The Role of Singular Value Decomposition in Data Analysis

Shmuel Friedland
University of Illinois at Chicago

Stanford, UC Berkeley and Tel Aviv University, 12.05

1 Outline of the talk

1. Fast low rank approximation of matrices using Monte-Carlo techniques.

2. A joint SVD decomposition of two or more matrices to compare several biological processes.

3. Estimation of missing values in given matrix data using the inverse eigenvalue problems techniques, and their applications to DNA microarrays and image processing.

Most of the results can be found in the following recent papers, which are available at

http://www.math.uic.edu/∼friedlan/research.html

2 SVD in inner product spaces

$\mathbf{U}_i$ is m_i-dimensional IPS over $\mathbb{C}$, with $\langle \cdot, \cdot \rangle_i, i = 1, 2$.
$T : \mathbf{U_1} \to \mathbf{U_2}$ linear operator. $T^* : \mathbf{U_2} \to \mathbf{U_1}$ the adjoint operator: $\langle T\mathbf{x}, \mathbf{y} \rangle_2 = \langle \mathbf{x}, T^*\mathbf{y} \rangle_1$.
$S_1 := T^*T : \mathbf{U_1} \to \mathbf{U_1}$,
$S_2 := TT^* : \mathbf{U_2} \to \mathbf{U_2}$.

S_1, S_2 self-adjoint: $S_1^* = S_1, S_2^* = S_2$ and nonnegative definite: $\langle S_i \mathbf{x}_i, \mathbf{x}_i \rangle_i \geq 0$.

$\sigma_1^2 \geq ... \geq \sigma_r^2 > 0$ positive eigenvalues of S_1 and S_2 and $r = \mathbf{rank}\, T = \mathbf{rank}\, T^*$. Let
$S_1 \mathbf{v}_i = \sigma_i^2 \mathbf{v}_i, \ \langle \mathbf{v}_i, \mathbf{v}_j \rangle_1 = \delta_{ij}, \ i, j =, 1, ..., r.$
Define $\mathbf{u}_i := \sigma_i^{-1} T \mathbf{v}_i, i = 1, ..., r$. Then
$\langle \mathbf{u}_i, \mathbf{u}_j \rangle_2 = \delta_{ij}, i, j = 1, \ldots, r.$

Complete $\{\mathbf{v_1}, ..., \mathbf{v}_r\}$ and $\{\mathbf{u_1}, ..., \mathbf{u}_r\}$ to orthonormal bases $[\mathbf{v_1}, ..., \mathbf{v}_{m_1}]$ and $[\mathbf{u_1}, ..., \mathbf{u}_{m_2}]$ in $\mathbf{U_1}$ and $\mathbf{U_2}$.

3 Matrix SVD

Let $A \in \mathbb{C}^{m\times n}$. Then $A : \mathbb{C}^n \to \mathbb{C}^m$. Assume $\mathbb{C}^n, \mathbb{C}^m$ equipped with standard inner product $\langle \mathbf{x}, \mathbf{y} \rangle := \mathbf{y}^*\mathbf{x}$.

Then $A = U\Sigma V^*$, where $U \in \mathbf{U}(m), V \in \mathbf{U}(n)$, $\Sigma = \mathbf{diag}(\sigma_1, \ldots, \sigma_{\mathbf{min}(m,n)}) \in \mathbb{R}_+^{m\times n}$.

U, V transition matrices from $[\mathbf{u_1}, ..., \mathbf{u}_m], [\mathbf{v_1}, ..., \mathbf{v}_n]$ to the standard bases in $\mathbb{C}^m, \mathbb{C}^n$ respectively.

For $k \leq r$ let $\Sigma_k = diag(\sigma_1, \ldots, \sigma_k) \in \mathbb{R}^{k\times k}$, and $U_k \in \mathbf{U}(m, k), V_k \in \mathbf{U}(n, k)$ having the first k columns of U, V respectively. Then $A_k := U_k\Sigma_k V_k^*$ the best rank k approximation in Frobenius and operator norm of A:

$\mathbf{min}_{B\in\mathcal{R}(m,n,k)} ||A - B|| = ||A - A_k||$.

$A = U_r\Sigma_r V_r^*$ is Reduced SVD

$(r \geq)\ \nu$ numerical rank of A if $\frac{\sigma_{\nu+1}}{\sigma_\nu} \approx \mathbf{0}$.

A_ν is a noise reduction of A.

Noise reduction has many applications in image processing, DNA-Microarrays analysis, data compression.

4 RANDOM k-SVD

Stable numerical algortihms of SVD introduced by Golub-Kahan 1965, Golub-Reinsch 1970:

Implicit QR Algo to reduce to upper bidiagonal form using Householder matrices, then Golub-Reinsch SVD algo to zero superdiagonal elements.

Complexity: $O(mn\min(m,n))$.

In applications for massive data:
$A \in \mathbb{R}^{m\times n}, m, n >> 1$ needed a good approximation $A_k = \sum_{i=1}^{k} \mathbf{x}_i \mathbf{y}_i^T, \mathbf{x}_i \in \mathbb{R}^m, \mathbf{y}_i \in \mathbb{R}^n, i = 1, \ldots, k << \min(m,n)$.

Random A_k approximation algo:

Find a good algo by reading l rows or columns of A at random and update the approximations.

Frieze-Kannan-Vempala FOCS 1998 suggest algo without updating.

5 FKNZ RANDOM ALGO [6]

Fast k-rank approximation and SVD algorithm

Input: positive integers m, n, k, l, N, $m \times n$ matrix A, $\epsilon > 0$.

Output: an $m \times n$ k-rank approximation B_f of A, with the ratios $\frac{||B_0||}{||B_t||}$ and $\frac{||B_{t-1}||}{||B_t||}$, approximations to k-singular values and k left and right singular vectors of A.

1. Choose k-rank approximation B_0 using k columns, (or rows), of A.

2. **for** $t = 1$ **to** N

- Select l columns, (or rows), from A at random and update B_{t-1} to B_t.

- Compute the approximations to k-singular values, and k left and right singular vectors of A.

- If $\frac{||B_{t-1}||}{||B_t||} > 1 - \epsilon$ let $f = t$ and finish.

Complexity: $O(mnk)$.

Each iteration $||A - B_{t-1}||_F \geq ||A - B_t||_F$.

6 DETAILS

Choose at random k columns of A. Apply modified Gram-Schmidt algo to obtain $\mathbf{x_1}, \ldots, \mathbf{x_q} \in \mathbb{R}^m, q \leq k$.
Set $B_0 := \sum_{i=1}^{q} \mathbf{x}_i (A^{\mathrm{T}} \mathbf{x}_i)^{\mathrm{T}}$.
$||A - B_0||_F^2 = \mathrm{tr}\, A^T A - \mathrm{tr}\, B_0^T B_0 =$
$\mathrm{tr}\, A^T A - \sum_{i=1}^{q} (A^{\mathrm{T}} \mathbf{x}_i)^{\mathrm{T}} (A^{\mathrm{T}} \mathbf{x}_i)$.

Choose at random another l columns of A: $\mathbf{w_1}, \ldots, \mathbf{w_l}$.
Apply modified Gram-Schmidt algo to
$\mathbf{x_1}, \ldots, \mathbf{x_q}, \mathbf{w_1}, \ldots, \mathbf{w_l}$ to obtain o.n.s.
$\mathbf{x_1}, \ldots, \mathbf{x_q}, \mathbf{x_{q+1}}, \ldots, \mathbf{x_p}$. Form
$C_0 := B_0 + \sum_{i=q+1}^{p} \mathbf{x}_i (A^T \mathbf{x}_i)$.

