PRACTICAL ALGORITHMS FOR LEARNING NEAR-ISOMETRIC LINEAR EMBEDDINGS*

JERRY LUO†, KAYLA SHAPIRO‡, HAO-JUN MICHAEL SHI§, QI YANG¶, AND KAN ZHU‖

Abstract. We propose two practical non-convex approaches for learning near-isometric, linear embeddings of finite sets of data points. Following Hegde, et. al. [6], given a set of training points $\mathcal{X}$, we consider the secant set $S(\mathcal{X})$ that consists of all pairwise difference vectors of $\mathcal{X}$, normalized to lie on the unit sphere. The problem can be formulated as finding a symmetric and positive semi-definite matrix ψ that preserves the norms of all the vectors in $S(\mathcal{X})$ up to a distortion parameter δ. Motivated by non-negative matrix factorization, we reformulate our problem into a Frobenius norm minimization problem, which is solved by the Alternating Direction Method of Multipliers (ADMM) and develop an algorithm, *FroMax*. Another method solves for a projection matrix ψ by minimizing the restricted isometry property (RIP) directly over the set of symmetric, postive semi-definite matrices. Applying ADMM and a Moreau decomposition on a proximal mapping, we develop another algorithm, *NILE-Pro*, for dimensionality reduction. Both non-convex approaches are then empirically demonstrated to be more computationally efficient than prior convex approaches for a number of applications in machine learning and signal processing.

Key words. dimensionality reduction, linear embeddings, compressive sensing, approximate nearest neighbors, classification

1. Introduction.

1.1. Motivation.

We are currently in a "data crisis" in which the size and complexity of raw data acquired and processed by diverse modalities poses a challenge to current state-of-the-art information processing systems. Since many machine learning algorithms' computational efficiency scale with the complexity of the data, machine learning researchers have introduced a family of algorithms for *dimensionality reduction* to address this issue. Dimensionality reduction algorithms devise a concise representation of high-dimensional data on a lower-dimensional subspace, with as minimal loss of intrinsic information as possible. This representation is often referred to as a low-dimensional *embedding*.

The canonical approach in statistics for constructing a linear embedding is principal components analysis (PCA) [10]. PCA is a linear embedding technique that projects data points onto a lower-dimensional subspace spanned by the principal components that contain the most variability within the data. PCA enjoys the benefits of being computationally efficient and easily generalizable to new data sets; however, it fails to preserve pairwise distances between sample data points, sometimes rendering two distinct points indistinguishable in the low-dimensional embedding space. This can potentially hamper the performance of PCA and other similar algorithms.

*This research was conducted as part of the California Research Training Program in Computational and Applied Mathematics 2014.

†Department of Mathematics, University of Arizona, Tucson, AZ 85721. (jerryluo@math.arizona.edu)

‡Department of Computing, Imperial College London, London SW7 2AZ, United Kingdom. (k.shapiro@berkeley.edu)

§Department of Mathematics, University of California, Los Angeles, Los Angeles, CA 90024. (hjmshi@ucla.edu)

¶Department of Mathematics, University of Southern California, Los Angeles, 90089. (yangq@usc.edu)

‖Department of Computer Science, Columbia University, New York, NY 10027. (kzhu9@ucla.edu)

Other popular nonlinear, manifold learning methods, such as ISOMAP and locally linear embedding (LLE), preserve geometric structure by approximating geodesics from k-nearest neighbors. However, most fail to preserve all pairwise distances between data points and produce embeddings which are easy to explicitly store and generalize. Note that linear embeddings can be explicitly stored using a matrix operator and can therefore be quickly applied to any new data point.

A linear embedding technique that preserves all pairwise distances is the method of *random projections*. Given $\mathcal{X}$, a cloud of Q data points in a high-dimensional Euclidean space $\mathbb{R}^N$, the Johnson-Lindenstrauss Lemma [7] states that there exists a linear, near-isometric, or distance preserving, embedding such that $\mathcal{X}$ can be mapped to a subspace of dimension $M = \mathcal{O}(\log Q)$ with high probability. Despite its conceptual simplicity, random projections suffers from probabilistic and asymptotic theoretical guarantees. A random projections mapping is also independent of the data under consideration, failing to utilize the geometric structure of the data.

1.2. Related Work. Using the geometric structure of the data, Hegde, et. al. developed a new deterministic approach, NuMax, to construct a near-isometric, linear embedding [6]. Given a training set $\mathcal{X} \subset \mathbb{R}^N$, the *secant set* is constructed by taking all pairwise difference vectors of $\mathcal{X}$, which are then normalized to lie on the unit sphere. Hegde, et. al. formulated an affine rank minimization problem to construct a *projection matrix* ψ that preserves norms of all vectors in $S(\mathcal{X})$ up to a distortion parameter δ. They then relax this problem to a convex program that can be solved using a tractable semidefinite program (SDP), with the help of column generation, and develop NuMax based on the Alternating Direction Method of Multipliers (ADMM). This framework deterministically produces a linear embedding that is near-isometric; however, the algorithm is computationally expensive due to the need to compute a singular value decomposition at each iteration to minimize the nuclear norm. Our proposed approaches build on their original framework by proposing non-convex problems which are solved to produce projection matrices in much faster time. Other algorithmic approaches for finding near-isometric linear embeddings are also described in [4, 5, 14].

1.3. Organization. The rest of the paper is organized as follows. We review the restricted isometry property and NuMax algorithm in §2. The FroMax algorithm is introduced in §3. NILE-Pro is discussed in §4. Rank adjustment and column generation methods which increase computational efficiency for large data sets is introduced in §5. Numerical simulations and runtime performance results are presented in §7. Lastly, §8 concludes the paper and gives direction for future work.

2. Background.

2.1. Restricted Isometry Property (RIP). E. Candes, et. al. introduce a formal, relaxed notion of isometry in [1] as follows:

DEFINITION 2.1. *Suppose $M \leq N$ and consider $\mathcal{X} \subset \mathbb{R}^N$. An embedding operator $\mathcal{P} : \mathcal{X} \to \mathcal{R}^M$ satisfies the restricted isometry property (RIP) on $\mathcal{X}$ if there exists a positive constant $\delta > 0$ such that, for every x, x' in $\mathcal{X}$, the following relation holds:*

$$(1-\delta)\|x - x'\|_2^2 \leq \|\mathcal{P}x - \mathcal{P}x'\|_2^2 \leq (1+\delta)\|x - x'\|_2^2. \tag{2.1}$$

We may also refer to δ as the *isometry constant*. Intuitively, this notion of near-isometry requires the distance of every pair of points in $\mathcal{X}$ to be nearly preserved.

