CS754 Assignment 4

Hitesh Kumar Punna, Sudhansh Peddabomma

190050093, 190050118

Question 1.

code for this question is in the folder code ,the results are in the folder images.

Method Used: (This method is implemented in the file Divide.m .)
We used the algorithm "Alternate Minimization" for the reconstruction of signals f1 and f2 from "f". It is done in the following way:-
Objective Function:

$$E(\theta_1, \theta_2) = ||f - A_1\theta_1 - A_2\theta_2||_2$$

$|s.t|\theta_1 \leq T_0$ —and— $\theta_2 \leq T_0$
Where,
$f1 = A_1\theta_1|s.t|A_1 = \phi\Psi_1$
$|s.t|$ A1 is DCT matrix of 256x256 &$|\theta_1|$is $|T_0|$sparse
$f2 = A_2\theta_2$ $|s.t|$ $A_1 = \phi\Psi_2$
Ψ_1 1d-DCT Matrix, $\Psi_1 \in R^{256\times256}$
Ψ_2 Identity Matrix, $\Psi_1 \in R^{256\times256}$
As we know that by C.S theory identity matrix is highly *incoherent* with DFT.
So here, ϕ is *Identity Matrix*, $\phi \in R^{256\times256}$
Therefore,
$A_1 = \Psi_1$, $A_2 = \Psi_2$
Algorithm:- For finding θ_1&θ_2, we will use OMP with T_0.
Initialize: θ_2 to some random value
Do the following till convergence (ith iteration)