Find the first left k-o.n. left singular vectors $\mathbf{v_1}, \ldots, \mathbf{v_k}$ of C_0. Then $B_1 := \sum_{i=1}^{k} \mathbf{v}_i (A^T \mathbf{v}_i)$ and
$\mathrm{tr}\, B_0^T B_1 \leq \mathrm{tr}\, B_1^T B_1$.

Obtain B_t from B_{t-1} as above.

7 Lifting body original

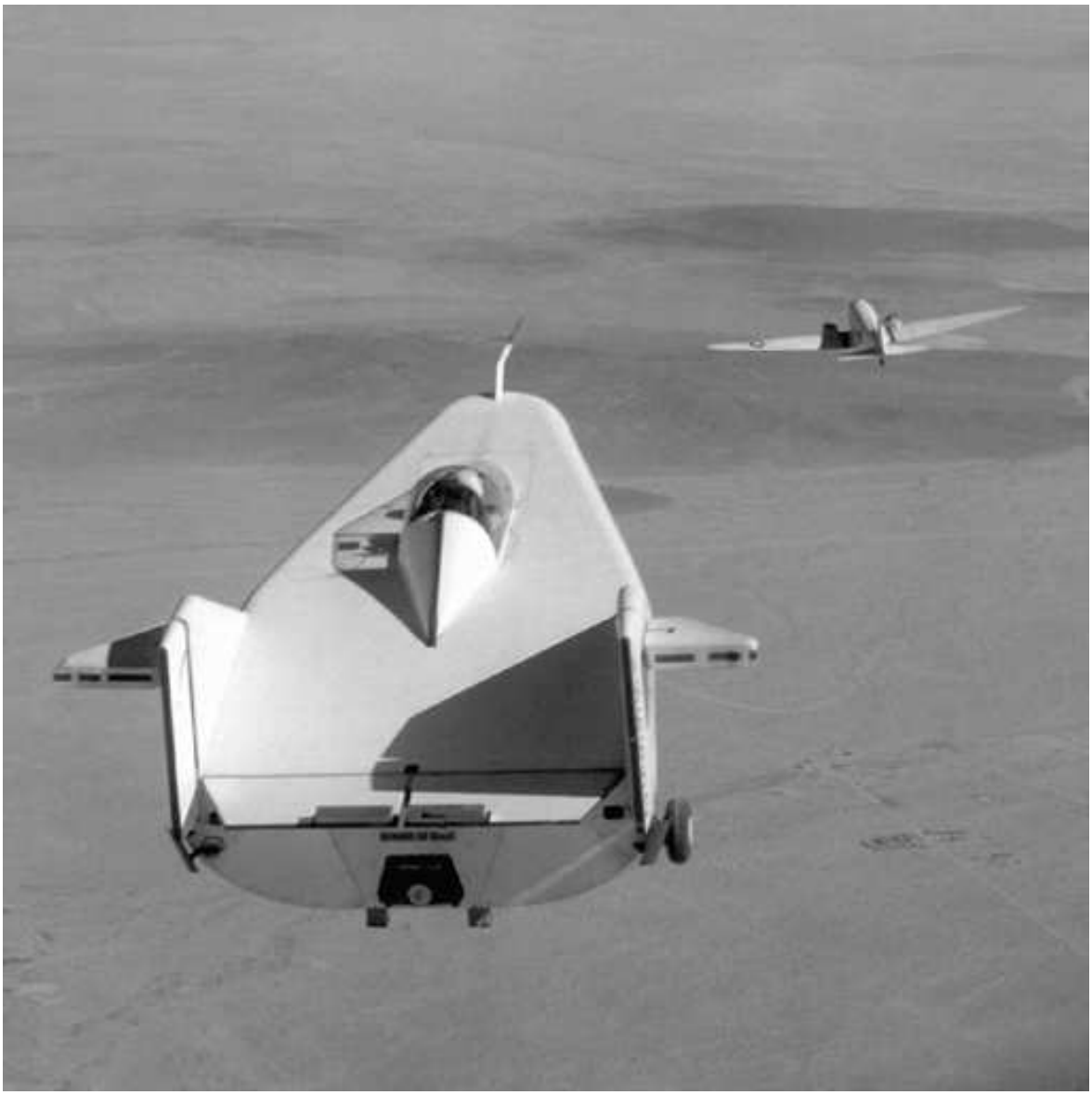

Figure 1: Lifting body image $\mathbf{512} \times \mathbf{512}$.

8 Lifting body compressed

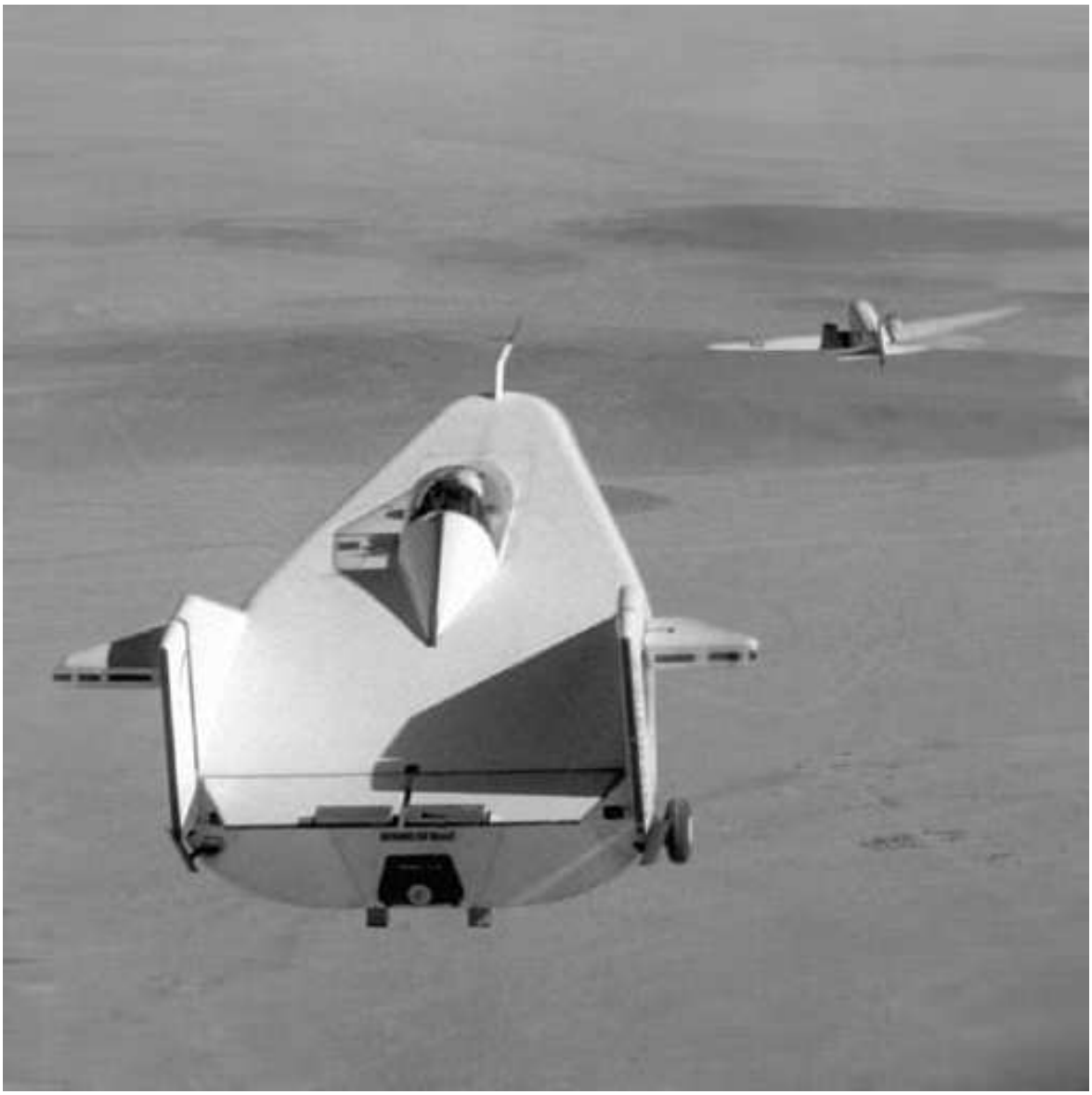

Figure 2: 80-rank approximation of Lifting body image 512×512.