Hegde, et. al. [6] develop a framework that seeks to find low rank matrices that satisfy the RIP.

2.2. NuMax. In this section, we review Hegde et. al. [6]'s work on NuMax. Given a data set $\mathcal{X} \subset \mathbb{R}^N$, Hegde et. al. formulate the secant set as follows:

$$S(\mathcal{X}) = \left\{ \frac{x - x'}{\|x - x'\|_2}, x, x' \in \mathcal{X}, x \neq x' \right\} \tag{2.2}$$

Hegde, et. al. [6] seeks to find a projection matrix $\psi \in \mathbb{R}^{M \times N}$ with the smallest possible rank that satisfies the RIP on $S(\mathcal{X})$ for a given $\delta > 0$. This problem is then cast as an optimization problem over all symmetric matrices which we denote as $\mathbb{S}^{N \times N}$. Let $P = \psi^T \psi \in \mathbb{S}^{N \times N}$ with $\text{rank}(P) = M$. Then for all secants $v_i \in S(\mathcal{X})$, we may rewrite the RIP constraint as:

$$(1 - \delta)\|v_i\|_2^2 \leq \|\psi v_i\|_2^2 \leq (1 + \delta)\|v_i\|_2^2 \tag{2.3}$$

$$|\|\psi v_i\|_2^2 - \|v_i\|_2^2| \leq \delta \tag{2.4}$$

$$|\|\psi v_i\|_2^2 - 1| \leq \delta \tag{2.5}$$

$$|v_i^T P v_i - 1| \leq \delta \tag{2.6}$$

Let 1_S denote the S-dimensional ones vector and $\mathcal{A} : X \to \{v_i^T X v_i\}_{i=1}^S$. This admits the rank minimization problem:

$$\begin{aligned} \underset{P}{\text{minimize}} \quad & \text{rank}(P) \\ \text{subject to} \quad & \|\mathcal{A}(P) - 1_S\|_\infty \leq \delta \\ & P \succeq 0 \end{aligned} \tag{2.7}$$

However, since rank minimization is a non-convex, NP-hard problem, a convex relaxation is performed on the objective to obtain the following nuclear-norm minimization:

$$\begin{aligned} \underset{P}{\text{minimize}} \quad & \|P\|_* \\ \text{subject to} \quad & \|\mathcal{A}(P) - 1_S\|_\infty \leq \delta \\ & P \succeq 0 \end{aligned} \tag{2.8}$$

where $\|P\|_*$ is the nuclear norm, which is the sum of the singular values of P. Then the desired linear embedding $\psi \in \mathbb{R}^{M \times N}$ can be found by taking a matrix square root of the minimizer $P^* = U \Gamma U^T$ by

$$\psi = \Gamma_M^{1/2} U_M^T \tag{2.9}$$

where $\Gamma_M = \text{diag}\{\lambda_1, ..., \lambda_M\}$ denotes the M leading (non-zero) eigenvalues of P^* and U_M are the corresponding eigenvectors.

Applying the Alternating Direction Method of Multipliers (ADMM), the optimization problem is rewritten by introducing auxilliary variables $L \in \mathbb{S}^{N \times N}$ and $q \in \mathbb{R}^S$:

$$\begin{aligned} \underset{P,L,q}{\text{minimize}} \quad & \|P\|_* \\ \text{subject to} \quad & P = L \\ & \mathcal{A}(L) = q, \\ & \|q - 1_S\|_\infty \leq \delta \\ & P \succeq 0 \end{aligned} \tag{2.10}$$

The linear constraints are then relaxed to form an augmented Lagrangian as follows:

$$L_A(P, L, q; \Gamma, \omega) = \|P\|_* + \frac{\beta_1}{2}\|P - L - \Gamma\|_F^2 + \frac{\beta_2}{2}\|\mathcal{A}(L) - q - \omega\|_2^2 \tag{2.11}$$

NuMax then solves the following augmented Lagrangian problem:

$$\begin{aligned} \underset{P,L,q,\Gamma,\omega}{\text{minimize}} \quad & L_A(P, L, q, \Gamma, \omega) \\ \text{subject to} \quad & \|q - 1_S\|_\infty \leq \delta \\ & P \succeq 0 \end{aligned} \tag{2.12}$$

where $\Gamma \in \mathbb{S}^{N\times N}$ and $\omega \in \mathbb{R}^S$ represent the scaled Lagrange multipliers. P, L and q are optimized in an alternating fashion, i.e. optimized one at a time with the others held fixed. This optimization can then be solved by three easier sub-problems, admitting a computationally efficient solution.

For more information regarding theoretical and empirical properties of NuMax, please refer to Hegde et. al. [6].

This framework, though slower than conventional methods such as PCA and random projections, admits a projection matrix satisfying the RIP. However, NuMax computes a singular value decomposition of P each iteration, which is computationally expensive. Furthermore, though minimizing the nuclear-norm tends to give low rank matrices, NuMax does not theoretically guarantee the lowest rank embedding for a given δ.

These issues motivate the pursuit of other practical algorithms that optimize similar non-convex problems that may admit low rank, near-isometric projection matrices that give faster, but sufficient (not necessarily optimal) results. Rather than solving both the rank minimization and near-isometry problems simultaneously, we solve a simpler non-convex problem quickly to find a near-isometric projection matrix and apply a rank adjustment heuristic to choose a minimal rank.

2.3. Non-Negative Matrix Factorization (NMF). One of our algorithms is motivated by ideas from *non-negative matrix factorization*. Non-negative matrix factorization (NMF) is a group of algorithms that factorize a non-negative matrix V into two low-rank non-negative matrices W and H [9]. More rigorously, let $V \in \mathbb{R}^{N\times M}$ be given, then we solve for $W \in \mathbb{R}^{M\times Q}$, and $H \in \mathbb{R}^{Q\times N}$ by solving the following optimization problem:

$$\begin{aligned} \underset{W,H}{\text{minimize}} \quad & \|WH - V\|_F^2 \\ \text{subject to} \quad & W_{ij} \geq 0, H_{ij} \geq 0, \forall\ i, j \end{aligned} \tag{2.13}$$

NMF motivates the problem formulation for our first algorithm, FroMax.

