1. (a) ay^2 + bxy + cx + dy + e = x^2

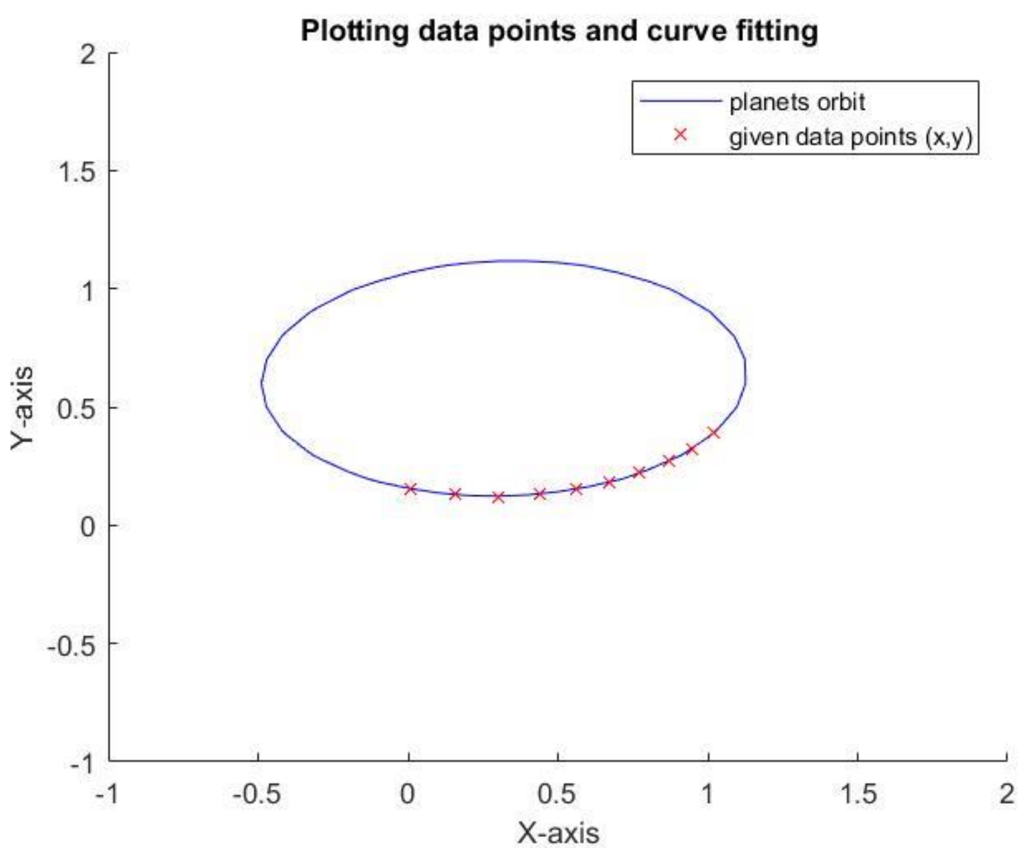
<u>Formulate a linear regression</u>:

A * (Coeff) = B
where A= [y^2 xy x y ones(size(x))]
Coeff = [a b c d e] ^T
B= x^2

Coeff = A\B (solves the system of linear equations A*x = B in mat lab)

(b) Coeff = [-2.6356
0.1436
0.5514
3.2229
-0.4329]

(c)



2. (a.) (i)

   1/3 + 1/3 = 2/3
   2/3 + 2/3 = 4/3
   1/3 + 2/3 = 3/3
   2/5 + 2/5 = 4/5
   3/5 + 1/5 = 4/5

   Column1 + Column2 = Column3

   The linear combination of first and second column gives third column. i.e. third column is linearly dependent. Also, observed first and second columns are linearly independent.

   Therefore, rank=2

   (ii) Computed the SVD

   Now, observing the diagonal values of Matrix S,

$$S = \begin{bmatrix} 2.5987 & 0 & 0 \\ 0 & 0.3682 & 0 \\ 0 & 0 & 0.0000 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

   Even taking into account double precision floating point error (using eps function in mat lab) we can observe the last column of S to be all zeros.

   Therefore, rank of the matrix A= 2

   (iii) Created a Diagonal matrix of 2000X2000 with each entry= (0.9) ^n
   By, using mat lab with precision 10^-16 we found rank(D) = 625
   That means after 625 the value of (0.9) ^n will be very negligible i.e.

