Practical Data Science
Assignment #1

Shikhar Vashishth

M.Tech CSA - 13374

Problem 1

Part (a):

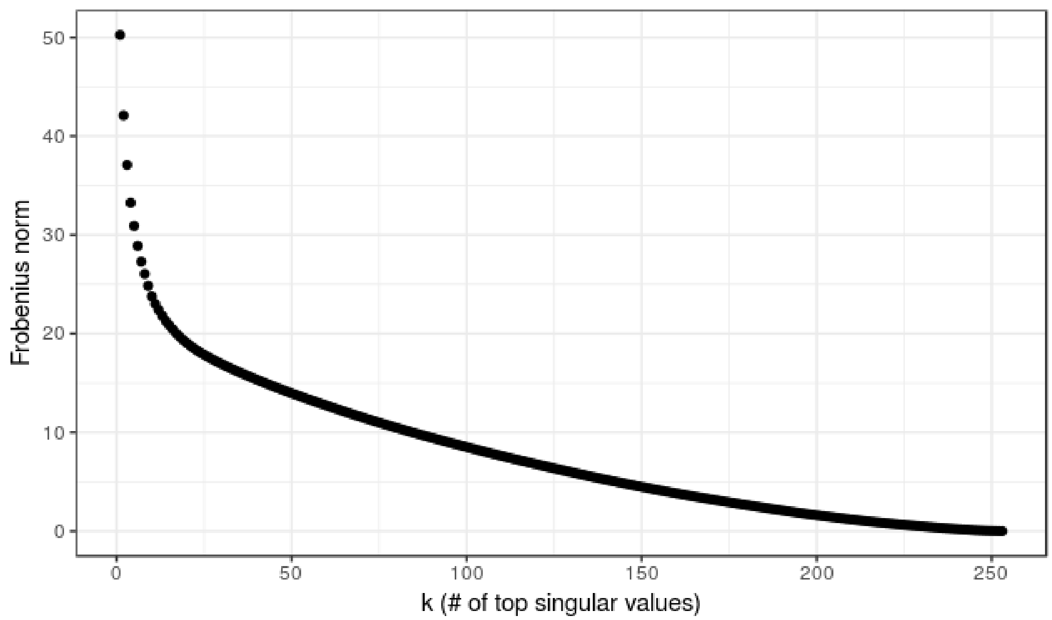

Figure 1: Plot of $||X - Z_k||_F$ vs k

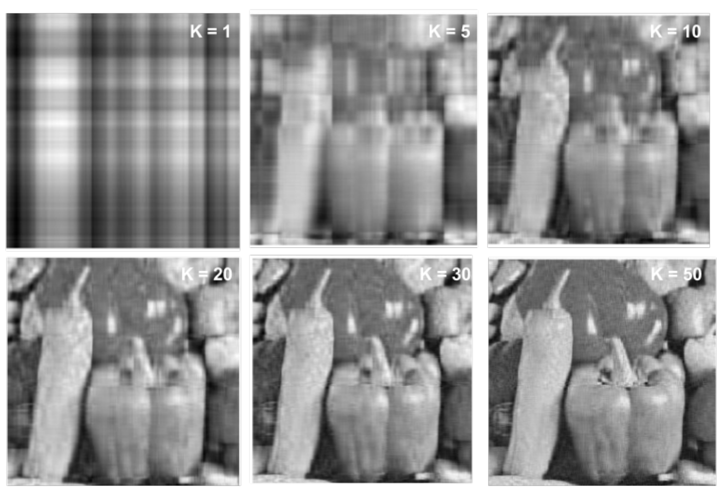

Figure 2: Approximation of image with different number of top singular values

Justification: We can see clearly from the above plot and images that as we increase the values of k, the number of top singular values used for approximating the original image, the quality of image improves. The plot was generated using R (ggplot2 library).