3. FroMax. Our first algorithm, *Frobenius norm minimization with Max-norm constraints*, or *FroMax* mixes ideas from NuMax and NMF to formulate a Frobenius norm minimization problem which we then solve based on ADMM, similar to NuMax [16]. Note that this algorithm does not discover the optimal rank for ψ. We combine FroMax with a rank adjustment heuristic to find low rank embeddings.

3.1. Optimization Framework. We formulate a specialized matrix factorization minimization problem to solve for a near-isometric linear embedding as follows:

Given a desired rank r, let $\psi \in \mathbb{R}^{r\times N}$. Here, we seek to solve:

$$\begin{aligned} \underset{P,\psi}{\text{minimize}} \quad & \frac{1}{2}\|P-\psi^T\psi\|_F^2 \\ \text{subject to} \quad & \|\mathcal{A}(P)-1_S\|_\infty \le \delta \end{aligned} \tag{3.1}$$

We introduce auxiliary variables to apply ADMM. In particular, let $Y=\psi \in \mathbb{R}^{r\times N}$, $X=Y^T \in \mathbb{R}^{N\times r}$ and $P \in \mathbb{R}^{N\times N}$. Then

$$\begin{aligned} \underset{P,X,Y,q}{\text{minimize}} \quad & \frac{1}{2}\|P-XY\|_F^2 \\ \text{subject to} \quad & \mathcal{A}(P)=q \\ & Y=X^T \\ & \|q-1_S\|_\infty \le \delta \end{aligned} \tag{3.2}$$

This gives $Y=\psi \in \mathbb{R}^{r\times N}$ such that the RIP holds for all secant vectors in the secant set $S(\mathcal{X})$ for an isometry constant δ. The optimization formulation for (3.1) is conceptually simple, only requiring the input data set $\mathcal{X}$, desired isometry constant $\delta>0$ and desired rank r.

An important caveat is that our optimization problem is non-convex. Thus, we cannot guarantee that FroMax will converge to the optimal solution of (3.1). However, various experiments in §7 indicate that FroMax yields excellent, stable results for real-world data sets and finds projection matrices much more quickly than NuMax and other convex approaches. We implement ADMM since Wang et. al. [15] indicate that ADMM is more likely to converge than the Augmented Lagrangian Method for nonconvex, nonsmooth problems.

3.2. ADMM. We develop our algorithm, FroMax, to solve (3.2) based on ADMM. We relax the linear constraints and form an augmented Lagrangian of (3.2) as follows:

$$\begin{aligned} L_A(X,Y,q,P) = {} & \frac{1}{2}\|P-XY\|_F^2 + \Gamma\cdot(\mathcal{A}(P)-q) + \Pi\cdot(Y-X^T) \\ & + \frac{\beta_1}{2}\|\mathcal{A}(P)-q\|_2^2 + \frac{\beta_2}{2}\|Y-X^T\|_F^2 + \iota_{\{q:\|q-1_S\|_\infty\le\delta\}} \end{aligned} \tag{3.3}$$

Here, $\Gamma \in \mathbb{R}^{N\times N}$ and $\Pi \in \mathbb{R}^{r\times N}$ represent the scaled Lagrange multipliers. The indicator function, $\iota_{\{q:\|q-1_S\|_\infty\le\delta\}}$, is defined as

$$\iota_{\{q:\|q-1_S\|_\infty\le\delta\}} = \begin{cases} 0 & \text{if } \|q-1_S\|_\infty \le \delta \\ \infty & \text{otherwise} \end{cases}$$

The optimization in (3.3) is carried out over $P \in \mathbb{R}^{N\times N}$, $X \in \mathbb{R}^{N\times r}$, $Y \in \mathbb{R}^{r\times N}$, and $q \in \mathbb{R}^S$, while Γ and Π are also iteratively updated. We optimize each variable in an alternating fashion like NuMax. The following steps below are performed until convergence.

Update q: Isolating the terms that involve q, we obtain a new estimate q_{k+1} as the solution of the constrained optimization problem

$$q_{k+1} \leftarrow \arg\min_q \ \Gamma\cdot(\mathcal{A}(P)-q) + \frac{\beta_1}{2}\|\mathcal{A}(P)-q\|_2^2 + \iota_{\{q:\|q-\mathbf{1}_S\|_\infty\le\delta\}} \tag{3.4}$$

Define $z = \mathcal{A}(P) - \Pi - 1_S$. Using a component-wise truncation procedure for entries in q, we easily see that

$$q_{k+1} = 1_S + \text{sign}(z) \cdot \min(|z|, \delta) \tag{3.5}$$

where the sign and min operators are applied component-wise.

Update P: Isolating the terms that involve P, we obtain a new estimate P_{k+1} as the solution of the constrained optimization problem

$$P_{k+1} \leftarrow \arg\min_{P} \frac{1}{2}\|P - XY\|_F^2 + \Gamma \cdot (\mathcal{A}(P) - q) + \frac{\beta_1}{2}\|\mathcal{A}(P) - q\|_2^2 \tag{3.6}$$

such that $P \succeq 0$. Since this is a least-squares problem, we can solve for the minimum by solving the linear system of equations

$$(P - XY) + \sum_{j=1}^{s} \Gamma_j v_j v_j^T + \beta_1 \mathcal{A}^*(\mathcal{A}(P) - q) = 0 \tag{3.7}$$

where $\mathcal{A}^*$ is the adjoint of $\mathcal{A}$.

Update X: Isolating the terms that involve X, we obtain a new estimate X_{k+1} as the solution of the constrained optimization problem

$$X_{k+1} \leftarrow \arg\min_{X} \frac{1}{2}\|P - XY\|_F^2 + \Pi \cdot (Y - X^T) + \frac{\beta_2}{2}\|Y - X^T\|_F^2 \tag{3.8}$$

It is easily seen that this can be solved similarly to the P update.

Update Y: Isolating the terms that involve Y, we obtain a new estimate Y_{k+1} as the solution of the constrained optimization problem

$$Y_{k+1} \leftarrow \arg\min_{Y} \frac{1}{2}\|P - XY\|_F^2 + \Pi \cdot (Y - X^T) + \frac{\beta_2}{2}\|Y - X^T\|_F^2 \tag{3.9}$$

It is easily seen that this can be solved similarly to the X update.

Update Γ, Π: Following standard augmented Lagrangian methods, we update Γ, Π according to the following equations

$$\Gamma_{k+1} \leftarrow \Gamma_k + \eta\beta_1(\mathcal{A}(P_{k+1}) - q_{k+1}) \tag{3.10}$$

$$\Pi_{k+1} \leftarrow \Pi_k + \eta\beta_2(Y_{k+1} - X_{k+1}^T) \tag{3.11}$$

Pseudocode for FroMax may be found in Algorithm 1. Convergence properties of FroMax are highly dependent on chosen parameters η, β_1, and β_2.