9 SIMULATIONS 1

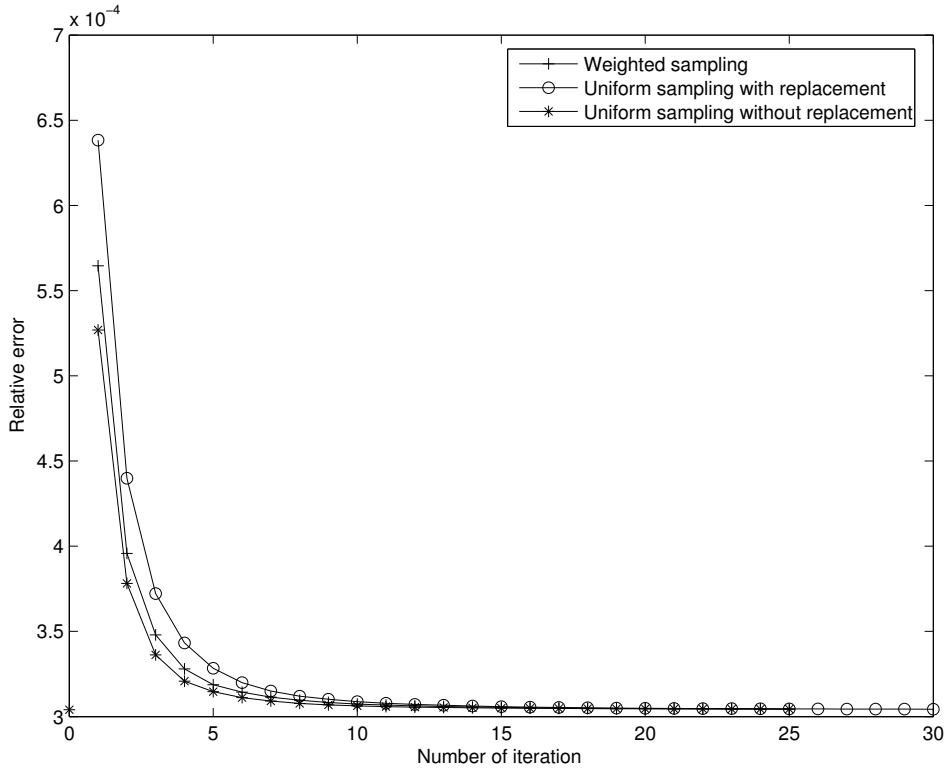

Figure 3: Convergence property of the Monte-Carlo method for Liftingbody image($\mathbf{512 \times 512}$), $\boldsymbol{k = 80}$.

10 SIMULATIONS 2

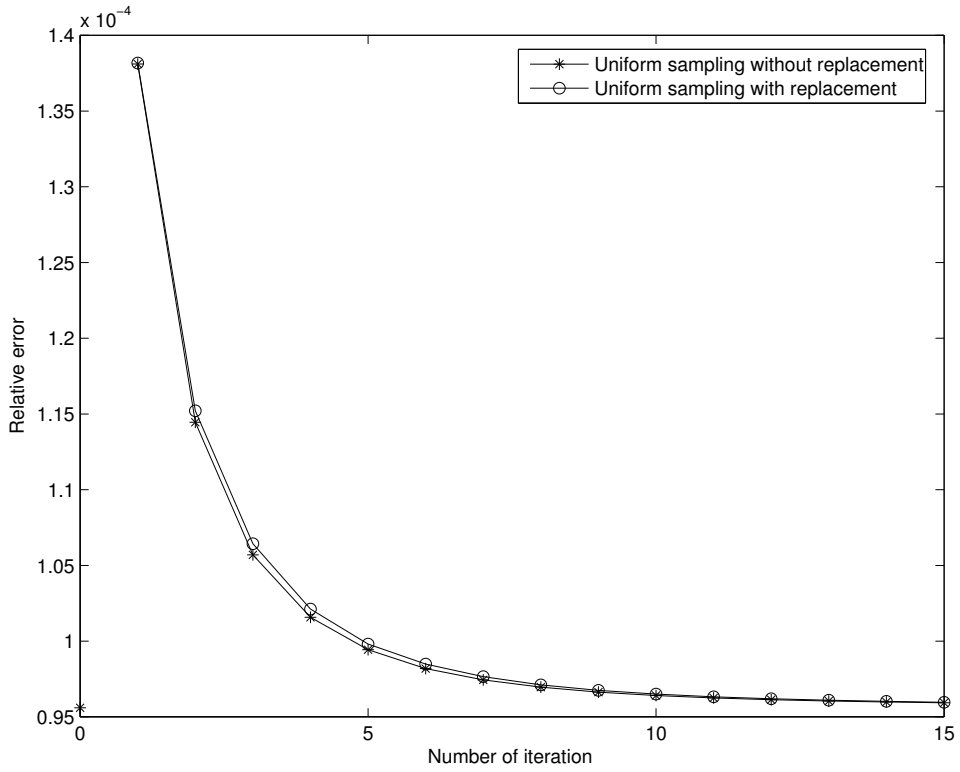

Figure 4: Convergence property of the Monte-Carlo method for random data matrix($\mathbf{3000} \times \mathbf{500}$).

11 COMPARISONS

Table 1: Comparison of relative error and speed up of our algorithm with optimum k-rank approximation algorithm

Data sets	Speed up	Re. ratio
Cameraman(256×256), $k = 80$	1.145	1.083
Liftingbody (512×512), $k = 100$	8	1.08
Map image(627×865) $k = 200$	3.33	1.067
Random matrix(8000×200) $k = 100$	42	1.1

12 Choosing columns of A

Frieze, Kannan and Vempala [10] suggest to choose column $\mathbf{c}_i(A)$ with probability $\frac{||\mathbf{c}_i(A)||^2}{||A||_F^2}$.

If $s \geq k$ are chosen then the k-approximation satisfies A_k

$||A - A_k||_F^2 \leq \sum_{i=k+1}^{m} \sigma_i(A)^2 + \frac{10k}{s}||A||_F^2$.

If $s \geq \frac{k}{10\epsilon}$ then

$||A - A_k||_F^2 \leq \sum_{i=k+1}^{m} \sigma_i(A)^2 + \epsilon||A||_F^2$.

Deshpande, Rademacher, Vempala and Wang [4] improved the sampling by modifying the sampling $\mathbf{c}_i(A)$ according to new probabilities $\frac{||\mathbf{c}_i(A-A_k)||^2}{||A-A_k||_F^2}$.

