Interest Zone Matrix Approximation

Gil Shabat and Amir Averbuch

February 19, 2011

Abstract

We present an algorithm for low rank approximation of matrices where only some of the entries in the matrix are taken into consideration. This algorithm appears in recent literature under different names, where it is described as an EM based algorithm that maximizes the likelihood for the missing entries without any relation for the mean square error minimization. When the algorithm is minimized from a mean-square-error point of view, we prove that the error produced by the algorithm is monotonically decreasing. It guarantees to converge to a local MSE minimum. We also show that an extension of the EM based algorithm for weighted low rank approximation, which appeared in recent literature, claiming that it converges to a local minimum of the MSE is wrong. Finally, we show the use of the algorithm in different applications for physics, electrical engineering and data interpolation.

Keywords: Singular Value Decomposition, matrix completion, matrix approximation

1 Introduction

Matrix completion and matrix approximation are important problems in a variety of fields such as statistics [1], biology [2], statistical machine learning [3], signal processing and computer vision/image processing [4]. Rank reduction by matrix approximation is important for example in compression where low rank indicates the existence of redundant information. Therefore, low rank matrices are better compressed. In statistics, matrix completion can be used for survey completion and in image processing for interpolation needs. Since in

general, low rank matrix completion is a NP-hard problem, some relaxations methods have been proposed. For example, instead of solving the problem

$$\begin{aligned} &\text{minimize } \text{rank}(\mathbf{X}) \\ &\text{subject to } X_{i,j} = M_{i,j}, \quad (i,j) \in \Omega \end{aligned} \tag{1.1}$$

it can be approximated by

$$\begin{aligned} &\text{minimize } ||\mathbf{X}||_* \\ &\text{subject to } X_{i,j} = M_{i,j}, \quad (i,j) \in \Omega \end{aligned} \tag{1.2}$$

where $||\mathbf{X}||_*$ specifies the nuclear norm of $\mathbf{X}$ which is equal to the sum of the singular values of $\mathbf{X}$. A small value of $||\mathbf{X}||_*$ is related with having a low rank [5]. An iterative solution, which is based on a singular value thresholding, was given in [6]. A completion algorithm based on the local information of the matrix was proposed in [7]. This powerful approach enables to divide a large matrix into a set of smaller blocks, which can be processed in parallel and thus it suits processing of large matrices.

In this paper, we are interested somehow in the "opposite" problem: we are looking for a matrix $\mathbf{X}$ that minimizes the difference between its entries to a set of given entries subject to a given rank. Mathematically, it is formulated as

$$\begin{aligned} &\text{minimize } \textstyle\sum_{(i,j)\in\Omega} |X_{i,j} - M_{i,j}|^2 = \|\mathcal{P}\mathbf{X} - \mathcal{P}\mathbf{M}\|_F^2 \\ &\text{subject to } \text{rank}(\mathbf{X}) = k. \end{aligned} \tag{1.3}$$

Solution for a special case, in which Ω specifies the entire entries of the matrix $\mathbf{M}$ with rank n approximated by the matrix $\mathbf{X}$ with rank $k < n$, is known as the Eckart-Young Theorem [8] and it is given by singular value decomposition (SVD). However, when only some entries participate in the process, the solution provides more degrees of freedom for the approximation. Hence, there are many possibilities to approximate the matrix and the solution is not unique.

A generalization of the Eckart-Young matrix approximation theorem is given in [9], where the low rank approximation of the matrix keeps a specified set of the unchanged columns. An algorithm for solving the Interest-Zone-Matrix-Approximation problem in Eq. 1.3 appears in recent literature under different names such as "SVD-Impute" [10] and "Hard-Impute" [11], where the motivation for the method came from maximizing the likelihood over the missing

entries by applying the EM algorithm and not from minimizing the mean-squared-root error (MSE). Then, the algorithm has the form

$$\mathbf{X}_n = \mathcal{D}_k((\mathcal{I} - \mathcal{P})\mathbf{X}_{n-1} + \mathcal{P}\mathbf{M}) \tag{1.4}$$

where $\mathcal{D}_k\mathbf{X}$ is the best rank k approximation (in Frobenius norm) for $\mathbf{X}$ and it is keeps the first k singular values of $\mathbf{X}$ while zeroing the rest, i.e. $\mathcal{D}_k\mathbf{X} = \mathbf{U}\mathbf{\Sigma_k}\mathbf{V^T}$ and $\text{diag}(\mathbf{\Sigma_k}) = (\sigma_1, ..., \sigma_k, 0, .., 0)$. Since the algorithm is EM based, it converges to a local maximum of the likelihood. However, this does not say anything about the MSE that we try to minimize. Along with the "Hard-Impute" algorithm in [11], "Soft-Impute" is an additional presented algorithm, which is very similar to the algorithm in Eq. 1.4. The only difference is that $\mathcal{D}_k$ is replaced by a softer operator $\mathcal{B}_\alpha$ which zeros only the singular values of a given matrix that exceed a certain threshold α. The "Soft-Impute" algorithm is the solution for the following problem:

$$\text{minimize}_X \frac{1}{2}\|\mathcal{P}\mathbf{X} - \mathcal{P}\mathbf{M}\|^2 + \lambda\|\mathbf{X}\|_*. \tag{1.5}$$

A convergence proof for Eq. 1.5 shows that the error is monotonically decreasing is given in [11]. However, there is no such proof for the algorithm described in Eq. 1.4.

An attempt to extend Eq. 1.4 to weighted low rank approximations such that the weights are not necessarily 0 or 1 ($\mathcal{P}$ operator) is made in [3] by modifying Eq. 1.4 to be

$$\mathbf{X}_n = \mathcal{D}_k((\mathbf{1} - \mathbf{W}) \otimes \mathbf{X}_{n-1} + \mathbf{W} \otimes \mathbf{M}) \tag{1.6}$$

where $\mathbf{W}$ is a matrix whose elements satisfy $0 \leq w_{i,j} \leq 1$ and $\otimes$ is pointwise multiplication (Hadamard product). Although the paper claims that the algorithm converges to a local minimum of the weighted MSE, numerical experiments show that this is not always the case.

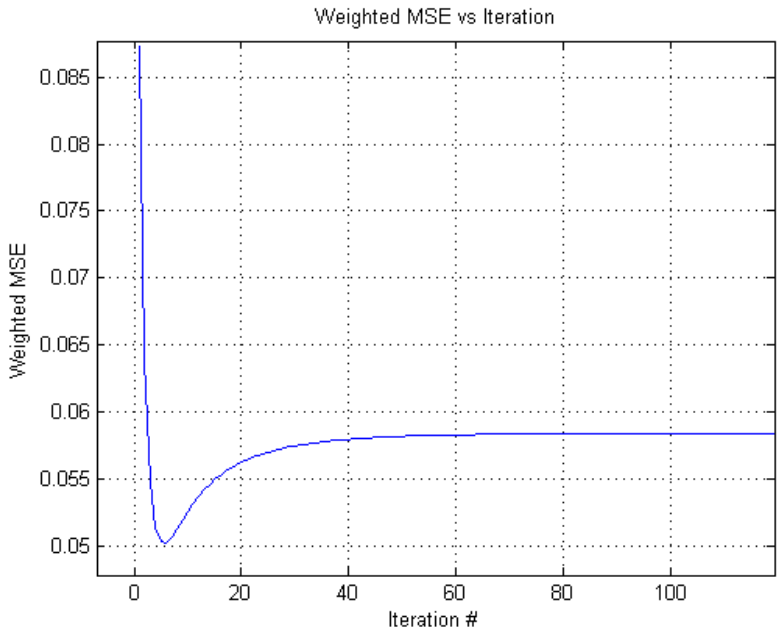


Figure 1.1: A weighted MSE for an arbitrary 2×2 matrix of rank 2 being approximated by a rank 1 matrix

Fig. 1.1 shows that the error decreases and at some point it begins to increase and finally converges to a point whose error is larger. The data we used for the illustration in Fig. 1.1 is:

$$\mathbf{M} = \begin{bmatrix} 0.86 & 0.0892 \\ 0.519 & 0.409 \end{bmatrix}; \mathbf{X_0} = \begin{bmatrix} 0.171 & 0.378 \\ 0.957 & 0.821 \end{bmatrix}; \mathbf{W} = \begin{bmatrix} 0.115 & 0.712 \\ 0.731 & 0.34 \end{bmatrix}.$$

The same phenomenon appears in other examples where matrices of different sizes are used. As will be shown later, this can never happen when the weights are strictly zero or one.

