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


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

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

$S_1 := T^* T : U_1 \to U_1$, $S_2 := TT^* : U_2 \to U_2$.

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

$\sigma_1^2 \geq \ldots \geq \sigma_r^2 > 0$ positive eigenvalues of $S_1$ and $S_2$ and $r = \text{rank } T = \text{rank } T^*$. Let 

$S_1 v_i = \sigma_i^2 v_i$, $\langle v_i, v_j \rangle_1 = \delta_{ij}$, $i, j = 1, \ldots, r$. 

Define $u_i := \sigma_i^{-1} T v_i, i = 1, \ldots, r$. Then 

$\langle u_i, u_j \rangle_2 = \delta_{ij}, i, j = 1, \ldots, r$.

Complete \{v_1, \ldots, v_r\} and \{u_1, \ldots, u_r\} to orthonormal bases $[v_1, \ldots, v_{m_1}]$ and $[u_1, \ldots, u_{m_2}]$ in $U_1$ and $U_2$. 
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 $\langle x, y \rangle := y^* x$.

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

$U, V$ transition matrices from $[u_1, \ldots, u_m], [v_1, \ldots, v_n]$ to the standard bases in $\mathbb{C}^m, \mathbb{C}^n$ respectively.

For $k \leq r$ let $\Sigma_k = \text{diag}(\sigma_1, \ldots, \sigma_k) \in \mathbb{R}^{k \times k}$, and $U_k \in U(m, k), V_k \in U(n, k)$ 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 $A$:

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

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

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

$A_\nu$ 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 algorithms 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: 

$A \in \mathbb{R}^{m \times n}$, $m, n \gg 1$ needed a good approximation 

$A_k = \sum_{i=1}^{k} x_i y_i^T$, $x_i \in \mathbb{R}^m$, $y_i \in \mathbb{R}^n$, $i = 1, \ldots, k < \min(m, n)$.

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

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

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

Obtain $B_t$ from $B_{t-1}$ as above.
Lifting body original

Figure 1: Lifting body image $512 \times 512$. 
8 Lifting body compressed

Figure 2: 80-rank approximation of Lifting body image $512 \times 512$. 
9 SIMULATIONS 1

Figure 3: Convergence property of the Monte-Carlo method for Liftingbody image $(512 \times 512)$, $k = 80$. 
Figure 4: Convergence property of the Monte-Carlo method for random data matrix (3000 $\times$ 500).
### 11 COMPARISONS

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

<table>
<thead>
<tr>
<th>Data sets</th>
<th>Speed up</th>
<th>Re. ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cameraman($256 \times 256$), $k = 80$</td>
<td>1.145</td>
<td>1.083</td>
</tr>
<tr>
<td>Liftingbody ($512 \times 512$), $k = 100$</td>
<td>8</td>
<td>1.08</td>
</tr>
<tr>
<td>Map image($627 \times 865$), $k = 200$</td>
<td>3.33</td>
<td>1.067</td>
</tr>
<tr>
<td>Random matrix($8000 \times 200$), $k = 100$</td>
<td>42</td>
<td>1.1</td>
</tr>
</tbody>
</table>
Choosing columns of $A$

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

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 $c_i(A)$ according to new probabilities $\frac{||c_i(A-A_k)||^2}{||A-A_k||^2_F}$.

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

Given a metric space $X, d : X \times X \rightarrow \mathbb{R}_+$ and $\mathcal{X} := \{x_1, \ldots, x_n\} \subset X$ are $n$ distinct points, associate $M := (d(x_i, x_j))_{i,j=1}^n \in \mathbb{R}_{+}^{n \times n}$.

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

There are many different approaches to solve this problem.

To use SVD one can consider the matrix $F(\alpha, \beta) := (e^{-\alpha d(x_i, x_j)^\beta}) \in \mathbb{R}^{n \times n}$, $\alpha, \beta > 0$ Usually $\beta = 1, 2$. Fix $\beta$, e.g. $(\beta = 1)$.

