NUCLEAR $p$-NORMS FOR LARGE TENSOR COMPLETION

Anonymous authors
Paper under double-blind review

ABSTRACT

We present algorithms for tensor completion using regularizers based on tensor nuclear $p$-norms. For the particular case of the nuclear $\infty$-norm, we generalize to higher-order tensors the theoretical guarantees of the max-norm for matrix completion. From a practical perspective, we present two algorithms based on stochastic gradients to regularize the canonical decomposition of tensors, and show on large-scale benchmark datasets for knowledge base completion that (a) contrary to what suggested prior results in the literature, the canonical decomposition of tensors can achieve state-of-the-art level performance on FB15K and WN, and (b) our new regularizations reach or outperform the state-of-the-art on task where the canonical decomposition alone is not reaching it. In particular, we provide evidence that the nuclear $3$-norm can replace the structures and/or regularization terms of existing link prediction models, and leads to better performance.

1 INTRODUCTION

The task of tensor completion is the extension of matrix completion to higher-order tensors which is one of the main tools for collaborative filtering, and, more generally, recommendation (Koren et al., 2009). Higher-order tensors capture multiple types of interactions between entities, and have been extensively used for link prediction or representation learning of relational databases such as knowledge bases (Nickel et al., 2016; Nguyen, 2017).

Matrix completion has been studied extensively in the past decade, and effective regularization techniques with strong statistical guarantees have been proved, such as the (weighted) nuclear-(or trace)-norm (Srebro et al., 2005; Candès & Recht, 2009; Foygel et al., 2011) or the max-norm (Srebro et al., 2005; Cai et al., 2016). These lead to convex optimization problems that can be solved directly (Cai et al., 2010; Jaggi et al., 2010), or to non-convex approximations based on stochastic or proximal gradients that scale better (Rennie & Srebro, 2005; Koren et al., 2009; Lee et al., 2010).

Unfortunately, these algorithms do not extend naturally to tensors. For tensors, several notions of rank exist and the canonical rank, which is the smallest number of rank one tensor whose sum is the original tensor is NP-hard to compute (see e.g. Kolda & Bader and references therein for a comprehensive review on tensor decompositions). Furthermore, in the context of matrix completion, a regularization of the Frobenius norm of the factors leads naturally to the nuclear norm, which is not the case for tensors. Given the difficulties associated with the tensor canonical rank, this has led several authors to propose tensor norms based on trace norms of matricizations (Tomioka et al., 2010; Gandy et al., 2011; Wimalawarne et al., 2014; Romera-Paredes et al.). However, the resulting algorithms do not scale to very large tensors.

We focus on norm-based regularization schemes for tensors, based on the so-called nuclear $p$-norms of tensors (Friedland & Lim, 2014), whose unit balls are the convex hull of rank-1 tensors with unit $p$-norm factors. These include the tensor trace norm (nuclear 2-norm) as studied by e.g., Yuan & Zhang (2016) and Cheng et al. (2016). However, it also includes two other norms of potential interest: for 3D tensors, the nuclear 3-norm has a smooth multi-convex variational form which is straightforward to minimize with stochastic gradient descent. Second, the nuclear $\infty$-norm, whose unit ball is the convex hull of rank-1 tensors with values in $\{-1, 1\}$. To the best of our knowledge, for matrices, the nuclear $\infty$-norm has not been considered as a regularizer. However, all recovery and generalization guarantees for the max-norm of matrices (Srebro & Shraibman, 2005; Cai et al., 2016) are based on the relationship between the max-norm and the nuclear $\infty$-norm. Whereas it is unclear how to extend the max-norm for general tensors, the nuclear $\infty$-norm is a tensor norm already; and
the statistical guarantees in terms of generalization error for matrices with max-norm regularization are straightforward to extend to tensors with nuclear $\infty$-norm regularization.

This paper presents several contributions. First, we propose alternative regularizers for the canonical decomposition of tensors, and show how they can be optimized at scale with stochastic gradient descent (SGD). Second, we perform extensive experiments on several datasets for link prediction in knowledge bases, which is an archetypical task for large-scale tensor completion, with a challenging state of the art [Nickel et al., 2011; Bordes et al., 2013; Trouillon et al., 2016; Kadlec et al., 2017a]. Our results show on one hand that the level of performance attainable with the classical CP model has been underestimated on datasets such as FB15K and WN, because too low ranks were systematically used. On other (smaller and larger) datasets which actually do require regularization, our results show that the usual methods for the canonical decomposition are extremely difficult to tune, and require a careful selection of the rank and early stopping to achieve acceptable results. On the other hand, the regularizations we propose and in particular the nuclear 3-norm, have a much better behavior and achieve similar or better results. In particular, they show little dependence on the optimization parameters and rank of the decomposition, as long as the rank is large enough, leaving the regularization parameter as the only critical tuning parameter of the method. We also provide empirical evidence that the nuclear 3-norm can be plugged in other models to replace the usual regularizers, with similar or better performances.

2 Tensor completion with nuclear $p$-norm regularization

Three dimensional tensor completion is the problem of estimating a tensor $\hat{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ given a training set of sampled entries $S = \{(i_1, j_1, k_1, y_1), \ldots, (i_{|S|}, j_{|S|}, k_{|S|}, y_{|S|})\}$, where each sample contains the indices in the tensor and an associated observed target binary value. There are two main objectives associated with matrix or tensor completion. The first one is to predict the values of missing entries, and is the extension to higher-order arrays of the ratings prediction task in collaborative filtering. In that case, a test set is composed of a triple of indices $(i, j, k)$ together with a desired value $y$ that is directly compared to the predicted value $\hat{X}_{i,j,k}$. The second objective is the extension to tensors of one-class collaborative filtering or link prediction (see e.g. [Pan et al., 2008]). In that case, we assume that there is an underlying binary tensor from which we only observe some positive entries and the goal is to rank unknown positive entries above (unknown) negative entries. The evaluation is then performed using a ranking performance metric. For instance, in link prediction in relational data, a (training) sample is a triple $(i, j, k)$ encoding that entity $i$ is linked to $k$ by relationship $j$, such as $(Washington, capital_of, USA)$. A common task is to predict the possible relationships between two entities. In this case, a test sample is defined by a pair of entities $(i, k)$ and a relationship $j$ that the system needs to predict. The performance is a function of the rank of $j$ in the list of relationships $j'$ ordered by decreasing values of $\hat{X}_{i,j',k}$. In the remainder of this work, tensor completion will always refer to link prediction, where missing entries are considered as zeros.