4. NILE-Pro. Our second algorithm, *Near-Isometric Linear Embedding via Proximal Mapping*, or *NILE-Pro* seeks to minimize the RIP constraint directly to solve for ψ. This minimization problem is solved using ADMM and a Moreau decomposition on a proximal mapping.

4.1. Optimization Framework. We formulate a new framework for NILE-Pro. We solve for our desired linear embedding ψ directly:

$$\underset{\psi}{\text{minimize}} \ \|\mathcal{A}(\psi^T\psi) - 1_S\|_\infty \tag{4.1}$$

Algorithm 1 FroMax

Inputs: Secant set $\mathcal{S}(\mathcal{X}) = \{v_i\}_{i=1}^S$, isometry constant δ, desired rank for P r, max iterations $m > 0$

for $k = 1$ **to** m **do**

$z \leftarrow \mathcal{A}(P_k) + \frac{\Gamma}{\beta_1} - 1_S$

$q_{k+1} \leftarrow \mathbf{1}_S + \text{sign}(z) \cdot \min(|z|, \delta)$

$P_{k+1} \leftarrow (I + \beta_1 \mathcal{A}^*\mathcal{A})^\dagger(\beta_1 \mathcal{A}^* q_{k+1} + X_k Y_k - \sum_{j=1}^s \Gamma_j v_j v_j^T)$

$X_{k+1} \leftarrow (\Pi_k^T + \beta_2 Y_k^T + P_{k+1} Y_k^T)(Y_k Y_k^T + \beta_2 I)^{-1}$

$Y_{k+1} \leftarrow (X_{k+1}^T X_{k+1} + \beta_2 I)^{-1}(X_{k+1}^T P_{k+1} - \Pi_k + \beta_2 X_{k+1}^T)$

$\Gamma_{k+1} \leftarrow \Gamma_k + \eta\beta_1(\mathcal{A}(P_{k+1}) - q_{k+1})$

$\Pi_{k+1} \leftarrow \Pi_k + \eta\beta_2(Y_{k+1} - X_{k+1}^T)$

if $\frac{1}{2}\|P_{k+1} - X_{k+1}Y_{k+1}\|_F^2 < \epsilon$ **then**

break

end if

end for

By introducing another variable q, we then have the following minimization problem:

$$\begin{aligned} &\underset{q,\psi}{\text{minimize}} \ \|q - 1_s\|_\infty \\ &\text{subject to } q = \mathcal{A}(\psi^T \psi) \end{aligned} \tag{4.2}$$

We apply ADMM and use a Moreau decomposition on a proximal mapping to solve for updates. Like FroMax, this optimization problem is non-convex and thus, we cannot guarantee that NILE-Pro will converge to the optimal solution of (4.1). However, we demonstrate in §7 that NILE-Pro may produce stable, excellent results for synthetic and real-world data sets at a much faster rate than both FroMax and NuMax due to the simplified problem it solves.

4.2. Proximal Mapping and Moreau Decomposition. We introduce some machinery to solve this minimization problem [12]:

DEFINITION 4.1. *The proximal mapping of a convex and proper function f is defined to be*

$$\text{prox}_f(x) = \arg\min_u (f(u) + \frac{1}{2}\|u - x\|_2^2)$$

THEOREM 4.2. *If a function $f : \mathbb{R}^n \to \mathbb{R}$ is proper, closed, and convex, then* $\text{prox}_f(x)$ *exists, well-defined, and unique for all x.*

Moreau's identity allows us to decompose any x into

$$x = \text{prox}_f(x) + \text{prox}_{f^*}(x)$$

where f^* is the convex conjugate of f. This decomposition, called the *Moreau decomposition*, generalizes the orthogonal decomposition on subspaces. We apply this machinery to help us solve for the update for q.

Algorithm 2 NILE-Pro

Inputs: Secant set $\mathcal{S}(\mathcal{X}) = \{v_i\}_{i=1}^{S}$, isometry constant δ, max iterations $m > 0$, initial rank r
for $k = 0, ..., m$ **do**
  $\tau \leftarrow \mathcal{A}(\psi_k^T \psi_k) - \omega - 1_s$
  $q_{k+1} \leftarrow \frac{1}{\beta}\Big(\beta\tau - \mathcal{P}_{\{\|x\|_1 \leq 1\}}(\beta\tau)\Big) + 1_S$
  $\psi_{k+1} \leftarrow \psi_k - 2\eta\psi_k \mathcal{A}^*(\mathcal{A}(\psi_k^T \psi_k) - q_{k+1} - \omega)$
  $\omega_{k+1} \leftarrow \omega_k - \beta(\mathcal{A}(\psi_{k+1}^T \psi_{k+1}) - q_{k+1})$
  $\epsilon_0 \leftarrow \|\mathcal{A}(\psi_{k+1}^T \psi_{k+1}) - 1_s\|_\infty$
  if $\epsilon_0 < \epsilon$ **then**
    break
  end if
end for

4.3. ADMM. Following a similar method as FroMax, we relax our linear constraints and find our augmented Lagrangian of (14):

$$L_A(\psi, q; \omega) = \|q - 1_S\|_\infty + \frac{\beta}{2}\|\mathcal{A}(\psi^T\psi) - q - \omega\|_2^2 \tag{4.3}$$

Here, $\omega \in \mathbb{R}^S$ is the scaled Lagrange multiplier. The optimization in (4.3) is carried out over $\psi \in \mathbb{R}^{r \times N}$ and $q \in \mathbb{R}^S$, while ω is updated. Each variable is updated in an alternating fashion. The following steps below are performed until convergence.

