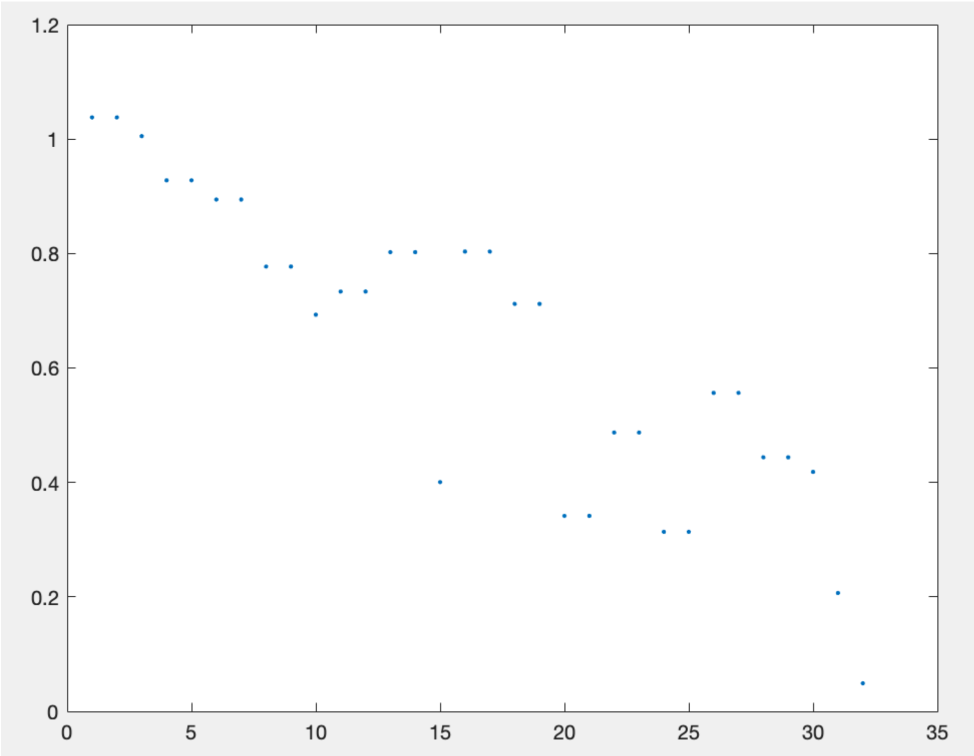
# HW5, Dept.: 수리과학과, NAME: 국윤범

## Problem 1 - Exercise 12.3

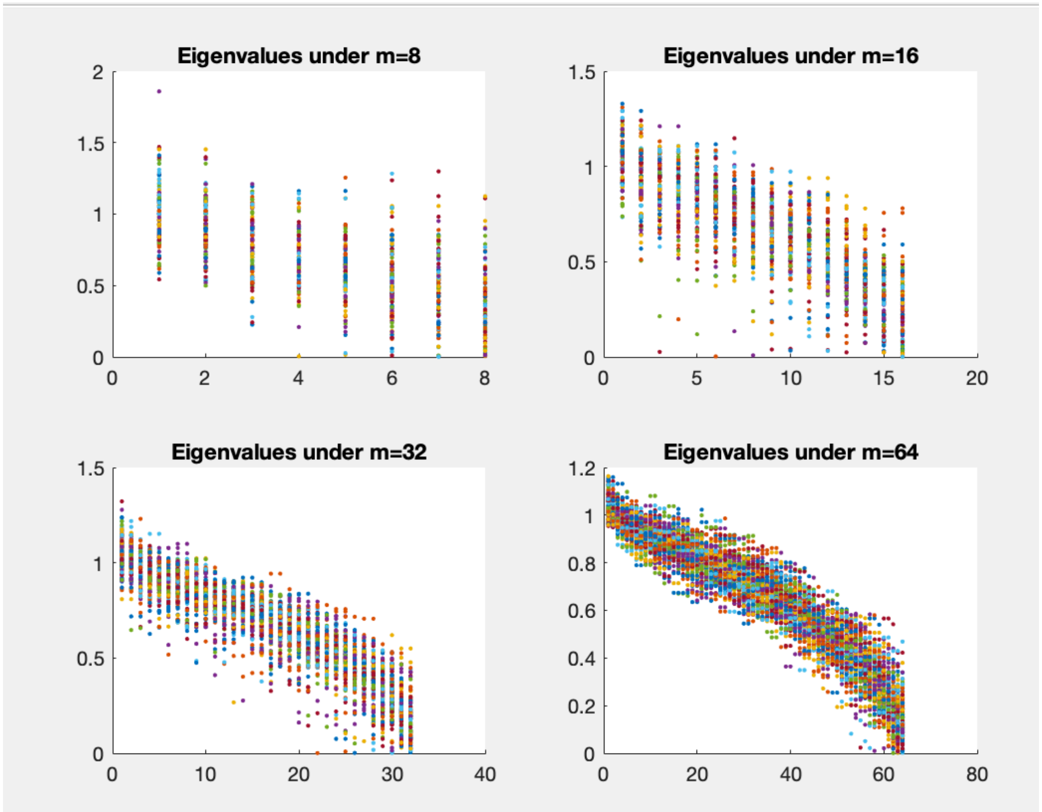
## (a)

```
%%%%% Exercise 12.1 %%%%%
% (a)
% What do eigenvalues of a random matrix look like?
A = randn(32,32)/sqrt(32);
plot(abs(eig(A)),'.')

% Superimpose all eigenvalues in a single plot for m = 8, 16, 32, 64
dim = [8, 16, 32, 64];

figure;
for d=1:4
    subplot(2,2,d); title(sprintf("Eigenvalues under m=%d",dim(d)));
    hold on
    for i=1:100
       A = randn(dim(d))/sqrt(dim(d));
       plot(abs(eig(A)),'.')
    end
    hold off
end

% Pattern of spectral norm(the largest eigenvalue) as m increases
figure;
spectral = zeros(1,100);
for i=1:100
    A = randn(10*i) / sqrt(10*i);
    eigen = eig(A); radius = abs(eigen(1));
    spectral(i) = radius;
end

plot([10:10:1000],spectral,'.'); title("Spectral Radius from m=1 to 1000")
```