Given the training set $S$, we propose to learn the tensor by solving a regularized empirical risk minimization problem of the form:

$$\hat{X} = \arg\min_{X \in \mathbb{R}^{n_1 \times n_2 \times n_3}} \sum_{(i,j,k,y) \in S} \ell(X; i, j, k, y) + \lambda \Omega(X),$$

where $\ell(X; i, j, k, y)$ is the instantaneous loss associated to the training sample $(i, j, k, y)$, $\Omega$ is a regularizer and $\lambda > 0$ is a regularization coefficient. For ratings prediction, $\ell$ is usually the square error loss $\ell(X; i, j, k, y) = (X_{i,j,k} - y)^2$. For link prediction, $S$ is only composed of positive examples, and the loss can be the multiclass log-loss: $\ell(X; i, j, k, l) = -X_{i,j,k} + \log \left( \sum_{j} \exp(X_{i,j,k}) \right)$.

We are interested in the nuclear $p$-norms for tensors, as defined by [Friedland & Lim, 2014], whose unit balls are the convex hull of rank-1 tensors with unit $p$-norm factors. More precisely, denoting by $\otimes$ the tensor product and by $\| \cdot \|_p$ the $p$-norm of a real-valued vector, the nuclear $p$-norm of $\|X\|_{\nu,p}$

---

4To simplify notations, we focus on order-3 tensors in this paper. However, our approach extends naturally to order-$D$ tensors.
\( \mathbf{X} \in \mathbb{R}^{n_1 \times n_2 \times n_3} \) for \( p \in [1, +\infty] \), is

\[
\| \mathbf{X} \|_{*,p} := \min \left\{ \sum_{r=1}^{R} \left| \sigma_r \right| \left| \mathbf{X} = \sum_{r=1}^{R} \sigma_r \mathbf{u}^{(1)}_r \otimes \mathbf{u}^{(2)}_r \otimes \mathbf{u}^{(3)}_r, \max_{d=1,..,R} \| \mathbf{u}^{(d)}_r \|_p \leq 1, R \in \mathbb{N} \right\}. \tag{2}
\]

where \( \sigma_r \in \mathbb{R} \) and \( \mathbf{u}^{(d)}_r \in \mathbb{R}^{n_d} \) for \( r \in \{1, \ldots, R\} \) and \( d \in \{1, 2, 3\} \). Friedland & Lim (2014) Proposition 4.3 guarantees that the minimum in (2) is attained for a rank \( R \leq n_1 n_2 n_3 \).

Note that nuclear \( p \)-norm is an atomic norm in the terminology of (Chandrasekaran et al., 2012) whose atoms are the rank-1 tensors \( \mathbf{u} \otimes \mathbf{v} \otimes \mathbf{w} \) with unit \( p \)-norm factors. Atomic norms are norms that induce solutions that are combinations of a small number of atoms. Indeed, here the 1-norm regularization on the weights \( \sigma_r \), is sparsifying and so will encourage a reduction of the number of terms in the sum, which correspond to learning a tensor with low canonical rank. One might question the choice of the 1-norm on \( \sigma \) and consider the possibility of using \( \alpha \)-norms. But for \( \alpha < 1 \) the corresponding tensor function is nonconvex and Friedland & Lim (2014) show that for \( \alpha > 1 \) the infimum is not attained and thus does not yield usable expansions.

The choice \( p = 2 \) in (2) yields the usual nuclear norm of tensors, or tensor trace norm, used for tensor completion by e.g. Yuan & Zhang (2016) and Cheng et al. (2016). It is the natural extension to tensors of the nuclear (or trace) norm for matrices, whose unit ball is the convex hull of rank-1 matrices with unit 2-norm factors. For reasons that we will describe later, in this paper we are mostly interested in the nuclear 3-norm and nuclear \( \infty \)-norm.

Scalable algorithms to find a global minimum of (1) with nuclear \( p \)-norm regularizers are not available, even though the problem is convex in \( \mathbf{X} \) as soon as \( \ell \) is. As for the case of matrix norms regularization, an alternative is to work with the variational formulations of the norms. To simplify notations, we introduce the set of canonical decompositions of a tensor of rank at most \( R \):

\[
\mathcal{U}_R(\mathbf{X}) = \left\{ (\mathbf{u}^{(d)}_r)_{d=1..3,r=1..R} \mid \mathbf{X} = \sum_{r=1}^{R} \mathbf{u}^{(1)}_r \otimes \mathbf{u}^{(2)}_r \otimes \mathbf{u}^{(3)}_r, \forall r, d, u^{(d)}_r \in \mathbb{R}^{n_d} \right\}. \tag{3}
\]

The nuclear \( p \)-norm admits the two following variational forms:

\[
\| \mathbf{X} \|_{*,p} = \min \left\{ \frac{1}{3} \sum_{r=1}^{R} \sum_{d=1}^{3} \| \mathbf{u}^{(d)}_r \|_p^3 \mid (\mathbf{u}^{(d)}_r) \in \mathcal{U}_R(\mathbf{X}), R \in \mathbb{N} \right\} \tag{4a}
\]

\[
= \min \left\{ \sum_{r=1}^{R} \max_{d=1..3} \| \mathbf{u}^{(d)}_r \|_p^3 \mid (\mathbf{u}^{(d)}_r) \in \mathcal{U}_R(\mathbf{X}), R \in \mathbb{N} \right\}. \tag{4b}
\]

The first one is proposed in Haefele & Vidal (2015). We prove both for the general case in the Appendix.