Update ψ: Isolating the terms that involve ψ, we obtain a new estimate ψ_{k+1} as the solution of the constrained optimization problem

$$\psi_{k+1} \leftarrow \arg\min_{\psi} \frac{\beta}{2}\|\mathcal{A}(\psi^T\psi) - q - \omega\|_2^2 \tag{4.4}$$

Update q: Isolating the terms that involve ψ, we obtain a new estimate ψ_{k+1} as the solution of the constrained optimization problem

$$q_{k+1} \leftarrow \arg\min_{q} \|q - 1_S\|_\infty + \frac{\beta}{2}\|\mathcal{A}(\psi^T\psi) - q - \omega\|_2^2 \tag{4.5}$$

Setting $X = q - 1_S$ and $\tau = \mathcal{A}(\psi^T\psi) - \omega - 1_S$, we apply a Moreau decomposition on a proximal mapping to solve for the q update:

$$X = \frac{1}{\beta}(\beta\tau - \text{prox}_{(\|X\|_1 \leq 1)}(\beta\tau)) \tag{4.6}$$

$$q = \frac{1}{\beta}(\beta\tau - \text{prox}_{(\|X\|_1 \leq 1)}(\beta\tau)) + 1_S \tag{4.7}$$

Update ω: Following standard augmented Lagrangian methods, we update ω according to the following equation

$$\omega_{k+1} \leftarrow \omega_k - \beta(\mathcal{A}(\psi_{k+1}^T\psi) - q_{k+1}) \tag{4.8}$$

Pseudocode for NILE-Pro may be found in Algorithm 2.

Algorithm 3 FroMax/NILE-Pro RA

Inputs: Secant set $\mathcal{S}(\mathcal{X}) = \{v_i\}_{i=1}^{S}$, isometry constant δ, max iterations for algorithm $m > 0$, initial rank R_0, max iterations for RA M, ψ_0
for k = 1, ..., M **do**
  $\psi \leftarrow$ FroMax/NILE-Pro$(S, \delta, m, R_0, \psi_0)$
  $P \leftarrow \psi^T \psi$
  $R_{k+1} \leftarrow R_k - 1$
  $[\Gamma, U] \leftarrow \text{eig}(P)$
  $\psi_0 \leftarrow \Gamma^{1/2} U^T$
  if FroMax/NILE-Pro does not converge **then**
    break
  end if
end for
return $\psi, r \leftarrow R_k$

5. Rank Adjustment and Column Generation. In this section, we discuss rank adjustment and column generation heuristics. We develop rank adjustment methods to discover the lowest optimal rank for both FroMax and NILE-Pro. Abbreviating rank adjustment to *RA*, we call our rank adjusted algorithms *FroMax RA* and *NILE-Pro RA*, respectively. We also use column generation techniques following Hegde et. al. [6] to work with subsets of $S(\mathcal{X})$ to lower the memory complexity of these algorithms, which we name *FroMax CG* and *NILE-Pro CG*, respectively. We discuss each heuristic algorithm in detail below.

5.1. Rank Adjustment. Though FroMax and NILE-Pro may dramatically decrease the time of solving for projection matrix ψ, both algorithms do not find an optimal rank for dimensionality reduction like NuMax. Hence, we propose a heuristic rank adjustment method that uses the discovered matrix $P = \psi^T \psi$ to give a good initialization for ψ of lower rank.

Given a sufficiently large rank, $R_0 \gg r$, the optimal rank, we run our dimensionality reduction algorithm for a maximum number of iterations or until convergence to find ψ. If our algorithm converges, we return $P = \psi^T \psi$ and find $\psi_0 = \Gamma_M^{1/2} U_M^T$, where $P = U \Gamma U^T$ from P's eigendecomposition. We then initialize our algorithm again with rank $R_1 = R_0 - 1$ and ψ_0 which we found in the last iteration and test again for convergence. We continue this process until we reach the maximum number of iterations within the algorithm and return the ψ given in the last iteration, considering its rank $r = R_k$ to be optimal. We summarize our rank adjustment heuristic in Algorithm 3.

5.2. Column Generation. Since FroMax and NILE-Pro use the secants of a given data set, applications involving millions of secants may be prohibited by the memory complexity of these algorithms. Some methods that are used to address large data sets include stochastic and online methods. Stochastic methods use random subsets of the data to learn an estimate for the entire data set. Online methods uses sequentially available data to update the current iterate then discards the information. Our column generation algorithms, FroMax CG and NILE-Pro CG, combines stochastic and online methods to estimate solutions to large-scale problems.

Similar to NuMax's column generation, which is based off of the Karush-Kuhn-Tucker (KKT) conditions, we apply a simple, greedy method to rapidly find the active constraints for (3.1) or (4.1).

Algorithm 4 FroMax/NILE-Pro CG

Inputs: Secant set $\mathcal{S}(\mathcal{X}) = \{v_i\}_{i=1}^{S}$, isometry constant δ, max iterations for algorithm $m > 0$, rank r, the FroMax or NILE-Pro algorithm
while not converged **do**
$\widehat{S} \leftarrow \{v_i \in S_0 : |v_i^T \psi^T \psi v_i - 1| \geq \delta\}$
$S_1 \leftarrow \{v_i \in S : v_i \notin S_0\}_{i=1}^{S''}$
$\widehat{S} \leftarrow \widehat{S} \bigcup \{v_i \in S_1 : |v_i^T \psi^T \psi v_i = 1| \geq \delta\}$
$\psi \leftarrow$ FroMax/NILE-Pro$(\widehat{S}, \delta)$
$S_0 \leftarrow \widehat{S}$
end while
return ψ

1. Solve (3.1) or (4.1) with a small subset $S_0 \subset S(\mathcal{X})$ using FroMax (Algorithm 1) or NILE-Pro (Algorithm 2), respectively to obtain an initial estimate $\widehat{\psi}$. Identify the set $\widehat{S}$ of secants that correspond to the active constraints:

$$\widehat{S} \leftarrow \{v_i \in S_0 : |v_i^T \widehat{\psi}^T \widehat{\psi} v_i - 1| \geq \delta\}$$

2. Select additional secants $S_1 \subset S$ not selected previously and identify all secants among S_1 that violate the constraint at the current estimate $\widehat{\psi}$. Then, append these secants to the set of active constraints $\widehat{S}$ to obtain an augmented set $\widehat{S}$

$$\widehat{S} \leftarrow \widehat{S} \bigcup \{v_i \in S_1 : |v_i^T \widehat{\psi}^T \widehat{\psi} v_i - 1| \geq \delta\}$$

3. Solve (3.1) or (4.1) with the new augmented set $\widehat{S}$ using FroMax or NILE-Pro to obtain a new estimate $\widehat{\psi}$.
4. Identify the secants that correspond to active constraints and repeat Steps 2 and 3 until convergence is reached for $\widehat{\psi}$.

Column generation allows us to perform a large numerical optimization procedure on smaller subsets of $S(\mathcal{X})$, resulting in significant computational gains. A key benefit of FroMax CG and NILE-Pro CG is that the subsets of secants used during each iteration never has to be explicitly stored in memory and can be generated on the fly. This leads to significant improvements in memory complexity.