In this paper, we bring a proof for the convergence of Eq. 1.4 and show that the MSE is monotonically decreasing. The proof neither comes from likelihood or probabilistic considerations nor from l_0 or l_1 minimization but rather from geometric considerations.

The paper has the following structure: Section 2 provides preliminary facts for the needed mathematical background. Section 3 describes the Interest Zone Matrix Approximation (IZMA) algorithm and proves its convergence. Section 4 demonstrates the application of the IZMA algorithm to different applications.

2 Preliminaries and Notation

Throughout the paper, the inner product $\langle \mathbf{X}, \mathbf{Y} \rangle$ for matrices refers to the standard inner product $\langle \mathbf{X}, \mathbf{Y} \rangle = \text{trace}(\mathbf{X}^* \mathbf{Y})$. The norm induced by this inner product is the Frobenius norm defined by $\|\mathbf{X}\|^2 = \text{trace}(\mathbf{X}^* \mathbf{X})$.

2.1 Singular Value Decomposition (SVD)

An $m \times n$ matrix $\mathbf{A}$ can be factorized as $\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^*$ where $\mathbf{U}$ is an $m \times m$ unitary matrix, $\mathbf{V}$ is an $n \times n$ unitary matrix and $\mathbf{\Sigma}$ is an $m \times n$ diagonal matrix with $\min(m, n)$ non-negative elements σ_i along the diagonal called "singular values". The number of positive singular values is equal to the rank of the matrix. In addition, the Frobenius norm is related to the singular values by $\|\mathbf{A}\|^2 = \text{trace}(A^* A) = \sum_{i=1}^{\min(m,n)} \sigma_i^2$. $\mathbf{A}$ can be written as a Fourier expansion of the SVD terms such that $\mathbf{A} = \sum_{i=1}^{\min(m,n)} \sigma_i \mathbf{u_i} \mathbf{v_i}^*$.

Eckart-Young Theorem ([8]) deals with an important property of the low rank approximation matrices using SVD: Let $\mathbf{U}\mathbf{\Sigma}\mathbf{V}^*$ be the SVD of $\mathbf{A}$ such that $\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^*$. The best approximation for $\mathbf{A}$ in Frobenius norm is given by $\tilde{\mathbf{A}} = \mathbf{U}\tilde{\mathbf{\Sigma}}\mathbf{V}^*$, where $\tilde{\mathbf{\Sigma}}$ is the same as $\mathbf{\Sigma}$ except it has only the first $k < \min(m, n)$ positive elements along the diagonal (the smallest $\min(m, n) - k$ elements along the diagonal are set to zero). We will use this fact later on in the paper. Another important property of SVD is that it is known to have the best "energy compaction" for all separable decompositions ([13, 17, 18]).

2.2 Definition of Operators

Before the discussion is continued, the following operators are defined:

1. Projection operator $\mathcal{P}$: Let $\mathbf{B}$ be a $m \times n$ matrix with entries $B_{i,j} \in \{0, 1\}$ (a binary matrix with entries 0 and 1) and let $\mathbf{X}$ be a $m \times n$ matrix. $\mathcal{P}\mathbf{X} \triangleq \mathbf{B} \otimes \mathbf{X}$ where $\otimes$ indicates the Hadamard product or pointwise product is defined as $(\mathbf{B} \otimes \mathbf{X})_{i,j} = B_{i,j} X_{i,j}$. It is easily verified that $\mathcal{P} = \mathcal{P}^2$ and that $\mathcal{P} = \mathcal{P}^*$. Note that $\mathcal{P}$ indicates the known entries locations we wish to approximate.

2. Rank reduction operator $\mathcal{D}$: Let $\mathbf{X}$ be an $m \times n$ matrix of rank r and let $\mathbf{U}\mathbf{\Sigma}\mathbf{V}^*$ be

its SVD. $\mathcal{D}\mathbf{X}$ is an $m \times n$ matrix of rank $k \leq r$ defined as $\mathcal{D}\mathbf{X} \triangleq \mathbf{U}\tilde{\mathbf{\Sigma}}\mathbf{V}^*$ where $\tilde{\mathbf{\Sigma}}$ is the same as $\mathbf{\Sigma}$ except that it has only the first k, instead of r, positive singular values along the diagonal. Note that $\mathcal{D}\mathbf{X}$ is the best Frobenius-norm rank k approximation of $\mathbf{X}$ and that it is a nonlinear transformation. $\mathcal{D}$ has a following properties:

(a) $\|\mathcal{D}\mathbf{X}\| = \|\tilde{\mathbf{\Sigma}}\|$ and since $\|\mathbf{X}\| = \|\mathbf{\Sigma}\|$ then $\|\mathcal{D}\mathbf{X}\| \leq \|\mathbf{X}\|$. i.e. $\mathcal{D}$ is bounded;

(b) $\langle \mathcal{D}\mathbf{X}, \mathbf{X} \rangle = \text{trace}(\mathbf{V}\tilde{\mathbf{\Sigma}}^*\mathbf{\Sigma}\mathbf{V}^*) = \text{trace}(\tilde{\mathbf{\Sigma}}^*\tilde{\mathbf{\Sigma}}) = \|\mathcal{D}\mathbf{X}\|^2 = \|\mathcal{D}\mathbf{X}\| \cdot \|\mathbf{X}\| \text{cos}\alpha$ where α is the angle between $\mathcal{D}\mathbf{X}$ and $\mathbf{X}$ and $\text{cos}\alpha = \frac{\sqrt{\sum_{i=1}^{k} \sigma_i^2}}{\sqrt{\sum_{i=1}^{r} \sigma_i^2}}$. $\mathcal{D}$ can be thought of as an operator that rotates $\mathbf{X}$ by α and shrinks it by $|\text{cos}\alpha|$.

3. Entries correction operator $\mathcal{W}$: Let $\mathbf{X}$ and $\mathbf{M}$ be $m \times n$ matrices. The operator $\mathcal{W}$ is defined as $\mathcal{W}\mathbf{X} \triangleq (\mathcal{I} - \mathcal{P})\mathbf{X} + \mathcal{P}\mathbf{M}$ where $\mathcal{I}$ is the identity operator ($\mathcal{I}\mathbf{X} = \mathbf{X}$). The matrix $\mathbf{M}$ can be considered as a parameter of the operator which replaces the entries in $\mathbf{X}$ by entries from $\mathbf{M}$ as indicated by the operator $\mathcal{P}$. $\mathcal{W}$ has the following properties:

(a) $\mathcal{P}\mathcal{W}\mathbf{X} = \mathcal{P}\mathbf{M}$;

(b) $(\mathcal{I} - \mathcal{P})\mathcal{W}\mathbf{X} = (\mathcal{I} - \mathcal{P})\mathbf{X}$;

(c) $\mathbf{X} - \mathcal{W}\mathbf{X} = \mathcal{P}\mathbf{X} - \mathcal{P}\mathbf{M}$.

For convenience, we define another operator $\mathcal{T}$ by $\mathcal{T}\mathbf{X} \triangleq \mathcal{D}\mathcal{W}\mathbf{X}$.

3 The Interest Zone Matrix Approximation (IZMA) algorithm

We are interested in an algorithm that finds the matrix $\mathbf{X}$ while minimizing the error function $\epsilon(\mathbf{X}) \triangleq \|\mathcal{P}\mathbf{X} - \mathcal{P}\mathbf{M}\|_F$, where $\text{rank}(\mathbf{X}) = k$, $\text{rank}(\mathbf{M}) = r$ and $k < r$. This algorithm minimizes the squared distance between the entries we wish to approximate. Using the definition in section 2.2, we suggest to solve the iterative algorithm $\mathbf{X}_{n+1} = \mathcal{T}\mathbf{X}_n$ with the initial matrix $\mathbf{X}_0$. A sufficient condition for its convergence is the fact that the sequence $\epsilon(\mathcal{T}^n\mathbf{X})$ is non-increasing and bounded from below ($\epsilon(\mathbf{X}) \geq 0$ by definition).

Theorem 3.1. $\epsilon(\mathcal{T}^{n+1}\mathbf{X}) \leq \epsilon(\mathcal{T}^{n}\mathbf{X})$ *for every* $\mathbf{X}$ *and* $n \geq 1$.

