li, and Yuan 2018). Since NMF is NP-hard, it is not star-convex in general; however, it is natural to conjecture that NMF is star-convex in the *typical* case. Such a property could explain the practical success of NMF on real-world datasets, which are far from worst-case. This is the working hypothesis of this paper, where the *typical* case is formalized probabilistically in Theorem 1. Indeed, one can verify numerically that NMF is typically star-convex for natural distributions and realistically sized matrices: see Figure 1 where we consider a rank 10 decomposition of $(100, 100)$-matrices with iid half-normal entries and a planted solution, sampled as per Assumption 1 stated in the next section. We dedicate the following sections to prove that NMF is star-convex with high probability in a planted model, and to confirm that this phenomenon generalizes to datasets from the real world, which are far from worst-case.

Proving Typical-Case Star-Convexity

Our aim now is to prove that the NMF loss-function is star-convex in the typical case for natural non-worst-case distributions of NMF instances. We consider a slightly weaker notation of star-convexity, where $f(\lambda \mathbf{x} + (1 - \lambda)\mathbf{x}^*) \leqslant$

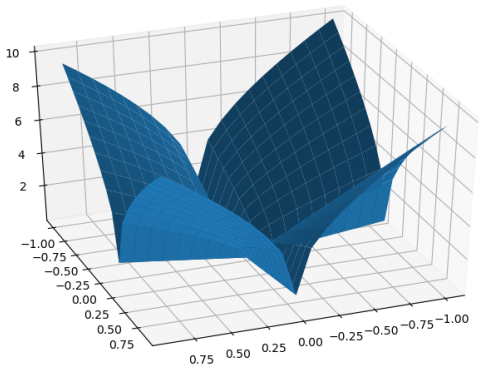

Figure 2: The function $(|x|^p + |y|^p)^{1/p}$ is an example of a star-convex function for $0 < p < 1$. It is non-convex in general, but convex towards $(0, 0)$.

$\lambda f(\mathbf{x}) + (1-\lambda) f(\mathbf{x}^*)$ holds not for all $\mathbf{x}$, but for random $\mathbf{x}$ with high probability. This is in fact the best achievable—an adversarial example of an NMF instance that isn't star-convex is simply $u_1 = 1, u^* = 0$ and $v_1 = 0, v^* = 1$. Our results show that NMF is convex on straight lines with high probability as the dimensionality of the problem increases, suggesting that the measure of such adversarial instances is small.

Inspired by the stochastic block model of social networks (Holland, Laskey, and Leinhardt 1983; Decelle et al. 2011) and the planted clique problem (Barak et al. 2016), we focus on a setting with a planted random solution. In the following section, we verify that the conclusions drawn from this model transfer to real-world datasets.

We assume that there is a planted optimal solution $(\mathbf{U}^*, \mathbf{V}^*)$ such that $\mathbf{X} = \mathbf{U}^*\mathbf{V}^*$, where entries of these matrices are sampled iid. This assumption follows from existing research on random input in neural networks (Li and Yuan 2017). Furthermore, we require matrices to be sampled from a class of distributions with good concentration properties, e.g., the half-normal distribution and bounded distributions. As is standard in random matrix theory (Vershynin 2010), we develop non-asymptotic results, which hold with a probability that grows as the matrices of shapes (n, r) and (r, m) increase in size. Consequently, we specify how r and m depend on n.

Assumption 1. *For $(\mathbf{U}, \mathbf{V}) \in R^{n\times r} \times R^{r\times m}$, we assume that the entries of the matrices $\mathbf{U}, \mathbf{V}$ are sampled iid from a continuous distribution with non-negative support that either (i) is bounded or (ii) can be expressed as a 1-Lipschitz function of a Gaussian distribution. As $n \to \infty$, we assume that r grows as n^γ up to a constant factor for $\gamma \in [1/2, 1]$, and m grows as n up to a constant factor.*

We are now ready to state our main result: the loss function in Equation 1 is convex on straight lines between points sampled as per Assumption 1, where one point can be the planted solution, with high probability. Thus, the loss satisfies our slightly weaker notion of star-convexity, and is convex on "most" straight lines. The probability increases as the size of the problem increases, suggesting a surprising benefit of high dimensionality. We also show similar results for the loss function in Equation 2 with unobserved data, under the assumption that the event of observing an entry occurs independently with constant probability p. The formal proof is given in the Appendix; we provide a proof sketch here.

Theorem 1. ***(Main)*** *Let matrices $\mathbf{U}_1, \mathbf{V}_1, \mathbf{U}_2, \mathbf{V}_2, \mathbf{U}^*, \mathbf{V}^*$ be sampled according to Assumption 1. Then there exists positive constants c_1, c_2, such that with probability $\geqslant 1 - c_1 \exp(-c_2 n^{1/3})$, the loss function $\ell(\mathbf{U}, \mathbf{V})$ in Equation 1 is convex on the straight line $(\mathbf{U}_1, \mathbf{V}_1) \to (\mathbf{U}_2, \mathbf{V}_2)$. The same holds along the line $(\mathbf{U}_1, \mathbf{V}_1) \to (\mathbf{U}^*, \mathbf{V}^*)$. It also holds if any entry (i, j) is observed independently with constant probability p, but with probability $\geqslant 1 - c_1 \exp(-c_2 r^{1/3})$.*

Proof Strategy Let us parameterize the NMF solution along the line $(\mathbf{U}_2, \mathbf{V}_2) \to (\mathbf{U}_1, \mathbf{V}_1)$ as

$$\hat{\mathbf{X}}(\lambda) = [\lambda\mathbf{U}_1 + (1-\lambda)\mathbf{U}_2]\,[\lambda\mathbf{V}_1 + (1-\lambda)\mathbf{V}_2]\,.$$