$$f' = f - A_2 * \theta_2^i$$

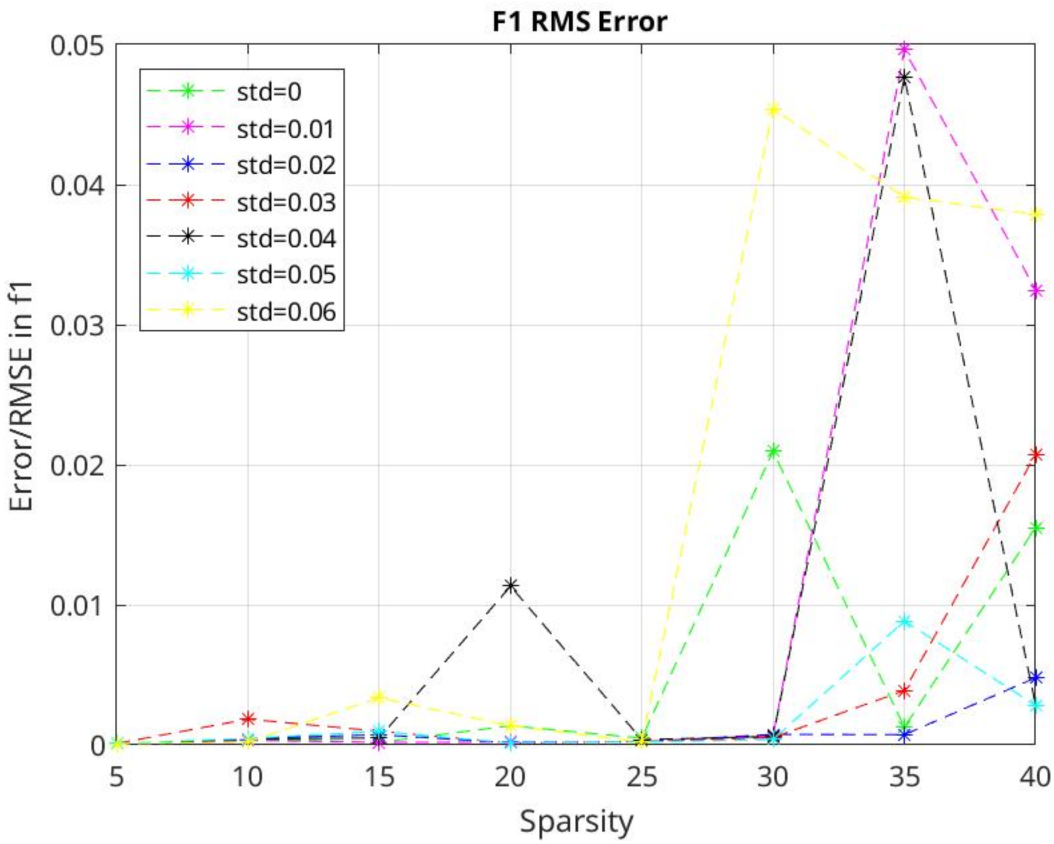
$$\theta_1^{i+1} = OMP(f', A_1, T_0)$$

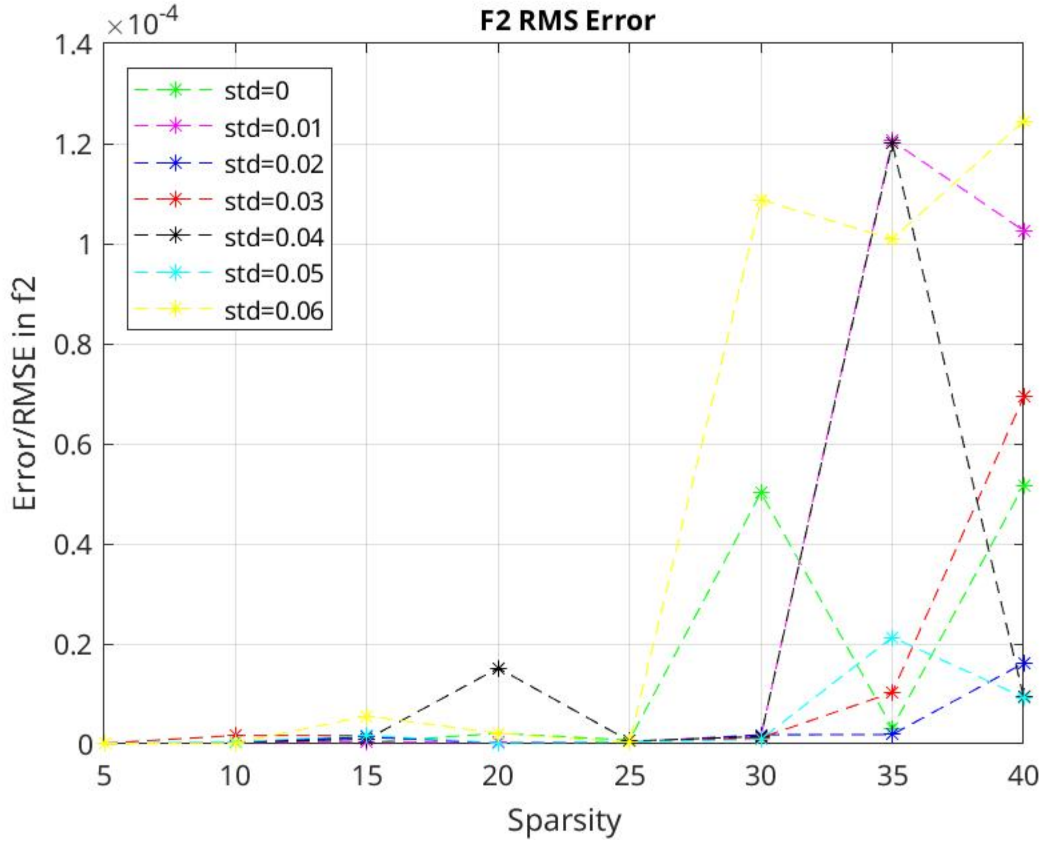
$$f'' = f - A_1 * \theta_1^{i+1}$$

$$\theta_2^{i+1} = OMP(f'', A_2, T_0)$$

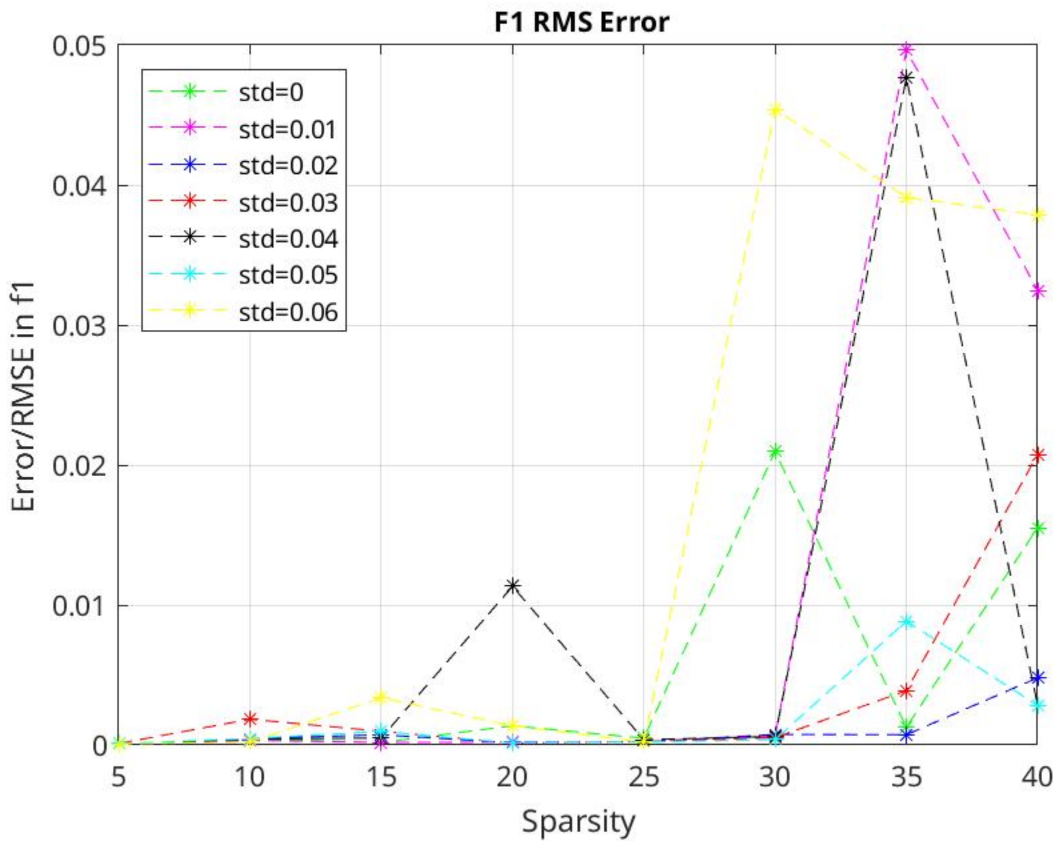
In this way we will find the reconstructed values for θ_1 and θ_2 from these two we will reconstructf_1 and f_2.
The corresponding plots we got are:-