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

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

$F(\infty, \beta) = I_n$ and $\mathcal{X}$ 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 $x_1, \ldots, x_n \in \mathbb{R}^r$

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 $A = (x_1, \ldots, x_n) \in \mathbb{R}^{r \times n}$. Let $\nu$ be the numerical rank of $A$. Let $y_1, \ldots, y_\nu \in \mathbb{R}^r$ be $\nu$ orthonormal eigenvectors of $B := A^T A$ corresponding to the first $\nu$ eigenvalues of $B$.

For each $i \in [1, \nu]$ cluster $y_i^T x_1, \ldots, y_i^T x_n \in \mathbb{R}$ as $\bigcup_{j=1}^{n_i} x_{i,j}$.

Use these $\nu$ clusters to obtain the final clustering.

For example intersect all the clusterings.
15 Generalized SVD

Let $A \in \mathbb{C}^{m \times n}$, $B \in \mathbb{C}^{l \times n}$. Then Van Loan 70s:

$A = F \Gamma R$, $B = G \Delta R$, $F \in U(m)$, $G \in U(l)$, $R \in GL(n, \mathbb{C})$, $\Gamma \in \mathbb{R}_+^{m \times n}$, $\Delta \in \mathbb{R}_+^{l \times n}$ diagonal matrices.

Numerical computations of GSVD are very unstable.

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

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

Reason: $T_A^*T_A + T_B^*T_B = I|_X \Rightarrow (T_A^*T_A)(T_B^*T_B) = (T_B^*T_B)(T_A^*T_A)$
Constructive way to obtain GSVD

\[ P = O \Omega^2 O^*, \; O \in U(n, r), \]
\[ \Omega = \text{diag}(\omega_1, \ldots, \omega_r), \; \omega_1 \geq \ldots \geq \omega_r > 0. \]

\[ Q_A := \Omega^{-1} O^* A^* A O \Omega^{-1}, \]
\[ Q_B := \Omega^{-1} O^* B^* B O \Omega^{-1} \in H(r). \]

As \( Q_A + Q_B = I_r, \)

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

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

**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

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

Choose random \( x_1, \ldots, x_{r+r_0} \in \mathbb{R}^n, \)

\( y_1, \ldots, y_{r_A} \in \mathbb{R}^m, z_1, \ldots, z_{r_B} \in \mathbb{R}^l \)

Then

\( A = \sum_{i=1}^{r_A} y_i x_i^T, \)

\( B = \sum_{i=1}^{r_0} z_i x_i^T + \sum_{i=r_0+1}^{r_B} z_i x_i^T + r_A - r_0 \)

We generated \( A_0 \in M_{8,7}(\mathbb{R}), B_0 \in M_{9,7}(\mathbb{R}) \) 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.
\[ A_0 = \]

\[
\begin{pmatrix}
1826 & 846 & 1516 & 1831 & 3060 & -577 & 1368 \\
-3452 & -1752 & -2182 & -2827 & -5970 & 119 & 3452 \\
5765 & 3573 & 745 & 2032 & 10755 & -246 & 1752 \\
-202 & -1818 & 7558 & 6964 & -2430 & 128 & 2182 \\
3873 & 1353 & 5193 & 5718 & 5955 & -911 & 2182 \\
-5206 & -2862 & -2306 & -3350 & -9270 & 196 & 5970 \\
-2060 & 1224 & -11470 & -11119 & -810 & -897 & 2827 \\
-2630 & -726 & -4390 & -4684 & -3810 & 48 & 3060
\end{pmatrix}
\]
\[ B_0 = \]
\[
\begin{pmatrix}
-3652 & -3486 & 640 & 2833 & -321 & 1424 \\
-8657 & -7471 & -2665 & 3283 & 1354 & 2669 \\
2420 & 2122 & 568 & -1063 & -289 & -776 \\
-3927 & -4161 & 2865 & 4833 & -1446 & 1899 \\
253 & -873 & 5837 & 4631 & -2952 & 895 \\
-4620 & -2044 & -11676 & -6664 & 5908 & -308 \\
2596 & 2388 & 20 & -1624 & -12 & -932 \\
-8624 & -7722 & -1180 & 4481 & 603 & 2908 \\
-7964 & -5438 & -10024 & -3195 & 5075 & 1176
\end{pmatrix}
\]
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 $A_0^T \mathbb{R}^8 \cap B_0^T \mathbb{R}^9$. 
The matrix $V \in \mathbb{M}_{7,3}(\mathbb{R})$ given by