For proving Theorem 1, it suffices to show that the loss function $\ell(\lambda) = \frac{1}{2}\|\hat{\mathbf{X}}(\lambda) - \mathbf{X}\|_F^2$ is convex in λ with high probability on $[0, 1]$. Our strategy is to employ a sum-of-squares lower bound on the second derivative, and then use concentration of measure from random matrix theory. For fixed matrices $\mathbf{U}_1, \mathbf{U}_2, \mathbf{U}^*, \mathbf{V}_1, \mathbf{V}_2, \mathbf{V}^*$, the function $\ell(\lambda)$ is a fourth-degree polynomial in λ, so its second derivative w.r.t. λ is a second-degree polynomial in λ. For a general second-degree polynomial $p(x) = ax^2 + bx + c$, we have $p(x) = \frac{1}{a}\left[\left(ax + \frac{b}{2}\right)^2 + \left(ac - \frac{b^2}{4}\right)\right]$. If $a > 0$, as is the case here (see the Appendix), proving that $p(x)$ is positive for all x can be done by showing $ac \geqslant \frac{b^2}{4}$. $\ell''(\lambda) > 0$ would imply that $\ell(\lambda)$ is convex for all λ. Thus, we need to show that

$$2\,\|\mathbf{W}_2\|_F^2 \left(\|\mathbf{W}_1\|_F^2 + 2\langle \mathbf{W}_0, \mathbf{W}_2\rangle\right) \geqslant 3\left(\langle \mathbf{W}_1, \mathbf{W}_2\rangle\right)^2 \tag{3}$$

where the matrices $\mathbf{W}_0, \mathbf{W}_1, \mathbf{W}_2$ are given as $\mathbf{W}_0 = \mathbf{U}_2\mathbf{V}_2 - \mathbf{U}^*\mathbf{V}^*$, $\mathbf{W}_1 = (\mathbf{U}_1 - \mathbf{U}_2)\,\mathbf{V}_2 + \mathbf{U}_2\,(\mathbf{V}_1 - \mathbf{V}_2)$, $\mathbf{W}_2 = (\mathbf{U}_1 - \mathbf{U}_2)\,(\mathbf{V}_1 - \mathbf{V}_2)$. With slight abuse of notation, we have used $\langle \mathbf{A}, \mathbf{B}\rangle$ to denote $\mathrm{Tr}(\mathbf{A}\mathbf{B}^T)$ for matrices $\mathbf{A}, \mathbf{B}$ of the same shape. By replacing terms in Equation 3 with their means, we get

$$\begin{aligned} 2(4rmn\sigma^4)\left(6rmn\sigma^4 + 4rmn\mu^2\sigma^2 + 2rmn\sigma^4\right) \\ \geqslant 3\left(-4rmn\sigma^4\right)^2. \end{aligned} \tag{4}$$

Here, σ^2 is the variance of the distribution of the entries in the matrices, and μ is the mean. By just counting terms of order $(rmn\sigma^4)^2$, we see that the LHS has 64 such terms while the RHS has only 48. Thus, if all matrices $\mathbf{W}_0, \mathbf{W}_1$ and $\mathbf{W}_2$ would exactly be equal to their means, the inequality in Equation 3 would hold. In proving that it holds in general, we use concentration of measure results from random matrix theory to show that the terms are concentrated around their means and that large deviations are exponentially unlikely.

Concentration of Measure Consider the matrix $\mathbf{W}_2 = (\mathbf{U}_1 - \mathbf{U}_2)\,(\mathbf{V}_1 - \mathbf{V}_2)$. Given that all matrices are iid, we can center the variables so that

$\mathbf{W}_2 = (\mathbf{U}_1 - \mathbf{U}_2)(\mathbf{V}_1 - \mathbf{V}_2) = (\bar{\mathbf{U}}_1 - \bar{\mathbf{U}}_2)(\bar{\mathbf{V}}_1 - \bar{\mathbf{V}}_2)$, where the bar denotes the centered matrices. The term $\|\mathbf{W}_2\|_F^2$ can then be written as $\mathrm{Tr}\left[(\bar{\mathbf{V}}_1 - \bar{\mathbf{V}}_2)^T (\bar{\mathbf{U}}_1 - \bar{\mathbf{U}}_2)^T (\bar{\mathbf{U}}_1 - \bar{\mathbf{U}}_2)(\bar{\mathbf{V}}_1 - \bar{\mathbf{V}}_2)\right]$. Given that all matrix entries are independent as per Assumption 1, we would expect some concentration of measure to hold. Although Bernstein-type inequalities turn out to be too weak for our purposes, the field of random matrix theory offers stronger results for matrices with independent sub-Gaussian entries (Ahlswede and Winter 2002; Tropp 2012; Meckes and Szarek 2012). Using concentration of measure for traces of random matrices, we achieve the following inequality (see the Appendix).

$$P\left(\left|\|\mathbf{W}_2\|_F^2 - \mathbb{E}\left[\|\mathbf{W}_2\|_F^2\right]\right| > trn^2\right) \leqslant c_3 \exp\left(-c_4 \min(t^2, t^{1/2})n\right) \tag{5}$$

where c_3, c_4 are positive constants. In expressions for some terms in Equation 3, however, we are not able to center all variables. For such expressions, we get similar but slightly weaker concentration results, where the exponent in the RHS of Equation 5 scales as $n^{1/3}$ instead of n (see the Appendix).