However, because FroMax and NILE-Pro are already both non-convex, column generation makes these algorithms even less predictable. Though these algorithms are not guaranteed to converge to an optimal solution, they appear to yield excellent results on large, real-world data sets, as we will show in §7.

Pseudocode for FroMax/Nile-Pro CG is found in Algorithm 4. Our column generation method converges when no additional secants violate our constraint.

6. Convergence of Algorithms. Since FroMax and NILE-Pro are derived from applying the ADMM to non-convex problems, the convergence properties of these algorithms can be understood based on the convergence properties of ADMM. For certain types of convex problems, ADMM has been shown to converge at a rate of $o(1/k)$ [3]. However, since our problems are non-convex, convergence analyses of ADMM do not apply.

		FroMax		NILE-Pro	
#Data	δ	Rank	Time	Rank	Time
60	0.4	9	1.3	9	0.7
	0.25	9	1.2	9	1.1
	0.1	13	1.4	13	1.5
200	0.4	16	511.2	16	109.1
	0.25	18	269.4	18	144.4
	0.1	27	74.5	27	448.5

Table 7.1

Comparison of runtime performance for FroMax and NILE-Pro on $S(\mathcal{X}_1)$ and $S(\mathcal{X}_2)$ given δ and rank.

7. Numerical Experiments. We demonstrate the performance of the FroMax and NILE-Pro algorithms in comparison to prior methods including NuMax. All of our experiments are performed on computers with Intel i5-650 processors and 4 GB of RAM unless otherwise specified. We test and compare the speed and accuracy of our algorithms through various tests on real-world and synthetic data sets.

7.1. Linear Low-Dimensional Embeddings. We first consider synthetic data sets $\mathcal{X}_1$ and $\mathcal{X}_2$ consisting of $7 \times 7 = 49$ and $14 \times 14 = 196$ dimensional images of translations of a white square on a black box respectively. We construct our training sets by randomly generating 60 49-dimensional images for $\mathcal{X}_1$ and 200 196-dimensional images for $\mathcal{X}_2$. We then construct secant sets $S(\mathcal{X}_1)$ and $S(\mathcal{X}_2)$ by computing the normalized pairwise difference vectors between different images. We compare FroMax and NILE-Pro's performance of producing linear, low-dimensional embeddings on these two data sets in Table 1.

Since NILE-Pro minimizes the RIP directly, NILE-Pro intuitively will converge faster for larger δ. Our experimental results match our theoretical intuition since NILE-Pro converges significantly faster for larger δ than lower δ.

FroMax experimentally converges faster for smaller δ than larger δ. Smaller δ restricts q to a smaller feasible set given by the RIP, leading to faster convergence.

Also note that both algorithms' computational complexity scale significantly with the size of the data set due to the use of the secant set. Our runtime results comparing $S(\mathcal{X}_1)$ and $S(\mathcal{X}_2)$ reflect this.

7.2. Linear Low-Dimensional Embeddings with Rank Adjustment. In §7.1, we input a given rank for FroMax and NILE-Pro and compare their run time. However, usually the optimal rank for dimension reduction is not known, motivating the development of rank adjustment heuristics. To analyze the performance of our rank adjustment heuristic, we consider a synthetic data set $\mathcal{X}$ comprised of $16 \times 16 = 256$ dimensional images of translations of a white square or a disk on a black box respectively, see figure 7.1. We construct a secant set $S(\mathcal{X})$ and compare PCA, Numax RA, FroMax RA and NILE-Pro RA's performance of producing linear, low-dimensional embeddings on this data set.

Figure 7.2 plots the variation of the number of measurements M as a function of the isometry constant δ. We observe that NILE-Pro RA achieves the desired isometry constant on the secants using by far the fewest number of measurements. FroMax RA performs better for small δ due to the correlation between δ and q, as we discussed before. Moreover, both Numax RA and Fromax RA greatly outperform PCA, a popular embedding technique in the literature.

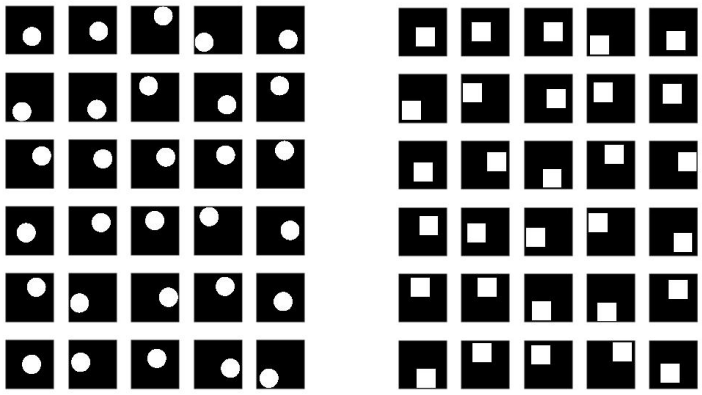

FIG. 7.1. *Our synthetic training set consists of sixty* 256*-dimensional random generated translating disks and squares figure.*

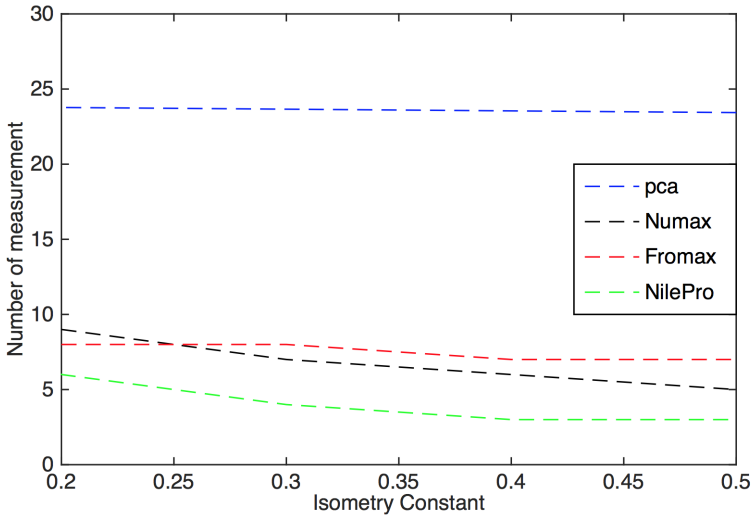

FIG. 7.2. *A comparison of the isometry constant* δ *with the number of measurements for PCA, Numax RA, FroMax RA and NILE-Pro RA's performance of producing linear, low-dimensional embeddings.*

7.3. Runtime Performance on MNIST with Rank Adjustment. In this experiment, we consider a more challenging, real-world data set, the MNIST data set, see figure 7.3. MNIST contains many digital images of handwritten digits and is a common benchmark data set for machine learning. We examine subsets of the training set for the digit "5". We take subsets consisting of 95, 200, and 500 data points with original dimension 49.