Part (b):

Time Comparison:

Data density %	CPU Time(sec) - SVT (250 iters)	CPU Time(sec) ISVD
10	4.1304	3.0991
20	3.9124	3.2581
30	4.2556	3.0166
40	4.1700	3.1476
50	4.1667	3.2274
60	4.1765	3.2299
70	4.3894	3.1999
80	4.1236	3.3031
90	4.1697	3.0940

Frobenius norm comparison ($||X_{org} - X_{apprx}||_F$):

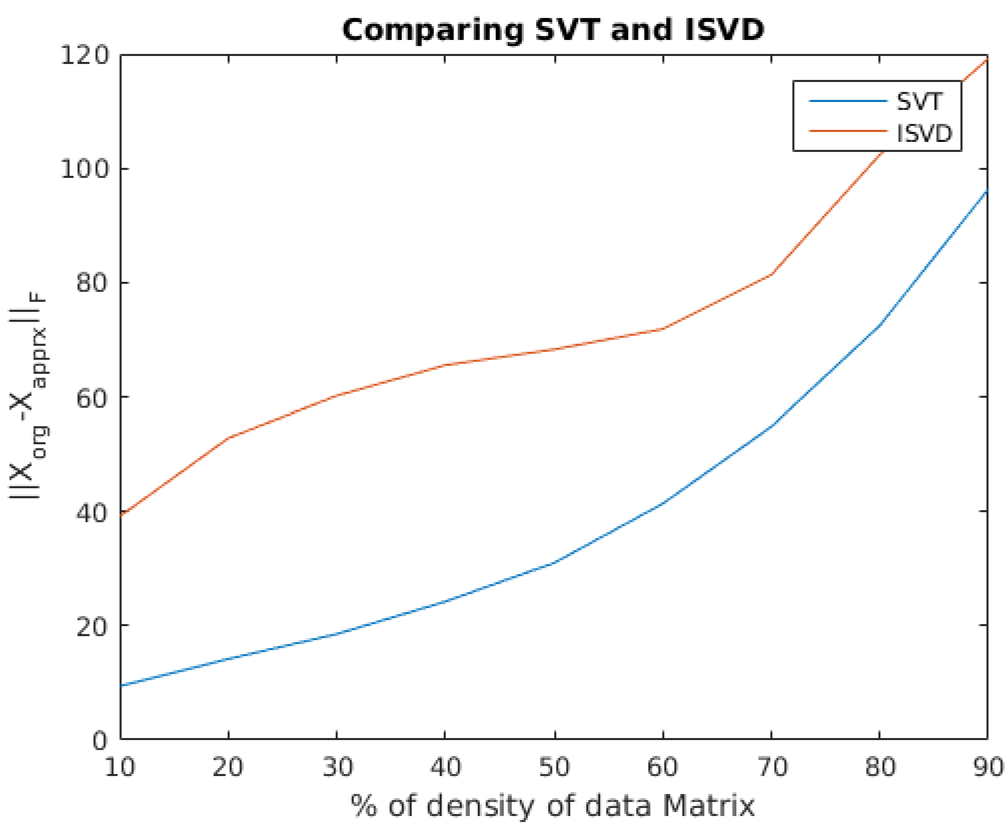

Figure 3: Comparing SVT and ISVD algorithms

Visual Comparison:

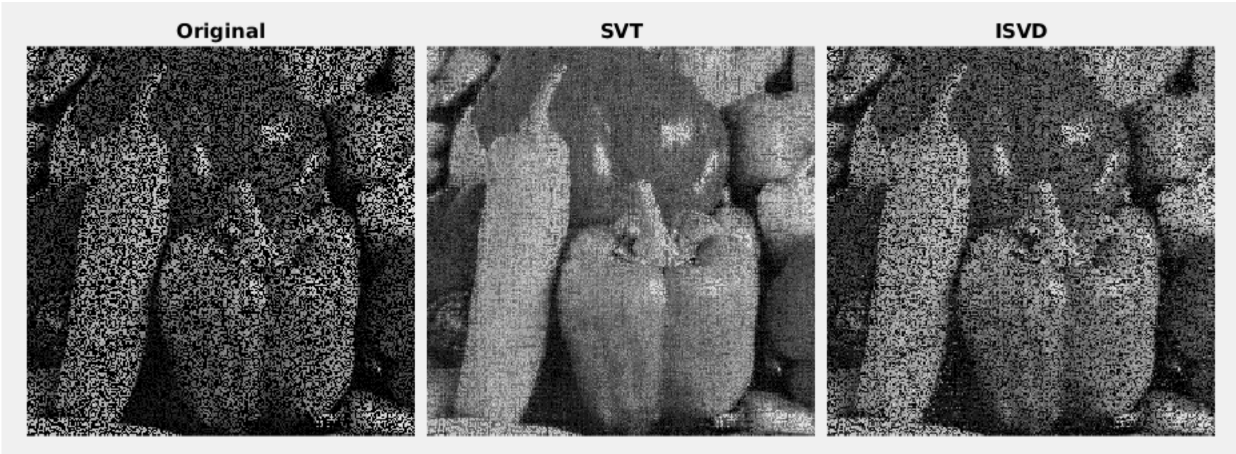

Figure 4: Comparing SVT and ISVD result for data density 50 %

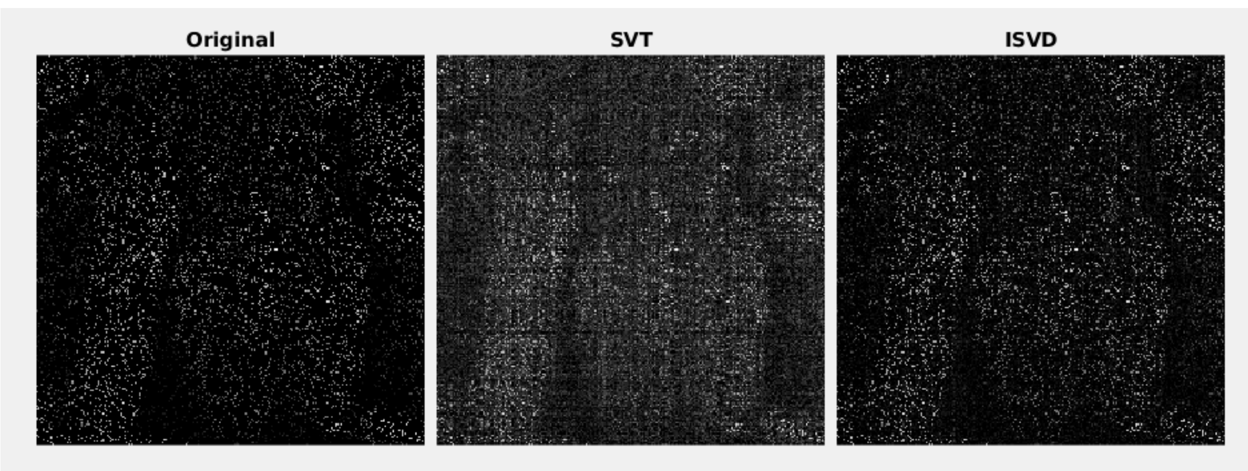

Figure 5: Comparing SVT and ISVD result for data density 10 %

Justification: From the given results we can conclude that although Singular value thresholding algorithm takes more time compared to ISVD algorithm but the performance of SVT is much better than ISVD. The frobenius norm of difference ($||X_{org} - X_{apprx}||_F$) clearly reveals this fact. The visual results also support this observation from the figure 4 and 5 we can see that how SVT algorithm's results are much superior than that of ISVD algorithm's results.

Problem 2

Formulation (A)

$$\begin{aligned} \underset{W,H}{\text{minimize}} \quad & \frac{1}{2}||X - WH||_F^2 + \frac{\tau}{2}(||W||_F^2 + ||H||_F^2) \\ \text{subject to} \quad & W, H \geq 0 \end{aligned}$$

Formulation (B)

$$\begin{aligned} \underset{W,H}{\text{minimize}} \quad & \frac{1}{2}||X - WH||_F^2 + \tau(||W||_1 + ||H||_1) \\ \text{subject to} \quad & W, H \geq 0 \end{aligned}$$

Formulation (C)

$$\begin{aligned} \underset{W,H}{\text{minimize}} \quad & \sum_{i,j} \left(X_{ij} \log \frac{X_{ij}}{(WH)_{ij}} - X_{ij} + (WH)_{ij} \right) + \frac{\tau}{2}(||W||_F^2 + ||H||_F^2) \\ \text{subject to} \quad & W, H \geq 0 \end{aligned}$$

Results:

	τ	Iters	CPU Time(sec)	Spar. % of W	Spar. % of H	$\|\|X - WH\|\|_F^2$
Form (A) Random	1000	100	1.427	-	-	-
	1	100	0.708	38.80	17.11	45.90
	.001	100	0.785	40.84	22.70	45.78
Form (A) NNDSVD	1000	100	1.700	-	-	-
	1	100	0.907	37.14	15.18	45.84
	.001	100	0.828	41.74	24.72	45.78
Form (B) Random	1000	100	0.781	-	-	-
	1	100	0.797	-	-	-
	.001	100	0.816	43.24	29.70	45.83
Form (B) NNDSVD	1000	100	0.933	-	-	-
	1	100	0.840	-	-	-
	.001	100	0.875	44.16	26.32	45.78
Form (C) Random	1000	50	791.28	54.85	42.55	46.05
	1	50	790.55	55.77	42.47	46.02
	.001	50	800.31	53.43	29.14	46.51
Form (C) NNDSVD	1000	50	854.63	55.18	34.75	46.50
	1	50	802.68	54.35	53.19	45.94
	.001	50	790.88	54.32	53.21	45.94

Sparsity is calculated as:

$$Sparsity = \frac{\text{\# of Zero entries}(< .000001)}{\text{Total entries}}$$

Formulation (A): Random Intialization

$\tau = 1000 \rightarrow$ No meaningful result
$\tau = 1 \rightarrow$

W_1	W_2	W_3	W_4	W_5
growth	mobile	yukos	mr	film
economy	people	said	labour	best
economic	music	russian	election	england
sales	said	oil	blair	game
year	digital	court	brown	win
said	phone	company	party	said
2004	technology	gazprom	said	won
prices	users	mr	howard	year
bank	broadband	firm	government	wales
rate	phones	russia	minister	play
business	**tech**	**business**	**politics**	**entertainment**

$\tau = 0.001 \rightarrow$

W_1	W_2	W_3	W_4	W_5
growth	mr	mobile	film	england
economy	labour	people	best	game
said	blair	music	awards	win
bank	election	said	award	wales
sales	brown	digital	actor	ireland
year	party	technology	oscar	cup
economic	said	phone	actress	said
oil	government	users	festival	team
prices	howard	broadband	won	play
2004	minister	software	films	players
business	**politics**	**tech**	**entertainment**	**sports**

Justification We can easily see that how the algorithm varies with different values of τ, when using $\tau = 1000$ the algorithm didn't give any meaningful results and even when τ was made 1000 the topics predicted were not complete (sports category missing). But with $\tau = .001$ the algorithm gave the desired results.

Formulation (A): NNDSVD Intialization

$\tau = 1000 \rightarrow$ No meaningful result
$\tau = 1 \rightarrow$

W_1	W_2	W_3	W_4	W_5
mr	england	growth	film	mobile
labour	game	economy	best	people
blair	win	said	awards	music
election	wales	year	award	said
brown	ireland	bank	actor	digital
party	said	sales	oscar	technology
said	cup	economic	actress	phone
government	team	oil	festival	users
howard	play	2004	won	broadband
minister	players	prices	films	software
politics	**sports**	**business**	**entertainment**	**tech**

$\tau = 0.001 \rightarrow$

W_1	W_2	W_3	W_4	W_5
mr	england	growth	film	mobile
labour	game	economy	best	people
blair	win	said	awards	music
election	wales	bank	award	said
brown	ireland	year	actor	digital
party	cup	sales	oscar	technology
said	said	economic	actress	phone
government	team	oil	festival	users
howard	play	prices	films	broadband
minister	players	2004	won	software
politics	**sports**	**business**	**entertainment**	**tech**

Justification The NNDSVD intialization greatly improved the performance of the algorithm, we can see clearly that although with random initialization and $\tau = 1$ the algorithm didn't give correct results but with NNDSVD intialization it gave the desired results for both values of τ (1 and .001) and there is improvement in frobenius norm as well.