Given an estimated upper bound on the optimal \( R \), the original problem can then be re-written as a non-convex problem using one of the variational forms (4a) or (4b). For example, using (4a):

\[
\min_{(u^{(d)}_r)_{d=1..3,r=1..R} \in \mathcal{U}_R(\mathbf{X})} \sum_{r=1}^{R} \ell \left( \sum_{i,j,k,y} \mathbf{X}_{i,j,k} (u^{(1)}_r \otimes u^{(2)}_r \otimes u^{(3)}_r)_{i,j,k,y} \right) + \lambda \sum_{r=1}^{R} \sum_{d=1}^{3} \| \mathbf{u}^{(d)}_r \|_p^3. \tag{5}
\]

Note that each loss term depends on the data only via \( X_{i,j,k} = \sum_{r=1}^{R} u^{(1)}_r u^{(2)}_r u^{(3)}_r \), which involves only \( 3r \) parameters (and where \( u^{(d)}_{r,i} \) denotes the \( i \)th entry of \( u^{(d)}_r \)).

If \( R \) is large enough the global optimum of (5) is the same as that of (1). While the original formulation is convex in its tensor argument \( \mathbf{X} \), it is NP-hard to optimize. The advantage of (5) is that whenever \( \ell \) is smooth with respect to \( \mathbf{X} \), the problem as a function of \( (\mathbf{u}^{(d)}_r)_{d,r} \) is a composite optimisation problem with a non-convex smooth part and a convex, possibly non-smooth, regularizer (it is easy to check that both \( \sum_{r=1}^{R} \sum_{d=1}^{3} \| \mathbf{u}^{(d)}_r \|_p^3 \) and \( \sum_{r=1}^{R} \max_{d=1..3} \| \mathbf{u}^{(d)}_r \|_p^3 \) are convex in \((\mathbf{u}^{(d)}_r)_{d,r}\)). Reddi et al. (2016) showed that in this setting, minibatch proximal stochastic gradient descent (with minibatches that increase in size) converges to a stationary point. Thus, SGD techniques can be used. In practice, the maximum rank \( R \) can be limited by the computational or memory budget.

We next discuss related work before presenting more precise formulations and algorithms.
### 3 Related Work

**Matrix completion** Norm-based regularization has been extensively studied in the context of matrix completion. The trace norm (or nuclear norm) has been proposed as a convex relaxation of the rank \( \{\text{Srebro et al., 2005}\} \) for matrix completion in the setting of ratings prediction, with strong theoretical guarantees \( \{\text{Candès & Recht, 2009}\} \). While efficient algorithms to solve the convex problems have been proposed (see e.g. \( \{\text{Cai et al., 2010; Jaggi et al., 2010}\} \), the practice is still to use the matrix equivalent of the nonconvex formulation \( \{\text{Srebro & Salakhutdinov, 2010, Section 5}\} \). The max-norm was subsequently proposed as an alternative to weighted trace norm that would also be robust to non-uniform sampling \( \{\text{Srebro & Shraibman, 2005; Cai et al., 2016}\} \). There is no known equivalent of the max-norm for tensors; however, the theoretical guarantees hold because the max-norm is related to the matrix equivalent of the nuclear \( \infty \)-norm \( \{\text{Srebro & Shraibman, 2005; Corollary 2}\} \). It is this observation that motivates our use of the nuclear \( \infty \)-norm for tensors.

**Tensor completion and decompositions** There is a large body of literature on low-rank tensor decompositions (see \( \{\text{Kolda & Bader}\} \) for a comprehensive review). Closely related to our work is the canonical decomposition of tensor (also called CANDECOMP/PARAFAC or CP) \( \{\text{Hitchcock, 1927}\} \), which solves a problem similar to \( \{\text{5}\} \) without the regularization (i.e., \( \lambda = 0 \)), and usually the square loss.

Quite a few norm-based regularizations for tensors have been proposed. Several are based on unfolding tensors along each of its mode to obtain matricizations, and either regularize by the sum of trace norms of all its matricizations \( \{\text{Tomioka et al., 2010}\} \) or write the original tensor as a sum of tensors \( T_k \) whose \( k \)th matricization alone is regularized in trace norm \( \{\text{Wimalawarne et al., 2014}\} \). However, in the large scale setting even rank-1 approximations of matricizations involve too many parameters to be tractable.

Recently, the tensor trace norm (nuclear 2-norm) has been proposed as a regularizer for tensor completion \( \{\text{Yuan & Zhang, 2016}\} \), and an algorithm based on the generalized conditional gradient has been proposed by \( \{\text{Cheng et al., 2016}\} \). This algorithm requires, in an inner loop, to compute a (constrained) rank-1 tensor that has largest dot-product with the gradient of the data-fitting term (gradient w.r.t. the tensor argument). This is possible in the one-class completion setup only with the square error loss, for which the gradient is a low-rank + sparse tensor when the argument is low-rank. However, on large-scale knowledge bases the state of the art is to use a binary log-loss or a multiclass log-loss \( \{\text{Trouillon et al., 2016; Kadlec et al., 2017a}\} \); in that case, the gradient is not properly structured and needs be computed explicitly, thereby making the approach of \( \{\text{Cheng et al., 2016}\} \) too costly.

**Link prediction in relational data** The task of link prediction on relational data has been studied extensively in the past decade \( \{\text{Nickel et al., 2016}\} \), in particular in the context of knowledge bases. We focus here on models of relational data based on tensor factorization. RESCAL \( \{\text{Nickel et al., 2011}\} \) is a constrained version of Tucker decomposition in which the two modes of the tensor that correspond to entities have the same factor matrices. That is, given the rank of the decomposition \( R \), the matrix corresponding to the (predicted) adjacency matrix of relationship \( j \) is \( \hat{X}_{:,j,:} = UR^{(j)}U^T \) where \( U \in \mathbb{R}^{E \times R} \) is the factor matrix for the entities common to each relationships, and \( R^{(j)} \in \mathbb{R}^{R \times R} \) is a matrix that depends on the relationship. TATEC \( \{\text{Garcia-Duran et al., 2015}\} \) adds to RESCAL an additional, simpler model containing only bi-linear terms to avoid overfitting on easier datasets. The latent factor model of \( \{\text{Jenatton et al., 2012}\} \) is a probabilistic model that has a factorization similar to RESCAL, but adds some structure for the relationships matrices \( R^{(j)} \) by enforcing them to be a sum of a small number of shared rank-1 matrices. DistMult \( \{\text{Yang et al., 2014}\} \) is a CP.
decomposition in which two modes of the tensor share the same factors; it can also be seen as RESCAL where the matrices \( R^{(3)} \) are diagonal. Even though the modeling assumption made by DistMult is incorrect on most data, because it predicts symmetric interactions for all relationships, it was recently shown to achieve surprisingly good performances on several large-scale knowledge base completion benchmarks (Kadlec et al., 2017a). The ComplEx model of (Trouillon et al., 2016) is a specific form of the CP decomposition with complex numbers and shared factors, which takes the form \( \hat{X}_{i,j,k} = \text{Re}(UR^{(j)}U^T) \), where \( U \in \mathbb{C}^{E \times R} \) is the shared factor matrix of the entities, \( R^{(j)} \in \mathbb{C}^{R \times R} \) is a diagonal (complex) matrix associated to relationship \( j \), and \( U \) is the complex conjugate of \( U \). The ComplEx decomposition is a means to share the embeddings between the two modes for the entities with a CP-like decomposition, without the incorrect modeling assumption of DistMult. Hayashi & Shimbo (2017) showed that the ComplEx decomposition is equivalent to the approach of holographic embeddings (Nickel et al., 2015).