Proof Sketch Given that $\mathbb{E}\left[\|\mathbf{W}_2\|_F^2\right] = 4rmn\sigma^4$, Equation 5 says that the probability of $\|\mathbf{W}_2\|_F^2$ deviating from its mean by a relative factor ϵ is less than $c_3 \exp\left(-c_5\epsilon^2 n\right)$ for some small ϵ. By applying similar arguments to terms $\langle \mathbf{W}_0, \mathbf{W}_2 \rangle$ and $\langle \mathbf{W}_1, \mathbf{W}_2 \rangle$, we show that the probability of them deviating by a relative factor ϵ is less than $c_6 \exp\left(-c_7\epsilon^2 n^{1/3}\right)$. $\|\mathbf{W}_1\|_F^2$ is a problematic term, containing a term of type $\mathrm{Tr}\left(\bar{\mathbf{V}}_1 - \bar{\mathbf{V}}_2\right)^T \mu_1^T \mu_1 \left(\bar{\mathbf{V}}_1 - \bar{\mathbf{V}}_2\right)$, which has weak concentration properties. Even so, since matrices of type $\mathbf{A}^T\mathbf{A}$ are p.s.d. due to non-negative traces, this term is non-negative. Moreover, we can simply omit $\|\mathbf{W}_1\|_F^2$ to lower bound the convexity because the term appears on the LHS of Equation 3. Using union bound, we bound the probability of at least one term deviating with a relative factor ϵ by $c_1 \exp\left(-c_8\epsilon^2 n^{1/3}\right)$ for positive constants c_1, c_8. Now, we set $\epsilon = 0.01$. If no term deviates by a factor of more than 0.01, then Equation 4 still holds as $0.99^2 \cdot 64 \geqslant 1.01^2 \cdot 48$. Thus, the inequality is violated with probability at most $c_1 \exp\left(-c_2 n^{1/3}\right)$ for positive c_1, c_2. ■

Proof Sketch for Unobserved Data If the entries in Equation 2 are "observed" independently with probability p, for fixed matrices $\mathbf{U}_1, \mathbf{U}_2, \mathbf{U}^*, \mathbf{V}_1, \mathbf{V}_2, \mathbf{V}^*$ such that Theorem 1 holds, we have

$$\begin{aligned}\mathbb{E}\left[\ell''(\lambda)\right] &= \mathbb{E}\left[\sum_{ij} \hat{1}_{(i,j)} \left(\hat{\mathbf{X}'}_{ij}^2 + \hat{\mathbf{X}''}_{ij}(\hat{\mathbf{X}}_{ij} - \mathbf{X}_{ij})\right)\right] \\ &= p \sum_{ij} \left(\hat{\mathbf{X}'}_{ij}^2 + \hat{\mathbf{X}''}_{ij}(\hat{\mathbf{X}}_{ij} - \mathbf{X}_{ij})\right) \\ &\geqslant 0.\end{aligned}$$

Thus, the expectation of $\ell(\lambda)$ is convex. To show that it is convex with high probability, we first observes that with high probability, no entry (i, j) in $\ell''(\lambda)$ is particularly large. Assuming this holds via union bound, for fixed matrices $\mathbf{U}_1, \mathbf{U}_2, \mathbf{U}^*, \mathbf{V}_1, \mathbf{V}_2, \mathbf{V}^*$ with elements that are "observed" independently with probability p, we get that $\ell''(\lambda)$ is concentrated around its convex mean via Hoeffding bounds. ■

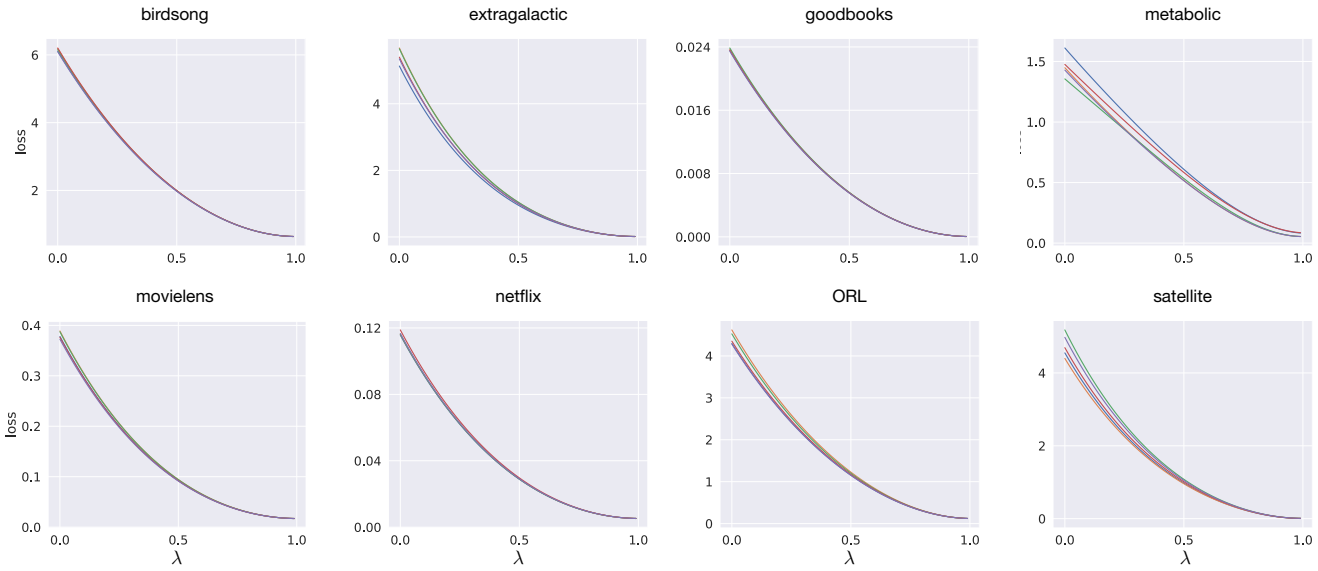

Figure 3: The NMF loss surface along the straight line from a random point w_0 to a local optima w^* found via gradient descent (from independent starting points). We overlap five independent lines; zoom in for detail. As our theoretical results predict, the loss surface is convex on these straight lines for all real-world datasets.