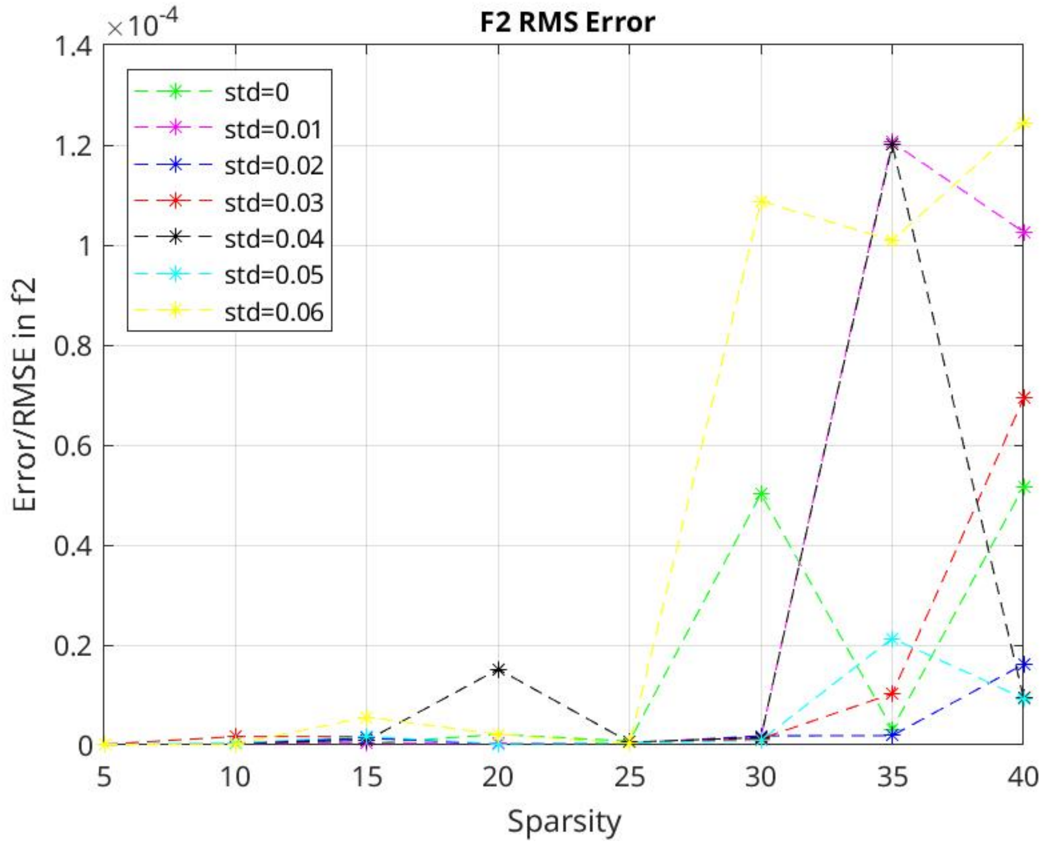
(a) We plotted the graph for sparsity values in the set $\begin{bmatrix} 5 & 10 & 15 & 20 & 25 & 30 & 35 \end{bmatrix}$ and we varied the standard deviation(σ) over the set $\{0, 0.01, 0.02, 0.03, 0.04, 0.05, 0.06\}$ and using the average value of $f_1 + f_2$
Inference: We can see from the graph that on Increasing "Sigma" the RMSE values of both f_1, f_2 increases. This is expected as the reconstruction quality decreases as noise increases.