These works on relational data aimed at finding modeling assumptions such that the overall tensor can be approximated with low rank. In practice, CP, RESCAL and ComplEx use an additional regularization based on the Frobenius norm of the factors. Whereas regularizing with the Frobenius norm of the factors is justified in the case of matrix factorization (because it corresponds to a trace norm regularization), there is no justification in the higher-order case beyond analogy.

The nuclear \( p \)-norm regularizers that we propose are intended to (a) provide tuning parameters that continuously trade-off between data fitting and limited model capacity, and (b) propose algorithms with performances that are less sensitive to other parameters such as the rank (the algorithm should yield the same results as soon as the rank is large enough) or the details of the optimization procedure.

4 FORMULATIONS AND ALGORITHMS FOR THE NUCLEAR 3 AND \( \infty \)-NORMS

4.1 NUCLEAR 3-NORM

The nuclear 3-norm (resp. \( D \)-norm) is particularly suited to order-3 (resp. order-\( D \)) tensors because its additive variational formulation becomes separable in each coefficient, given that \( \|u_r^{(d)}\|_3^3 = \sum_{i=1}^{n_d} |u_r^{(d)}|^3 \). This leads to very cheap SGD updates.

Given the separability of the norm and that only a subset of triples are sampled, we generalize to the tensor case the modification of the variational form that turns the trace-norm into the weighted trace-norm (Srebro & Salakhutdinov, 2010); we keep only in the regularization the terms corresponding to the sampled triplets. This leads to a formulation of the form

\[
\min_{(u_r^{(d)})_{d,r}, (i,j,k,y) \in S} \sum_{r=1}^{R} \left( \sum_{i,j,k,y} \|u_r^{(3)}\|_3^3 + \lambda \left( |u_r^{(1)}| + |u_r^{(2)}| + |u_r^{(3)}| \right) \right).
\]

For a specific example \((i,j,k,y)\), only those parameters useful for estimating \( \hat{X}_{i,j,k} \) will be regularized, leading to the same computational complexity as the currently used Frobenius norm regularizer.

Viewing ComplEx as a form of canonical decomposition over \( \mathbb{C} \), we could easily extend this regularizer to the complex setting by replacing the gradient of the squared 2-norms with the gradient of the cubed 3-norms over the moduli, which keeps the computational cost of the updates the same.

In the following, the nuclear 3-norm will always refer to the weighted version described here.

4.2 NUCLEAR \( \infty \)-NORM

Our choice of the nuclear \( \infty \)-norm is motivated by previous work on matrix completion (Srebro & Shraibman, 2005), suggesting the max-norm as an alternative to the trace norm which is robust to non-uniform sampling. We show in Appendix B.2 that the max-norm and \( \infty \)-norm have similar generalization guarantees. As far as we know, the nuclear \( \infty \)-norm is the only norm that generalizes the guarantees of the max-norm to tensors of order \( D \).
Despite the fact that the nuclear tensor $p$-norms are in theory all intractable to compute, we show that for $p = \infty$, a stationary point of variational form (4b) can be found efficiently using proximal SGD.

Indeed, the regularizer $\max_{d=1..3} \| \tilde{u}_r^{(d)} \|_\infty^3$ being convex, we can consider its proximal operator:

$$\text{Prox}(u^{(d)}, \lambda) = \text{argmin}_{\tilde{u}^{(d)}} \sum_{r} \| \tilde{u}_r^{(d)} - u_r^{(d)} \|_2^2 + \lambda \max_{d=1..3} \| \tilde{u}_r^{(d)} \|_\infty^3.$$ (6)

But this problem is separable in $r$:

$$\text{Prox}_r(u_r^{(d)}, \lambda) = \text{argmin}_{\tilde{u}_r^{(d)}} \sum_{d} \| \tilde{u}_r^{(d)} - u_r^{(d)} \|_2^2 + \lambda \max_{d=1..3} \| \tilde{u}_r^{(d)} \|_\infty^3.$$ (7)

Since $\max_{d=1..3} \| \tilde{u}_r^{(d)} \|_\infty^3 = \| \tilde{u}_r \|_\infty^3$, with $\tilde{u}_r$ the concatenation over $d$ of all the $\tilde{u}_r^{(d)}$, we can set $\tilde{x} = \tilde{u}_r$ and $x = u_r$, to re-write the problem as:

$$\text{Prox}_{\tilde{x}}(x, \lambda) = \text{argmin}_{\tilde{x}} \| \tilde{x} - x \|_2^2 + \lambda \| \tilde{x} \|_\infty^3 = \text{argmin}_{C, \tilde{x}} \| \tilde{x} - x \|_2^2 + \lambda C^3.$$ (7)

Solving this proximal operator is similar to what one would do for the usual $\ell_\infty$ norm regularization. It is actually possible to find the value of $C$ analytically. The exact derivation and algorithm are given in the appendix. In practice, since the solution requires to sort all the parameters, we apply the proximal operator after a fixed number of gradient updates. This is a slight departure from the theory which would require minibatch updates of increasing sizes.