   (0.9) ^626, (0.9) ^627 …. (0.9) ^2000 are all negligible and approximated to zero.

   Hence rank of the Diagonal Matrix, D= 276

2.(b)(i)

The new coefficients are going to change each time we execute the program. Since we are adding noise to each coordinate randomly distributed on the interval [−0.005, 0.005].

<u>For a particular instance:</u>

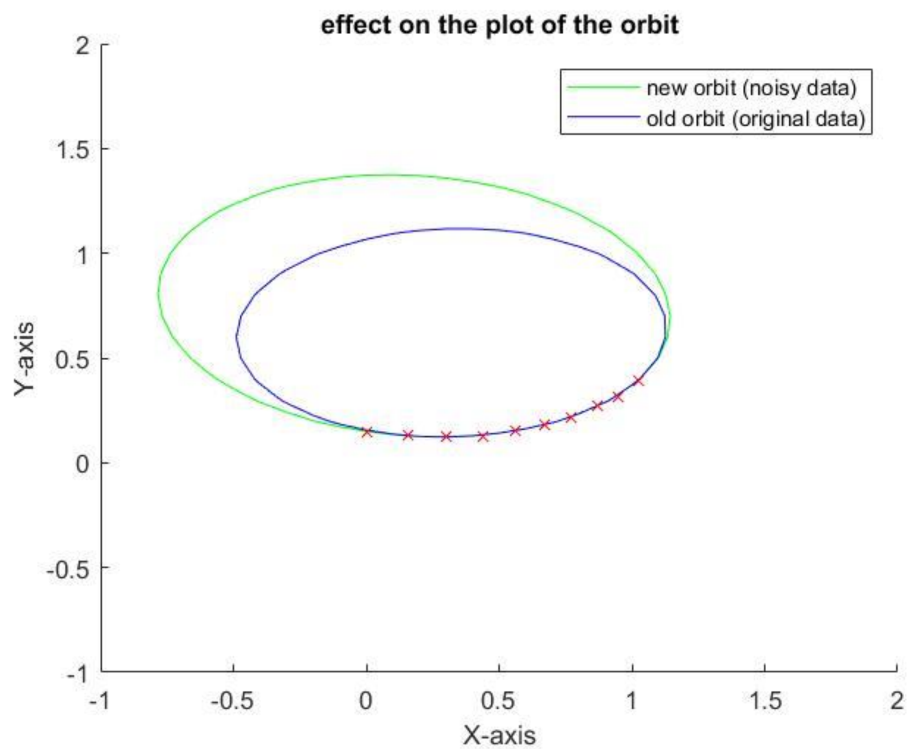
New coefficients = [-2.4737
-0.2792
0.5929
3.6411
-0.4946]

Old coefficients = [-2.6356
0.1436
0.5514
3.2229
-0.4329]

Compare coefficients = New coefficients – Old coefficients
= [0.1619
-0.4228
0.0415
0.4182
-0.0617]

Though the differences in the noisy data and the original data are very small, the coefficients obtained are very sensitive to perturbations. There is a considerable change in the value of the coefficients.

Plot:



The resulting trajectory is another ellipse which also fit the perturbed data very well.

Even the plot changes for each execution of the program (Because the coefficients change each time). So, for a particular instance, we can observe the above graph.

ay^2 + bxy + cx + dy + e = x^2 where new coefficients, a=-2.4737; b=-0.2792; c= 0.5929; d=3.6411; e=-0.4946

The green Curve represents the new orbital formed by the noise added data
Whereas the blue curve represents the old orbital (problem 1) of the planet.

new_x = 1.0212; 0.9544; 0.8685; 0.7691; 0.6748; 0.5645; 0.4418; 0.3049; 0.1627; 0.0084
new_y = 0.3912; 0.3244; 0.2685; 0.2191; 0.1848; 0.1545; 0.1318; 0.1249; 0.1327; 0.1484

The new data points (new_x, new_y) are plotted on the graph.

We can observe,

If new coefficients > old coefficients, then the size of new orbital (noisy data) is larger than the old orbital (original data).

If new coefficients < old coefficients, then the size of new orbital (noisy data) is smaller than the old orbital (original data).
If new coefficients are approximately equal to old coefficients, then the size of new orbital (noisy data) is almost equal to the old orbital (original data).

2.(b) (II) solving the problem, for both the original and the noisy data, using a low rank approximation based on the SVD. Since we are adding noise to each coordinate randomly distributed on the interval [−0.005, 0.005]. The coefficients get changed each time we execute the program.