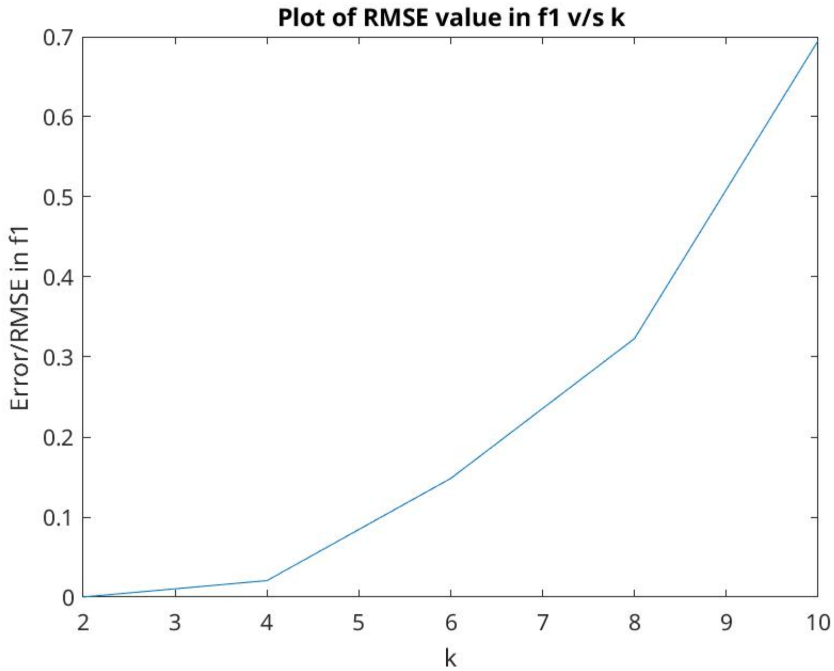


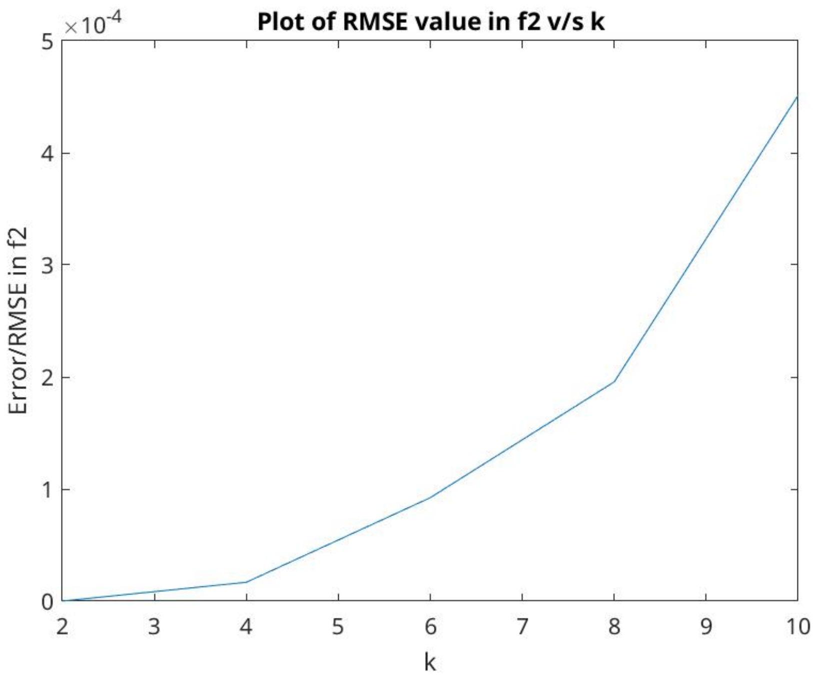
(b) We infer the observations from the same graphs as before.
and we varied the sparsity(s) over the set $\{5, 10, 15, 20, 25, 30, 35, 40\}$ and used the average value of $f_1 + f_2$ to determine the noise
Inference: We can see from the graph that on Increasing "Sparsity" the RMSE values of f_1 wont get affected much in the start when the noise is low . We see a sudden increase and decrease (due to random effects) with higher sparsity.
Increasing the "Sparsity" the RMSE values of f_2 is also increasing


F1 RMS Error
std=0
std=0.01
std=0.02
std=0.03
std=0.04
std=0.05
std=0.06
Error/RMSE in f1
Sparsity
0.05
0.04
0.03
0.02
0.01
0
5
10
15
20
25
30
35
40



c) We plotted the graph for fixed sparsity value,sparsity= 5
and we varied 'k' over the set $\{2, 4, 6, 8, 10\}$
Inference: We can see from the graph that on Increasing "k" the RMSE values of both f_1, f_2 increases.


Plot of RMSE value in f1 v/s k
Error/RMSE in f1
k
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
2
3
4
5
6
7
8
9
10


Plot of RMSE value in f2 v/s k
$\times 10^{-4}$
Error/RMSE in f2
k
0
1
2
3
4
5
2
3
4
5
6
7
8
9
10

Question 2.