Formulation (B): Random Intialization

$\tau = 1000 \rightarrow$ No meaningful result
$\tau = 1 \rightarrow$ No meaningful result
$\tau = 0.001 \rightarrow$

W_1	W_2	W_3	W_4	W_5
mobile	film	growth	mr	yukos
people	best	economy	labour	said
music	england	economic	election	russian
digital	game	sales	blair	court
phone	win	year	brown	company
technology	said	said	party	oil
said	won	prices	said	gazprom
broadband	year	2004	howard	law
users	wales	bank	government	firm
phones	play	rate	minister	mr
tech	**sports**	**business**	**politics**	**business**

Formulation (B): NNDSVD Intialization

$\tau = 1000 \rightarrow$ No meaningful result
$\tau = 1 \rightarrow$ No meaningful result
$\tau = 0.001 \rightarrow$

W_1	W_2	W_3	W_4	W_5
mr	england	growth	film	mobile
labour	game	economy	best	people
blair	win	said	awards	music
election	wales	bank	award	said
brown	ireland	year	actor	digital
party	cup	sales	oscar	technology
said	said	economic	actress	phone
government	team	oil	festival	users
howard	play	prices	films	broadband
minister	players	2004	won	software
politics	**sports**	**business**	**entertainment**	**tech**

Justification With formulation (B), the gradient descent approach didn't perform well with τ value as 1000 and 1, for both of them it gave meaningless results. But with τ value as .001 the algorithm gave the desired results. With NNDSVD intializaiton the frobenius norm value decreased.

Formulation (C): Random Intialization

$\tau = 1000 \rightarrow$

W_1	W_2	W_3	W_4	W_5
music	england	said	film	mr
said	game	growth	best	said
people	team	economy	awards	labour
users	wales	market	award	election
software	ireland	year	won	blair
games	said	bank	said	party
band	players	company	actor	government
online	injury	sales	year	people
technology	chelsea	oil	oscar	brown
microsoft	rugby	china	star	minister
tech	**sports**	**business**	**entertainment**	**politics**

$\tau = 1 \rightarrow$

W_1	W_2	W_3	W_4	W_5
england	mr	mobile	film	said
game	said	people	best	growth
win	government	music	awards	sales
cup	labour	digital	award	economy
said	election	phone	band	year
club	blair	technology	said	2004
match	party	said	year	bank
team	brown	games	star	prices
wales	minister	tv	album	market
injury	tax	broadband	festival	economic
sports	**politics**	**tech**	**entertainment**	**business**

$\tau = 0.001 \rightarrow$

W_1	W_2	W_3	W_4	W_5
said	game	mr	film	mobile
economy	england	said	best	people
growth	win	labour	awards	said
government	said	blair	band	technology
mr	cup	party	music	music
year	play	election	award	digital
bank	match	government	album	users
economic	team	people	said	software
oil	club	howard	star	phone
market	players	minister	actor	net
business	**sports**	**politics**	**entertainment**	**tech**

Formulation (C): NNDSVD Intialization

$\tau = 1000 \rightarrow$ No meaningful result

W_1	W_2	W_3	W_4	W_5
mr	film	said	game	people
said	best	growth	england	mobile
labour	awards	market	win	said
election	award	economy	said	technology
blair	music	year	cup	users
government	band	company	club	software
party	star	bank	match	music
brown	album	sales	team	digital
minister	festival	oil	injury	computer
people	actor	firm	players	games
politics	**entertainment**	**business**	**sports**	**tech**

$\tau = 1 \rightarrow$

W_1	W_2	W_3	W_4	W_5
mr	film	said	england	people
said	best	growth	game	mobile
labour	awards	economy	win	said
election	award	bank	said	technology
blair	music	market	club	software
government	band	company	cup	users
party	star	year	match	digital
brown	year	sales	team	games
minister	said	oil	injury	phone
people	album	china	players	music
politics	**entertainment**	**business**	**sports**	**tech**

$\tau = 0.001 \rightarrow$

W_1	W_2	W_3	W_4	W_5
mr	film	said	england	people
said	best	growth	game	mobile
labour	awards	economy	win	said
election	award	bank	said	technology
blair	music	market	club	software
government	band	company	cup	users
party	star	year	match	digital
brown	year	sales	team	games
minister	said	oil	injury	phone
people	album	china	players	music
politcs	**entertainment**	**business**	**sports**	**tech**

Justification With formulation (C), the algorithm gave the desired results for all values of τ (1000, 1, .001). The sparsity of computed W and H matrices was found to be much better as compared to the other two formulations. But in terms of computation time the algorithm is the worst.

Conclusion: From all the formulations the KL-divergence formulation is the most stable, as it gives the desired results for all values of τ. Moreover, the sparsity of the generated W and H matrices also improves with this formulation. But the compuational time required for it is considerably higher as compared to other two formulations.