## 5 Experimental Results

We begin in Section 5.1 with a comprehensive comparative study of regularized CP and show that our nuclear $p$-norms regularizers are better behaved than the one used in practice. We then show scalability by applying these algorithms to larger datasets in Section 6. Finally, we plug-in our nuclear 3-norm into a state of the art model, ComplEx, and show that we gain both in ease of tuning and performances on the SVO dataset. Detailed grid search parameters can be found in appendix 7.4.

### 5.1 Small datasets

NATION, UMLS and KINSHIP ((Kemp et al. (2006))) are three small datasets on which we can study tensor completion in the setting of ratings prediction as in Nickel et al. (2011). We know from experiments in Nickel et al. (2011) that a good regularization is key on these datasets or CP will overfit. We study the effects of regularization on two tensor decomposition methods: CP and ComplEx. The regularizer used in practice for these two methods is the squared $L_2$ norm of the factors. We report on Figure 1 the cross-validated validation MSE for different values of $\lambda$ on UMLS with three different regularizers and after 50 epochs (without early-stopping). While the curves for the nuclear norms are nicely unimodal in $\lambda$ and have invisible error bars, for the values of $\lambda$ that are sufficiently large to prevent overfitting, the performance of $L_2$ has large variance, which is most probably due to
the non-convexity of this regularizer. This unstable behavior makes it difficult to find good values of the regularization coefficient. As a confirmation, a grid-search over \(\{epochs, rank, \lambda\}\) for the CP-\(L_2\) baseline chose a low \(\lambda\) combined with a small number of epochs thus preferring early-stopping to prevent overfitting.

Furthermore, and in stark contrast with \(L_2\), we observe in Figures 2a and 2b that the rank and the number of epochs play no significant roles when we use the nuclear norms.

Finally, for the three datasets, we split the data in 10-fold, use 8 folds for training, one for validation (choice of hyperparameters) and one for testing, and repeat 10 times on different folds. Similarly to Nickel et al. (2011) and Bailly et al. (2015), we use the ratings prediction setting, and unobserved values are set to zero before fitting the tensor with an \(L_2\) loss. Final mean and standard errors for each method are reported in Table 1. For the baselines CP and ComplEx, we grid search over the rank and \(\lambda\), setting a low ADAGRAD learning rate and test after each epoch for early stopping based on validation set performance. For the nuclear norm regularized methods, we train for a fixed number of epochs that is sufficient to reach convergence on the training set and without any early stopping. The rank and learning rate are picked based on preliminary tests.

We compare the AUC for each baseline on the three datasets with that of the same model with our proposed regularizers. We include in the comparison the results for two state-of-the-art methods, SITAR (Bailly et al., 2015) and LFM (Jenatton et al., 2012). We observe a consistent increase of performance for both CP and ComplEx when regularized with nuclear norms, except for CP-\(\infty\) on NATION.

Using ComplEx with our regularizer, we match the state of the art results of SITAR on two out of three datasets.

<table>
<thead>
<tr>
<th>Method</th>
<th>NATION</th>
<th>UMLS</th>
<th>KIN</th>
</tr>
</thead>
<tbody>
<tr>
<td>CP-(L_3)</td>
<td>0.857 ± 0.01</td>
<td>0.975 ± 0.004</td>
<td>0.949 ± 0.004</td>
</tr>
<tr>
<td>CP-(\infty)</td>
<td>0.825 ± 0.02</td>
<td>0.978 ± 0.003</td>
<td>0.950 ± 0.003</td>
</tr>
<tr>
<td>CP- baseline</td>
<td>0.846 ± 0.02</td>
<td>0.963 ± 0.005</td>
<td>0.944 ± 0.005</td>
</tr>
<tr>
<td>ComplEx- baseline</td>
<td>0.869 ± 0.02</td>
<td>0.972 ± 0.005</td>
<td>0.969 ± 0.004</td>
</tr>
<tr>
<td>ComplEx- (L_3)</td>
<td>0.878 ± 0.02</td>
<td>0.982 ± 0.004</td>
<td>0.971 ± 0.005</td>
</tr>
<tr>
<td>RESCAL</td>
<td>0.84</td>
<td>0.98</td>
<td>0.95</td>
</tr>
<tr>
<td>LFM*</td>
<td>0.909 ± 0.009</td>
<td>0.990 ± 0.003</td>
<td>0.946 ± 0.005</td>
</tr>
<tr>
<td>SITAR</td>
<td>0.890 ± 0.019</td>
<td>0.977 ± 0.003</td>
<td>0.969 ± 0.004</td>
</tr>
</tbody>
</table>

Table 1: 10-fold cross validated AUC on the three datasets in ratings prediction. *LFM does not consider missing elements as negative, which is an easier setting.
5.2 Big Datasets

5.2.1 FB15K and WN18

We consider FB15K and WN18, two datasets used frequently in the link prediction literature (Bordes et al., 2013). We follow the same protocol as in Trouillon et al. (2016) and select our best model based on the Mean Reciprocal Rank (see e.g. Kadlec et al. (2017b) for a definition). Precisely, we learn separate representations of the relations and their inverse, using one or the other to rank the left hand side or right hand side.

All factorization-based methods perform as well as the state of the art (DistMult, Kadlec et al. (2017b)), as shown in Table 2 (for a more complete overview of the state-of-the-art on these datasets and a description of the metrics, see Kadlec et al. (2017a)).

Note that the results reported here for CP are without regularization.

We tried larger ranks (up to 10000), but never needed regularization, because our models never seemed to overfit. The fact that regularization seems somewhat useless may be because the data exhibits very low noise, or because overfitting may require ranks that are too large to try with our computational budget.

This suggest that FB15K and WN might not be the best datasets to test new structured formulations.