Perhaps our algorithm can be combined with above sampling of columns to get better results.

13 Clustering

Given a metric space $\mathbf{X}, d : \mathbf{X} \times \mathbf{X} \to \mathbb{R}_+$ and $\boldsymbol{\mathcal{X}} := \{\mathbf{x}_1, \ldots, \mathbf{x}_n\} \subset \mathbf{X}$ are n distinct points, associate $M := (d(\mathbf{x}_i, \mathbf{x}_j))_{i,j=1}^n \in \mathbb{R}_+^{n \times n}$.

Problem: Partition $\boldsymbol{\mathcal{X}}$ to clusters $\boldsymbol{\mathcal{X}} = \cup_{j=1}^m \boldsymbol{\mathcal{X}}_j$ using M.

There are many different approaches to solve this problem.

To use SVD one can consider the matrix

$F(\alpha, \beta) := (e^{-\alpha d(\mathbf{x}_i, \mathbf{x}_j)^\beta}) \in \mathbb{R}^{n \times n}, \alpha, \beta > 0$

Usually $\beta = 1, 2$. Fix β, e.g. ($\beta = 1$).

$F(0, \beta) = J_n, \lambda_1(J_n) = n, \lambda_i(J_n) = 0, i \geq 2.$

For $\alpha = 0$ $\boldsymbol{\mathcal{X}}$ is one cluster.

$F(\infty, \beta) = I_n$ and $\boldsymbol{\mathcal{X}}$ consists of n clusters.

For a right choice of $\alpha > 0$ $F(\alpha, \beta)$ will have numerical rank $r \in [1, n]$, which gives the number of clusters.

To identify the clusters additional analysis is needed.

(For small r) Approximate $F(\alpha, \beta)$ by rank r symmetric matrix G and try to cluster the columns of G as n vectors in $\mathbb{R}^r$.

14 Clustering of $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathbb{R}^r$

Case $r = 1$.

This case should be done by assuming the underlying probability model on the distribution of sampling points: uniform, normal, poisson, ...

General case:

Form $A = (\mathbf{x}_1, \ldots, \mathbf{x}_n) \in \mathbb{R}^{r \times n}$. Let ν be the numerical rank of A. Let $\mathbf{y}_1, \ldots, \mathbf{y}_\nu \in \mathbb{R}^r$ be ν orthonormal eigenvectors of $B := A^{\mathbf{T}} A$ corresponding to the first ν eigenvalues of B.

For each $i \in [1, \nu]$ cluster $\mathbf{y}_i^{\mathbf{T}} \mathbf{x}_1, \ldots, \mathbf{y}_i^{\mathbf{T}} \mathbf{x}_n \in \mathbb{R}$ as $\cup_{j=1}^{n_i} \mathcal{X}_{i,j}$.

Use these ν clusters to obtain the final clustering.

For example intersect all the clusterings.

15 Generalized SVD

Let $\boldsymbol{A} \in \mathbb{C}^{m\times n}, \boldsymbol{B} \in \mathbb{C}^{l\times n}$. Then Van Loan $\mathbf{70}\boldsymbol{s}$:
$\boldsymbol{A} = \boldsymbol{F}\boldsymbol{\Gamma}\boldsymbol{R},\ \boldsymbol{B} = \boldsymbol{G}\boldsymbol{\Delta}\boldsymbol{R},$
$\boldsymbol{F} \in \mathbf{U}(m),\ \boldsymbol{G} \in \mathbf{U}(l),\ \boldsymbol{R} \in \mathbf{GL}(n,\mathbb{C}),$
$\boldsymbol{\Gamma} \in \mathbb{R}_+^{m\times n}, \boldsymbol{\Delta} \in \mathbb{R}_+^{l\times n}$ diagonal matrices.

Numerical computations of GSVD are very unstable.

Thm ([5]). Let $\boldsymbol{P} := \boldsymbol{A}^*\boldsymbol{A} + \boldsymbol{B}^*\boldsymbol{B}$ and $r := \mathbf{rank}\,\boldsymbol{P}$.
Then $\boldsymbol{A} = \boldsymbol{U}\boldsymbol{\Phi}\boldsymbol{V}^*, \boldsymbol{U} \in \mathbf{U}(m,r), \boldsymbol{V} \in \mathbb{C}^{n\times r},$
$\boldsymbol{B} = \boldsymbol{W}\boldsymbol{\Psi}\boldsymbol{V}^*, \boldsymbol{W} \in \mathbf{U}(l,r),$
$\boldsymbol{\Phi} = \mathbf{diag}(\phi_1,\ldots,\phi_r),$
$\boldsymbol{\Psi} = \mathbf{diag}(\psi_1,\ldots,\psi_r) \in \mathbb{R}_+^{r\times r}$ and
$\boldsymbol{\Phi}^2 + \boldsymbol{\Psi}^2 = \boldsymbol{I}_r.$

Hence $\boldsymbol{P} = \boldsymbol{V}\boldsymbol{V}^*$ and the columns of $\boldsymbol{V}$ form an orthonormal basis of the subspace $\mathbf{X}$, spanned by the columns of $\boldsymbol{A}^*, \boldsymbol{B}^*$ with respect to the inner product $\langle \mathbf{x}, \mathbf{y}\rangle := \mathbf{y}^*\boldsymbol{P}\mathbf{x}$ on $\mathbf{V}$.

Reason: $\boldsymbol{T}_A^*\boldsymbol{T}_A + \boldsymbol{T}_B^*\boldsymbol{T}_B = \boldsymbol{I}|_{\mathbf{X}} \Rightarrow$
$(\boldsymbol{T}_A^*\boldsymbol{T}_A)(\boldsymbol{T}_B^*\boldsymbol{T}_B) = (\boldsymbol{T}_B^*\boldsymbol{T}_B)(\boldsymbol{T}_A^*\boldsymbol{T}_A)$

Constructive way to obtain GSVD

$P = O\Omega^2 O^*, O \in \mathbf{U}(n,r),$
$\Omega = \mathbf{diag}(\omega_1, ..., \omega_r),\ \omega_1 \geq ... \geq \omega_r > 0.$
$Q_A := \Omega^{-1} O^* A^* A O \Omega^{-1},$
$Q_B := \Omega^{-1} O^* B^* B O \Omega^{-1} \in \mathbf{H}(r).$

As $Q_A + Q_B = I_r,$

$Q_A = T\Phi^2 T^*, T \in \mathbf{U}(r),$
$\Phi = \mathbf{diag}(\phi_1, ..., \phi_r), \phi_i \geq 0,\ i = 1, \ldots, r,$
$Q_B = T\Psi^2 T^*, \Psi = \mathbf{diag}(\psi_1, ..., \psi_r),$
$\psi_i \geq 0,\ i = 1, \ldots, r,$
$\phi_i^2 + \psi_i^2 = 1, i = 1, \ldots, r.$

$V = O\Omega T$ and U, W obtained from
$U\Phi = AO\Omega^{-1}T, W\Psi = BO\Omega^{-1}T.$

Claim. Any GSVD decomposition given by Theorem F-05 is obtained as described above.

Microrrays Interpretation: A, B represent two different sets of genes under the same number of experiments n. r is the number of total acting functions. $\frac{\phi_i}{\psi_i}$ relative importance of function i in the first set versus the second set.

