APPLICATIONS OF THE SINGULAR VALUE DECOMPOSITION

PROJECT 2

MA580 FALL98

PROFESSOR M. CHU

BY

JORDAN E. MASSAD

DATE: OCTOBER 30, 1998

ABSTRACT

This project examines five applications of the singular value decomposition (SVD) of matrices. Implementation of the SVD in constructing the generalized matrix inverse, in determining numerical rank, and in fitting curves by least-squares is described. Finally, the application of the SVD in image compression and in voice recognition is demonstrated.

Table of Contents

1. INTRODUCTION 1

2. APPLICATIONS OF THE SVD 1

2.1. The Generalized Matrix Inverse 1
2.2. Numerical Rank 2
2.3. Least-Squares Curve Fitting 3
2.4. Image Compression 4
2.5. Voice Recognition 7

3. CONCLUSION 11

4. REFERENCES 12

1. Introduction

The singular value decomposition (SVD) of a real $m \times n$ matrix A is

$$\mathbf{A} = \mathbf{U}\Sigma\mathbf{V}^T \tag{1}$$

where U is an $m \times m$ orthogonal matrix, Σ is an $m \times n$ diagonal matrix, and V is an $n \times n$ orthogonal matrix. Such a factorization exists for all matrices [6]. The diagonal elements of Σ are $\{\sigma_1, \ldots, \sigma_p\}$, where $p = \min\{m,n\}$, and such that $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_p \geq 0$. The σ_i are called the singular values of A and the number of singular values p corresponds to the rank of A.

For the purposes of this assignment, I will assume that the reader has a reasonable knowledge of the SVD and its computation. In addition, I will consider real matrices in all formulations.

My emphasis here will be in specific applications of the SVD. In §2.1, I describe the construction of the generalized matrix inverse and in §2.2, I describe the assessment of matrix rank in numerical computations. Sections 2.3-2.5 hinge upon these two fundamental applications. In §2.3, I demonstrate curve fitting via the least-squares implementation of the SVD (related to the generalized inverse solution to rectangular systems). I apply numerical rank reduction in §2.4 and §2.5 where I discuss image compression and noise filtering for voice recognition systems.

2. Applications of the SVD

2.1. The Generalized Matrix Inverse

The inverse of full-rank, square matrices is well defined. The matrix B is the inverse of $n \times n$ A if $AB=I$ where I is the $n \times n$ identity matrix. Of course, the notion of such a matrix B for general $A \in \Re^{m \times n}$ is absurd. Nevertheless, using the SVD, one can determine a matrix $X \in \Re^{m \times n}$ such that the following properties hold [6]:

$$\mathbf{AXA} = \mathbf{A} \qquad \mathbf{XAX} = \mathbf{A} \qquad (\mathbf{AX})^T = \mathbf{AX} \qquad (\mathbf{XA})^T = \mathbf{XA} \tag{2}$$

X is called the Moore-Penrose generalized matrix inverse, or the pseudoinverse of A and is denoted A^+. Using the factorization defined by the SVD in (1), the matrix $A^+ \in \Re^{n \times m}$ that satisfies (2) is

$$\mathbf{A}^+ = \mathbf{V}\Sigma^+\mathbf{U}^T \tag{3}$$

Σ^+ is a diagonal $n \times m$ matrix with the reciprocal singular values of A on its diagonal. The pseudoinverse is considered a generalized matrix inverse since, by the existence of the SVD for any $m \times n$ matrix, the pseudoinverse also exists for any $m \times n$ matrix. If a matrix is square and invertible, then the pseudoinverse is exactly the usual matrix inverse.

Here is an example of a 2x3 matrix and its pseudoinverse determined by its SVD:

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \xrightarrow{SVD} \mathbf{A} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

then

$$\mathbf{A}^+ = \begin{bmatrix} 0 & 1 \\ 1 & 0 \\ 0 & 0 \end{bmatrix}$$

In §2.3, the pseudoinverse will provide a least-squares solution to an overdetermined linear system.