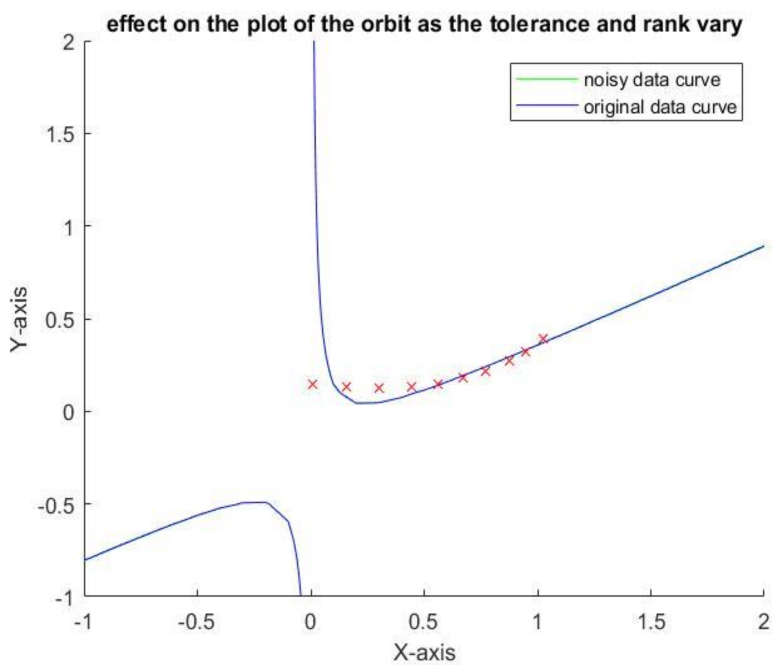
<u>For a particular instance:</u>

<u>Coefficients_SVD_originaldata</u>	**<u>Coefficients_SVD_noisydata</u>**
Rank = 1, tolerance=10^-1 0 1.8211 0.4072 0 -0.0574	Rank = 1, tolerance =10^-1 0 1.8408 0.3992 0 -0.0559
Rank = 2, tolerance =10^-2 0 -1.3036 0.7713 3.2736 -0.4955	Rank = 2, tolerance =10^-2 0 -1.7476 0.7885 3.8555 -0.5630
Rank = 3,4,5; tolerance =10^-3, 10^-4, 10^-5 -2.6356 0.1436 0.5514 3.2229 -0.4329	Rank = 3,4,5; tolerance =10^-3, 10^-4, 10^-5 -3.1416 0.0495 0.5244 3.7016 -0.4785

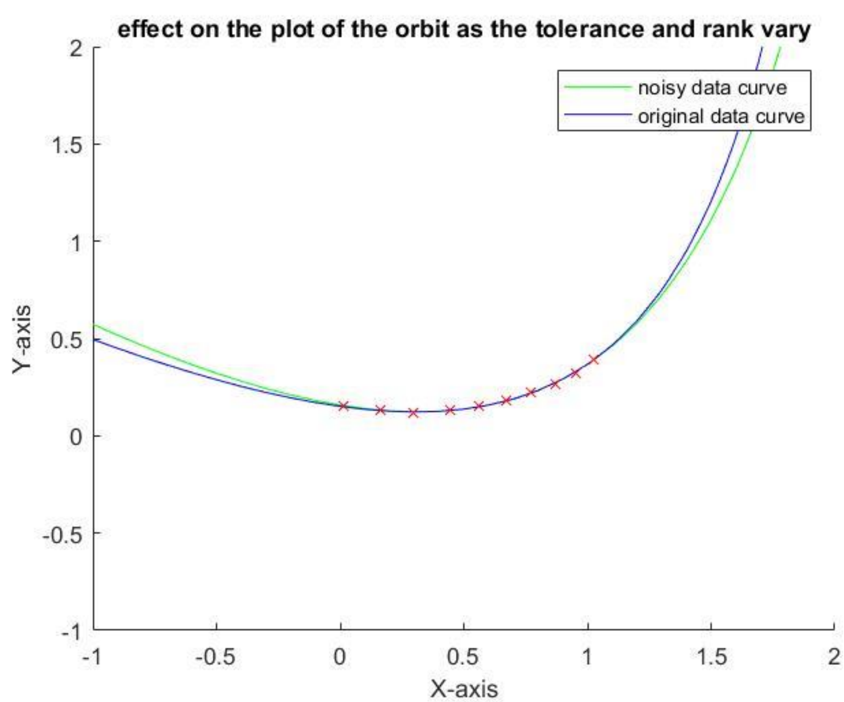
<u>Comparing Coefficients:</u> Results are similar in nature when compared with the original data. Althought the results obtained are different for the same rank approximation.