Title: CoSaMP: Iterative signal recovery from incomplete and inaccurate samples
Link: Here
Author: D. Needell , J.A. Tropp
Published on: 22 July 2008
Published at : Department of Mathematics, University of California at Davis
Code :

Algorithm 1. CoSaMP recovery algorithm

CoSaMP($\boldsymbol{\Phi}$, $\boldsymbol{u}$, s)
Input: Sampling matrix $\boldsymbol{\Phi}$, noisy sample vector $\boldsymbol{u}$, sparsity level s
Output: An s-sparse approximation $\boldsymbol{a}$ of the target signal

```
a^0 ← 0                         {Trivial initial approximation}
v ← u                           {Current samples = input samples}
k ← 0

repeat
  k ← k + 1
  y ← Φ* v                      {Form signal proxy}
  Ω ← supp(y_2s)                {Identify large components}
  T ← Ω ∪ supp(a^{k-1})         {Merge supports}

  b|_T ← Φ_T^† u                {Signal estimation by least-squares}
  b|_{T^c} ← 0

  a^k ← b_s                     {Prune to obtain next approximation}
  v ← u − Φ a^k                 {Update current samples}

until halting criterion true
```

For the halting we criteria we will find

$$v = ||x - x_s||_2 + \frac{1}{\sqrt{s}}||x - x_s||_1 + ||e||_2$$

and the halting criterion is $||v||_2 \leq \epsilon$, ϵ is some constant derived from bounds of noise vector 'e'.

Error Bound :

$$||x - a||_2 \leq C.max\{\eta, \frac{1}{\sqrt{s}}||x - x_{s/2}||_1 + ||e||_2\}$$

- s = Sparsity of the signal

- Reconstruction is done over the equation $u = \Phi x + e$
- e = noise in the measurement $e \in R^m$
- u = vector of samples of an arbitrary signal x measured using the sensing matrix ϕ
- x = true signal / vector, $x \in R^N$
- m = number of measurements
- a = an s-sparse approximation vector estimated value using CoSaMp reconstruction
- Φ = m $\times$ N sampling matrix with restricted isometry constant $\delta_{2s} \leq c$.
- $x_{s/2}$ is a best (s/2)-sparse approximation to x
- η = A given precision parameter (constant value taken from various observations)

Time Complexity : $O(L \log(||x||_2/\eta))$
Here 'L' , bounds the cost of a matrix– vector multiply with Φ or $\Phi*$. Working storage is O(N).

Question 3.

Dictionary $\mathcal{D}$ sparsely represents the class of images $\mathcal{S}$.
Let I be the image vector and it is given by the equation $I = \mathcal{DS}$

(a) Class $\mathcal{S}_1$ obtained by applying derivative filter on $\mathcal{S}$
Let δ be the derivative kernel. Then for image I in $\mathcal{S}$ can be converted to an image I' in $\mathcal{S}_1$ with the following equation

$$I' = \delta * I$$

where $*$ represents the convolution operation. Since this is a convolution operation we can represent the above equation in the following way.

$$I' = AI$$

Where the matrix "A" can be derived from the δ kernel, and as we can see that the matrix A will only depend on δ it's not affected by the dictionary matrix.
From the above equations we can say that

$$I' = AI = AD\theta$$

$$I' = B\theta, B = AD$$

θ is the sparse vector, from this we can say that new learned dictionary matrix B($B = AD$) can be used for images in class $\mathcal{S}_1$

(b) Class $\mathcal{S}_2$ which consists of images obtained by rotating a subset of the images in class $\mathcal{S}$ by a known fixed angle α, and the other subset by another known fixed angle β.
As we know Rotation of Matrices can be expressed by Matrix multiplication(using Rotation Matrix), we can do the following,

$$A_{rotated} = R_\theta A$$

where $A_{rotated}$ is the matrix in the rotated frame, A is the original matrix, R_θ is the Rotation matrix when rotated with angle θ, and we know that Rotation matrix R_θ depends only on θ.
Now, let $\mathcal{S}_2^1$ be the Class of Images when rotated with angle α.
$\mathcal{S}_2^2$ be the Class of Images when rotated with angle β.
Let I^1 the images in the class $\mathcal{S}_2^1$,I^2 the images in the class $\mathcal{S}_2^2$, from the previous equations we can say that,

$$I^1 = D_\alpha\theta = R_\alpha D\theta^1$$

$$I^2 = D_\beta\theta = R_\beta D\theta^2$$

Now the for any image I' in set S_2 it should belong Union of sets S_2^1, S_2^2, so we can express I' in the following way

$$I' = I^1 + I^2$$

$$I' = R_\alpha D\theta^1 + R_\beta D\theta^2$$

$$I' = \begin{bmatrix} R_\alpha D & R_\beta D \end{bmatrix} \begin{bmatrix} \theta^1 \\ \theta^2 \end{bmatrix}$$

We know that θ_1, θ_2 are sparse vectors, from this we can say that $\begin{bmatrix}\theta^1 \\ \theta^2\end{bmatrix}$ will also be a sparse vector for the Image I'
From this we can that the matrix $\begin{bmatrix}R_\alpha D & R_\beta D\end{bmatrix}$will be the Dictionary matrix for the Class of Images $\mathcal{S}_2$

(c) Class $\mathcal{S}_3$ is obtained by applying the intensity transformation

$$I^i_{new}(x,y) = \alpha(I^i_{old}(x,y))^2 + \beta(I^i_{old}(x,y)) + \gamma$$

to the images I in $\mathcal{S}$, where α, β, γ are known constants.