2.2. Numerical Rank

The number of independent rows or columns of a matrix characterizes its rank. However, when dealing with machine computations, this characterization is unsuitable. Consider the following matrix:

$$\mathbf{A} = \begin{bmatrix} 1 & 1 \\ 0 & \varepsilon \end{bmatrix} \tag{4}$$

for some small ε>0. At a set tolerance (or ultimately when ε is less than machine precision), A is reasonably rank deficient. The SVD yields a quantitative measure of the qualitative notion of rank [11]. In particular, *the smallest singular value of A is the 2-norm distance of A to the set of all rank-deficient matrices* [6].

The singular values of (4) are $\sqrt{2} + O(\varepsilon^2)$ and $\frac{\sqrt{2}}{2}\varepsilon + O(\varepsilon^3)$. Therefore, given a numerical tolerance less than ε, (4) will have rank one. Of course, this example is quite simple. In cases of large matrices, the rank deficiency may not be so obvious. The SVD, however, will amplify the subtle hints of rank deficiency in the singular values, no matter how complicated the matrix. In §2.5 and §2.6, I demonstrate the SVD's handiness with numerical rank pertaining to image compression and noise filtering.

2.3. Least-Squares Curve Fitting

Given any three distinct points in a plane, one can determine a unique parabola containing those points. In general, given more than three distinct points in a plane, a parabola fitting all the points may not exist. This scenario corresponds to an overdetermined system $Ax=b$ where A is an $m\times3$ ($m\geq3$ points) matrix, x is a 3×1 coefficient vector, and b is a $m\times1$ zero vector (See (5ab)).

$$ax^2+bx+c=0 \tag{5a}$$

$$\begin{bmatrix} x_1^2 & x_1 & 1 \\ x_2^2 & x_2 & 1 \\ \vdots & \vdots & \vdots \\ x_m^2 & x_n & 1 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{5b}$$

The SVD of matrix A produces a best solution to (5b). It can be shown that $\|\mathbf{A}x-b\|_2$ is minimized over all solutions x when $x = (V\Sigma^{+}U^{T})b$ [6,14]. Therefore, according to §2.1, the least-squares best solution to the system in (5b) is

$$x_{LS} = A^{+}b \tag{6}$$

In Figure 1, I have generated sets of random points in a 2D Cartesian coordinate system. For each set, I have plotted the parabola that solves (5b).