name	description	shape (n, m, r)	sparsity	reference
birdsong	bird call time series	$(5120, 1246, 88)$		(Flenner and Hunter 2017)
extragalactic	spectra of extragalactic sources	$(2760, 2820, 10)$		(Zhu 2016)
goodbooks	book ratings	$(10000, 43461, 50)$	0.0022	(Kula 2017)
metabolic	yeast cell metabolic activity	$(9335, 36, 3)$		(Li and Ngom 2013)
movielens	movie ratings	$(3953, 6041, 20)$	0.0419	(Harper and Konstan 2016)
netflix	movie/tv-show ratings	$(47928, 8963, 20)$	0.0121	(Zhou et al. 2008)
ORL faces	black and white facial images	$(400, 10304, 49)$		(Li et al. 2001).
satellite	hyperspectral satellite images	$(162, 94249, 4)$		(Erichson et al. 2018).

Table 1: Dataset details. References contain suggested rank r and previous usage (see the Appendix for details).

Experiments

Verifying Theoretical Predictions

To verify that the conclusions from our theoretical model hold more broadly, we now empirically study real-world datasets previously used in NMF literature. A few datasets have ranks outside the scope of our theoretical model, but they still display star-convexity properties, indicating that star-convexity might be a more general phenomenon. We focus on a handful of representative datasets spanning image analysis, scientific applications, and collaborative filtering. In Table 1, we list these datasets together with their sparsity. We use decomposition ranks as per the values previously reported in the literature. We perform a non-negative matrix factorization via gradient descent, starting with randomly initialized data. To enable comparison between datasets, we scale all data matrices so that the variance of observed entries is one, and divide the loss function by the number of (observed) entries. We initialize decomposition matrices using the half-normal distribution, which is scaled so that the mean matches with that of the dataset. For simplicity, we use the same learning rate of $1\mathrm{e}{-5}$ for all datasets and run gradient descent until the rate of relative improvement in the loss falls below $1\mathrm{e}{-7}$. This procedure gives good convergence for all datasets (see the Appendix). As is standard in NMF, we compute the loss only over observed entries for the collaborative filtering datasets with unobserved ratings (movielens, netflix, and goodbooks) (Zhang et al. 2006). In Figure 3, we plot the loss function from an initialization point to an independent local optima. In Figure 4, we plot the loss function between two random points drawn from the initialization distribution—observe that the loss is convex. These results agree with our theoretical model, and we conclude that many real-world matrices can be decomposed as a low-rank matrix $\mathbf{U}^*\mathbf{V}^*$ with the convexity properties our theoretical results suggest, plus a "noise term" that must have a small norm since the loss $\ell(\mathbf{U}^*, \mathbf{V}^*)$ is small (see the Appendix).

Ablation Experiments

Theorem 1 suggests that, as the matrices become larger, NMF is increasingly likely to be star-convex. To test if this is the

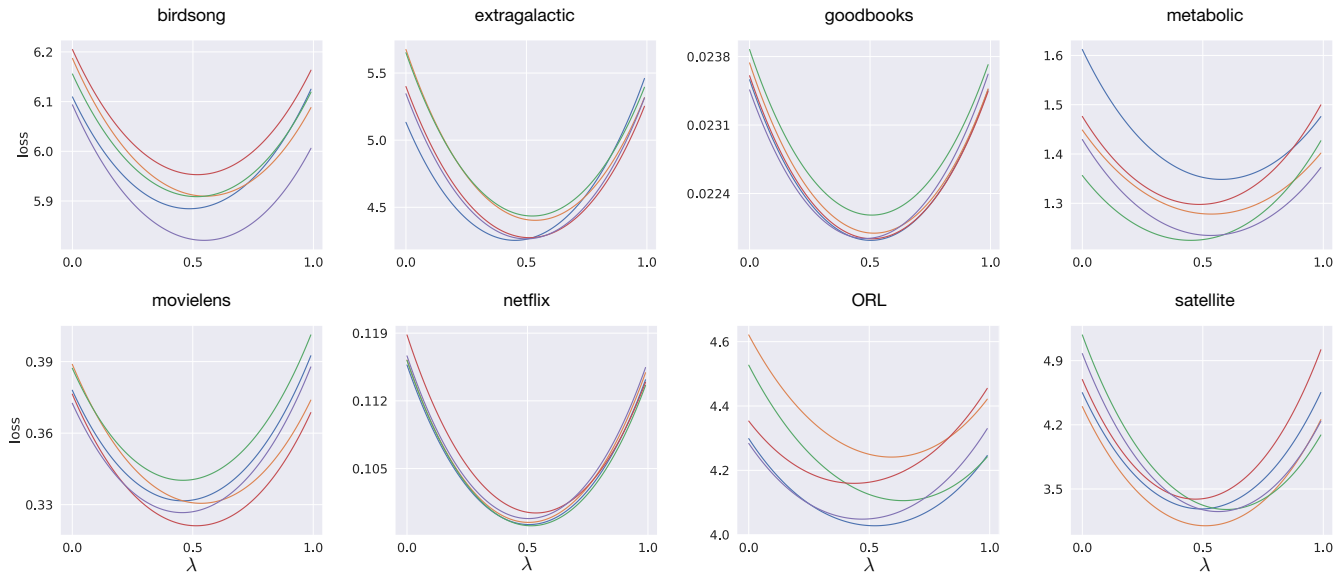

Figure 4: We here illustrate the NMF loss surface on straight paths connecting two random points for 8 real-world datasets. We overlap five independent lines for each dataset. Note that the curves are always convex, suggesting that the loss surface is "typically" convex as our theoretical results suggest.