\[
\begin{pmatrix}
9975.156 & 2218.778 & -16518.709 \\
4258.080 & 5446.091 & -13181.085 \\
9910.168 & -17951.929 & -10755.839 \\
11513.101 & -16136.565 & 1610.055 \\
16275.444 & 9076.818 & 5451.311 \\
-2894.442 & -3832.434 & 4134.750 \\
8306.914 & -8471.697 & -15231.563
\end{pmatrix}
\]
$U_1$ has the first two columns of $U \in M_{8,3}(\mathbb{R})$:

$$
\begin{pmatrix}
0.1797 & 0.0217 \\
-0.3322 & -0.0908 \\
0.5229 & 0.3627 \\
0.0653 & -0.5648 \\
0.4031 & -0.0979 \\
-0.4902 & -0.2086 \\
-0.3105 & 0.6860 \\
-0.2832 & 0.1293
\end{pmatrix}
$$
$W_1$ has the last two columns of $W \in \mathbb{M}_{9,3}(\mathbb{R})$

$$W_1 = \begin{pmatrix} -0.2125 & 0.2001 \\ -0.2093 & 0.5034 \\ 0.0709 & -0.1395 \\ -0.3819 & 0.2001 \\ -0.3995 & -0.0545 \\ 0.6105 & 0.3397 \\ 0.1176 & -0.1455 \\ -0.3124 & 0.4913 \\ 0.3408 & 0.5156 \end{pmatrix}.$$
Robustness of GSVD

Let $A := A_0 + X$, $B := B_0 + Y$, where $X \in M_{8,7}(\mathbb{R})$, $Y \in \mathbb{R}_{9,7}(\mathbb{R})$ with random entries and relatively small $\ell_2$ norm with respect to $\ell_2$ norms of $A$, $B$ respectively. $X$, $Y$ have integer entries in $[-99, 99]$. $X$ is

$$
\begin{pmatrix}
-14 & -73 & 65 & 3 & -14 & 16 & 9 \\
-10 & 8 & 90 & -94 & -22 & -24 & 0 \\
78 & 32 & -48 & -6 & 80 & -18 & -63 \\
32 & 9 & 41 & -95 & -28 & -90 & -63 \\
-23 & -72 & -84 & -84 & -58 & -37 & -40 \\
35 & 14 & -29 & 76 & -62 & -82 & -5 \\
18 & -40 & -51 & 11 & 87 & 66 & -46
\end{pmatrix}
$$
The singular values of $X, Y$ rounded off to three significant digits are:

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

So $\|X\| \sim 0.01\|A_0\|$, $\|Y\| \sim 0.01\|B_0\|$.

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

$(27490,\ 17450,\ 233,\ 130,\ 119,\ 70.0,\ 18.2)$,
$(29884,\ 19183,\ 250,\ 187,\ 137,\ 102,\ 19.7)$
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 $P = A_2^T A_2 + B_2^T B_2$ (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 $\nu = 3$.

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

$Q_{A_2} := \Omega^{-1} O^T A_2 A_2 O \Omega^{-1} = T \Phi^2 T^T,$

$Q_{B_2} := \Omega^{-1} O^T B_2 B_2 O \Omega^{-1} = T \Psi^2 T^T$ where $T \in O(3), V_2 = O \Omega T$ 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 $\nu = 4$. 