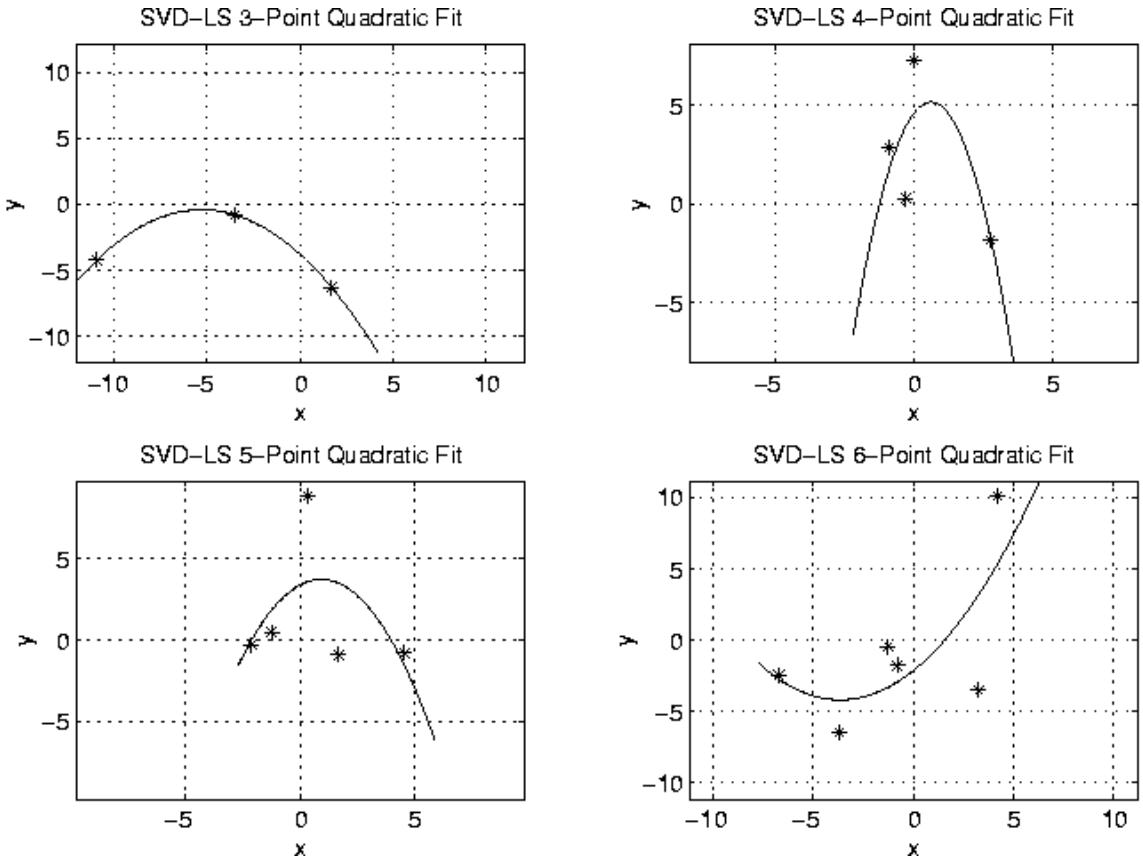


Figure 1

2.4. Image Compression

In information processing systems, it is crucial to transfer data quickly and efficiently. In most instances, optimal transfer is achieved by transmitting a minimal, reduced amount of data while still conveying the information. Such is the case for image processing. For most applications, the image in consideration possesses a pattern or underlying symmetry. If the image is encoded as set of data in a matrix, the symmetry will manifest itself in the rank of that matrix. In this application, rank deficiency reveals redundancies among the data that comprise the image. As descried in §2.2, the SVD effectively exposes rank deficiency in the singular values. Accordingly, a simple algorithm for image compression is as follows:

1) Retrieve the image data in a predefined matrix format
2) Decompose the image matrix with SVD
3) Reduce the system by eliminating small singular values and their corresponding rows/columns in U, V

4) Transmit the data in its reduced, decomposed form
5) Receive and reconstruct the image via the inverse SVD

Note that the final image contains the same amount of data as the original image. If q singular values are kept, then the decomposition is of the form

$$[m\times q]\cdot[q\times q]\cdot[q\times n]$$

where $q \leq \min\{m,n\}$. Then only $(mq + q + nq)$ data are transmitted[1]; whereas in no compression, mn data are sent. I define the compression ratio as

$$Compression\ Ratio = 1 - q\frac{(m+n+1)}{mn}. \tag{7}$$

Note that the compression ration can be negative. A negative compression ratio occurs when too many singular values are kept. In this case, the SVD contains more data than the original data matrix[2]. The maximum compression ratio is attained when q=1 and this value varies with matrix dimension. Of course, determining the optimal compression ratio is crucial for a successful transfer. Refer to §2.5 for discussion on determining rank for the analogous case of noise filtering.

In Figures 2ab, I have plotted a 33×41 image. The image data is indexed in a 33×41 matrix of surface height values. The data matrix has 31 singular values (rank of 31). Using the **svd(·,k)** command in MATLAB, I computed three, two, and one singular values. Then I recomposed the compressed image matrices. Accordingly, the rank reductions yielded 83%, 89%, and 94% compression ratios. There were no noticeable degradations for the 83% compression, and relatively moderate distortions for the 89% compression. In Figure 3, I plotted the difference between the uncompressed image and the 94% reduced image. From this figure, it is more apparent where the major distortions occurred (up to 30% distortion in the corners).

[1] One only needs the diagonal values of Σ for transmission.

[2] The worst-case results from keeping m singular values of an $m\times m$ matrix: Ratio = -(1+1/m).