We test runtime performance of FroMax and NILE-Pro RA on these data sets. Our results may be found in Table 2.

Our experimental results show that NILE-Pro RA may perform significantly faster

FIG. 7.3. *Examples of 5 images from the MNIST dataset.*

		NILE-PRO		FROMAX		NUMAX	
δ	#DATA	RK	TIME	RK	TIME	RK	TIME
	95	7	25	9	102	12	71
0.4	200	9	96	15	520	21	311
	500	11	710	27	2490	25	3477
	95	11	28	11	111	14	56
0.2	200	14	130	16	569	18	557
	500	18	751	40	1498	27	3517
	95	15	41	15	91	16	21
0.1	200	20	165	19	823	21	279
	500	25	1285	44	650	30	3410

TABLE 7.2

Comparison of runtime performance for FroMax RA, NILE-Pro RA, and NuMax on subsets of "5" images from MNIST.

and give a better optimal rank than NuMax while FroMax RA converges slower for larger data sets. This may be due to the nature of the local minima found in FroMax; the estimate for $P = \psi^T \psi$ given for a larger rank does not correspond to the local minima for lower ranks so that this initialization is beneficial.

Our results for FroMax RA reveal another issue with our rank adjustment method. FroMax RA appears to struggle with determining the optimal rank, sometimes performing worse than NuMax. We believe that our algorithm may be converging to local minima, which makes our rank adjustment ineffective. This issue motivates us to look into other rank adjustment methods that start at a sufficiently low rank and examine higher ranks to discover the optimal rank.

Also, since rank adjustment for NILE-Pro is still based on the core NILE-Pro algorithm, we see that NILE-Pro RA converges in much slower time for smaller δ.

The former caveat motivates us to continue looking for better rank adjustment methods for FroMax.

7.4. Nearest Neighbor Classification on MNIST. The MNIST data set consists of 60,000 training data points and 10,000 test data points of handwritten digits [8]. The dataset contains 10 classes corresponding to each digit from 0-9. For this experiment, we use the $N = 20 \times 20 = 400$-dimensional data set that excludes extra space at the boundaries. We use NuMax CG and FroMax CG to embed our

δ	Rank	NuMax CG	Time (hrs)	FroMax CG	Time (hrs)
0.4	72	3.09%	0.926	3.00%	0.411
0.25	98	3.15%	1.273	3.26%	0.811
0.1	167	3.31%	2.664	3.42%	1.358

Table 7.3

Comparison of misclassification rates and run-time performance of approximate nearest neighbor classifiers using NuMax CG and FroMax CG for given δ and rank on the MNIST test set.

MNIST training set into a lower dimensional space and nearest neighbor classification.

The misclassification rate of nearest neighbor classification on the unchanged data set is 3.47%. Table 7.3 gives the nearest neighbor classification misclassification rate for NuMax CG and FroMax CG for given δ and rank applied on the MNIST data set. In particular, though NuMax CG and FroMax CG give similar misclassification rates, FroMax CG has significantly better runtime performance than NuMax CG. Though a combined rank adjustment and column generation method has not been implemented for FroMax, the results suggest that FroMax may find a sufficiently good projection matrix in much less time.

7.5. Approximate Nearest Neighbors. Given a data set modeled by points in Euclidean space and a query point, *nearest neighbors* identifies the k closest points in the data set [2]. These points are usually used for further processing, such as unsupervised or supervised regression and classification.

However, as the dimension N of the data set grows, the computational cost of identifying the k nearest neighbors also becomes increasingly expensive. An alternative to computing nearest neighbors directly is to embed the data into a lower-dimensional subspace while preserving near-isometry, then applying nearest neighbor techniques. This method is called *approximate nearest neighbors*. Since NuMax, FroMax, and NILE-Pro construct low rank, near-isometric linear embeddings for a given distortion δ, they may potentially enable efficient ANN computations for high-dimensional data sets.

For this experiment, we use the LabelMe data set consisting of 4000 images of indoor and outdoor scenes [13]. We then computed GIST descriptors for each image, which are vectors of size $N = 512$ that roughly describe the overall spatial statistics of the image [11]. We then used NuMax CG and FroMax CG to estimate low rank, near-isometric linear embeddings for this data set for a given distortion parameter δ. Then we perform ANN computations on 1000 test data points in the corresponding low dimensional space. We compute embeddings of various ranks for FroMax CG to compare performance between different ranks.

Figure 7.4 demonstrates that FroMax CG generally attains similar if not better performance than NuMax CG for the same rank. In fact, our results suggest that FroMax CG could perform similarly at a lower rank than NuMax CG. We leave further investigation for future research.

8. Discussion.

8.1. Research Overview. In this paper, we construct two comprehensive algorithmic frameworks for finding near-isometric linear embeddings of high-dimensional data sets. Based on the convex optimization formulation in NuMax, we proposed two non-convex minimization approaches which approximately preserve the norms of all pairwise secants of the given dataset. In particular, we developed two algorithms,