16 Numerical Examples

In this examples we choose

$l, m, n, r_A, r_B, r, r_0 + 1 \in \mathbb{N}$, such that

$r_0 \leq r_A \leq m, r_0 \leq r_B \leq l$,

$r = r_A + r_B - r_0 \leq n$ and random matrices

$A \in \mathbb{R}^{m \times n}, B \in \mathbb{R}^{l \times n}$ of ranks r_A, r_B such that

$\dim(A^{\mathbf{T}}\mathbb{R}^m \cap B^{\mathbf{T}}\mathbb{R}^l) = r_0$.

Choose random $\mathbf{x}_1, \ldots \mathbf{x}_{r+r_0} \in \mathbb{R}^n$,

$\mathbf{y}_1, \ldots, y_{r_A} \in \mathbb{R}^m, \mathbf{z}_1, \ldots, z_{r_B} \in \mathbb{R}^l$

Then $A = \sum_{i=1}^{r_A} \mathbf{y}_i \mathbf{x}_i^{\mathbf{T}}$,

$B = \sum_{i=1}^{r_0} \mathbf{z}_i \mathbf{x}_i^{\mathbf{T}} + \sum_{i=r_0+1}^{r_B} \mathbf{z}_i \mathbf{x}_{i+r_A-r_0}^{\mathbf{T}}$

We generated $A_0 \in \mathbf{M}_{8,7}(\mathbb{R}),\ B_0 \in \mathbf{M}_{9,7}(\mathbb{R})$ with

$r_0 = 1, r_{A_0} = r_{B_0} = 2$ as above.

We used Maple routine to generate random vectors and matrices with integer entries in the range $[-99, 99]$.

Hence the matrices A_0 and B_0 have integer entries.

$$A_0 = \begin{pmatrix} 1826 & 846 & 1516 & 1831 & 3060 & -57 \\ -3452 & -1752 & -2182 & -2827 & -5970 & 119 \\ 5765 & 3573 & 745 & 2032 & 10755 & -246 \\ -202 & -1818 & 7558 & 6964 & -2430 & 128 \\ 3873 & 1353 & 5193 & 5718 & 5955 & -91 \\ -5206 & -2862 & -2306 & -3350 & -9270 & 196 \\ -2060 & 1224 & -11470 & -11119 & -810 & -89 \\ -2630 & -726 & -4390 & -4684 & -3810 & 48 \end{pmatrix}$$

$$B_0 = \begin{pmatrix} -3652 & -3486 & 640 & 2833 & -321 & 1424 \\ -8657 & -7471 & -2665 & 3283 & 1354 & 2669 \\ 2420 & 2122 & 568 & -1063 & -289 & -776 \\ -3927 & -4161 & 2865 & 4833 & -1446 & 1899 \\ 253 & -873 & 5837 & 4631 & -2952 & 895 \\ -4620 & -2044 & -11676 & -6664 & 5908 & -308 \\ 2596 & 2388 & 20 & -1624 & -12 & -932 \\ -8624 & -7722 & -1180 & 4481 & 603 & 2908 \\ -7964 & -5438 & -10024 & -3195 & 5075 & 1176 \end{pmatrix}$$

The first three singular values of A_0, B_0 are

$$27455.509,\ 17374.683,\ 3.141 \times 10^{-12},$$
$$29977.543,\ 19134.384,\ 3.524 \times 10^{-12},$$

The four first singular values of P are

$$1.322 \times 10^{9},\ 6.044 \times 10^{8},$$
$$3.943 \times 10^{8},\ 1.346 \times 10^{-7}$$

The 3 generalized singular values of A_0, B_0:

$$\phi_1 = 1,\ \phi_2 = 0.681, \phi_3 = 3.778 \times 10^{-9},$$
$$\psi_1 = 0,\ \psi_2 = 0.732,\ \psi_3 = 1$$

So A_0, B_0 have one common function corresponding to one dimensional common subspace $A_0^{\mathbf{T}}\mathbb{R}^8 \cap B_0^{\mathbf{T}}\mathbb{R}^9$.

The matrix $V \in \mathbf{M_{7,3}}(\mathbb{R})$ given by

$$\begin{pmatrix} \mathbf{9975.156} & \mathbf{2218.778} & \mathbf{-16518.709} \\ \mathbf{4258.080} & \mathbf{5446.091} & \mathbf{-13181.085} \\ \mathbf{9910.168} & \mathbf{-17951.929} & \mathbf{-10755.839} \\ \mathbf{11513.101} & \mathbf{-16136.565} & \mathbf{1610.055} \\ \mathbf{16275.444} & \mathbf{9076.818} & \mathbf{5451.311} \\ \mathbf{-2894.442} & \mathbf{-3832.434} & \mathbf{4134.750} \\ \mathbf{8306.914} & \mathbf{-8471.697} & \mathbf{-15231.563} \end{pmatrix}$$

$\boldsymbol{U_1}$ has the first two columns of $\boldsymbol{U} \in \mathbf{M_{8,3}}(\mathbb{R})$:

$$\begin{pmatrix} \mathbf{.1797} & \mathbf{0.0217} \\ \mathbf{-.3322} & \mathbf{-0.0908} \\ \mathbf{.5229} & \mathbf{.3627} \\ \mathbf{0.0653} & \mathbf{-.5648} \\ \mathbf{.4031} & \mathbf{-0.0979} \\ \mathbf{-.4902} & \mathbf{-.2086} \\ \mathbf{-.3105} & \mathbf{.6860} \\ \mathbf{-.2832} & \mathbf{.1293} \end{pmatrix}$$

$\boldsymbol{W_1}$ has the last two columns of $\boldsymbol{W} \in \mathbf{M_{9,3}}(\mathbb{R})$

$$\boldsymbol{W_1} = \begin{pmatrix} \mathbf{-.2125} & \mathbf{.2001} \\ \mathbf{-.2093} & \mathbf{.5034} \\ \mathbf{0.0709} & \mathbf{-.1395} \\ \mathbf{-.3819} & \mathbf{.2001} \\ \mathbf{-.3995} & \mathbf{-0.0545} \\ \mathbf{.6105} & \mathbf{.3397} \\ \mathbf{.1176} & \mathbf{-.1455} \\ \mathbf{-.3124} & \mathbf{.4913} \\ \mathbf{.3408} & \mathbf{.5156} \end{pmatrix}.$$

17 Robustness of GSVD

Let $A := A_0 + X, B := B_0 + Y$, where $X \in \mathbf{M_{8,7}}(\mathbb{R}),\ Y \in \mathbb{R}_{9,7}(\mathbb{R})$ with random entries and relatively small ℓ_2 norm with respect to ℓ_2 norms of A, B respectively. X, Y have integer entries in $[-99, 99]$. X is

$$\begin{pmatrix} -14 & -73 & 65 & 3 & -14 & 16 & 9 \\ -10 & 8 & 90 & -94 & -22 & -24 & 0 \\ 78 & 32 & -48 & -6 & 80 & -18 & -63 \\ 66 & -13 & 88 & -45 & -92 & -69 & -43 \\ 32 & 9 & 41 & -95 & -28 & -90 & -63 \\ -23 & -72 & -84 & -84 & -58 & -37 & -40 \\ 35 & 14 & -29 & 76 & -62 & -82 & -5 \\ 18 & -40 & -51 & 11 & 87 & 66 & -46 \end{pmatrix}$$