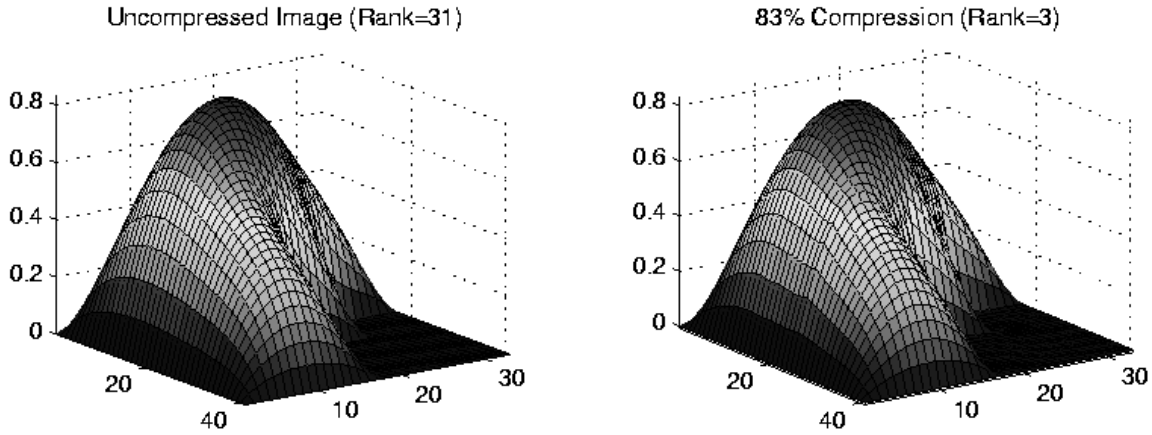


Figure 2a

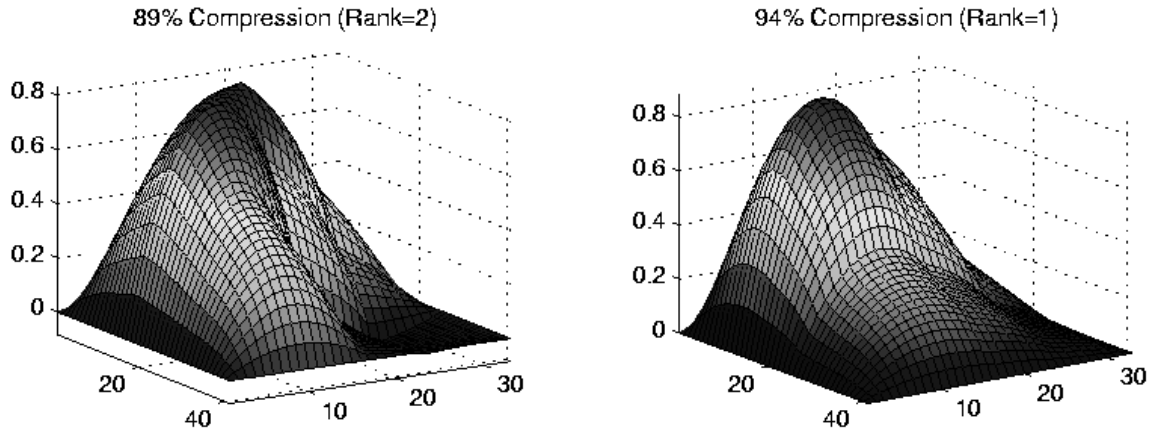


Figure 2b

Similar to the elimination of superfluous data discussed here, in §2.5 I discuss the elimination of erroneous data from a retrieved signal.