Let I be the vectorized form of an image in $\mathcal{S}$. Now we know that,

$$I = \mathcal{D}\theta$$

where θ is sparse and $\mathcal{D}$ has size $n \times k$. Since, we are dealing with squared intensities, we can write the following,

$$I(i) = \sum_j^k \mathcal{D}_{ij}\theta_j$$

where k is the dimension of sparse vector θ. Now,

$$I^2(i) = \sum_j^k \sum_w^k \mathcal{D}_{ij}\theta_j\mathcal{D}_{iw}\theta_w = \sum_j^k \sum_w^k (\mathcal{D}_{ij}\mathcal{D}_{iw})\theta_j\theta_w$$

where $\mathcal{D}_i$ is the ith row of dictionary $\mathcal{D}$. This gives an idea of constructing a dictionary of the following form. Consider $\mathcal{F}$ of size $n \times k^2$ defined as follows

$$\mathcal{F}_i = \texttt{vectorized}(D_i^T D_i)$$

for $i \in [n]$. D_i has the dimension $1 \times k$. Therefore, $D_i^T D_i$ has the dimension $k \times k$ and $\texttt{vectorized}(D_i^T D_i)$ has the size $k^2 \times 1$. Notice that,

$$I^2 = \mathcal{F}\theta'$$

where $\theta' = \texttt{vectorized}(\theta^T\theta)$. As θ is sparse, θ' is also sparse. Now, construct the dictionary $\mathcal{D}'$ of size $n \times (k^2 + k + 1)$ that represents images in $\mathcal{S}_3$ sparsely in the following way

$$\mathcal{D}' = \begin{bmatrix}\alpha\mathcal{F} & \beta\mathcal{D} & \gamma\mathcal{U}\end{bmatrix}$$

where $\mathcal{U}$ is a column vector of size $n \times 1$ with all entries as 1. Any image I' in $\mathcal{S}_3$ can be represented as

$$I' = \begin{bmatrix}\alpha\mathcal{F} & \beta\mathcal{D} & \gamma\mathcal{U}\end{bmatrix} \cdot \begin{bmatrix}\theta' \\ \theta \\ 1\end{bmatrix} = \alpha\mathcal{F}\theta' + \beta\mathcal{D}\theta + \gamma\mathcal{U} = \alpha(I^2) + \beta(I) + \gamma$$

Therefore, as $\begin{bmatrix}theta' & \theta & 1\end{bmatrix}^T$ is sparse, $\mathcal{D}'$ represent $\mathcal{S}_3$ sparsely.

(d) Class $\mathcal{S}_4$ consists of images obtained by applying a known blur kernel to the images in Class $\mathcal{S}$. Let δ' be the blur kernel. Then for image I in $\mathcal{S}$ can be converted to an image I' in $\mathcal{S}_4$ with the following equation

$$I' = \delta' * I$$

where $*$ represents the convolution operation. Since this is a convolution operation we can represent the above equation in the following way.

$$I' = A'I$$

Where the matrix "A'" can be derived from the δ' kernel, and as we can see that the matrix A' will only depend on δ' it's not affected by the dictionary matrix.
As we know that blur filter can be expressed as matrix multiplication, we can say that

$$I' = A'I = AD\theta$$

$$I' = B\theta, B = A'D$$

θ is the sparse vector, from this we can say that new learned dictionary matrix B ($B = A'D$) can be used for images in class $\mathcal{S}_4$

(e) Class $\mathcal{S}_5$ consists of images obtained by applying a blur kernel which is known to be a linear combination of blur kernels belonging to a known set B, to the images in Class $\mathcal{S}$. Let $B = \{\delta_1, \delta_2, \delta_3, ..., \delta_n\}$, I_0 be the original Image and $\mathcal{I}$ be the Image after applying blur kernel. and the blur kernel formed using linear combination of elements in set $\mathcal{B}$ be δ'

$$\delta' = a_1\delta_1 + a_2\delta_2 + a_3\delta_3 + ... + a_n\delta_n$$

here $a_1, a_2, ..., a_n$ are constants.

$$I = \delta' * I_0$$

where $*$ represents the convolution operation.
As $*$ and $+$ are linear operators we can represent the previous equation in given form.

$$I = a_1\delta_1 * I_0 + a_2\delta_2 * I_0 + a_3\delta_3 * I_0 + ... + a_n\delta_n * I_0$$