Proof. Let $\mathcal{H}$ be a Hilbert space of all $m \times l$ matrices equipped with the standard inner product $\langle \mathbf{X}, \mathbf{Y} \rangle = \text{trace}(\mathbf{X}^*\mathbf{Y})$, which induces the standard Frobenius norm $\|\mathbf{X}\|^2 = \text{trace}(\mathbf{X}^*\mathbf{X})$. Assume $\mathbf{X}$ is an arbitrary matrix in $\mathcal{H}$ and let $\mathbf{M}$ be the matrix whose entries we wish to approximate according to the projection operator $\mathcal{P}$. Since $n \geq 1$ then $\text{rank}(\mathcal{T}^n\mathbf{X}) = k$. Let $\mathbf{Q}$ be the locus of all matrices $\mathbf{Y}$ that satisfy $\mathcal{P}\mathbf{Y} = \mathcal{P}\mathbf{M}$. $\mathbf{Q}$ can be thought as a line parallel to the $\mathcal{I} - \mathcal{P}$ axis and perpendicular to the $\mathcal{P}$ axis - see Fig. 3.1. Note that the error $\epsilon(\mathbf{X})$ is the distance between the matrix point $\mathbf{X}$ and the line $\mathbf{Q}$. Applying $\mathcal{W}$ to $\mathcal{T}^n\mathbf{X}$, denoted by $\mathcal{W}\mathcal{T}^n\mathbf{X}$, which is a zero error matrix and $\mathcal{W}\mathcal{T}^n\mathbf{X}$ on $\mathbf{Q}$ does not necessarily has rank k. Applying $\mathcal{D}$ to $\mathcal{W}\mathcal{T}^n\mathbf{X}$ produces $\mathcal{T}^{n+1}\mathbf{X}$ which is the best rank k approximation for $\mathcal{W}\mathcal{T}^n\mathbf{X}$. Therefore, it must be inside a ball that is centered in $\mathcal{W}\mathcal{T}^n\mathbf{X}$ with radius $\|\mathcal{T}^n\mathbf{X} - \mathcal{W}\mathcal{T}^n\mathbf{X}\|$ so that $\|\mathcal{T}^{n+1}\mathbf{X} - \mathcal{W}\mathcal{T}^n\mathbf{X}\| \leq \|\mathcal{T}^n\mathbf{X} - \mathcal{W}\mathcal{T}^n\mathbf{X}\|$ (otherwise $\mathcal{T}^n\mathbf{X}$ is a better approximation for $\mathcal{W}\mathcal{T}^n\mathbf{X}$ which contradicts the Eckart-Young Theorem) - see Fig. 3.1. Thus, we obtain:

$$\begin{aligned} \|\mathcal{T}^{n+1}\mathbf{X} - \mathcal{W}\mathcal{T}^n\mathbf{X}\|^2 &= \|(\mathcal{I} - \mathcal{P})\mathcal{T}^{n+1}\mathbf{X} - (\mathcal{I} - \mathcal{P})\mathcal{W}\mathcal{T}^n\mathbf{X}\|^2 + \\ \|\mathcal{P}\mathcal{T}^{n+1}\mathbf{X} - \mathcal{P}\mathcal{W}\mathcal{T}^n\mathbf{X}\|^2 &\leq \|\mathcal{T}^n\mathbf{X} - \mathcal{W}\mathcal{T}^n\mathbf{X}\|^2 \\ &= \|\mathcal{P}\mathcal{T}^n\mathbf{X} - \mathcal{P}\mathbf{M}\|^2 \end{aligned} \tag{3.1}$$

where in Eq. 3.1 we used the third property of $\mathcal{W}$ and since (according to the first property of $\mathcal{W}$) $\|\mathcal{P}\mathcal{T}^{n+1}\mathbf{X} - \mathcal{P}\mathcal{W}\mathcal{T}^n\mathbf{X}\| = \|\mathcal{P}\mathcal{T}^{n+1}\mathbf{X} - \mathcal{P}\mathbf{M}\|$ we finally obtain: $\|\mathcal{P}\mathcal{T}^{n+1}\mathbf{X} - \mathcal{P}\mathbf{M}\| \leq \|\mathcal{P}\mathcal{T}^n\mathbf{X} - \mathcal{P}\mathbf{M}\|$.

Equality is obtained if and only if $(\mathcal{I} - \mathcal{P})\mathcal{T}^{n+1}\mathbf{X} = (\mathcal{I} - \mathcal{P})\mathcal{W}\mathcal{T}^n\mathbf{X} = (\mathcal{I} - \mathcal{P})\mathcal{T}^n\mathbf{X}$. □

Geometrically, the algorithm means that in each iteration, our current matrix is projected onto $\mathbf{Q}$. Then, it is approximated to a rank k matrix by $\mathcal{D}$. The new rank k matrix must be inside a ball centered at the current point in $\mathbf{Q}$ and its radius is the distance to the previous rank k matrix iteration. The new point is projected again onto $\mathbf{Q}$. It continues this way such that the radius of each ball is becoming smaller after each iteration. This is illustrated in Fig. 3.1. This means that the algorithm eventually converges. The convergence speed depends on the convergence value $\kappa_n = \|(\mathcal{I} - \mathcal{P})\mathcal{T}^{n+1}\mathbf{X} - (\mathcal{I} - \mathcal{P})\mathcal{T}^n\mathbf{X}\|$. If this value becomes smaller then the algorithm will converge slowly. When $\kappa = 0$, it means that the algorithm reached

a convergence point. One can suggest different methods for measuring the convergence rate, that can come from geometry. For example, a good relative measure is $\frac{\text{dist}(\mathcal{P}\mathbf{X_{k-1}},\mathbf{Q})}{\text{dist}(\mathcal{P}\mathbf{X_k},\mathbf{Q})}$

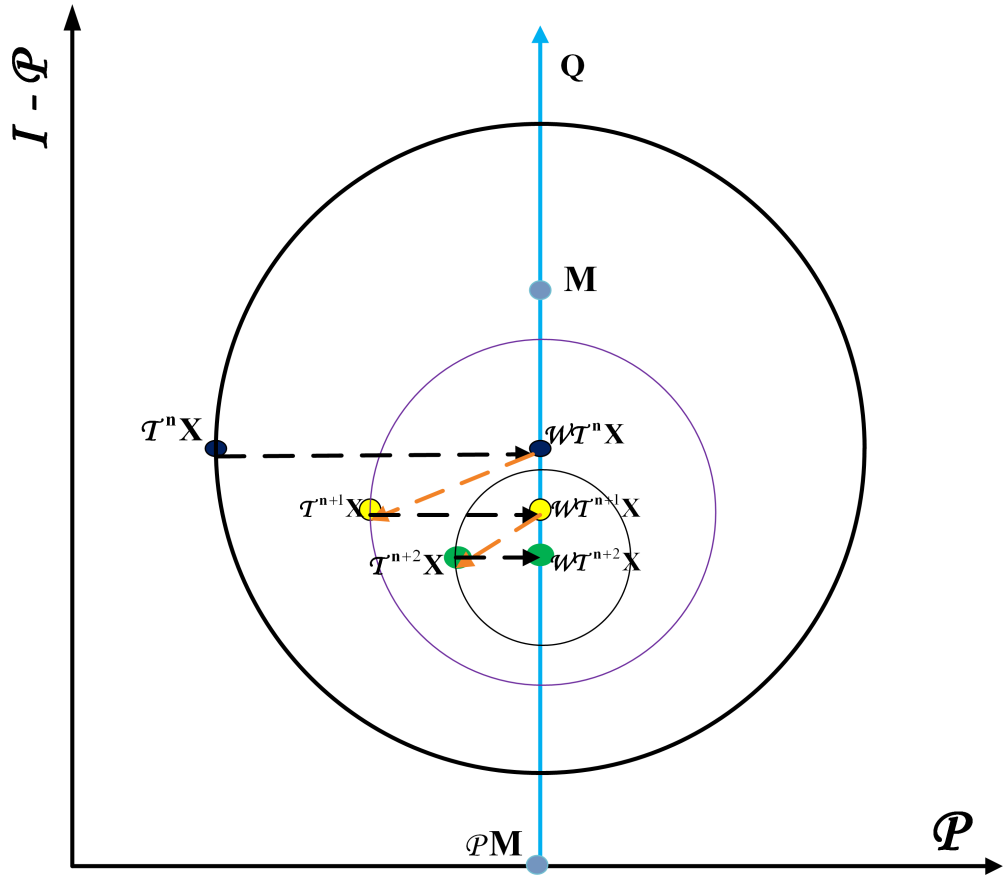


Figure 3.1: Geometric illustration how the radius of each ball in the proof of theorem 3.1 is getting smaller and smaller

Algorithm 3.1 *Interest Zone Matrix Approximation* (IZMA)

Inputs:

$\mathbf{M}$ - matrix to approximate, $\mathbf{B}$ - binary matrix that specifies important entries

$\mathbf{X}_0$ - initial guess, k - desired rank.