Plot of the orbits as the rank and tolerance varies:

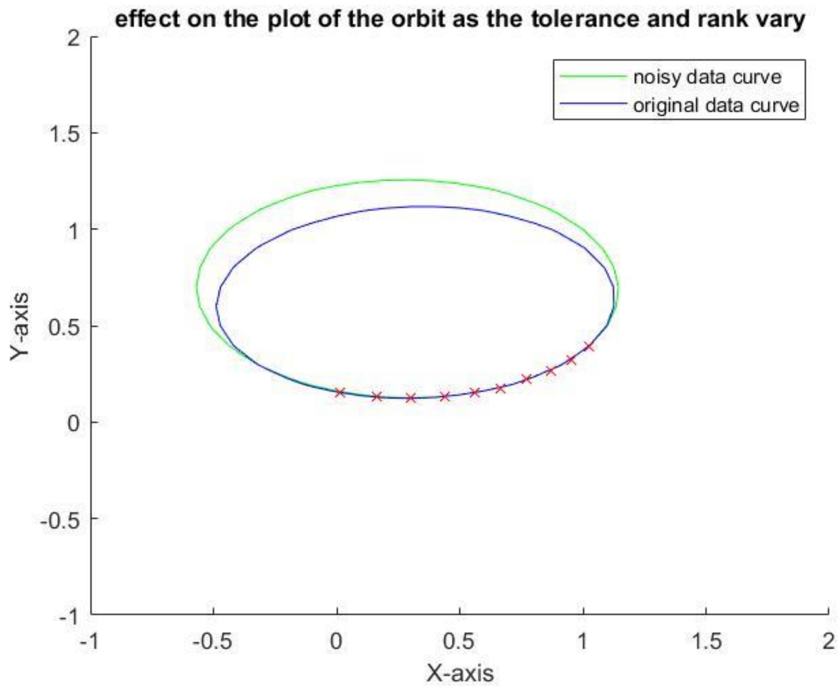
For rank=1, tolerance=10^-1



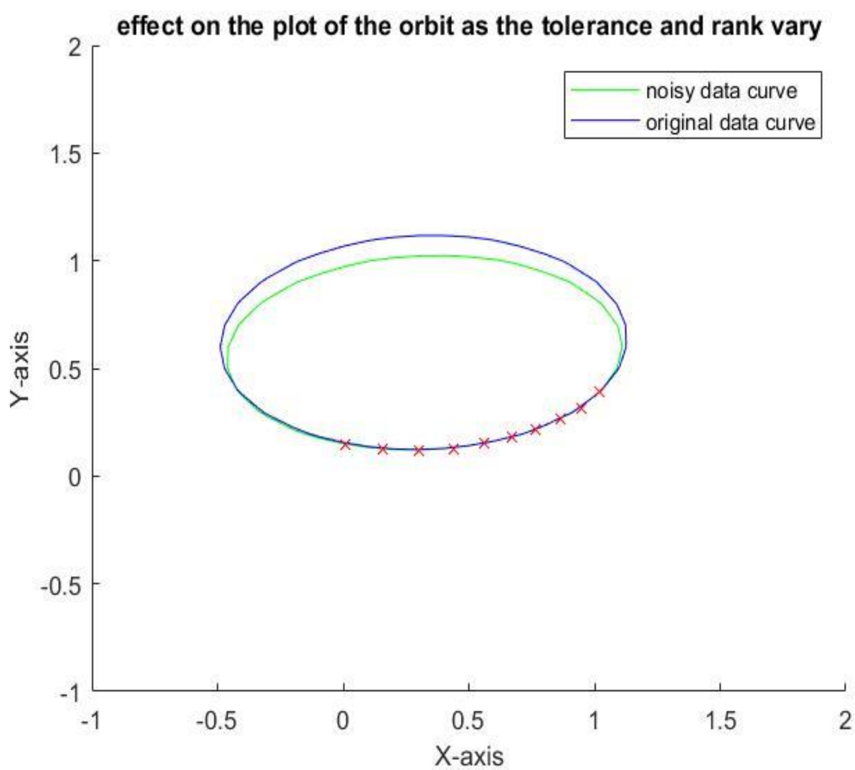
For rank=2, tolerance=10^-2



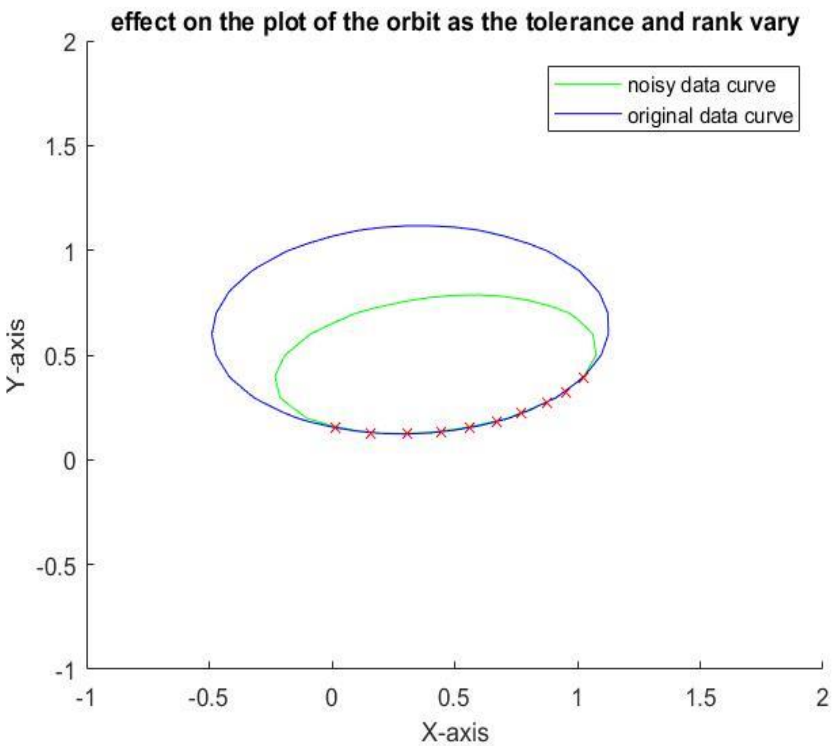
For rank=3, tolerance=10^-3



For rank=4, tolerance=10^-4



For rank=5, tolerance=10^-5



We can observe, low-ranking approximations i.e. rank=1 & rank=2, do not give parameters that describe elliptical trajectories and their trajectories