$\boldsymbol{Y}$ is

$$\begin{pmatrix} -14 & -83 & 65 & -22 & -80 & 43 & -56 \\ -55 & -50 & 68 & 66 & 9 & -26 & -58 \\ -63 & -32 & -25 & 96 & 90 & -5 & 28 \\ -49 & 31 & -17 & -27 & 46 & -5 & 37 \\ 81 & -99 & -98 & 22 & 58 & -68 & -37 \\ 59 & 57 & 65 & -64 & 65 & 84 & 41 \\ -68 & 36 & -63 & 7 & -58 & 53 & 90 \\ 95 & -27 & 54 & -16 & -46 & -18 & 46 \\ 23 & 10 & -64 & 58 & 58 & -73 & 97 \end{pmatrix}$$

The singular values of $\boldsymbol{X}, \boldsymbol{Y}$ rounded off to three significant digits are:

$(\mathbf{266},\ \mathbf{183},\ \mathbf{165},\ \mathbf{151},\ \mathbf{99.1},\ \mathbf{36.0},\ \mathbf{14.1})$,
$(\mathbf{259},\ \mathbf{229},\ \mathbf{198},\ \mathbf{153},\ \mathbf{116},\ \mathbf{86.8},\ \mathbf{46.2})$

So $||\boldsymbol{X}|| \sim \mathbf{0.01}||\boldsymbol{A_0}||,\ ||\boldsymbol{Y}|| \sim \mathbf{0.01}||\boldsymbol{B_0}||$.

The singular values of $\boldsymbol{A}, \boldsymbol{B}$ rounded off to three significant digits are:

$(\mathbf{27490}, \mathbf{17450}, \mathbf{233}, \mathbf{130}, \mathbf{119}, \mathbf{70.0}, \mathbf{18.2})$,
$(\mathbf{29884}, \mathbf{19183}, \mathbf{250}, \mathbf{187}, \mathbf{137}, \mathbf{102}, \mathbf{19.7})$

Assume that the numerical rank of A, B is $\nu = 2$.

Replace A, B by A_2, B_2 of rank two. Then two nonzero singular values of A_2, B_2 are

$(27490, 17450), (29883, 19183)$,

(rounded to 5 significant digits.)

The singular values of $P = A_2^{\mathrm{T}} A_2 + B_2^{\mathrm{T}} B_2$ (up to 3 significant digits:)

$(1.32 \times 10^9, 6.07 \times 10^8, 3.96 \times 10^8, 1.31 \times 10^4, 0.068, 9.88 \times 10^{-3}, 6.76 \times 10^{-3})$

Assume first that the numerical rank of P is $\nu = 3$.

$P_3 = O \Omega O^{\mathrm{T}}, O \in \mathrm{O}(7, 3)$,

$Q_{A_2} := \Omega^{-1} O^{\mathrm{T}} A_2^{\mathrm{T}} A_2 O \Omega^{-1} = T \Phi^2 T^{\mathrm{T}}$,
$Q_{B_2} := \Omega^{-1} O^{\mathrm{T}} B_2^{\mathrm{T}} B_2 O \Omega^{-1} = T \Psi^2 T^{\mathrm{T}}$ where $T \in \mathrm{O}(3)$, $V_2 = O \Omega T$ and U_2, W_2 obtained from $U_2 \Phi = A_2 O \Omega^{-1} T, W_2 \Psi = B_2 O \Omega^{-1} T$.

The three generalized singular values of A_2, B_2 are $(1.0000, .6814704276, 0.7582 \times 10^{-8})$, $(0., .7318456506, 1.0)$, match the generalized singular values of A_0, B_0 at least up to four significant digits. The relative matching of V and V_2, and the computable columns of U, U_2 and W, W_2 is good up 4 digits.

Assume second that the numerical rank of P is $\nu = 4$.
$P_4 = O\Omega O^{\mathrm{T}}, O \in \mathrm{O}(7,4)$,
$Q_{A_2} := \Omega^{-1} O^{\mathrm{T}} A_2^{\mathrm{T}} A_2 O \Omega^{-1} = T \Phi^2 T^{\mathrm{T}}$,
$Q_{B_2} := \Omega^{-1} O^{\mathrm{T}} B_2^{\mathrm{T}} B_2 O \Omega^{-1} = T \Psi^2 T^{\mathrm{T}}$ where
$T \in \mathrm{O}(4)$, $V_2 = O\Omega T$ and U_2, W_2 obtained from
$U_2 \Phi = A_2 O \Omega^{-1} T, W_2 \Psi = B_2 O \Omega^{-1} T$. Then the four generalized singular values of A, B up to six significant digits are $(1,1,0,0), (0,0,1,1)$!

In particular each matrix has two uncorrelated functions

Replace A, B by A_3, B_3 of rank three. The singular values of the matrix P up to three significant digits are:
$(1.32 \times 10^9, 6.07 \times 10^8, 3.96 \times 10^8, 4.74 \times 10^4, 3.83 \times 10^4, 5.39 \times 10^3, 9.70 \times 10^{-3})$.
Assume first that P has numerical rank $\nu = 6$. Then the generalized singular values of A_3, B_3 are $(1,1,1,0,0,0)$ and $(0,0,0,1,1,1)$ up to six significant digits!.

Assume finally that $\nu = 3$. Then the generalized singular values of A, B are close to gsv of A_0, B_0:
$(.9999796224, .6814701987, 0.005232470265)$,
$(0.006383948621, .7318458638, .9999863106)$.

18 GSVD for many matrices

Let $A_i \in \mathbb{C}^{m_i \times n}$ for $i = 1, ...k \geq 3$.

What is the corresponding GSVD?

Form $P := \sum_{i=1}^{k} A_i^* A_i \in \mathbb{C}^{n \times n}$ and $r := \operatorname{rank} P$ (or r is the numerical rank of P).

$$P_r = O\Omega^2 O^*, O \in \mathrm{U}(n, r),$$
$$\Omega = \operatorname{diag}(\omega_1, ..., \omega_r),\ \omega_1 \geq ... \geq \omega_r > 0.$$
$$Q_i := \Omega^{-1} O^* A_i^* A_i O \Omega^{-1}, i = 1, \ldots, k.$$
$$Q_i = T_i \Phi_i^2 T_i^*, T = (\mathrm{t}_{1,i}, \ldots, \mathrm{t}_{r,i}) \in \mathrm{U}(r),$$
$$\Phi_i = \operatorname{diag}(\phi_{1,1}, ..., \phi_{i,r}),$$
$$\phi_{1,i} \geq \ldots \geq \phi_{r,i} \geq 0,\ j = 1, \ldots, r,$$
$$\sum_{j=1}^{p} \sum_{i=1}^{k} \phi_{j,i}^2 \geq p,\ p = 1, \ldots, r.$$

The relative importance of $j-th$ function in A_i with respect to all $A_1, \ldots, A_l$ is measured by the value $k\phi_{j,i}^2$.
The relative importance of $j-th$ function in A_i with respect to the $j-th$ function in A_l is given by $\frac{\phi_{j,i}^2}{\mathrm{t}_{j,i}^* A_l \mathrm{t}_{j,i}}$.

19 MISSING ENTRIES PROBLEM

In DNA Microarrays experiments one measure thousands of genes $i = 1, \ldots, m$ in n different conditions, typically $n \in [3, 20]$.

$0 \leq a_{ij}$ measures the intensity of gene i in $j - th$ experiment. The results are recorded in the matrix $A = (a_{ij}) \in \mathbb{R}^{m \times n}$.

Sometimes the entries a_{ij} are missing (corrupted, up to 20%).

Let $\mathcal{T} \subset \{1, \ldots, n\} \times \{1, \ldots, m\}$ missing entries set.

Set $a_{ij} = 0$ if $(i, j) \in \mathcal{T}$.