Output:

$\mathbf{X}$ - Approximated matrix with rank k for entries in $\mathbf{M}$ specified by $\mathbf{B}$

1 Set $\mathbf{X} \leftarrow \mathcal{D}\mathbf{X}_0$ (set $\mathbf{X}$ to have the best rank k approximation of $\mathbf{X}_0$ using SVD)

2 repeat

2.1 $\mathbf{X} \leftarrow \mathcal{W}\mathbf{X}$ (Replace the entries we want to approximate in $\mathbf{X}$ by the known entries from $\mathbf{M}$ according to $\mathbf{B}$)

2.2 $\mathbf{X} \leftarrow \mathcal{D}\mathbf{X}$ (set $\mathbf{X}$ to have the best rank k approximation of $\mathbf{X}$ using SVD)

until $\|\mathcal{P}\mathbf{X} - \mathcal{P}\mathbf{M}\|$ converges

3 Return $\mathbf{X}$

Both the convergence speed and the final matrix the algorithm converges to depend on the initial matrix $\mathbf{X}_0$. If the large singular values of $\mathbf{X}_0$ mainly approximate the values of $(\mathcal{I} - \mathcal{P})\mathbf{X}_0$, then the application of $\mathcal{D}$ will not change $(\mathcal{I} - \mathcal{P})\mathbf{X}_0$ significantly but will change $\mathcal{P}\mathbf{X}_0$. To avoid it, the values of $(\mathcal{I} - \mathcal{P})\mathbf{X}_0$ should be at the same order of magnitude such as $\mathcal{P}\mathbf{M}$. Application of $\mathcal{W}$ will bring it back very close to the previous iteration. Thus, the algorithm will iterate near two points that are changed very slowly if at all. To avoid having the algorithm converges to a local minimum, we suggest the following strategy: Suppose we have a mapping $\mathcal{U} : \mathcal{H} \rightarrow \mathcal{H}$ that maps a point in $\mathcal{H}$ to its low rank approximation with respect to $\mathcal{P}$, i.e. $\mathcal{U}(\mathbf{X})$ is the closest low rank approximation to $\mathbf{X}$ such that $\epsilon(\mathcal{U}(\mathbf{X}))$ is minimal. Applying $\mathcal{U}$ to the line $\mathbf{Q}$ will map all its points to be on a hollow cylinder of radius r, where r indicates the minimal possible error. If we start from several random distant initial points and obtain different final points with the same error, this may serve as an indication that we reached the global minimum. As an example, we present the following numerical example which shows that the algorithm does not always converge to the global minimum but rather depends on the starting point. Suppose we wish to approximate by a rank 2 matrix the following full rank 3×3 matrix:

$$\mathbf{M} = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0.75 & 0.25 \\ 0 & 0.25 & 0.75 \end{bmatrix}$$

where the interest points are indicated by

$$\mathbf{B} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 1 & 1 \end{bmatrix}.$$

If we take as an initial guess $\mathbf{X}_0 = \mathbf{M}$ then, after the first iteration we obtain the matrix

$$\mathbf{X}_1 = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0.5 & 0.5 \\ 0 & 0.5 & 0.5 \end{bmatrix}$$

which is a rank 2 matrix that mapped by $\mathcal{T}$ to itself, i.e. $\kappa = 0$ and $\mathbf{X}_i = \mathbf{X}_1$ for $i \geq 1$. In the rank reduction part of the algorithm, the operator we named $\mathcal{D}$ reduces the rank of $\mathcal{W}\mathbf{X}_1$ but the values of $(\mathcal{I} - \mathcal{P})\mathcal{W}\mathbf{X}_1$ remain unchanged. For example, if we start from a random matrix

$$\mathbf{X}_0 = \begin{bmatrix} 0.553 & 0.133 & -1.58 \\ -0.204 & 1.59 & -0.0787 \\ -2.05 & 1.02 & -0.682 \end{bmatrix}$$

then eventually we will get the matrix

$$\mathbf{X}_{100} = \begin{bmatrix} 0.854 & 0.685 & -1.25 \\ -1.32 & 0.75 & 0.25 \\ -1.37 & 0.25 & 0.75 \end{bmatrix}$$

which has the error $\epsilon(\mathbf{X}_{100}) = \|\mathcal{P}\mathbf{X}_{100} - \mathcal{P}\mathbf{M}\| = 0$.

3.1 Numerical example that illustrates step-by-step the convergence

Suppose we are given the following matrix:

$$\mathbf{M} = \begin{bmatrix} 1 & 2 \\ 3 & x \end{bmatrix}.$$

We want to complete the matrix such that the new matrix will have rank 1. Since completion is not the primary goal of the algorithm, it might happen that the known entries will change. In this case, $M_{2,2}$ is unknown (or we do not want to approximate it). The projection operator $\mathcal{P}$ will zero $M_{2,2}$ to become

$$\mathcal{P}\mathbf{M} = \begin{bmatrix} M_{1,1} & M_{1,2} \\ M_{2,1} & 0 \end{bmatrix} = \begin{bmatrix} 1 & 2 \\ 3 & 0 \end{bmatrix}.$$

Obviously, the solution we are looking for is

$$\tilde{\mathbf{M}} = \begin{bmatrix} 1 & 2 \\ 3 & 6 \end{bmatrix}$$

which is of rank 1 and $\|\mathcal{P}\tilde{\mathbf{M}} - \mathcal{P}\mathbf{M}\| = 0$. Suppose we start with the following initial guess:

$$\mathbf{X}_0 = \begin{bmatrix} 1 & 2 \\ 3 & 500 \end{bmatrix} \Rightarrow \mathcal{T}\mathbf{X}_0 = \begin{bmatrix} 0.0121 & 2.01 \\ 3 & 500 \end{bmatrix} \Rightarrow \mathcal{W}\mathcal{T}\mathbf{X}_0 = \begin{bmatrix} 1 & 2 \\ 3 & 500 \end{bmatrix} \Rightarrow \mathcal{T}^2\mathbf{X}_0 = \begin{bmatrix} 0.0121 & 2.01 \\ 3 & 500 \end{bmatrix}.$$

We can see that $\kappa_1 = \|(\mathcal{I} - \mathcal{P})\mathcal{T}\mathbf{X} - (\mathcal{I} - \mathcal{P})\mathcal{T}^2\mathbf{X}\| \approx 2.3e - 5$. This indicates that the algorithm will converge slowly and it will iterate between two points that are very close to $\mathcal{T}\mathbf{X}$ and $\mathcal{W}\mathcal{T}\mathbf{X}$ and it drifts slowly away from them such that κ_n will change slowly. A rank 1 matrix is obtained after 50,000 iterations to be:

$$\mathbf{X}_{50,000} = \begin{bmatrix} 0.0121 & 2.0059 \\ 3.004 & 498.806 \end{bmatrix}.$$