do not pass near the data points. They are not good approximations for the problem.

Also observe for rank=3,4,5 gives exactly the same set of coefficients as before for original data. Whereas for noisy data the coefficients change but the trajectories are different ellipses that fit the data points well.

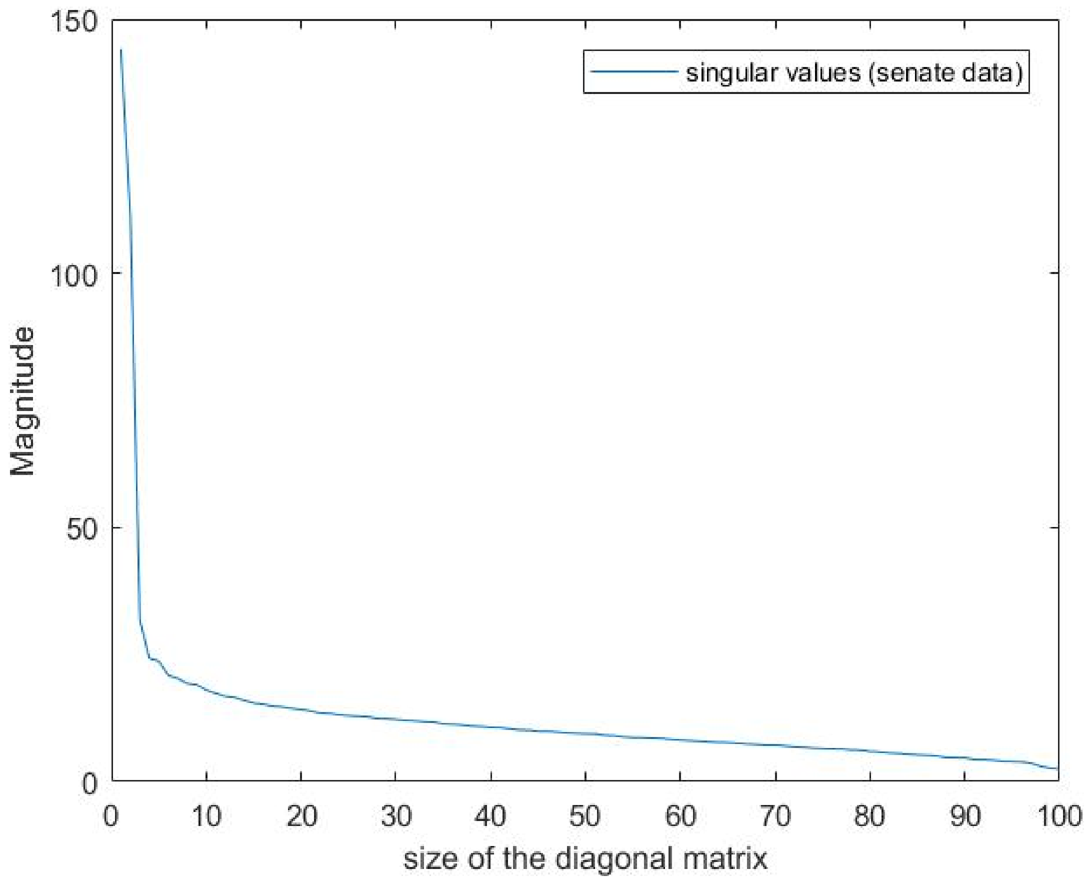
From all the above graphs we can observe, that the graph with rank=3 and tolerance =10^-3 is the best low rank approximation.