And we can express each $\delta_i * I_0, i \in \{1, 2, 3, ..., n\}$ as some A_iI_0 where A_i is a matrix used to represent the kernel δ_i. (i.e $\delta_i * I_0 = A_iI_0, i \in \{1, 2, 3, ..., n\}$)
Therefore,

$$I = a_1A_1I_0 + a_2A_2I_0 + a_3A_3I_0 + ... + a_nA_nI_0$$

$$I = (a_1A_1 + a_2A_2 + a_3A_3 + ... + a_nA_n)I_0$$

as $I_0 = D\theta$ where θ is a sparse vector, D is the dictionary matrix.
we can see that

$$I = (a_1A_1 + a_2A_2 + a_3A_3 + ... + a_nA_n)I_0 = ((a_1A_1 + a_2A_2 + a_3A_3 + ... + a_nA_n)D)s$$

$$I = B\theta, B = (a_1A_1 + a_2A_2 + a_3A_3 + ... + a_nA_n)D$$

from this we can say that new learned dictionary matrix B ($B = (a_1A_1 + a_2A_2 + a_3A_3 + ... + a_nA_n)D$) can be used for images in class $\mathcal{S}_5$

Question 4.

1. Minimize $J(A_r) = \|A - A_r\|_F^2$, where A_r is a rank-r matrix.
   We know A and we are trying to find its rank-r approximation that minimizes the Frobenius norm of the difference.

   Eckart Young Theorem.
   Let A be an $m \times n$ matrix. To find the rank $r(r < \min n, m)$ matrix A_r which is closest to A wrt to the Frobenius norm of the difference, construct A_r in the following manner.

   A can be written as $A = U \cdot S \cdot V^T$ using Singular Value Decomposition. U is an orthonormal matrix of size $m \times m$. V is an orthonormal matrix of size $n \times n$. S is an $m \times n$ diagonal matrix with diagonal entries equal to the singular values of A. WLOG, assume the singular values are in decreasing order of magnitude (i.e, $S(1,1)$ is the largest singular value of A). Let S be the set of values corresponding to the r largest singular values of A.

   Now, $A_r = U_r \cdot S_r \cdot V_R^T$. U_r is an $m \times r$ matrix with left singular values corresponding to S. V_r is an $n \times r$ matrix with right singular values corresponding to S. S_r is an $r \times r$ matrix with diagonal entries from S.

   Therefore, $A_r = U_r \cdot S_r \cdot V_R^T$ minimizes $J(A_r)$.

   This optimization is needed in the KSVD algorithm. We try to find the rank 1 approximation of one of the intermediate matrices (E_k) while learning the dictionary. The equation that we try to minimize is

   $$\|Y - AS\|_F^2 = \left\| Y - \sum_{j=1}^{K} a_j s^j \right\|_F^2 = \left\| Y - \sum_{j \neq k}^{K} a_j s^j - a_k s^k \right\|_F^2 = \left\| E_k - a_k s^k \right\|_F^2$$

   where S is the coefficient vector and A is the dictionary. We obtain

   $$E_k = U \Lambda V^T, a_k = u_1, s^k = \lambda(1,1) v_1^T$$

   The dictionary learnt through this method is popularly used in Image denoising, Image inpainting, Image deblurring, Blind compressive sensing etc.

2. Minimize $J(R) = \|A - RB\|_F^2$ where $A, B \in \mathbb{R}^{n \times m}, R \in \mathbb{R}^{n \times n}, m > n$ and R is orthonormal.

   We are trying to obtain $\min_R \|A - RB\|^2$ s.t. $R \cdot R^T = I$. This is very similar to what we have done in Method 3 of Dictionary Learning - Union of Orthonormal bases. Let $R^* = \min_R \|A - RB\|^2$ s.t. $R \cdot R^T = I$. Now,

   $$\min_R \|A - RB\|^2 = \min_R \texttt{trace}((A - RB)^T \cdot (A - RB)) = \min_R \texttt{trace}(A^T A - 2A^T RB + B^T B)$$

   Notice that, we need to maximise the middle term in the above equation.

   $$R^* = \max_R \texttt{trace}(A^T RB) = \max_R \texttt{trace}(RBA^T)$$

   as $\texttt{trace}(FG) = \texttt{trace}(GF)$. Now, we can write $BA^T = Q = UDV^T$ using Singular Value Decomposition. Hence,

   $$R^* = \max_R \texttt{trace}(RUDV^T) = \max_R \texttt{trace}(V^T RUD) = \max_R \texttt{trace}(Z(R)D)$$

where $Z(R) = V^T RU$

$$R^* = \max_A \sum_i z_{ii} d_{ii} \leq \sum_i d_{ii}$$

as $Z(R) \cdot Z(R)^T = I$. Now, the maximum is achieved when $Z(R) = I$. Therefore,

$$V^T RU = I \implies R = VU^T$$

Therefore, $R = VU^T$ minimizes $J(R)$ where, $BA^T = USV^T$ using SVD.

This optimization is required in the Union of Orthonormal bases method of Dictionary Learning. That is, we want to find an over-complete dictionary which has the form $D = [D_1|D_2|...|D_n]$ where D_i's are orthonormal bases. In one of the intermediate steps, we have to solve the following

$$X_m = X - \sum_{j \neq m} A_j S_j$$

$$S_m X_m^T = U \Lambda V^T$$

$$A_m = VU^T$$

S is the coefficient matrix of the form $\begin{bmatrix} S_1 & S_2 & ... & S_M \end{bmatrix}^T$ and A is an over-complete dictionary of the form $\begin{bmatrix} A_1 & A_2 & ... & A_M \end{bmatrix}$. We obtain A_m using the above method and the equation in the 2nd line is related to the Orthogonal Proscrustes problem.