Let $\mathcal{X}$ be all $X = (x_{ij}) \in \mathbb{R}^{m \times n}$ where $x_{ij} = 0$ if $(i, j) \notin \mathcal{T}$.

Assume that the completed matrix of the experiment should have the numerical rank ν. Then we complete the entries by solving the problem:

$(1) \quad \min_{X \in \mathcal{X}} \sum_{i=\nu+1}^{n} \sigma_i^2(A + X) =$
$\min_{X \in \mathcal{X}} \sum_{i=\nu+1}^{n} \lambda_i((A + X)^{\mathrm{T}}(A + X))$

20 FRAA

Fixed Rank Approximation Algorithm: [8]

Let $G_p \in \mathcal{X}$ be the p^{th} approximation to a solution of optimization problem (1). Let $A_p := G_p^{\mathrm{T}} G_p$ and find an orthonormal set of eigenvectors for A_p, $\mathbf{v}_{p,1}, \ldots, \mathbf{v}_{p,m}$. Then G_{p+1} is a solution to the following minimum of a convex nonnegative quadratic function

$\min_{X \in \mathcal{X}} \sum_{q=l+1}^{m}$
$(X\mathbf{v}_{p,q})^{\mathrm{T}}(X\mathbf{v}_{p,q})$.

Flow chart of the algorithm:

Fixed Rank Approximation Algorithm (FRAA)

Input: integers $m, n, L, iter$, the locations of non-missing entries $\mathcal{S}$, initial approximation G_0 of $n \times m$ matrix G.

Output: an approximation G_{iter} of G.

for $p = 0$ **to** $iter - 1$

- Compute $A_p := G_p^{\mathrm{T}} G_p$ and find an orthonormal set of eigenvectors for A_p, $\mathbf{v}_{p,1}, \ldots, \mathbf{v}_{p,m}$.

- G_{p+1} is a solution to the minimum problem (1) with $\nu = L - 1 = l$.

Let $f_l(X) := \sum_{i=\nu+1}^{n} \sigma_i^2(A+X)$. In each step of the algorithm $f_l(G_p) \geq f_l(G_{p+1})$. $G_p, p = 1, \ldots$ converges to a critical point $\tilde{G}$. FRAA gives a good approximation of $\tilde{G}$. In many simulations $\tilde{G} = G^*$.

FRAA is an adaptation of an algo for IEP:

Inverse Eigenvalue Problem: *Find the values of the missing entries of* G *such that the nonnegative definite matrix* $G^{\mathbf{T}}G$ will have $m - l$ smallest eigenvalues equal to zero.

IEP appear often in engineering. See [9] for examples of IEP and a number of good algorithms to solve these problems.

FRAA is a robust algorithm which performs good, but not as well as KNNimpute, BCPA and LSSimpute.

All other algo reconstruct the missing values of each gene from similar genes.

21 IFRAA - FKNZ

Improved Fixed Rank Approximation Algorithm [7].

First use FRAA to find a completion G.

Then use a cluster algorithm,

(We used K-means repeating & refining cluster size),

to find a reasonable number of clusters of similar genes, each cluster is a relatively smaller matrix having an effective low rank.

For each cluster of genes apply FRAA separately to recover the missing entries in this cluster.

These results suggest that IFRAA has a potential for being an effective algorithm to recover blurred spots in digital images.

22 SIMULATIONS 1

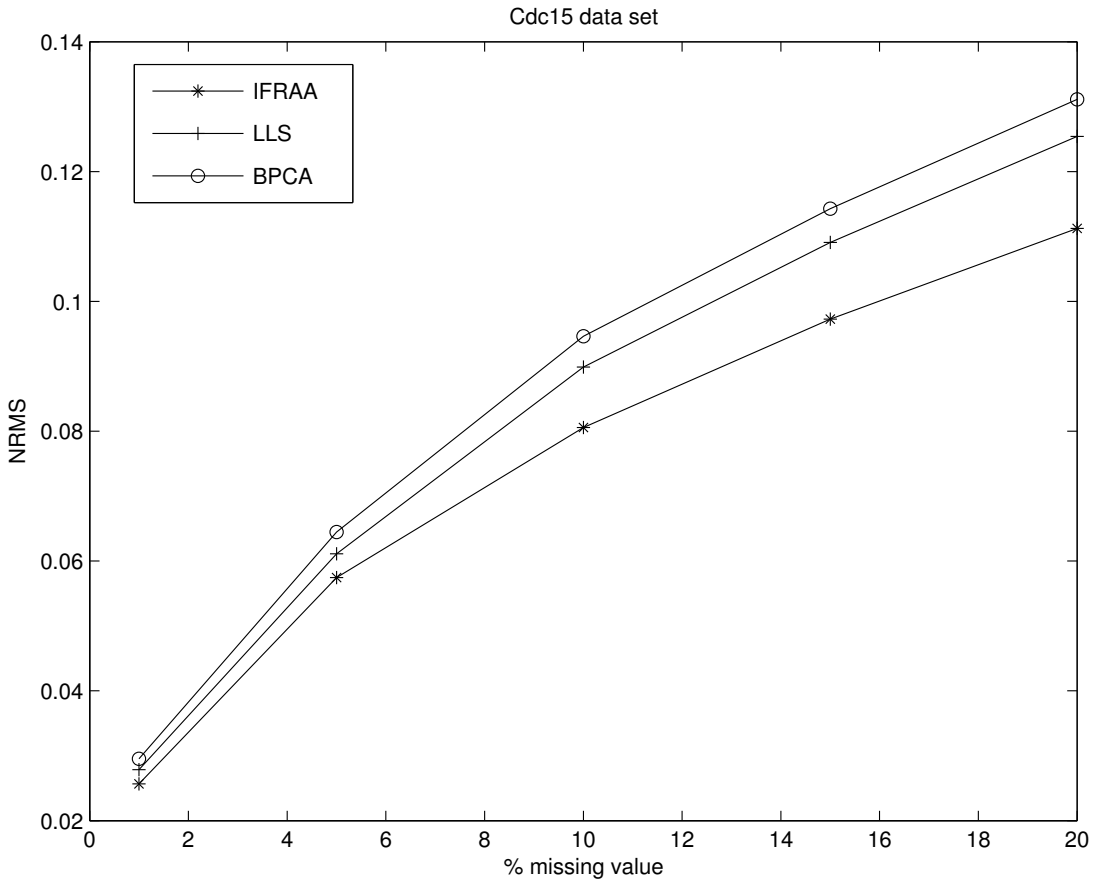

Figure 5: Comparison of NRMSE against percent of missing entries for three methods: IFRAA, BPCA and LLS. Cdc15 data set in [17] with 24 samples.