I would consider one that is less sensitive to small perturbations in the data as a better solution. Because despite of the noise added they are forming different trajectories that fit the data points well.

3.)

<u>SENATE DATA:</u>

b.)



The singular values or the diagonal values get reduced to zero as the size of the matrix increases.

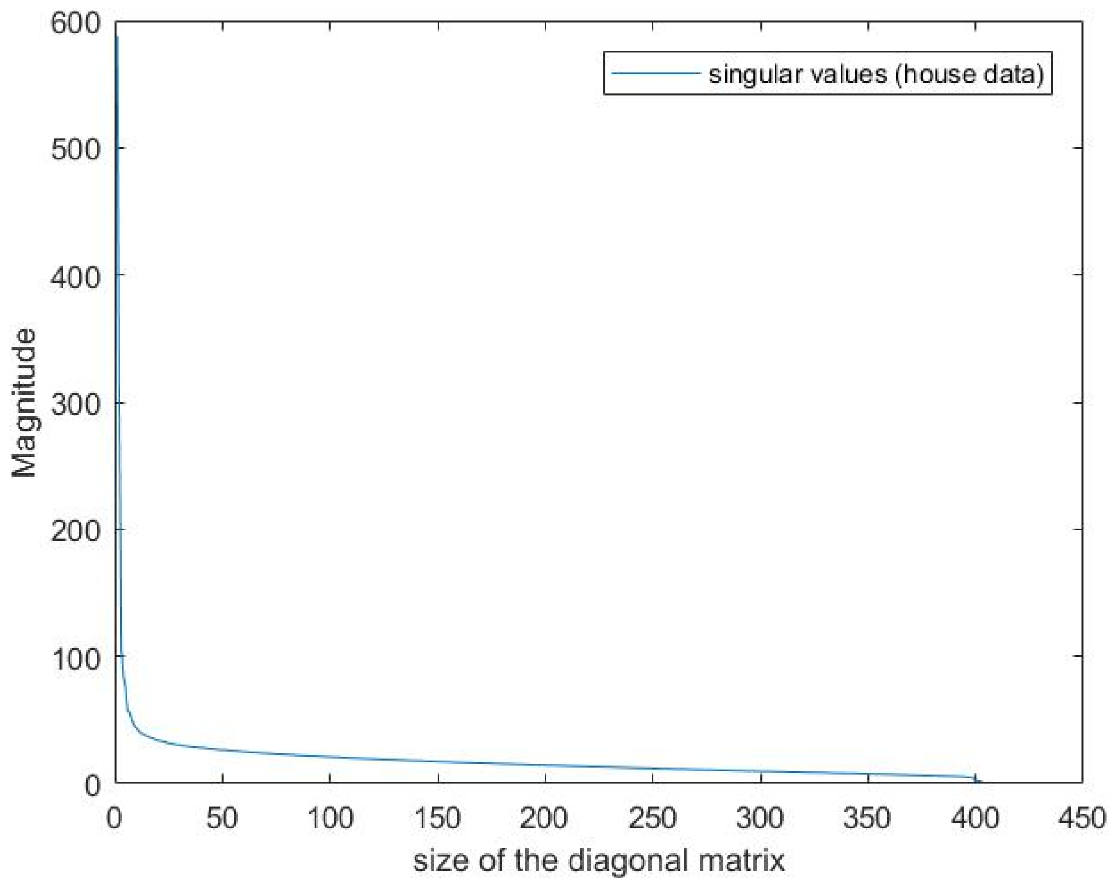
At first the magnitude of the singular values is 145 when the size of the matrix is (1,1).