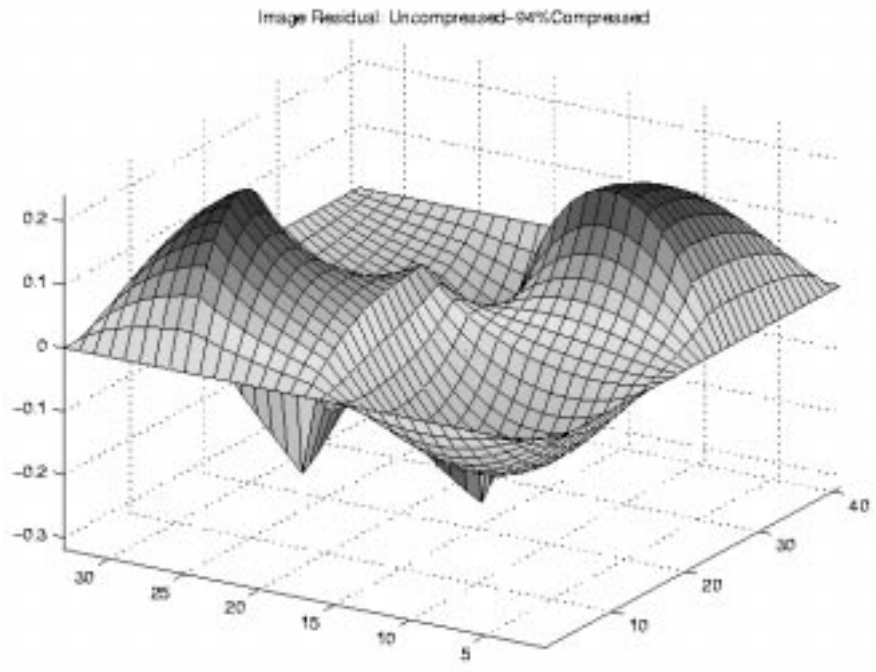


Figure 3

2.5. Voice Recognition

Background noise in practical environments is a major obstacle for voice recognition systems. Therefore, recorded speech needs to be preconditioned prior to the signal analysis in the recognition device. One can apply the SVD to this preconditioning problem if the additive "white" noise and the signal are uncorrelated and if the noise is small (overall) compared to the desired signal. The major principle considered here is that small, random, normally distributed white noise contributes mainly or wholly to the smaller singular values of the signal spectrum [1,3,8,9].

For the purposes of this paper, I will describe the basic process in which one can filter noise from an audio signal. I employed the following algorithm for filtering noise:

1) Retrieve and store the noisy audio signal.[3]
2) Partition the whole audio signal into smaller time/sample frames.
3) Construct a square (or nearly square) Hankel matrix from the signal vectors in each frame[4].
4) Compute the SVD of each Hankel matrix.

[3] The signal is imported as a vector of amplitude values indexed by signal sample.

[4] The j^{th} antidiagonal of each Hankel matrix corresponds to the j^{th} element in the signal vector.

5) Employ a rank reduction scheme to each matrix.
6) Re-compose reduced matrices of each frame.
7) Employ Hankel structure restoring scheme[5].
8) Unwind frame Hankel matrices into signal frames
9) Adjoin frames to form whole signal
10) Store filtered signal and analyze for recognition.

Using MATLAB, I applied the algorithm above on a voice sample of a female saying "goodbye." The sound file had a sampling frequency of 8kHZ at 8kB/sec. After importing the sound file, I added random noise so that the new noisy signal had a signal-to-noise ratio (SNR) of 10dB (See Figure 4)[6]. Then, I partitioned the signal vector into 22 frames of 256 samples. From these smaller signal vectors, I generated twenty-two 125×132 Hankel matrices.