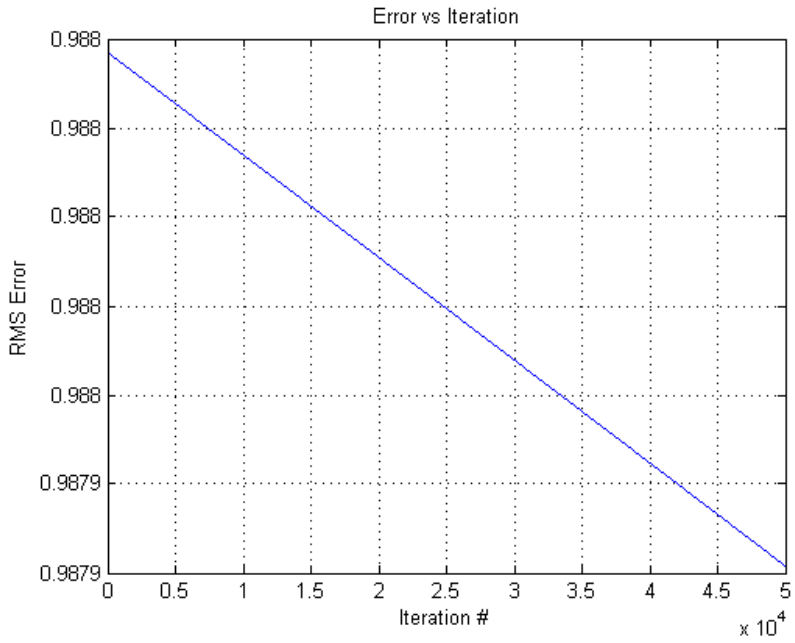


Figure 3.2: l_2-error vs iteration number

From Fig. 3.2, we see that the residual error is relatively big and the convergence is slow. The same experiment is repeated with another initial matrix $\mathbf{X}_0$:

$$\mathbf{X}_0 = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \Rightarrow \mathcal{T}\mathbf{X}_0 = \begin{bmatrix} 1.27 & 1.81 \\ 2.88 & 4.09 \end{bmatrix} \Rightarrow \mathcal{W}\mathcal{T}\mathbf{X}_0 = \begin{bmatrix} 1 & 2 \\ 3 & 4.09 \end{bmatrix} \Rightarrow \mathcal{T}^2\mathbf{X}_0 = \begin{bmatrix} 1.26 & 1.82 \\ 2.89 & 4.16 \end{bmatrix}. \tag{3.2}$$

We can see that $\mathbf{X}_{0_{2,2}}$ is changed between the iterations and $\kappa_1 \approx 0.07$ is better than the previous case where $\mathbf{X}_{0_{2,2}} = 500$. After 500 iterations, the error is $6.2e-8$ (in this case the solution is unique). Moreover, the problem can also be thought of as looking for a rank 1 matrix approximation to a rank 2 matrix with the requirement to approximate only 3 entries instead of the whole matrix. Rank 1 matrix is obtained and it accurately approximated the matrix while the application of the Eckart-Young theorem to the entire matrix produced worse results in approximating the interesting points. Figure 3.3 shows the reduction of the error by the application of the IZMA algorithm.

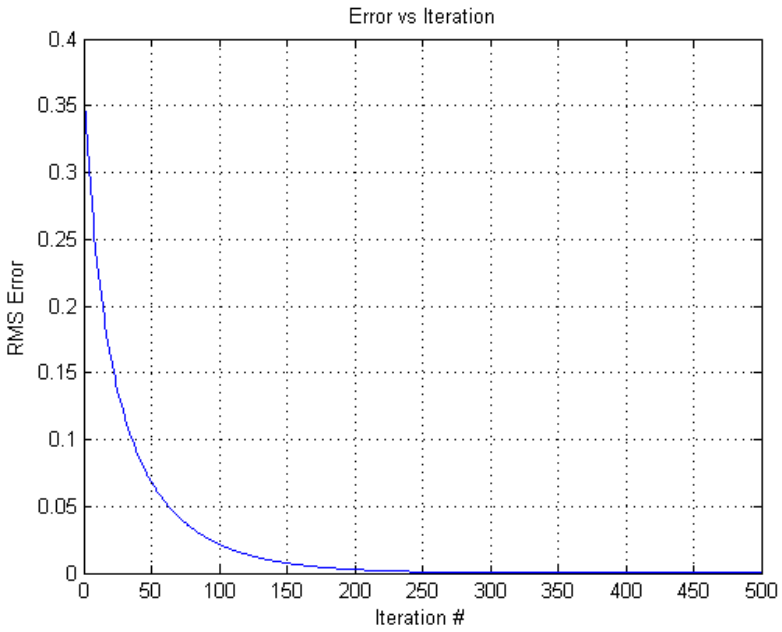


Figure 3.3: l_2-error vs iteration number starting from a "good" guess in Eq. 3.2

4 Applications

4.1 Image interpolation and approximation

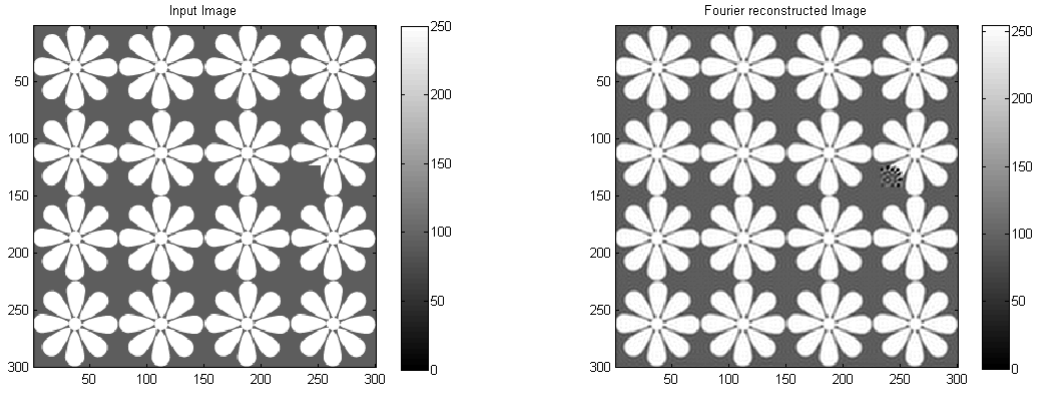
Interpolation reconstructs a discrete function $I[m, n]$ (or a continuous function $I(x, y)$) from a discrete set Ω. Most interpolation methods try to restore the function by assuming it can be spanned by a set of basis functions (called "kernel"). Typical basis functions are splines, polynomials, trigonometric functions, wavelets, radial functions, etc. For example, in order to approximate $I(\mathbf{x}) = I(x, y)$ with a Gaussian radial basis function such as $\phi(r) = \exp(-\beta r^2)$ for some $\beta > 0$, then the approximating function can be written as $Y(\mathbf{x}) = \sum_{i=1}^{N} a_i \phi(\|(\mathbf{x} - c_i)\|_2)$ where $\{c_i\}_{i=1}^{N}$ are the centers in which we lay the radial functions on. $\{a_i\}_{i=1}^{N}$ are the coefficients of the functions, which can be found by solving $\mathbf{a}^* = \text{argmin}\|Y(\mathbf{x}) - I(\mathbf{x})\|_2, \quad \mathbf{x} \in \Omega$. This solves the standard least squares problem on the discrete set Ω.

As was stated above, the same procedure can be repeated for different kernels by minimizing a different metric such as l_1, l_2 or l_∞. It is important to mention that different kernels produce different results. A-priori knowledge about the physical nature of the function we

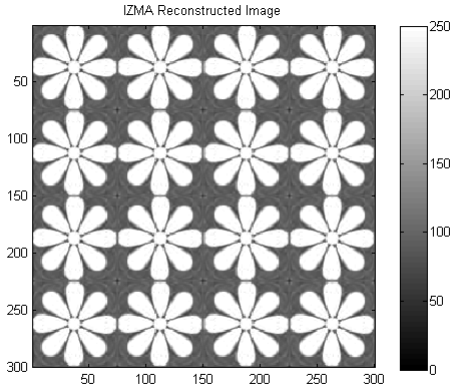
wish to interpolate can be an important input for choosing the interpolation kernel. For example, audio signals are usually spanned well (i.e. they require a small number of coefficients) using trigonometric functions, where other signals, such as Chirp or Linear FM used in radar systems [16] are better adjusted to wavelets or Gabor functions. However, since SVD has the best energy compaction property from all separable functions, it can be used to find on the fly the appropriate basis functions.

Our approach, which is based on SVD, does not require any a-priori knowledge for the interpolation procedure. It finds it from the available data. A disadvantage of this method is that it is not suitable for sparse data reconstruction. When the data is too sparse, there is not enough information for extracting the most suitable basis functions.

The example in Fig. 4.1 compares between the approximations of missing data through the application of the IZMA algorithm and a standard approximation method that uses the GP interpolation method with Fourier basis functions.