When the size of the matrix is (10,10) the magnitude reaches 20 and then

As the size of the matrix reaches (100,100) the magnitude of the singular value approximately approaches zero.

3 (f)

HOUSE DATA:



The singular values or the diagonal values get reduced to zero as the size of the matrix increases and reaches 400.

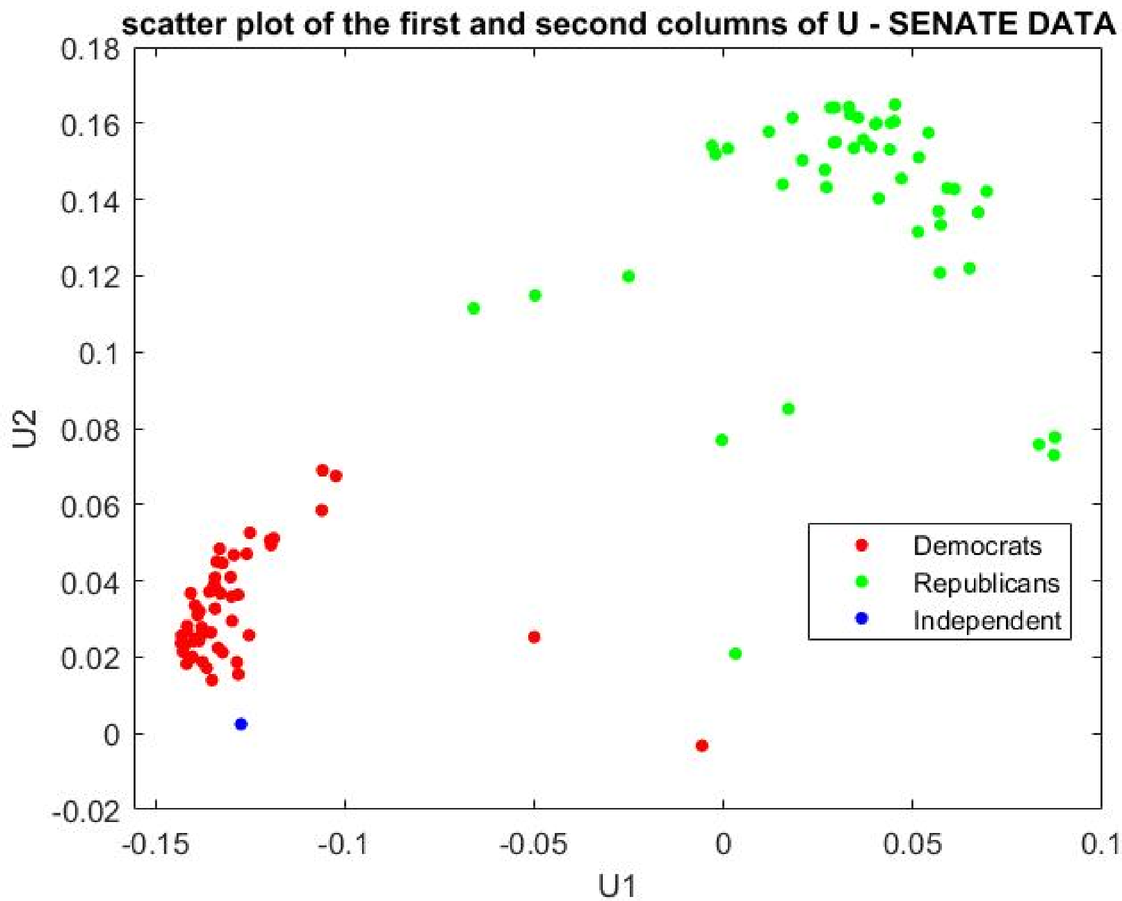
At first the magnitude of the singular values is 580 when the size of the matrix is (1,1).