Figure 5: We illustrate how the relative deviation $\frac{\sigma}{\mu}$ of the curvature in Equation 6 depends on the dataset's size. We normalize by μ to avoid uniform scaling. For all datasets, the relative deviations decrease with more samples, suggesting that the (positive) curvature becomes increasingly concentrated around its mean for larger matrices.

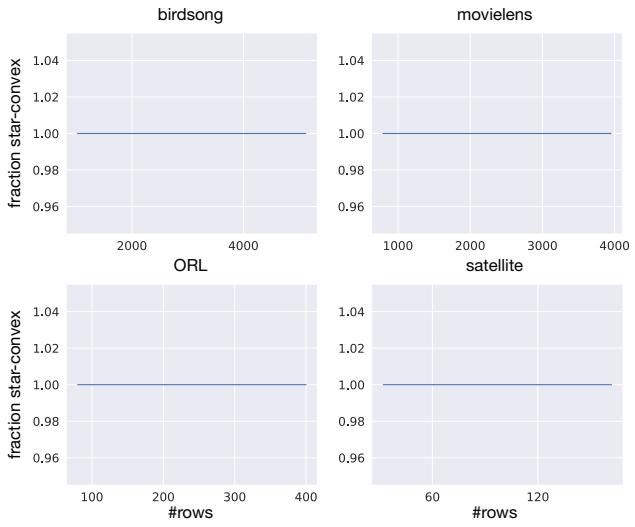

Figure 6: We here show the fraction of sampled curvatures (as in Equation 6) that are positive as the dimensionality of the dataset is varied. Note that it is always 1, implying that we have star-convexity even for smaller problems, even though the curvature typically fluctuates more for such problems as per Figure 5.

case for our real-world datasets, we perform ablation experiments by varying the dimensions of the matrices. We decrease the number of data points n by subsampling rows and columns uniformly randomly. Our measure of curvature at a point $\mathbf{x}$, given some optimal solution $\mathbf{x}^*$, is

$$\alpha(\mathbf{x}) = \min_{\lambda \in [0,1]} \ell''\left(\lambda \mathbf{x} + (1-\lambda)\mathbf{x}^*\right). \tag{6}$$

Note that $\alpha \geqslant 0$ implies star-convexity. In practice, we obtain $\mathbf{x}^*$ from gradient descent; finding the absolute minima remains a challenge. For each dataset and subsample rate, we find 50 optima and evaluate the curvature from 50 random points, thus obtaining 2500 samples of α. Figure 5 shows how the relative deviation $\frac{\sigma}{\mu}$ of α decreases as the dataset becomes larger. Figure 6 that shows the fraction of non-negative curvature as a function of input dimensionality—we confirm that the sampled curvatures typically are positive. This can also be considered as a quantitative depiction of Figure 3. Figures 5 and 6 together show that the curvature becomes increasingly concentrated around its positive mean for larger matrices, suggesting that the star-convexity phenomenon is valid beyond our simplistic theoretical model. In the Appendix, we also illustrate the spectrum of singular values of $\mathbf{U}^*$ for various datasets, and of random matrices of the same shapes as $\mathbf{U}^*$. The spectra generally share the quality of having a single large singular value and a tail of smaller values; however, real-world solutions typically have a tail with larger values.

Implications for Neural Networks

We have seen how increasing the number of parameters makes NMF problems more likely to be star-convex, while also making the curvature tend towards its positive mean, as displayed in Figure 5. Theorem 1 suggests that this is a result of concentration of measure, and it is natural to believe that a similar phenomenon would occur in the context of neural networks. It has previously been observed how neural networks are locally convex (Izmailov et al. 2018), and also how overparameterization is important in deep learning (Arora, Cohen, and Hazan 2018; Frankle and Carbin 2018). Based on our observations in NMF, we hypothesize that a major benefit of overparameterization is in making the loss surface more convex via concentration of measure w.r.t. the weights.

To verify this hypothesis, we consider image classification on CIFAR10 with Resnet networks (He et al. 2016) trained with standard parameters (see the Appendix). Networks are

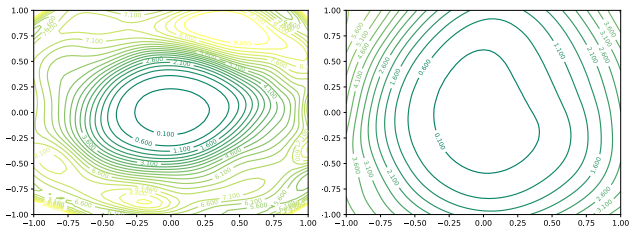

Figure 7: The loss landscape of a 110-layer Resnet architecture at epoch 200 along two random directions, visualized as in (Li et al. 2018). The network in the bottom image is four times as wide (i.e. has four times as many channels per layer), and its loss landscape is increasingly convex. In Table 2, we generalize this idea and show that the length scale of local convexity increases with network width.