Since eigenvalues are possibly complex numbers in general, so I plot the magnitude of those eigenvalues.

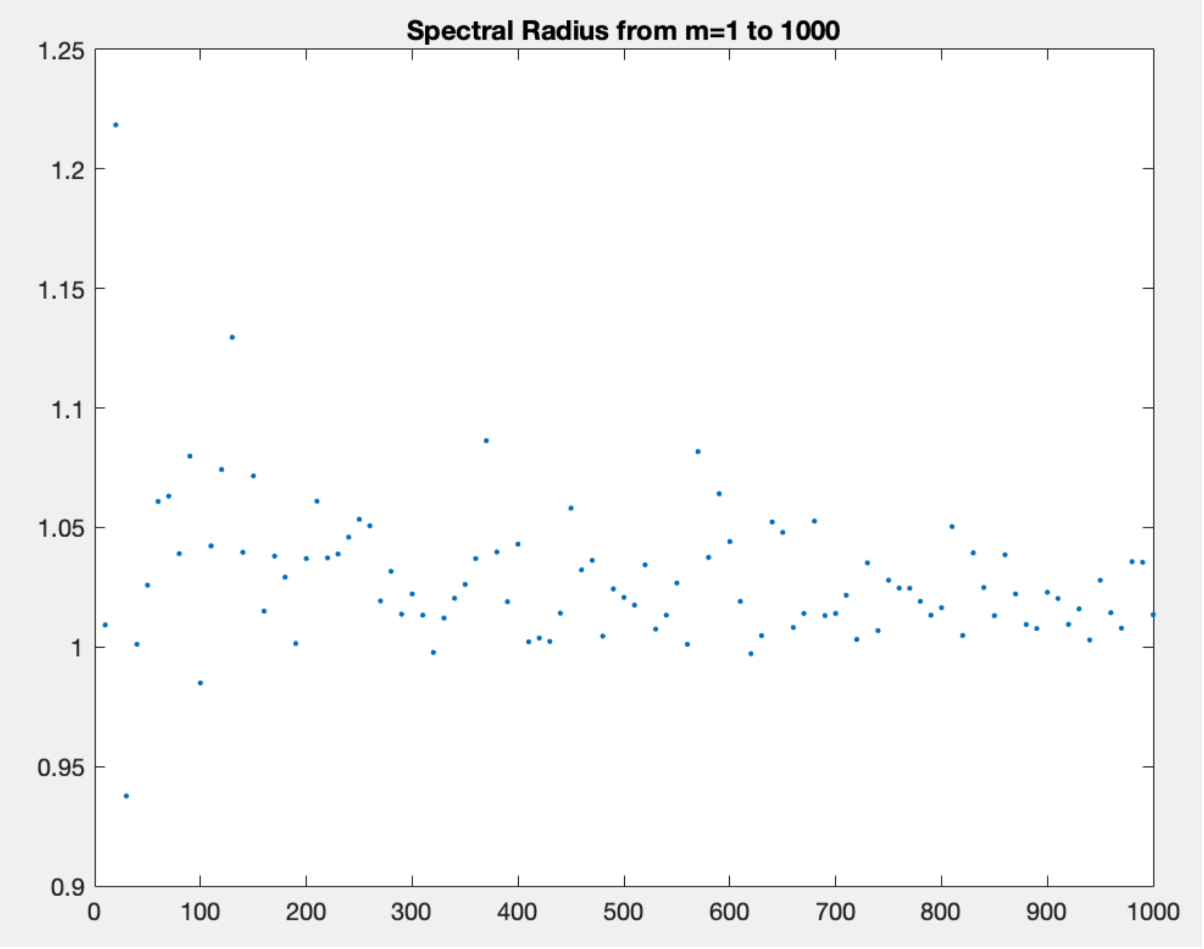


We can observe that the largest eigenvalue is close to 1 and the magnitude of all eigenvalues seem to decrease in linear scale. Then, I superimpose all eigenvalues of 100 random matrices in a single plot, changing the size (m) of matrices.



The observation that I mentioned right above shows up consistently for all random matrices, regardless of the size of matrices. In detail, the largest eigenvalue seem to be drawn from normal distribution with mean 1 and some variance which get small as the size