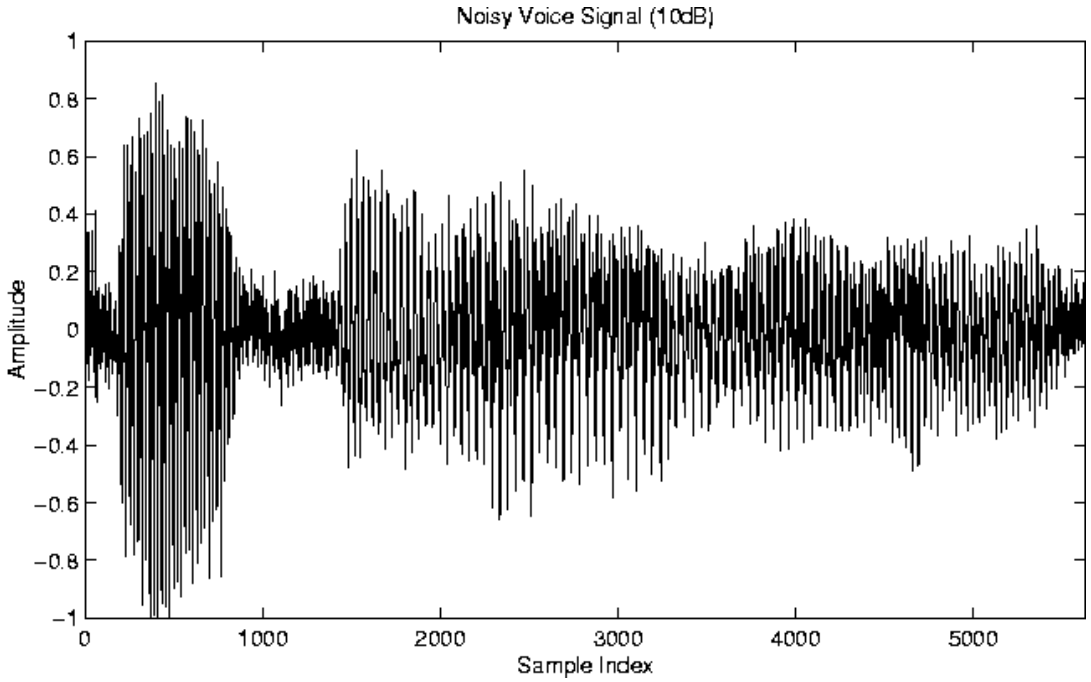


Figure 4: "Goodbye" with 10dB white noise.

Next, I determined the SVD of each Hankel matrix (i.e., of each frame). For this example, I chose a constant rank reduction scheme in which I uniformly reduced the rank of each Hankel matrix by a predetermined number.[7]

---

[5] The process of rank reduction destroys the Hankel structure of the signal matrices. To retrieve the filtered signal from the rank-reduced matrices, one must restore the Hankel structure. See [1,3,4,8,9]. In [2], Chu, et. al. Discuss structure-preserving rank reduction methods.

[6] The SNR is the ratio of the (mean) square amplitude of the signal to the (mean) square amplitude of the noise. To convert to decibel (dB) units, take $10\log_{10}(\text{SNR})$.

[7] There is ongoing research to develop algorithms that determine the optimal rank reduction given signal data [1,3,4,8,9].

The next step is to reconstruct the filtered signal Hankel matrix. To restore the Hankel structure of the matrices, I averaged the antidiagonal elements. The mean values of the antidiagonals represent the filtered signal elements (sample amplitudes)[8]. With the filtered signals in Hankel form, I obtained the signal vectors and adjoined them to form a whole, filtered signal.

Figure 5 compares the waveforms of the noise present in the signal before and after filtering. The SNR increased 36% from 10dB to 13.6dB in this case (chosen rank of 17).