23 SIMULATIONS 2

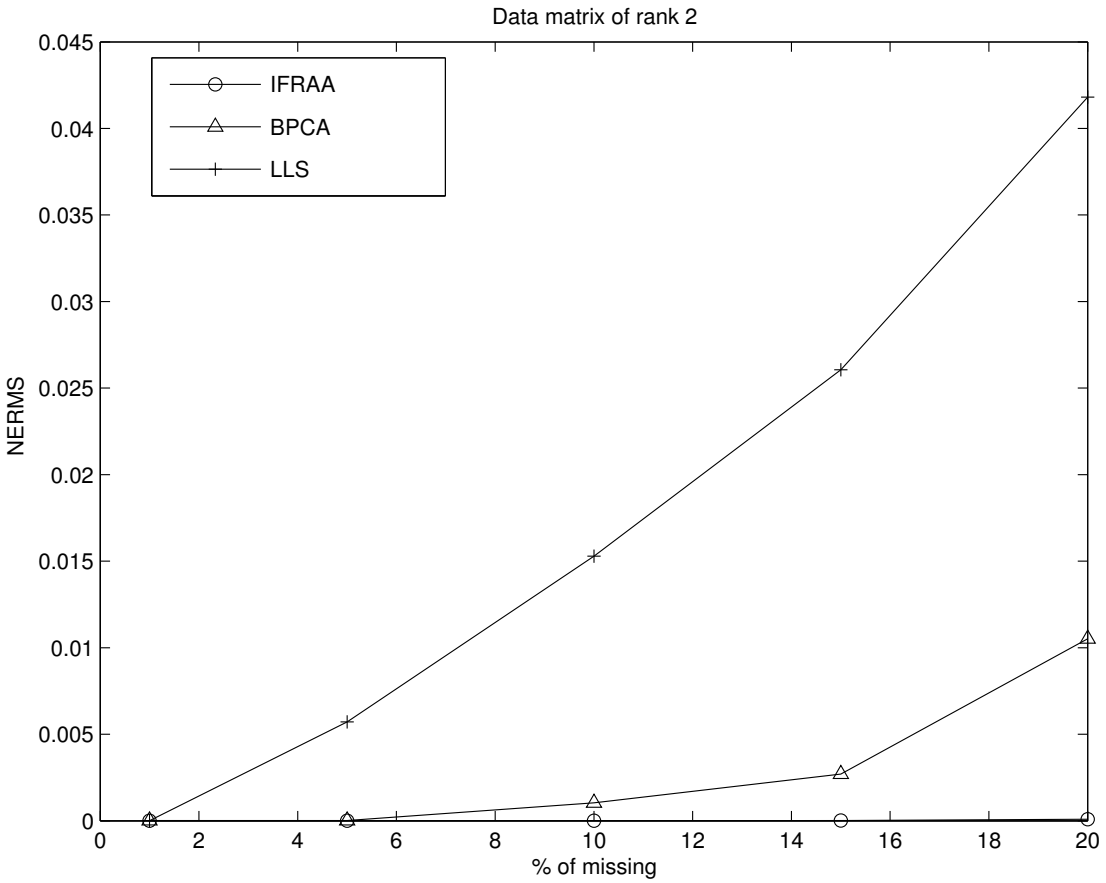

Figure 6: Comparison of NRMSE against percent of missing entries for three methods: IFRAA, BPCA and LLS. Data set was a $\mathbf{2000} \times \mathbf{20}$ randomly generated matrix of rank 2.

Bayesian principal component analysis-BPCA [15]: A global method consisting of three components. First, principal component regression, which is basically a low rank approximation of the data set is performed. Second, Bayesian estimation, which assumes that the residual error and the projection of each gene on principal components behave as normal independent random variables with unknown parameters, is carried out. Third, Bayesian estimation follows by iterations based on the expectation-maximization (EM) of the unknown Bayesian parameters.

Local least squares imputation method LLS [14]: A local methods, which use similarity structure of the data to impute the missing values. LLS has two versions to find similar genes whose expressions are not corrupted: the L_2-norm and the Pearson's correlation coefficients. After a group of similar genes C are identified, the missing values of the gene are obtained using least squares applied to the group C. The recovery of missing data is done independently, i.e. the estimation of each missing entry does not influence the estimation of the other missing entries.

24 TABLE

The performance of the BCPA, IFRAA and LLS algorithms depends on the unknown distribution of missing position of the entries.

Table 2: Comparison of NRMSE for three methods: IFRAA, LLS and BPCA for actual missing values distribution for three gene expression data sets with different percentage of missing values.

Data sets	IFRAA	LLS	BPCA
Cdc15 data set %0.81 missing	0.0175	0.0200	0.0216
Evolution data set %9.16	0.0703	0.0969	0.1247
Calcineurin data set %3.68	0.0421	0.0445	0.0453

References

[1] O. Alter, P.O. Brown and D. Botstein, Singular value decomposition for genome-wide expression data processing and modelling, *Proc. Nat. Acad. Sci. USA* 97 (2000), 10101-10106.

[2] O. Alter, P.O. Brown and D. Botstein, Generalized singular decomposition for comparative analysis of genome-scale expression data sets of two different organisms, *Proc. Nat. Acad. Sci. USA* 100 (2003), 3351-3356.

[3] O. Alter, G.H. Golub, P.O. Brown and D. Botstein, Novel genome-scale correlation between DNA replication and RNA transcription during the cell cycle in yeast is predicted by data-driven models, 2004 *Miami Nature Winter Symposium*, Jan. 31 - Feb. 4, 2004, to appear.

[4] A. Deshpande, L. Rademacher, S. Vemapala and G. Wang, Matrix Approximation and Projective Clustering via Volume Sampling, *SODA*, 2006.

[5] S. Friedland, A New Approach to Generalized Singular Value Decomposition, to appear in *SIMAX*.

[6] S. Friedland, M. Kaveh, A. Niknejad and H. Zare, Fast Monte-Carlo Low Rank Approximations for Matrices, preprint 2005, submitted, (Arxiv...)

[7] S. Friedland, M. Kaveh, A. Niknejad and H. Zare, An Algorithm for Missing Value Estimation for DNA Microarray Data, preprint 2005, submitted, (Arxiv...)

[8] S. Friedland, A. Niknejad and L. Chihara, A simultaneous reconstruction of missing data in DNA microarrays, to appear in *Linear Algebra and Its Applications*.

[9] S. Friedland, J. Nocedal and M. Overton, The formulation and analysis of numerical methods for inverse eigenvalue problems, SIAM J. Numer. Anal. 24 (1987), 634-667.

[10] A. Frieze, R. Kannan and S. Vempala, Fast Monte-Carlo algorithms for finding low rank approximations, *Proceedings of the 39th Annual Symposium on Foundation of Computer Science*, 1998.

[11] G.H. Golub and C.F. Van Loan, *Matrix Computation*, John Hopkins Univ. Press, 3rd Ed., 1996.

[12] R.A. Horn and C.R. Johnson, *Matrix Analysis*, Cambridge Univ. Press, 1987.

[13] R.A. Johnson, D. W. Wichern, Applied Multivariate Statistical Analysis, Prentice Hall, New Jersey, 4th edition (1998).

[14] H. Kim, G.H. Golub and H. Park, Missing value estimation for DNA microarray gene expression data: local least squares imputation, *Bioinformatics* 21 (2005), 187-198.

[15] S. Oba, M. Sato, I. Takemasa, M. Monden, K. Matsubara and S. Ishii, A Baesian missing value estimation method for gene expression profile data, *Bioinformatics* 19 (2003), 2088-2096.

[16] C.C. Paige and M. A. Saunders, Towards a generalized singular value decomposition, *SIAM J. Numer. Anal.* 18 (1981), 398-405.

[17] P.T. Spellman, G. Sherlock, M.Q. Zhang, V.R. Iyer, K. Anders, M.B. Eisen, P.O. Brown, D. Botstein and B. Futcher, Comprehensive identification of cell cycle-regulated genes of the yeast *Saccharomyces*

cerevisiae by microarray hybridization, Mol. Biol. Cell, **9** (1998), 3273-3297.

[18] G.W. Stewart, A method for computing the generalized singular value decomposition, *Matrix Pencils*, B. K**a**gström and A. Ruhe, *Lecture Notes in Mathematics*, 973 (1982), 207-220.