<table>
<thead>
<tr>
<th></th>
<th>FB15K</th>
<th></th>
<th>WN</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>H@1</td>
<td>H@10</td>
<td>MRR</td>
</tr>
<tr>
<td>TransE†</td>
<td>50.0</td>
<td>83.2</td>
<td>0.633</td>
</tr>
<tr>
<td>DistMult*</td>
<td>79.7</td>
<td>89.3</td>
<td>0.798</td>
</tr>
<tr>
<td>ComplEx†</td>
<td>74.8</td>
<td>87.0</td>
<td>0.779</td>
</tr>
<tr>
<td>CP</td>
<td>77.7</td>
<td>87.0</td>
<td>0.810</td>
</tr>
</tbody>
</table>

Table 2: All methods are roughly equivalent on FB15K and WN18, given a high enough rank. *Results for DistMult include a specifically selected ensemble model for H@1 and a single model selected on H@10 rather than MRR. † Results are for our re-implementation of these two models, giving better results than what is reported in the literature.

5.2.2 SVO

Subject-Verb-Object (SVO) is a large dataset introduced in Jenatton et al. (2012). Contrary to what we observed on FB15K and WN18, learning on this dataset requires careful regularization. Pictured in figure 3 is a fine grid-search over $\lambda$ for two versions of the ComplEx model trained with a ranking-loss. Everything else being equal, we observe a gap in validation performances that the Frobenius regularized version cannot bridge.

We use this dataset to showcase how the nuclear 3-norm can improve performances on large tensor completion without any additional effort on an off-the-shelf algorithm like ComplEx. This improvement puts ComplEx trained with a ranking loss at 79H@5% just shy of the state of the art, 80H@5% in Garcia-Duran et al. (2015), without any additional efforts. Results on this dataset are similar for CP, we report experiments for this setting in the appendix.

In terms of timings, our implementation of ComplEx with a nuclear 3-norm regularizer runs at about 40s/epoch for a ranking log-loss on 20 cores, and converges in around 1000 epochs, as seen on figure 3. We do not notice any difference in runtime for the nuclear norms compared to the usual Frobenius norm, provided that the proximal operator for the nuclear $\infty$-norm is applied after $n_{\text{entities}}$ epochs, which is frequent enough in practice.

6 Conclusion

We studied two particular norms in the nuclear $p$-norms family, and their practical use in a completion settings. The nuclear $\infty$-norm extends the generalization guarantees of the matrix max-norm to higher
order tensor and has a fast proximal descent algorithm. For the practitioner, the nuclear 3-norm can be plugged without effort into models that are related to CP, and leads to consistent improvements in generalization performances. Having a regularizer that is insensitive to the rank allowed us to try higher ranks without fear of overfitting for CP, which we re-introduce as a competitive method, matching the state of the art on FB15K and WN18. An interesting direction for future work would be to derive the proximal operator for a weighted version of the tensor trace norm (or nuclear 2-norm). We used ADAGRAD in a non-convex setting, a rigorous proof of its convergence to stationary points in this setting would be helpful. Another interesting question is to explore the statistical guarantees for the nuclear 3-norm.

REFERENCES

Raphaël Bailly, Antoine Bordes, and Nicolas Usunier. Semantically Invariant Tensor Factorization. 2015.


Rodolphe Jenatton, Nicolas Le Roux, Antoine Bordes, and Guillaume Obozinski. A latent factor model for highly multi-relational data. December 2012. URL [https://hal.inria.fr/hal-00776335/document](https://hal.inria.fr/hal-00776335/document).


7 Appendix

7.1 Proof of Eqns. 4a and 4b

We will prove a version for tensors of any order $D$:

**Proposition 1.** For $X$ a tensor or order $D$, the nuclear $p$-norm has the following equivalent variational forms:

\[
\|X\|_{*,p} = \min \left\{ \frac{1}{D} \sum_{r=1}^{R} \sum_{d=1}^{D} \|u_r^{(d)}\|_p^D \mid (u_r^{(d)}) \in U_R(X), R \in \mathbb{N} \right\} \tag{8a}
\]

\[
= \min \left\{ \sum_{r=1}^{R} \max_{d=1..D} \|u_r^{(d)}\|_p^D \mid (u_r^{(d)}) \in U_R(X), R \in \mathbb{N} \right\}. \tag{8b}
\]

**Proof.** We first show that

\[
\|X\|_{*,p} \geq \min \left\{ \frac{1}{D} \sum_{r=1}^{R} \sum_{d=1}^{D} \|u_r^{(d)}\|_p^D \mid (u_r^{(d)}) \in U_R(X), R \in \mathbb{N} \right\} \tag{9a}
\]

\[
\geq \min \left\{ \sum_{r=1}^{R} \max_{d=1..D} \|u_r^{(d)}\|_p^D \mid (u_r^{(d)}) \in U_R(X), R \in \mathbb{N} \right\}. \tag{9b}
\]

For $X$, let $(u_r^{(d)})$, be the decomposition that realizes the nuclear $p$-norm given in equation 2 where we got rid of $\sigma$ by scaling one of the factors. For a fixed $r$, we can rescale the factors with a vector $c$ such that $c \in C = \{ c \in \mathbb{R}_+^*, c_1 c_2 \ldots c_D = 1 \}$, leaving the product of the norms unchanged.

\[
\forall d, \|\tilde{u}_r^{(d)}\|_p = c_d \|u_r^{(d)}\|_p = \frac{D}{\|u_r^{(1)} \ldots u_r^{(D)}\|_p}. \tag{10}
\]

Now that the factors have equal norm:

\[
\prod_{d=1}^{D} \|\tilde{u}_r^{(d)}\| = \frac{1}{D} \sum_{d=1}^{D} \|\tilde{u}_r^{(d)}\|_p^D = \max_{d=1..D} \|u_r^{(d)}\|_p^D. \tag{11}
\]

Thus proving the inequalities. For the opposite inequality, we start with a decomposition $(u_r^{(d)})$ realizing the min in either 4a or 4b. Normalizing each factor $u_r^{(d)}$ to have unit $p$-norm and writing the product of the scalings as $\sigma_d$, we can use Equation 10 to go back to the original nuclear $p$-norm formulation. \hfill \square