When the size of the matrix is (50,50) the magnitude reaches 26 and then

As the size of the matrix reaches (400,400) the magnitude of the singular value approaches zero.

c.)

SENATE DATA:



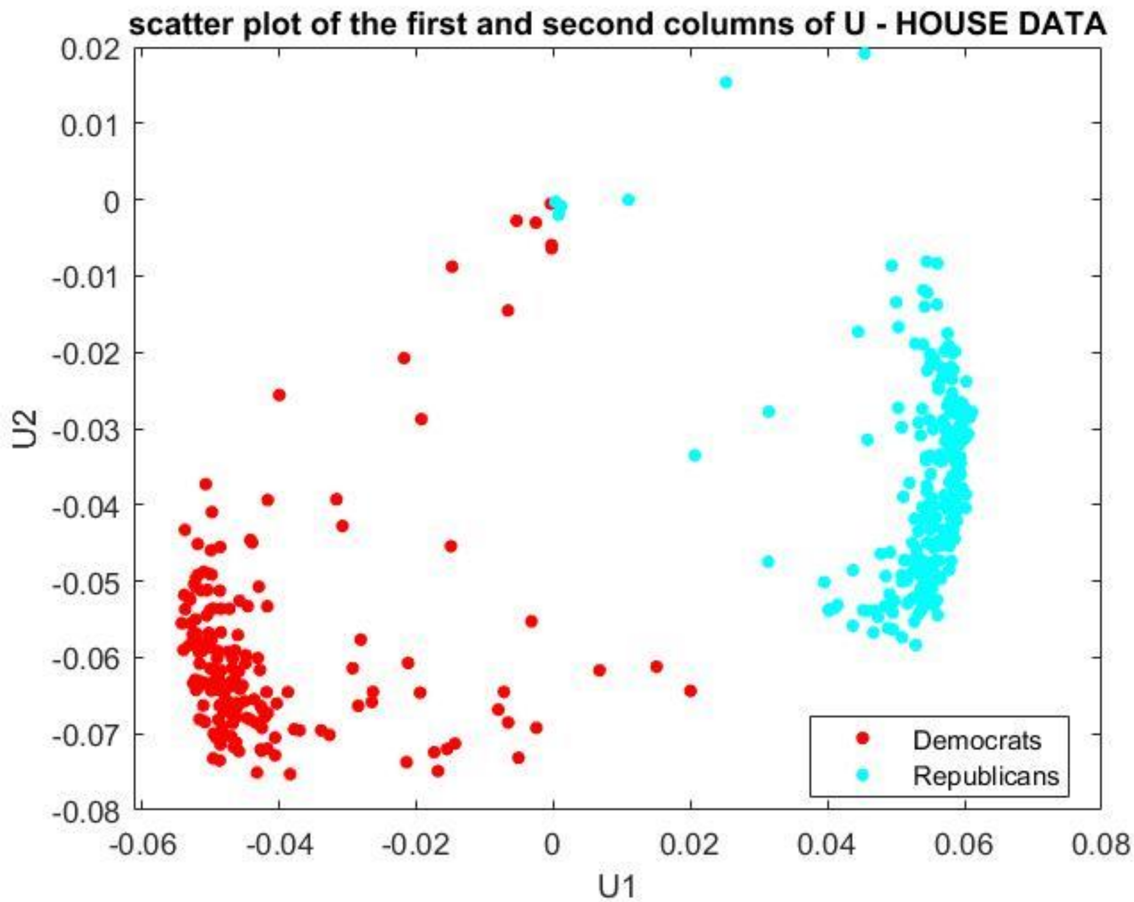
3.) (d.)

SENATE DATA: If we observe the above scatter plot carefully we can observe all the democrat's density is located on the left side of the plot while the Republican density is located on the right side of the plot. It can be inferred that voting patterns of Democrats and Republicans are different from each other irrespective of the bill. Also it can be observed that independent candidate is with Democrats for a particular bill.

U1 is capturing the voting pattern of Democrats while U2 is capturing the voting pattern of Republicans.

3(f)

HOUSE DATA:



HOUSE DATA: If we observe the above scatter plot carefully we can observe all the democrat's density is located on the left side of the plot while the Republican density is located on the right side of the plot. It can be inferred that voting patterns of Democrats and Republicans are different from each other irrespective of the bill.

U1 and U2 both are capturing the voting pattern of Democrats.

3 (g.) It can be inferred that voting patterns of Democrats and Republicans are different from each other irrespective of the bill.