Question 5.

1. The aim of Hyperspectral Unmixing is to identify the materials present in the captured scene, as well as their compositions, by using high spectral resolution of hyperspectral images. The equation model is given by

$$y[n] = \sum_{i=1}^{N} a_i s_i[n] + \nu[n] = As[n] + \nu[n]$$

for $n = 1, ..., L$, where each $a_i \in \mathbb{R}^M$, $i = 1, .., N$ is called an *endmember signature vector*, which contains the spectral components of a specific material (indexed by i) in the scene; N is the number of endmembers and $A \in R^{M\times N}$ is the endmember matrix; $s_i[n]$ describes the contribution of material i at pixel n; $s[n] = [s_1[n], ..., s_N[n]] \in \mathbb{R}^N$ is called the *abundance vector* at pixel n; L is the number of pixels; and $\nu[n] \in \mathbb{R}^M$ is noise. The goal os unmixing is to solve

$$\hat{s}[n] = arg \min_{s[n]\in S} \|y[n] - As[n]\|_2^2$$

for $n = 1, ..., L$ and S is the feasible set of abundance vectors.

2. NMF is an approach to solve blind HU. NMF is a very flexible formulation for blind HU. Now, we can model the blind HU problem as an NMF optimization of the form

$$\min_{A,S\geq 0} \|Y - AS\|_F^2$$

NMF can be used to solve $\min_{A,S\geq 0} \|Y - AS\|_F^2$. In blind HU, the connection is that the NMF factors obtained, A and S, can serve as estimates of the endmembers and abundances, respectively as they are non-negative by nature. As discussed in the later parts of the paper, we can modify NMF to include regularizers g and h. The abundance regularizer h usually follows the design principle of sparsity. For example, in Dictionary Learning, $\min_{A,S\geq 0} \|Y - AS\|_F^2 + \mu \|S\|_{1,1}$. In DL, the dictionary size is often set to be large, and should be larger than the true number of endmembers; the number of endmembers is determined by the row sparsity of $S = \|S\|_{row-0}$. More formally, we consider

$$\min_{A\geq 0, S\in S^L} \|Y - AS\|_F^2 + \lambda \cdot g(A) + \mu \cdot \|S\|_{row-0}$$

3. NMF can be used to solve $\min_{A,S\geq 0} \|Y - AS\|_F^2$. However, this does not guarantee uniqueness and the optimization schemes are rather pragmatic. This is a serious issue to blind HU as the NMF solution may not be necessarily true. Therefore, we modify the optimization criterion as

$$\min_{A\geq 0, S\in S^L} \|Y - AS\|_F^2 + \lambda \cdot g(A) + \mu \cdot h(S)$$

where $S^L = S|s[n] \geq 0, 1^T s[n] = 1, 1 \leq n \leq L$, g and h are regularizers. Different works use different g, h. For example, the volume constrained NMF uses

$$\min_{A\geq 0, S\in S^L} \|Y - AS\|_F^2 + \lambda \cdot vol(B)$$

where $vol(B)$ is the simplex volume corresponding to A, in which $b_i = C^i(a_i - d)$ for all i. As $vol(B)$ is non-convex, ICE uses $g(A) = \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} \|a_i - a_j\|_2^2$ and $h(S) = 0$. Another example of modified NMF is inspired from Dictionary Learning. From an NMF based blind HU perspective, we can use row-sparsity to provide joint endmember number, endmember and abundance estimation. That is, consider SPICE, which uses

$$\min_{A \geq 0, S \in S^L} \|Y - AS\|_F^2 + \lambda \cdot \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} \|a_i - a_j\|_2^2 + \mu \cdot \sum_{i=1}^{N} \gamma_i \left\|s^i\right\|_1$$

That is, $g(A) = \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} \|a_i - a_j\|_2^2$ and $h(S) = \sum_{i=1}^{N} \gamma_i \|s^i\|_1$. Here, the number of columns of A, given by N, is chosen to be a number greater than the true number of endmembers and we use $\|S\|_{row-0}$ to represent the endmember number. In $h(S)$, γ_i represent weights that are iteratively updated; this regularizer is a convex surrogate of $\|S\|_{row-0}$. Similarly, CoNMF also aims at row sparsity, using a nonconvex surrogate of $h(S) = \sum_{i=1}^{K} \|s^i\|_2^p$, $0 < p \leq 1$. Some more examples of g and f are given below,

Algorithm	$g(A)$	$h(S)$
DL[60]	0	$\|S\|_{1,1}$
$L_{1/2}$-NMF [71]	0	$\|S\|_{1/2,1/2}^{1/2}$