7.2 Generalization Guarantees for the Nuclear $\infty$-Norm

**Proposition 2.** For all target tensors $Y \in \{ \pm 1 \}_{n_1 \times \ldots \times n_D}$ and an i.i.d sample $S$ of $|S|$ entries in $Y$ sampled under $\mathcal{P}$, with probability $1-\delta$ over the sample selection, the following holds for an $L$-Lipschitz loss and $X$ such that $\|X\|_{*,\infty} \leq M$:

\[
D_P(X; Y) < D_S(X; Y) + 6LM \sqrt{\frac{\sum_{d=1}^{D} n_d}{|S|}} + \frac{\log \frac{4}{\delta}}{2|S|} \tag{12}
\]

**Proof.** We follow the proof in [Srebro 2004] Section 6.2.4. Our class of tensor $B_M = \{ X \mid \|X\|_{*,\infty} \leq M \}$ is directly the scaled convex hull of a finite set of rank-one sign tensors of size $|B_M| \leq 2 \sum_{d=1}^{D} n_d$. Using [Srebro 2004] Theorem 47), we can bound its Rademacher Complexity by

\[
\hat{R}_S(B_M) \leq M \sqrt{\frac{\sum_{d=1}^{D} n_d + \log |S|}{|S|}} \leq 6M \sqrt{\frac{\sum_{d=1}^{D} n_d}{|S|}} \tag{13}
\]

Where we bounded $\log |S|$ by its maximum value: $\sum_{d=1}^{D} n_d$, and used $\sqrt{35} < 6$. \hfill \square
7.3 Nuclear $\infty$-norm proximal descent

We derive the updates for the slightly more complicated (diagonal) ADAGRAD algorithm [Duchi et al. (2011)], in the case of composite mirror descent. Our proof follows what is done in Section 5 of [Duchi et al. (2011)]. To simplify notations in the remainder, as was done in the paper, we use the separability of the loss to work on a fixed $r$, and set $x$ the concatenation of $u_{r}^{(d)}$ for all $d$. At time $t$, let $x_t$ be the parameter vector, $g_t$ the gradient of the loss, $\eta$ the step-size, and $H_t = \delta I + diag(G_t)^{1/2}$ the diagonal adagrad weights (for further details : [Duchi et al. (2011)]). With $\lambda$ our regularization parameter, we can write the composite mirror descent update:

$$x_{t+1} = \text{argmin}_x \{ \langle g_t, x \rangle + \eta \lambda \Omega(x) + \frac{1}{2} \langle x, H_t x \rangle - \langle H_t x_t, x \rangle \}$$  \hspace{0.5cm} (13)

Setting $u = \eta g_t - H_t x_t$, we can rewrite this as:

$$x_{t+1} = \text{argmin}_x \{ \langle u, x \rangle + \eta \lambda \Omega(x) + \frac{1}{2} \langle x, H_t x \rangle \}$$  \hspace{0.5cm} (14)

With $z = H_t^{1/2} x$ and $a = diag(H_t^{-1/2})$, this amounts to solving:

$$x_{t+1} = a \odot \text{argmin}_z \{ \| z + a \odot u \|_2^2 + \eta \lambda \Omega(a \odot z) \}$$  \hspace{0.5cm} (15)

With $v = -a \odot u$, we need to describe how to solve, for $a \succeq 0$ and $z \in \mathbb{R}^d$:

$$\text{argmin}_z \{ \| z - v \|_2^2 + \eta \lambda \Omega(a \odot z) \}$$  \hspace{0.5cm} (16)

The specific regularizer we are interested in is the nuclear $\infty$-norm. We can rewrite our problem with constraints, adding a $\frac{1}{3}$ factor to the regularizer to simplify the final algorithm’s form:

$$\text{argmin}_{x, \lambda \in C, \| a \odot z \|_\infty \leq C} \{ \| z - v \|_2^2 + \eta \frac{\lambda}{3} C^3 \}$$  \hspace{0.5cm} (17)

Introducing dual variable $\mu \succ 0$, our Lagrangian can then be written:

$$\mathcal{L}(z, \mu) = \| z - v \|_2^2 + \eta \frac{\lambda}{3} C^3 + \sum_{i=1}^{d} \mu_i (\sigma(z_i) - C)$$  \hspace{0.5cm} (18)

The KKT conditions give us:

$$\nabla_z \mathcal{L}(z, \mu) = 0 \iff 2(z_i - v_i) + \mu_i a_i \text{sign}(z_i) = 0$$

$$\mu_i > 0 \iff |z_i| = \frac{C}{a_i}$$

Using these conditions we find that $z_i = \text{sign}(v_i) \frac{C}{a_i}$ if $|v_i| > \frac{C}{a_i}$, and $z_i = v_i$ otherwise. Let $\sigma$ be a non-increasing ordering of the sequence $\{|v_i| a_i\}$, then for $\rho$ defined as the largest index such that $C \in I_\rho = [\|v_{\sigma(\rho-1)} a_{\sigma(\rho-1)}\|, \|v_{\sigma(\rho)} a_{\sigma(\rho)}\|)$, we can write the objective as a convex function of $C$:

$$f(C) = \sum_{i \prec \rho} \left( \frac{C}{a_{\sigma(i)}} - v_{\sigma(i)} \right)^2 + \eta \frac{\lambda}{3} C^3$$  \hspace{0.5cm} (19)

Setting the gradient to zero gives the condition:

$$\nabla_C f(C) = 0 \iff \sum_{i \prec \rho} 2 \frac{C}{a_{\sigma(i)}} \left( \frac{C}{a_{\sigma(i)}} - v_{\sigma(i)} \right) + \eta \lambda C^2 = 0$$  \hspace{0.5cm} (20)