(a) (b)



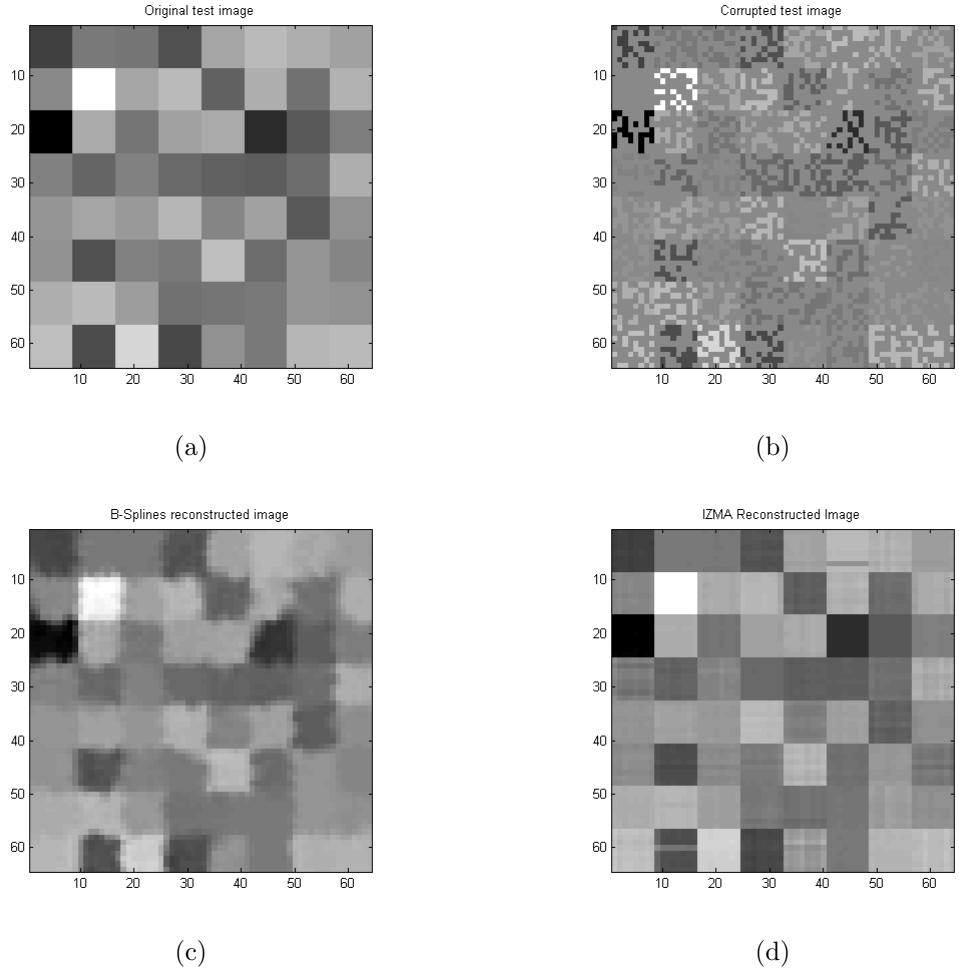
(c)

Figure 4.1: (a) The original input image for the interpolation process. (b) Approximation by GP. (c) Approximation by the IZMA algorithm

We see from Fig. 4.1 that the IZMA algorithm completed the flower image (of size 300×300 pixels) correctly since the basis functions, which were used, are flowers components. The Fourier basis functions, on the other hand, failed to reconstruct the flower. The Fourier l_2 error (MSE) is 0.066 (normalized by the number of gray-levels) while the IZMA l_2 error (MSE) is 0.05. Also, from the rank perspective, the Fourier based reconstructed image rank is 131 and the IZMA based reconstructed algorithm produced a rank 15 matrix. The original image rank was 223.

Another example is illustrated in Fig. 4.2 where the test image was produced from a

combination of Haar-wavelet basis functions. 60% of the data was missing. It was restored by the IZMA algorithm and the multilevel B-Splines ([15]). Image size is 64×64.



(a) (b)

(c) (d)

Figure 4.2: (a) The original input image. (b) The image where 60% of the entries are 0. (c) The reconstruction by the application of B-Splines. (d) The reconstruction by the application of the IZMA algorithm

The RMS error (normalized by the number of gray-levels) after 100 iterations using the IZMA algorithm was 0.016 compared with 0.036 by the multilevel B-splines algorithm. It indicates that the IZMA algorithm found the suitable basis functions and thus achieved a smaller error with better visual effect.

4.2 Reconstruction of Physical Signals

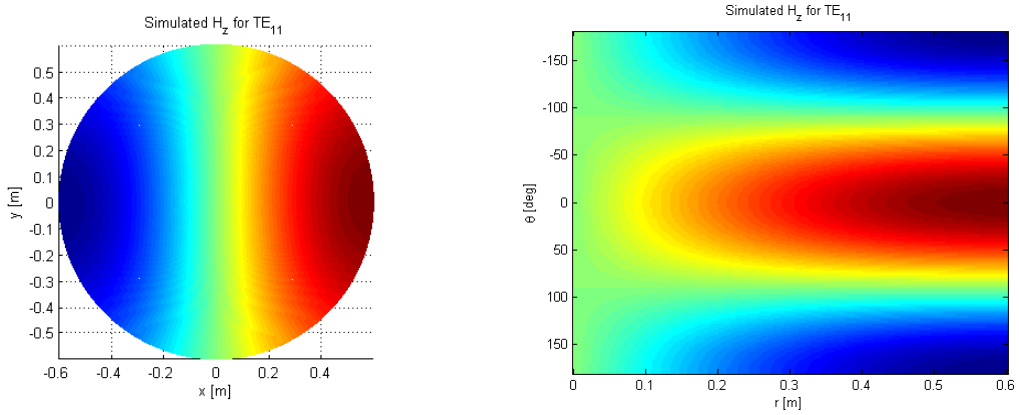
A typical family of matrices that have low rank can be originated from PDEs that are solved by separation of variables. In this case, the solution is given as a sum such as $U(x,y) = \sum_{n=1}^{N} X_n(x)Y_n(y)$. Note that when the solution is stored as a matrix, then the element $X_n(x)Y_n(y)$ is discretized and stored as $\mathbf{XY^T}$ where $\mathbf{X}$ and $\mathbf{Y}$ are column vectors and $\mathbf{XY^T}$ is a matrix of rank 1. After summation, the obtained rank is N (as the functions of the solution are linear independent). As an example, we examine the propagation of an electromagnetic wave inside a cylindrical waveguide of radius R. The electromagnetic waves travelling inside the waveguide are called *modes* and they depend on the input frequency and the geometry of the waveguide. Usually waveguides are designed to support only one mode. We assume that this is the case. The primary mode and the most important one for cylindrical waveguide is the first Transverse Electric mode denoted as TE_{11}. TE modes do not have electric field in the z direction but only the magnetic field H_z that is called the "generating field". The rest of the fields can be derived from it. For more information see [19]. H_z is found by solving the Hemholtz equation

$$\nabla^2 H_z + k^2 H_z = 0 \quad H_z(R,\theta,z) = 0 \tag{4.1}$$

where ∇^2 is the Laplacian operator in cylindrical coordinates (r,θ,z), $k = \frac{2\pi}{\lambda}$ is the wavenumber and λ is the wavelength. The solution of Eq. 4.1 is known and for TE_{11} it is given by:

$$H_z(r,\theta,z) = (A\sin\theta + B\cos\theta)J_1(k_c r)e^{-i\beta z} \tag{4.2}$$

where $J(x)$ is the Bessel function of the first kind, k_c is the cut-off wavenumber which for TE_{11} is the first zero of $J_1'(x)$ divided by R (in our case $k_c = \frac{1.84}{R}$) and $\beta^2 = k^2 - k_c^2$. For a mode to exist in the waveguide, its cut-off wavenumber k_c must be smaller than k. Hence, λ can be chosen such that only the first mode will excite in the waveguide. The z-axis has only phase accumulation along the waveguide and this is not very interesting. We will investigate the modes as a function of (r,θ).



(a) (b)

Figure 4.3: Simulated magnetic field $H_z = \cos(\theta)J_1(k_c r)$, $R = 0.6$m in different coordinate systems. (a) H_z of TE_{11} mode in Cartesian coordinates. (b) H_z of TE_{11} mode in Polar coordinates. Both images are 200×200 pixels.

Assume that the image in Fig. 4.3 is corrupted such that 60% of the data is missing as shown in Fig. 4.4 and it has to be restored . Note that neither information on the geometry of the waveguide nor the wavelength is needed. The only parameter is *the number* of modes, which as we saw earlier, is equal to the rank of the matrix.