	32-layers			44-layers		
epoch	k=1	k=2	k=4	k=1	k=2	k=4
0	**1.0** $\pm$ 0.0	**1.0** $\pm$ 0.0	**1.0** $\pm$ 0.0	**1.0** $\pm$ 0.0	**1.0** $\pm$ 0.0	**1.0** $\pm$ 0.0
100	0.77 $\pm$ 0.035	0.8 $\pm$ 0.031	**0.84** $\pm$ 0.026	0.72 $\pm$ 0.041	0.79 $\pm$ 0.037	**0.83** $\pm$ 0.028
200	0.61 $\pm$ 0.036	0.68 $\pm$ 0.036	**0.8** $\pm$ 0.031	0.66 $\pm$ 0.033	0.68 $\pm$ 0.034	**0.76** $\pm$ 0.033
300	0.55 $\pm$ 0.037	0.68 $\pm$ 0.037	**0.82** $\pm$ 0.032	0.57 $\pm$ 0.036	0.68 $\pm$ 0.036	**0.78** $\pm$ 0.032
	56-layers			68-layers		
epoch	k=1	k=2	k=4	k=1	k=2	k=4
0	**1.0** $\pm$ 0.0	**1.0** $\pm$ 0.0	**1.0** $\pm$ 0.0	**1.0** $\pm$ 0.0	**1.0** $\pm$ 0.0	**1.0** $\pm$ 0.0
100	0.7 $\pm$ 0.036	0.81 $\pm$ 0.03	**0.84** $\pm$ 0.03	0.71 $\pm$ 0.032	0.76 $\pm$ 0.03	**0.87** $\pm$ 0.026
200	0.63 $\pm$ 0.039	0.67 $\pm$ 0.034	**0.8** $\pm$ 0.031	0.6 $\pm$ 0.036	0.71 $\pm$ 0.031	**0.8** $\pm$ 0.029
300	0.57 $\pm$ 0.035	0.66 $\pm$ 0.035	**0.79** $\pm$ 0.033	0.58 $\pm$ 0.036	0.67 $\pm$ 0.033	**0.81** $\pm$ 0.03
	80-layers			110-layers		
epoch	k=1	k=2	k=4	k=1	k=2	k=4
0	**1.0** $\pm$ 0.0	**1.0** $\pm$ 0.0	**1.0** $\pm$ 0.0	**0.94** $\pm$ 0.016	**0.94** $\pm$ 0.018	**0.94** $\pm$ 0.016
100	0.72 $\pm$ 0.036	**0.85** $\pm$ 0.03	0.81 $\pm$ 0.027	0.79 $\pm$ 0.032	0.75 $\pm$ 0.04	**0.91** $\pm$ 0.019
200	0.59 $\pm$ 0.036	0.7 $\pm$ 0.038	**0.77** $\pm$ 0.031	0.71 $\pm$ 0.037	0.71 $\pm$ 0.036	**0.82** $\pm$ 0.03
300	0.63 $\pm$ 0.036	0.71 $\pm$ 0.036	**0.79** $\pm$ 0.03	0.63 $\pm$ 0.037	0.68 $\pm$ 0.034	**0.82** $\pm$ 0.033

Table 2: Typical length scales of local convexity for Resnet networks with various width (indicated by k), depth, and training. We sample 25 random "lines" of length 1 in parameter space, centered on current parameters, and report mean length of convex subset of such "lines" and the std of this statistic. Increasing width makes the loss surface increasingly locally convex.

typically only locally convex, a property we quantify as the length of subsets of random "lines" in parameter space along which the training loss function is convex. Formally, we sample a random direction $\mathbf{r}$ and then consider an interval of length one along this direction, centered around the current parameters $\mathbf{w}$, i.e., $\mathbf{w} + \lambda\mathbf{r}$ for $\lambda \in [-1/2, 1/2]$. We then define the "convexity length scale" as the length of the maximal sub-interval containing $\mathbf{w}$ on which $\ell(\mathbf{w} + \lambda\mathbf{r})$ is convex. Directions are sampled from Gaussian distributions and then normalized for each network filter f to have the same norm as the weights of f. Table 2 shows how this length scale of local convexity varies with depth, width, and training, where width is varied by multiplying the number of channels by k. Indeed, increasing width makes the landscape increasingly locally convex, supporting our hypothesis.

Related Work

As the success of deep learning has become widespread, many researchers have empirically investigated its behavior on real-world datasets (Li and Yuan 2017; Zhang et al. 2016). In the context of sharp vs flat local minima (Keskar et al. 2016), Li et al. (2018) illustrate how increasing the width improved flatness in a Resnet network, an observation that Table 2 quantifies. Our work was initially motivated by studies on local and star-convexity in neural networks due to Kleinberg, Li, and Yuan (2018), Izmailov et al. (2018) and Zhou et al. (2019). Whereas such previous work empirically observes star-convexity and investigates its implications, we prove that this benign property arises simply from concentration of measure, albeit in the simpler NMF case. We intentionally focus on dense NMF problems to explain its practical success, leaving e.g., sparsity for future work (Richard and Montanari 2014). A common theme in non-convex optimization more generally is that functions with only saddle points and global minima can be solved via SGD (Ge et al. 2015). We note that problems with such properties, for example, tensor decomposition, can be efficiently optimized. Our work, on the other hand, addresses an NP-hard optimization problem, utilizing statistical assumptions on the input to achieve positive results. There is extensive work on non-worst-case analyses of algorithms and machine learning models, and on what problem distributions can guarantee tractability (Bilu and Linial 2012; Ackerman and Ben-David 2009; Afshani, Barbay, and Chan 2017). On the positive side, Arora et al. (2012) have proposed an exact algorithm for NMF that runs in polynomial time for small constant r, and there are positive results for so-called "separable" NMF (Donoho and Stodden 2004). Our work is also related to the analysis of algorithms where instances have "planted" solutions, for instance, the planted clique problem (Barak et al. 2016) and the stochastic block model (Holland, Laskey, and Leinhardt 1983; Decelle et al. 2011).