We are looking for roots of this polynomial that lies in the interval $I_\rho$, leading to Algorithm[1]
Algorithm 1 Step for the nuclear $\infty$-norm and adaptive composite mirror descent

\begin{algorithm}
\textbf{procedure} \textsc{Step}(x_t, H_t, g_t, a, \eta, \lambda) \\
\hspace{1em} v \leftarrow a \odot (H_t x_t - \eta g_t) \\
\hspace{1em} \sigma \leftarrow \text{argsort}([v_i | a_i]) \quad \triangleright \text{Descending order} \\
\hspace{1em} \textbf{for} \rho \leq \text{len}(x) \textbf{do} \\
\hspace{2em} (a, \beta, \gamma) \leftarrow (\eta \lambda, 2 \sum_{i<\rho} a_{\sigma(i)}^2, -2 \sum_{i<\rho} \frac{|v_{\sigma(i)}|}{a_{\sigma(i)}}) \\
\hspace{2em} \textbf{if} \alpha |x_{\sigma(\rho)}|^2 + \beta |x_{\sigma(\rho)}| + \gamma \leq 0 \textbf{then} \\
\hspace{3em} C \leftarrow \frac{-\beta + \sqrt{\beta^2 - 4 \alpha \gamma}}{2 \alpha} \\
\hspace{3em} \textbf{break} \\
\hspace{2em} \textbf{end if} \\
\hspace{2em} \textbf{end for} \\
\hspace{1em} \textbf{for} i < \rho \textbf{do} \\
\hspace{2em} y_{\sigma(i)} \leftarrow \text{sign}(v_{\sigma(i)}) C \\
\hspace{2em} \textbf{end for} \\
\hspace{1em} \textbf{for} i \geq \rho \textbf{do} \\
\hspace{2em} y_{\sigma(i)} \leftarrow a_{\sigma(i)} v_{\sigma(i)} \\
\hspace{2em} \textbf{end for} \\
\hspace{1em} \textbf{return} y \\
\textbf{end procedure}
\end{algorithm}

7.4 Grid Searches

7.4.1 Small Datasets

- \textbf{KINSHIP}:
  - \textsc{CP} Baseline : $\lambda \in \{0, 1e-6, 5e-6, 1e-5, 5e-5, 1e-4\}$, $R \in \{50, 100, 150, 200, 250, 300\}$, $\eta = 1e-2$. \\
  - \textsc{CP} nuclear 3-norm: $R = 200$, $epoch = 50$, $\lambda \in \{5e-4, 6e-4, 7e-4, 8e-4, 9e-4, 1e-3, 2e-3, 3e-3\}$, $\eta = 1e-1$. \\
  - \textsc{CP} nuclear $\infty$-norm: $R = 200$, $epoch = 50$, $\lambda \in \{1e-6, 5e-6, 1e-5, 5e-5, 1e-4, 5e-4, 1e-3, 5e-3\}$, $mb = 100$, $\eta = 1e-1$. \\
  - \textsc{ComplEx} Baseline : $\lambda \in \{0, 1e-6, 5e-6, 1e-5, 5e-5, 1e-4\}$, $R \in \{50, 100, 150, 200\}$, $\eta = 1e-2$. \\
  - \textsc{ComplEx} nuclear 3-norm: $R = 200$, $epoch = 50$, $\lambda \in \{5e-6, 7.5e-6, 1e-5, 2.5e-5, 5e-5, 7.5e-5, 1e-4\}$, $\eta = 1e-1$. \\

- \textbf{UMLS}:
  - \textsc{CP} Baseline : $\lambda \in \{0, 1e-6, 5e-6, 1e-5, 5e-5, 1e-4\}$, $R \in \{50, 100, 150, 200, 250, 300\}$, $\eta = 1e-2$. \\
  - \textsc{CP} nuclear 3-norm: $R = 200$, $epoch = 50$, $\lambda \in \{5e-5, 7.5e-5, 1e-4, 3e-4, 5e-4, 7e-4, 9e-4, 1e-3, 2e-3\}$, $\eta = 1e-1$. \\
  - \textsc{CP} nuclear $\infty$-norm: $R = 200$, $epoch = 50$, $\lambda \in \{0.8, 0.9, 1, 2, 3, 5, 10, 20, 30\}$, $mb = 10000$, $\eta = 1e-1$. \\
  - \textsc{ComplEx} Baseline : $\lambda \in \{0, 1e-6, 5e-6, 1e-5, 5e-5, 1e-4\}$, $R \in \{50, 100, 150, 200\}$, $\eta = 1e-2$. \\
  - \textsc{ComplEx} nuclear 3-norm: $R = 200$, $epoch = 50$, $\lambda \in \{5e-5, 7.5e-5, 1e-4, 2e-4, 3e-4, 4e-4, 5e-4, 6e-4, 7e-4, 8e-4, 9e-4, 1e-3, 2e-3\}$, $\eta = 1e-1$. \\

- \textbf{NATION}:
  - \textsc{CP} Baseline : $\lambda \in \{0, 1e-6, 5e-6, 1e-5, 5e-5, 1e-4\}$, $R \in \{50, 100, 150, 200, 250, 300\}$, $\eta = 1e-2$. \\
  - \textsc{CP} nuclear 3-norm: $R = 200$, $epoch = 50$, $\lambda \in \{6e-3, 7e-3, 8e-3, 9e-3, 1e-2, 1.25e-2, 2e-2, 3e-2\}$, $\eta = 1e-1$. \\
  - \textsc{CP} nuclear $\infty$-norm: $R = 200$, $epoch = 50$, $\lambda \in \{1, 1.2, 1.4, 1.6, 1.8, 2, 2.2, 2.4, 2.6\}$, $mb = 50$, $\eta = 1e-1$. 

14
Under review as a conference paper at ICLR 2018

- ComplEx Baseline : $\lambda \in \{0, 1e-6, 5e-6, 1e-5, 5e-5, 1e-4\}$, $R \in \{50, 100, 150, 200\}$, $\eta = 1e-2$.
- ComplEx nuclear 3-norm: $R = 200$, $epoch = 50$, $\lambda \in \{5e-3, 6e-3, 7e-3, 8e-3, 9e-3, 1e-2, 2e-2, 3e-3, 4e-4, 5e-5\}$, $\eta = 1e-1$.

7.4.2 SVO

- CP (baseline and nuclear 3-norm) : $R = 1000$, $\lambda \in \{1e-6, 5e-6, 1e-5, 5e-5, 1e-4, 5e-4, 1e-3, 5e-3, 1e-2, 5e-2\}$, $\eta = 1e-1$.
- ComplEx (baseline and nuclear 3-norm) : $R = 2000$, $\lambda \in \{5e-4, 6e-4, 7e-4, 8e-4, 9e-4, 1e-3, 2e-3, 3e-3, 4e-3, 5e-3\}$, $\eta = 1e-1$. 