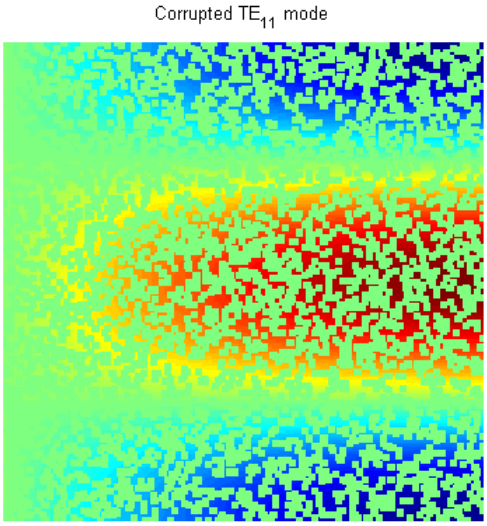
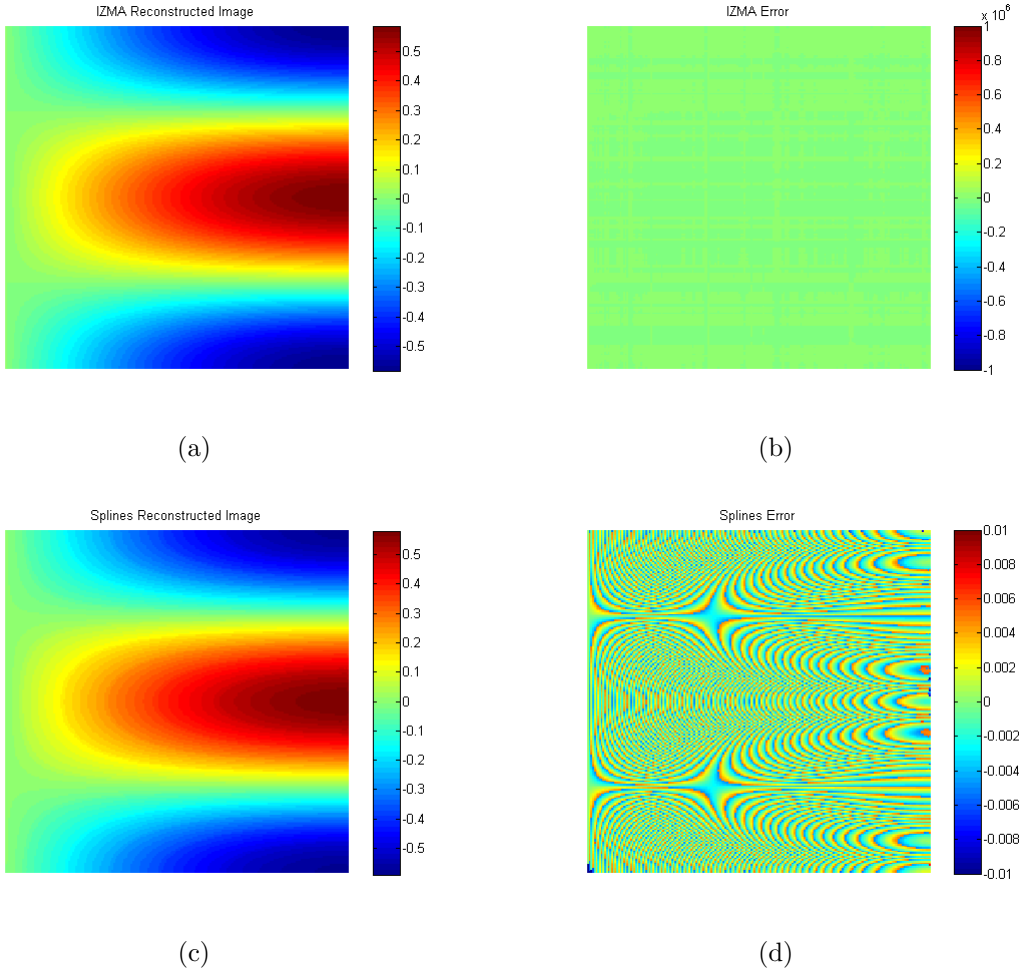


Figure 4.4: Corrupted TE_{11} mode in a circular waveguide

The results from 55 iterations of the IZMA algorithm is compared with the results from the application of the multilevel B-Splines.



(a) (b)

(c) (d)

Figure 4.5: (a) The reconstruction from the application of the IZMA algorithm. (b) Error from the reconstructed image after the application of the IZMA algorithm. (c) The reconstructed image after the application of the multilevel B-splines. (d) Error from the reconstructed image after the application of the multilevel B-Splines.

4.3 Masked SVD

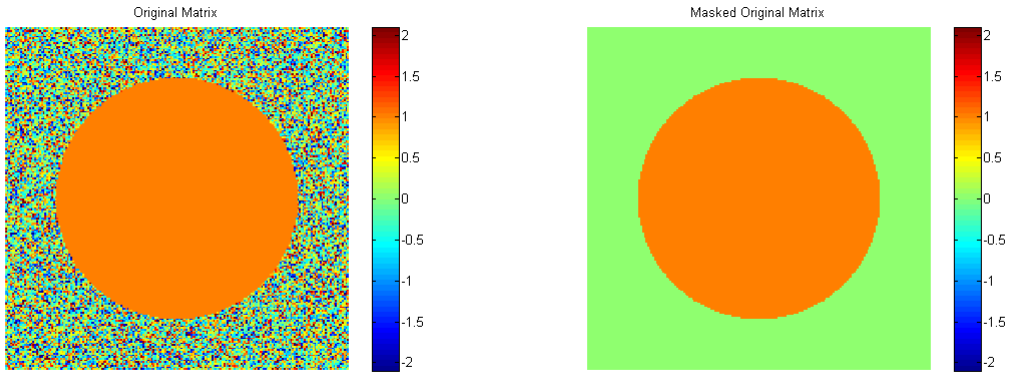
Another interesting use of the IZMA algorithm is to calculate the SVD only on a certain region of a matrix. For example, a matrix can be full rank but contains a circular region

which can be considered as "rank 1". The interest zone (or the shape) is defined by the operator $\mathcal{P}$. For example, suppose $\mathbf{M}$ is an $m \times n$ matrix of rank m but there may exist a matrix $\mathbf{X} = \mathbf{U\Sigma V^T}$ of rank $k < m$ such that

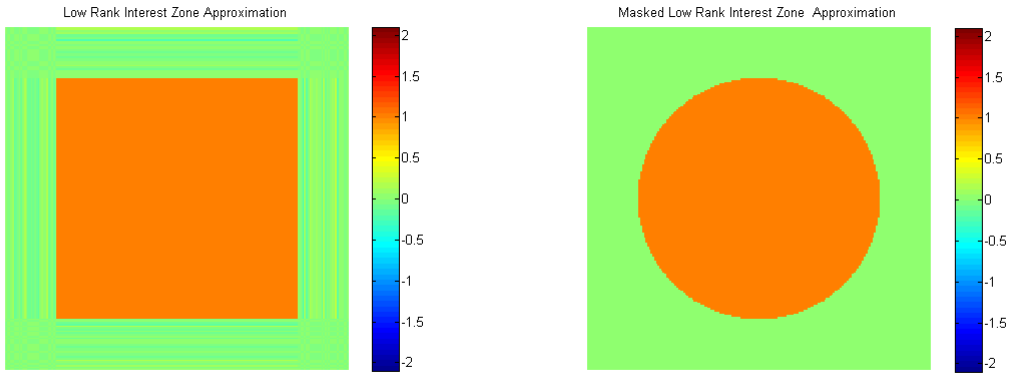
$$\mathcal{P}\mathbf{M} = \mathcal{P}\mathbf{X} = \mathcal{P}(\mathbf{U\Sigma V^T}). \tag{4.3}$$

Equation 4.3 can be thought as a way to determine the rank of a sub-region of a matrix and its SVD is calculated when only a certain region is taken into consideration. Note that not always there exists a matrix $\mathbf{X}$ that satisfies Eq. 4.3 with a lower rank.

Figure 4.6 shows a 200×200 matrix $\mathbf{M}$ of rank 200 created by Gaussian noise with zero mean and standard deviation of 1, where the center was replaced by a circle of values one as shown in Fig. 4.6a. Figure 4.6c shows a rank 1 matrix $\mathbf{M}$ that approximates the matrix perfectly within the circle so that $\mathcal{P}\mathbf{M} = \mathcal{P}\mathbf{X}$.



(a) (b)



(c) (d)

Figure 4.6: (a) Original matrix $\mathbf{M}$, rank($\mathbf{M}$) = 200. (b) The projected matrix (zeros outside the circle) $\mathcal{P}\mathbf{M}$. (c) Interest zone approximated rank 1 matrix $\mathbf{X}$. (d) $\mathcal{P}\mathbf{X}$ matrix.