Conclusions

This paper revisits NMF, a non-convex optimization problem in machine learning. We have shown that NMF is typically star-convex, provably for a natural average-case model and empirically on an extensive set of real-world datasets. Additionally, we have shown how network width improves local convexity of neural networks. Our results support the counter-intuitive observation that optimization might sometimes be *easier* in higher dimensions due to concentration of measure.

Acknowledgements

This material is based upon work supported by the National Science Foundation under Grant Number CCF-1522054. This material is also based upon work supported by the Air Force Office of Scientific Research under award number FA9550-18-1-0136. This research is supported in part by the grants from Facebook, the National Science Foundation (III-1618134, III-1526012, IIS1149882, IIS-1724282, and TRIPODS- 1740822), the Office of Naval Research DOD (N00014- 17-1-2175), Bill and Melinda Gates Foundation. We are thankful for generous support by Zillow and SAP America Inc. We are also grateful from generous support from the TTS foundation. This work was partially supported by the Cornell Center for Materials Research with funding from the NSF MRSEC program (DMR-1719875).

Ethics Statement

Our work extends the research community's understanding of the fundamental properties of non-convex optimization, which is ubiquitously used in critical problems nowadays. Our proofs and empirical observations advance research in NMF and deep learning, which have proved instrumental in many real-life problems. We do not perceive any entity to be directly put at a disadvantage or to be harmed due to any system failure.

References

Ackerman, M.; and Ben-David, S. 2009. Clusterability: A theoretical study. In *Artificial Intelligence and Statistics*, 1–8.

Afshani, P.; Barbay, J.; and Chan, T. M. 2017. Instance-optimal geometric algorithms. *Journal of the ACM (JACM)* 64(1): 3.

Ahlswede, R.; and Winter, A. 2002. Strong converse for identification via quantum channels. *IEEE Transactions on Information Theory* 48(3): 569–579.

Arora, S.; Cohen, N.; and Hazan, E. 2018. On the optimization of deep networks: Implicit acceleration by overparameterization. *arXiv preprint arXiv:1802.06509* .

Arora, S.; Ge, R.; Kannan, R.; and Moitra, A. 2012. Computing a nonnegative matrix factorization–provably. In *Proceedings of the forty-fourth annual ACM symposium on Theory of computing*, 145–162. ACM.

Barak, B.; Hopkins, S. B.; Kelner, J.; Kothari, P.; Moitra, A.; and Potechin, A. 2016. A nearly tight sum-of-squares lower bound for the planted clique problem. In *2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)*, 428–437. IEEE.

Berne, O.; Joblin, C.; Deville, Y.; Smith, J.; Rapacioli, M.; Bernard, J.; Thomas, J.; Reach, W.; and Abergel, A. 2007. Analysis of the emission of very small dust particles from Spitzer spectro-imagery data using blind signal separation methods. *Astronomy & Astrophysics* 469(2): 575–586.

Bilu, Y.; and Linial, N. 2012. Are stable instances easy? *Combinatorics, Probability and Computing* 21(5): 643–660.

Blum, A.; and Rivest, R. L. 1989. Training a 3-node neural network is NP-complete. In *Advances in neural information processing systems*, 494–501.

Candès, E. J.; and Recht, B. 2009. Exact matrix completion via convex optimization. *Foundations of Computational mathematics* 9(6): 717.

Cortes, C.; and Vapnik, V. 1995. Support-vector networks. *Machine learning* 20(3): 273–297.

Decelle, A.; Krzakala, F.; Moore, C.; and Zdeborová, L. 2011. Asymptotic analysis of the stochastic block model for modular networks and its algorithmic applications. *Physical Review E* 84(6): 066106.

Donoho, D.; and Stodden, V. 2004. When does non-negative matrix factorization give a correct decomposition into parts? In *Advances in neural information processing systems*.

Erichson, N. B.; Mendible, A.; Wihlborn, S.; and Kutz, J. N. 2018. Randomized nonnegative matrix factorization. *Pattern Recognition Letters* 104: 1–7.

Flenner, J.; and Hunter, B. 2017. A deep non-negative matrix factorization neural network. *Semantic Scholar* .

Frankle, J.; and Carbin, M. 2018. The lottery ticket hypothesis: Finding sparse, trainable neural networks. *arXiv preprint arXiv:1803.03635* .

Ge, R.; Huang, F.; Jin, C.; and Yuan, Y. 2015. Escaping from saddle points—online stochastic gradient for tensor decomposition. In *Conference on Learning Theory*.

Gillis, N. 2012. Sparse and unique nonnegative matrix factorization through data preprocessing. *Journal of Machine Learning Research* 13(Nov): 3349–3386.

Harper, F. M.; and Konstan, J. A. 2016. The movielens datasets: History and context. *Acm transactions on interactive intelligent systems (tiis)* 5(4): 19.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Holland, P. W.; Laskey, K. B.; and Leinhardt, S. 1983. Stochastic blockmodels: First steps. *Social networks* 5(2): 109–137.

Izmailov, P.; Podoprikhin, D.; Garipov, T.; Vetrov, D.; and Wilson, A. G. 2018. Averaging weights leads to wider optima and better generalization. *arXiv preprint arXiv:1803.05407* .