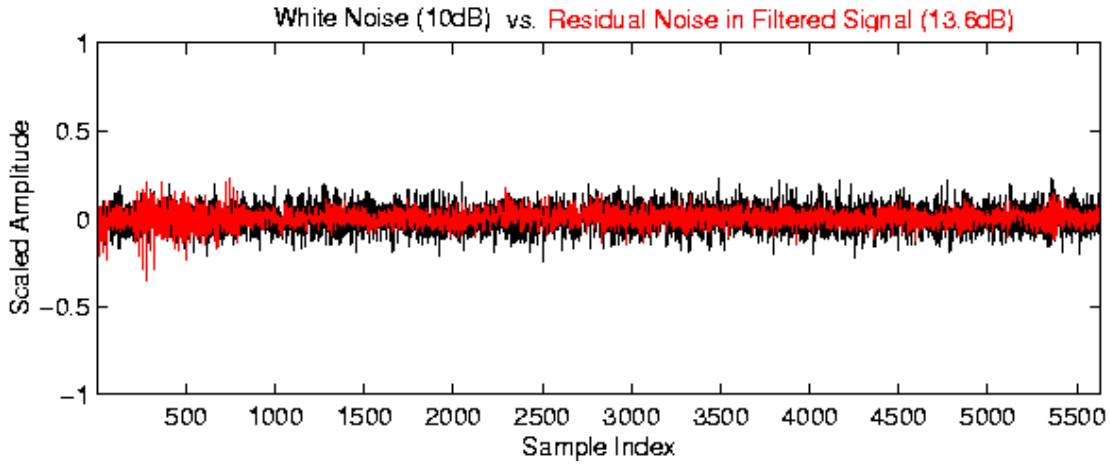


Figure 5: Original white noise and noise remaining after filter.

I also exhaustively computed the resulting SNRs from all possible uniform rank reductions. In Figure 6, it is evident that keeping a rank of 17 was optimal for 10dB white noise. Note that the uniform scheme produced bad results in the second and third frames (Figure 5) where the reduction actually had adverse effects on the signal. In this region, where the noise nearly matched the signal, a rank of 17 was probably unsuitable. I highly expect a non-uniform rank selection scheme to improve the SNR significantly.

[8] Actually, this symmetrization process in turn destroys the rank reduction. The iterative process of reducing rank and averaging converges to a rank-reduced, Hankel matrix. Whether the resulting Hankel matrix produces the cleanest signal is to-be-determined.

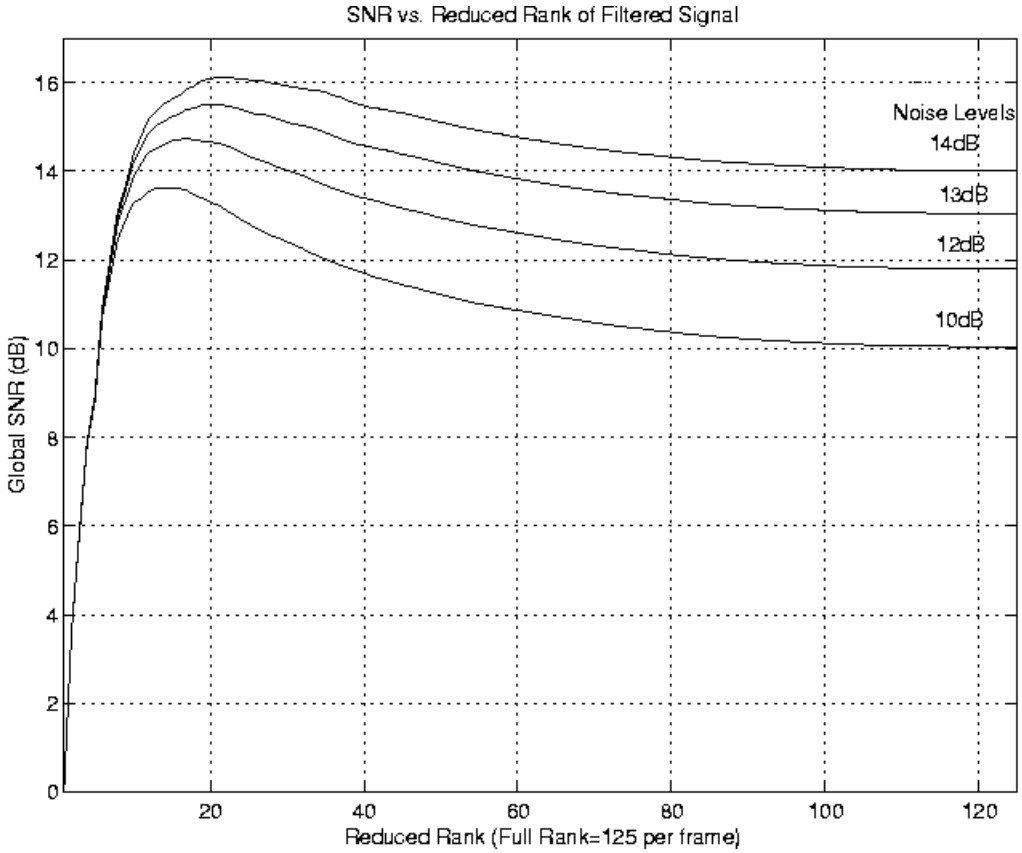


Figure 6: SNR of the whole signal vs. chosen frame matrix rank. Larger SNR values indicate less noise (better filtering). Of course, choosing too low of a rank will decrease the SNR, as the signal itself gets filtered out.