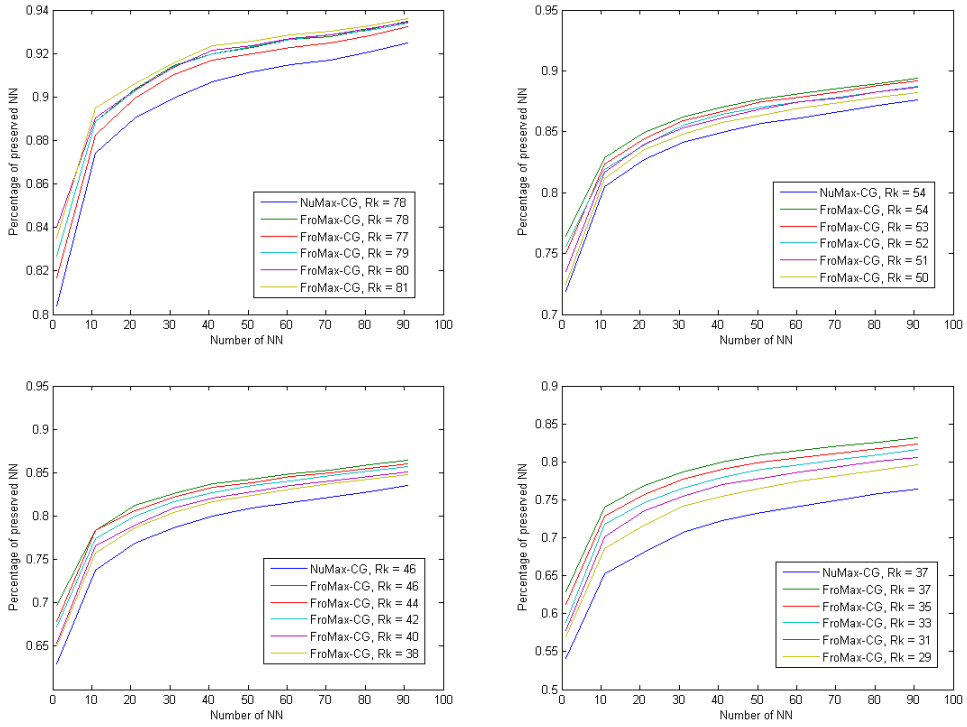

FIG. 7.4. *Comparison of FroMax CG and NuMax CG on preserving nearest neighbors.*

FroMax and NILE-Pro, that may construct the desired embedding with smaller computational complexity than NuMax.

Since NuMax automatically discovers the optimal rank, we created a rank adjustment method for finding the best rank for our algorithms. We also implemented column generation in addition to FroMax and NILE-Pro so our algorithms can be adapted to perform on larger data sets.

Constructing linear, information-preserving embeddings of high-dimensional signals to lower-dimensional signals have become of significant importance for a wide range of machine learning and compressive sensing applications. However, little is known about near-isometric linear embeddings beyond the Johnson-Lindenstrauss Lemma. The frameworks discussed in this paper build on the convex, deterministic approach of NuMax to produce practical, potentially more computationally efficient dimension reduction algorithms that are both information-preserving and feasible for a broad range of applications. Though we do not provide an analytical foundation to our work due to the non-convex nature of our algorithms, we hope to initiate work in developing a theoretical basis for similar work.

8.2. Future Work. There are still many challenges left to tackle. As discussed in §7, we still need to further develop rank adjustment methods for FroMax and column generation techniques for NILE-Pro. We would also like to incorporate both rank adjustment and column generation together. One direction is to consider an eigengap heuristic for rank adjustment, in which some heuristic is set based on the difference between singular values of the matrix P to determine the next chosen rank.

In addition, further testing and parameter-tweaking is necessary to analyze and optimize the stability and performance of our algorithms on various data sets. Other heuristics for non-convex optimization, such as applying perturbations to avoid local minima, may also be applied to give better solutions. We defer the study of these challenges and heuristics for future research.

Acknowledgements. The work of Jerry Luo, Kayla Shapiro, and Hao-Jun Michael Shi were supported in part by the California Research Training Program for Computational and Applied Mathematics 2014 under NSF Grant DMS-1045536. The work of Qi Yang was supported in part by USC Provost's Undergraduate Research Fellowship and the WiSE Research Experience for Undergradutes. Thanks to Dr. Ming Yan and Dr. Wotao Yin for their consistent advice, support, and mentorship throughout the entirety of this project.

REFERENCES

[1] Emmanuel J Candes and Terence Tao, *Decoding by linear programming*, Information Theory, IEEE Transactions on, 51 (2005), pp. 4203–4215.

[2] Thomas M Cover and Peter E Hart, *Nearest neighbor pattern classification*, Information Theory, IEEE Transactions on, 13 (1967), pp. 21–27.

[3] Wei Deng, Ming-Jun Lai, and Wotao Yin, *On the o (1/k) convergence and parallelization of the alternating direction method of multipliers*, arXiv preprint arXiv:1312.3040, (2013).

[4] Edward Grant, Chinmay Hegde, and Piotr Indyk, *Nearly optimal linear embeddings into very low dimensions*, in Global Conference on Signal and Information Processing (GlobalSIP), 2013 IEEE, IEEE, 2013, pp. 973–976.

[5] Chinmay Hegde, Aswin C Sankaranarayanan, and Richard G Baraniuk, *Near-isometric linear embeddings of manifolds*, in Statistical Signal Processing Workshop (SSP), 2012 IEEE, IEEE, 2012, pp. 728–731.

[6] C. Hegde, A. C. Sankaranarayanan, W. Yin, and R. G. Baraniuk, *NuMax: a convex approach for learning near-isometric linear embeddings*, IEEE Transactions on Signal Processing, 63 (2015).

[7] William Johnson and Joram Lindenstrauss, *Extensions of Lipschitz mappings into a Hilbert space*, in Conference in modern analysis and probability (New Haven, Conn., 1982), vol. 26 of Contemporary Mathematics, American Mathematical Society, 1984, pp. 189–206.

[8] Yann LeCun, Corinna Cortes, and Christopher JC Burges, *The MNIST database of handwritten digits*, 1998.

[9] Daniel D. Lee and H. Sebastian Seung, *Algorithms for non-negative matrix factorization*, in Advances in Neural Information Processing Systems 13, T.K. Leen, T.G. Dietterich, and V. Tresp, eds., MIT Press, 2001, pp. 556–562.

[10] Bruce Moore, *Principal component analysis in linear systems: Controllability, observability, and model reduction*, Automatic Control, IEEE Transactions on, 26 (1981), pp. 17–32.

[11] Aude Oliva and Antonio Torralba, *Modeling the shape of the scene: A holistic representation of the spatial envelope*, International journal of computer vision, 42 (2001), pp. 145–175.

[12] R. Tyrrell Rockafellar, *Convex analysis*, 1970.

[13] Bryan C Russell, Antonio Torralba, Kevin P Murphy, and William T Freeman, *Labelme: a database and web-based tool for image annotation*, International journal of computer vision, 77 (2008), pp. 157–173.

[14] Ali Sadeghian, Bubacarr Bah, and Volkan Cevher, *Energy-aware adaptive bi-lipschitz embeddings*, in 10th International Conference on Sampling Theory and Applications (SampTA), 2013.

[15] Yu Wang, Wotao Yin, and Jinshan Zeng, *Global convergence of admm in nonconvex nonsmooth optimization*, arXiv preprint arXiv:1511.06324, (2015).

[16] Yangyang Xu, Wotao Yin, Zaiwen Wen, and Yin Zhang, *An alternating direction algorithm for matrix completion with nonnegative factors*, Frontiers of Mathematics in China, 7 (2012), pp. 365–384.