Keskar, N. S.; Mudigere, D.; Nocedal, J.; Smelyanskiy, M.; and Tang, P. T. P. 2016. On large-batch training for deep learning: Generalization gap and sharp minima. *arXiv preprint arXiv:1609.04836* .

Kleinberg, R.; Li, Y.; and Yuan, Y. 2018. An Alternative View: When Does SGD Escape Local Minima? *arXiv preprint arXiv:1802.06175* .

Koren, Y.; Bell, R.; and Volinsky, C. 2009. Matrix factorization techniques for recommender systems. *Computer* (8): 30–37.

Kula, M. 2017. Mixture-of-tastes Models for Representing Users with Diverse Interests. *arXiv preprint arXiv:1711.08379* .

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

Lee, D. D.; and Seung, H. S. 2001. Algorithms for non-negative matrix factorization. In *Advances in neural information processing systems*, 556–562.

Lee, J. C.; and Valiant, P. 2016. Optimizing star-convex functions. In *2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)*, 603–614. IEEE.

Li, H.; Xu, Z.; Taylor, G.; Studer, C.; and Goldstein, T. 2018. Visualizing the loss landscape of neural nets. In *Advances in Neural Information Processing Systems*, 6389–6399.

Li, S. Z.; Hou, X.; Zhang, H.; and Cheng, Q. 2001. Learning spatially localized, parts-based representation. *CVPR (1)* 207: 212.

Li, Y.; and Ngom, A. 2013. The non-negative matrix factorization toolbox for biological data mining. *Source code for biology and medicine* 8(1): 10.

Li, Y.; and Yuan, Y. 2017. Convergence analysis of two-layer neural networks with relu activation. In *Advances in Neural Information Processing Systems*, 597–607.

Lu, C.; and Tang, X. 2015. Surpassing human-level face verification performance on LFW with GaussianFace. In *Twenty-ninth AAAI conference on artificial intelligence*.

Luo, X.; Zhou, M.; Xia, Y.; and Zhu, Q. 2014. An efficient non-negative matrix-factorization-based approach to collaborative filtering for recommender systems. *IEEE Transactions on Industrial Informatics* 10(2): 1273–1284.

Mao, Y.; Saul, L. K.; and Smith, J. M. 2006. Ides: An internet distance estimation service for large networks. *IEEE Journal on Selected Areas in Communications* 24(12): 2273–2284.

Meckes, M.; and Szarek, S. 2012. Concentration for noncommutative polynomials in random matrices. *Proceedings of the American Mathematical Society* 140(5): 1803–1813.

Mnih, V.; Kavukcuoglu, K.; Silver, D.; Rusu, A. A.; Veness, J.; Bellemare, M. G.; Graves, A.; Riedmiller, M.; Fidjeland, A. K.; Ostrovski, G.; et al. 2015. Human-level control through deep reinforcement learning. *Nature* 518(7540): 529.

Pardalos, P. M.; and Vavasis, S. A. 1991. Quadratic programming with one negative eigenvalue is NP-hard. *Journal of Global Optimization* 1(1): 15–22.

Richard, E.; and Montanari, A. 2014. A statistical model for tensor PCA. In *Advances in Neural Information Processing Systems*, 2897–2905.

Schmidt, M. N.; Larsen, J.; and Hsiao, F.-T. 2007. Wind noise reduction using non-negative sparse coding. In *2007 IEEE workshop on machine learning for signal processing*, 431–436. IEEE.

Suram, S. K.; Xue, Y.; Bai, J.; Le Bras, R.; Rappazzo, B.; Bernstein, R.; Bjorck, J.; Zhou, L.; van Dover, R. B.; Gomes, C. P.; et al. 2016. Automated phase mapping with AgileFD and its application to light absorber discovery in the V–Mn–Nb oxide system. *ACS combinatorial science* 19(1): 37–46.

Trindade, G. F.; Abel, M.-L.; and Watts, J. F. 2017. Non-negative matrix factorisation of large mass spectrometry datasets. *Chemometrics and Intelligent Laboratory Systems* 163: 76–85.

Tropp, J. A. 2012. User-friendly tail bounds for sums of random matrices. *Foundations of computational mathematics* 12(4): 389–434.

Vavasis, S. A. 2009. On the complexity of nonnegative matrix factorization. *SIAM Journal on Optimization* 20(3): 1364–1377.

Vershynin, R. 2010. Introduction to the non-asymptotic analysis of random matrices. *arXiv preprint arXiv:1011.3027* .

Zhang, C.; Bengio, S.; Hardt, M.; Recht, B.; and Vinyals, O. 2016. Understanding deep learning requires rethinking generalization. *arXiv preprint arXiv:1611.03530* .

Zhang, S.; Wang, W.; Ford, J.; and Makedon, F. 2006. Learning from incomplete ratings using non-negative matrix factorization. In *Proceedings of the 2006 SIAM international conference on data mining*, 549–553. SIAM.

Zhou, Y.; Wilkinson, D.; Schreiber, R.; and Pan, R. 2008. Large-scale parallel collaborative filtering for the netflix prize. In *International conference on algorithmic applications in management*, 337–348. Springer.

Zhou, Y.; Yang, J.; Zhang, H.; Liang, Y.; and Tarokh, V. 2019. SGD converges to global minimum in deep learning via star-convex path. *arXiv preprint arXiv:1901.00451* .

Zhu, G. 2016. Nonnegative Matrix Factorization (NMF) with Heteroscedastic Uncertainties and Missing data. *arXiv preprint arXiv:1612.06037* .