Audio files accompany this report. The files include the pure, noisy and filtered voices from various initial noise levels. Upon listening to the filtered samples, one notices the presence of tonal aberrations that were not present in the pure or noisy signals[9]. Other researchers have experienced this anomaly [3,9]. I conjecture that the tones stem from the distorting of signal data by restoring the Hankel structure to the reduced-rank matrices.

[9] Unlike the white noise, these aberrations have a definite audible structure.

3. Conclusion

The singular value decomposition is a versatile marvel of matrix theory and numerical analysis. Within the last 20 years, the engineering discipline has devised new ways and technologies based on the SVD's abilities to obtain least-squares solutions and its ability to discern numerical rank. I urge the reader to consult the references listed in §4 for other recent SVD applications.

4. References

[1] S. Bakamidis, M. Dendrinos, G. Carayannis, *SVD analysis by synthesis of harmonic signals*, IEEE Trans. on Signal Processing, **39**, No. 2 (1991), pp. 472-477.

[2] M. Chu, R. Funderlic, & R. Plemmons, (in press), *Structured Lower Rank Approximation*, 1998.

[3] M. Dendrinos, S. Bakamidis, G. Carayannis, *Speech enhancement from noise: A regenerative approach*, Speech Communication, **10** (1991), pp. 45-57.

[4] A. Ericsson-Jackson, P. Bainum, G. Xing, *Application of singular value decomposition method to digitally controlled large orbiting platforms*, J. of Guidance, Control, & Dynamics, **21**, No. 4 (1998), pp. 551-558.

[5] C. Gerald, P. Wheatley, *Applied Numerical Analysis.*, pp. 43-46, Addison Wesley Pub Co Inc, New York, 1994

[6] G. Golub, C. Van Loan, *Matrix Computations*, John Hopkins University Press, Baltimore, MD, 1996.

[7] J. Haueisen, et. al., *Classification of cardiac diseases based on singular value decomposition of the magnetocardiogram*, 18th Annual International Conf. Of the IEEE Engineering in Medicine and Biology Society, Amsterdam 1996, pp. 808-810.

[8] S. Jensen, et. al., *Reduction of broad-band noise in speech by truncated QSVD*, IEEE Trans. On Speech and Audio Processing, **3**, No. 6 (1995), pp. 439-448.

[9] B. Lilly, K. Paliwal, *Robust speech recognition using singular value decomposition based speech enhancement*, 1997 IEEE TENCON Speech & Image Technologies for Computing and Telecommunications, (1997), pp. 257-260.

[10] J. Luo, C. Chen, *Singular value decomposition for texture analysis*, Proc. Of the SPIE: Applications of Digital Image Processing XVII, **2298**, (1994), pp. 407-418.

[11] M. Lupu, D. Todor, *Linear Prediction and Singular Value Decomposition in NMR Signal Analysis*, in Signal Treatment and Signal Analysis in NMR, D.N. Rutledge, ed., Elsevier Science B.V., Amsterdam, Netheralnds, 1996, pp. 164-190.

[12] J. Navarro-Esbrí, et. al., *Reactor noise analysis based on the singular value decomposition (SVD)*, Annals of Nuclear Energy, **25**, No. 12, (1998), pp. 907-921.

[13] R. Nishi, et al., *New computed tomography algorithm of electrostatic force microscopy based on the singular value decomposition combined with the discrete Fourier transform*, Jpn. J. Appl. Phys., **37**, Pt. 2, No. 4A, (1998), pp. 417-419.

[14] G. Strang, *Linear Algebra and Its Applications*, Harcourt Brace Jovanovich College Publishers, Orlando, FL, 1988.