$P_4 = O\Omega O^T, O \in O(7, 4)$.

$Q_{A_2} := \Omega^{-1}O^TA_2^TA_2O\Omega^{-1} = T\Phi^2T^T,$

$Q_{B_2} := \Omega^{-1}O^TB_2^TB_2O\Omega^{-1} = T\Psi^2T^T$ where $T \in O(4), V_2 = O\Omega T$ and $U_2, W_2$ obtained from

$U_2\Phi = A_2O\Omega^{-1}T, W_2\Psi = B_2O\Omega^{-1}T$. 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 $A_0, B_0$: 

$(.9999796224, .6814701987, 0.005232470265),
(0.006383948621, .7318458638, .9999863106)$. 

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

What is the corresponding GSVD?

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

\[
P_r = O \Omega^2 O^*, \quad O \in U(n, r),
\]
\[
\Omega = \text{diag}(\omega_1, \ldots, \omega_r), \quad \omega_1 \geq \ldots \geq \omega_r > 0.
\]
\[
Q_i := \Omega^{-1} O^* A_i^* A_i O \Omega^{-1}, \quad i = 1, \ldots, k.
\]
\[
Q_i = T_i \Phi_i^2 T_i^*, \quad T = (t_{1,i}, \ldots, t_{r,i}) \in U(r),
\]
\[
\Phi_i = \text{diag}(\phi_{1,i}, \ldots, \phi_{i,r}),
\]
\[
\phi_{1,i} \geq \ldots \geq \phi_{r,i} \geq 0, \quad j = 1, \ldots, r,
\]
\[
\sum_{j=1}^{p} \sum_{i=1}^{k} \phi_{j,i}^2 \geq p, \quad p = 1, \ldots, r.
\]

The relative importance of the \( j \)-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 the \( j \)-th function in \( A_i \) with respect to the \( j \)-th function in \( A_l \) is given by \( \frac{\phi_{j,i}^2}{t_{j,i}^* A_l t_{j,i}} \).
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 $T \subset \{1, \ldots, n\} \times \{1, \ldots, m\}$ missing entries set.

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

Let $\mathcal{X}$ be all $X = (x_{ij}) \in \mathbb{R}^{m \times n}$ where $x_{ij} = 0$ if $(i, j) \not\in T$.

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

\[
(1) \quad \min_{X \in \mathcal{X}} \sum_{i=\nu+1}^{n} \sigma_i^2 (A + X) = \min_{X \in \mathcal{X}} \sum_{i=\nu+1}^{n} \lambda_i ((A + X)^T(A + X))
\]
Fixed Rank Approximation Algorithm: [8]

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

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

Flow chart of the algorithm:

---

Fixed Rank Approximation Algorithm (FRAA)

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

Output: an approximation $G_{iter}$ of $G$.

for $p = 0$ to $iter - 1$
- Compute $A_p := G_p^T G_p$ and find an orthonormal set of eigenvectors for $A_p, v_{p,1}, \ldots, v_{p,m}$.
- $G_{p+1}$ 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 $G^T G$ 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.*
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.
Figure 6: Comparison of NRMSE against percent of missing entries for three methods: IFRAA, BPCA and LLS. Data set was a $2000 \times 20$ 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.
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.

<table>
<thead>
<tr>
<th>Data sets</th>
<th>IFRAA</th>
<th>LLS</th>
<th>BPCA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cdc15 data set %0.81 missing</td>
<td>0.0175</td>
<td>0.0200</td>
<td>0.0216</td>
</tr>
<tr>
<td>Evolution data set %9.16</td>
<td>0.0703</td>
<td>0.0969</td>
<td>0.1247</td>
</tr>
<tr>
<td>Calcineurin data set %3.68</td>
<td>0.0421</td>
<td>0.0445</td>
<td>0.0453</td>
</tr>
</tbody>
</table>
References