4.4 Approximation of Random Matrices

In this example, we approximate a 1000×1000 random matrix whose entries are Gaussian white noise of zero mean and standard deviation 1. The initial rank of the matrix is 1000. By removing randomly $\approx 2\%$ $(20,017)$ entries, it is possible to achieve a rank 930 matrix with RMS error of less than 0.002 for the other 98% of the entries after 200 iterations. Figure 4.7 shows the results for the upper-left 40×40 block of the white noise matrix as well as the error on the interest points vs number of iterations. The figures show that small errors appear on the locations we wished to approximate.

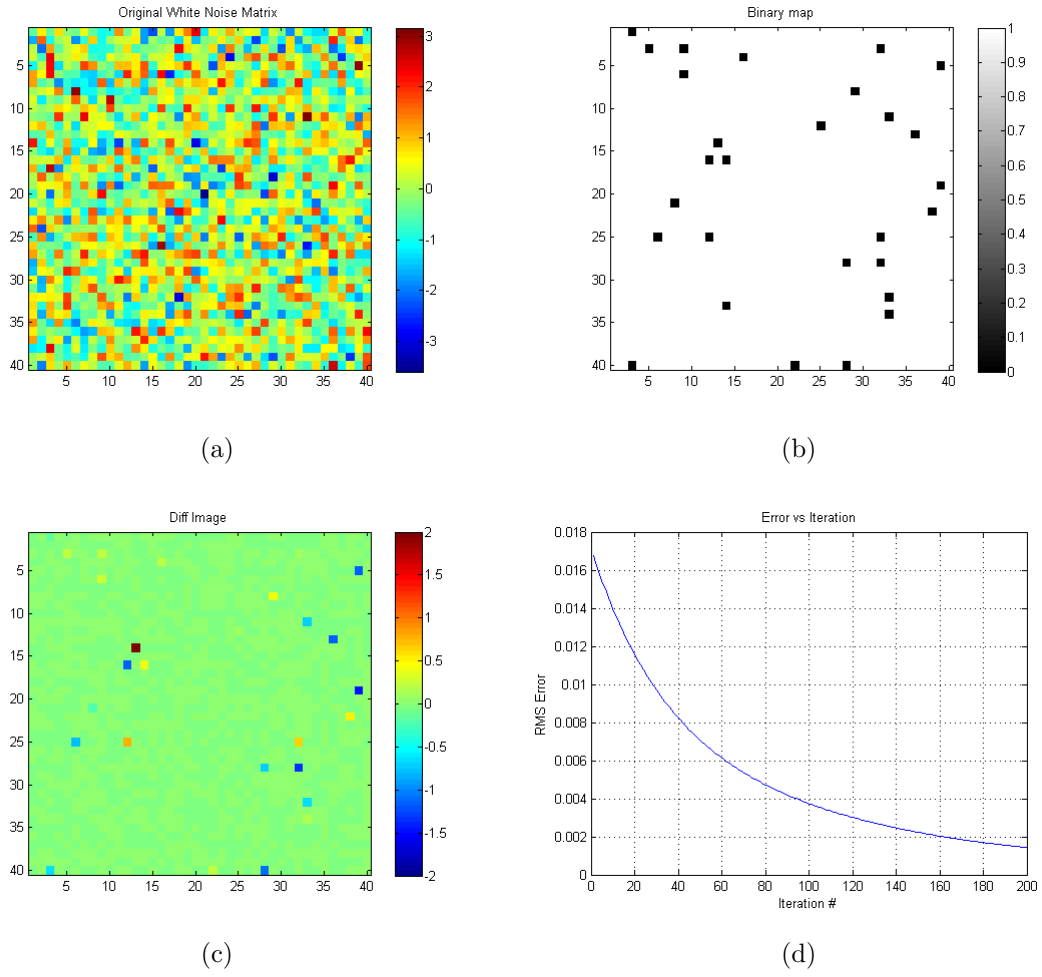


Figure 4.7: (a) The original white noise matrix (upper-left 40×40 block). (b) The matrix where some of its entries were removed (black indicates entries to be ignored). (c) The difference between the original matrix and the approximated matrix. (d) RMS error vs the number of iterations.

5 Conclusion

In this paper, the IZMA algorithm was introduced for low rank approximation of certain entries in a matrix. The convergence of the algorithm is proved. It also explains how the convergence speed is affected by the given data and by the choice of the initial starting point. In addition, the performance of the proposed IZMA algorithm on images and random data

were demonstrated. This is a new interpolation approach that uses adaptive basis functions and it is suitable for a variety of images and data originated from physical problems.

ACKNOWLEDGMENT: This research was partially supported by the Israel Science Foundation (Grant No. 1041/10).

References

[1] T.A. Louis, *Finding the observed information matrix when using the EM algorithm*, Journal of the Royal Statistical Society, Series B. (Methodological), Vol. 44, No. 2, pp. 226-233, 1982.

[2] T. Hastie, R. Tibshirani, G. Sherlock, M. Eisen, P. Brown, and D. Botstein, *Imputing missing data for gene expression arrays*. Technical report; Division of Biostatistics, Stanford University, 1999.

[3] N. Srebro and T. Jaakkola, *Weighted low-rank approximations*, Preceeding of the 20th International Conference on Machine Learning (ICML-2003), Washington DC, 2003.

[4] J. Mairal, M. Elad, G. Sapiro, *Sparse representation for color image restoration*, IEEE Transactions on Image Processing, Vol. 17, No. 1, pp.53-69, 2008.

[5] M. Fazel, *Matrix Rank Minimization with Applications. PhD thesis*, Stanford University, 2002.

[6] J.F. Cai, E.J. Candes and Z. Shen, *Singular Value Thresholding Algorithm for Matrix Completion*, SIAM Journal on Optimization, Vol. 20, No. 4, pp. 1956-1982, 2010.

[7] Feng Nan, *Low Rank Matrix Completion*, Master thesis, Massachusetts Institute of Technology, 2009.

[8] G. Eckart and G. Young, *The approximation of one matrix by another of lower rank*, Psychometrika, Vol. 1, No. 3, pp.211-218, 1936.

[9] G.H. Golub, A. Hoffman, G.W. Stewart, *A generalization of the Eckart-Young-Mirsky matrix approximation theorem*, Linear Algebra and Its Applications, Vol. 88-89, pp. 317-327, Elsevier, 1987.

[10] O. Troyanskaya, M. Cantor, G. Sherlock, P. Brown, T. Hastie, R. Tibshirani, D. Botstein, and R. B. Altman, *Missing value estimation methods for DNA microarrays. Bioinformatics*, Vol. 17, No.6, pp520525, 2001.

[11] R. Mazumder, T. Hastie and R. Tibshirani, *Spectral Regularization Algorithms for Learning Large Incomplete Matrices*, Journal of Machine Learning Research 11 (2010) 2287-2322.

[12] A. Papoulis, *A new algorithm in spectral analysis and band-limited interpolation*, IEEE Transactions on Circuits and Systems, Vol. 22 No. 9, pp. 735-742, 1975.

[13] H. Andrews, C. Patterson, *Singular Value Decomposition Image Coding*, IEEE Transactions on communications, Vol. 24, No. 4, pp. 425-432, 1976.

[14] A.P. Dempster, N.M. Laird, D.B. Rubin *Maximum likelihood from incomplete data via the EM algorithm*, Journal of the Royal Statistical, Series B. (Methodological) Vol. 39, No. 1, pp. 1-38, 1977.

[15] S. Lee, G. Wolberg, S.Y. Shin, *Scattered Data Interpolation with Multilevel B-Splines*, IEEE Transactions on Visualization and Computer Graphics, Vol. 3, No. 3, pp. 228-244, 1997.

[16] N. Levanon, *Radar Principles*, Wiley-Interscience, 1988.

[17] P. Waldemar, T.A. Ramstad, *Hybrid KLT-SVD image compression*, IEEE International Conference on Acoustics, Speech, and Signal Processing, Vol. 4, pp. 2713 - 2716, 1997.

[18] M. Tian, S.W. Luo, L.Z. Liao, *An investigation into using singular value decomposition as a method of image compression*, IEEE Proceedings of International Conference on Machine Learning and Cybernetics, 2005.

[19] D.M. Pozar, *Microwave engineering* 3rd Edition, Wiley, 2004.