grows. Specifically, as the size of matrices increases, the variance of eigenvalues at  value decreases. (Fix 된 x value 기준으로 eigenvalue 의 분포가 좀 더 dense 해짐).

Now, I plot the spectral radius of random matrices, increasing the size of matrices. Here, note that the spectral radius is exactly equal to the largest eigenvalues.

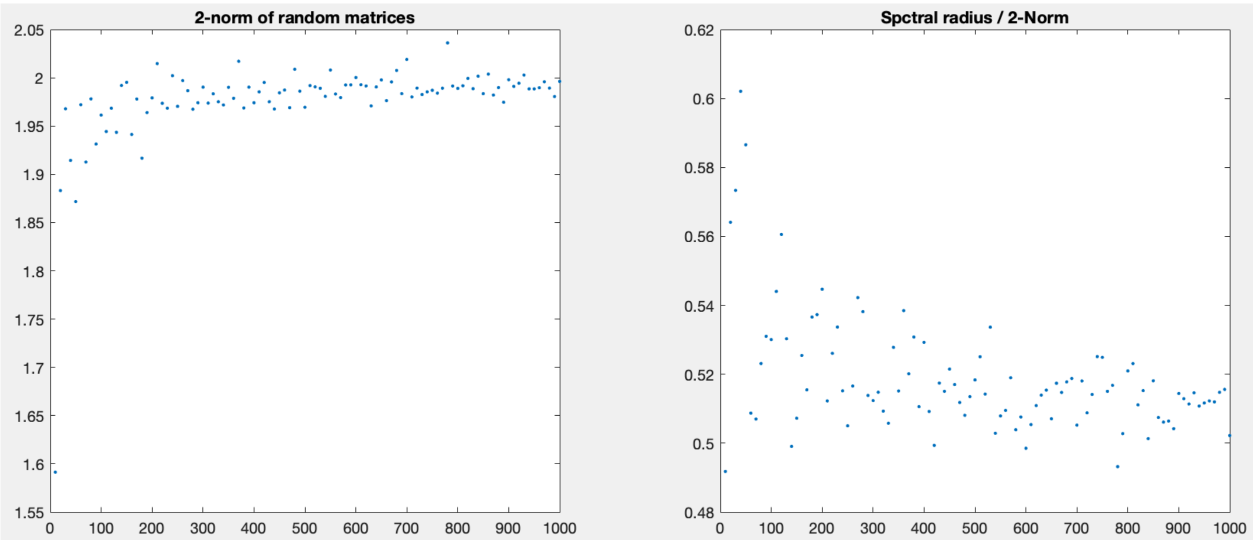



Definitely, the spectral radius is approaching 1 as the size of matrices also increases. Furthermore, as I mentioned right above, it seems that the spectral radius is less scattered from 1 as the size of matrices grows.

## (b)

```
%(b)

figure;
norm2 = zeros(100,1);
sm_sig = zeros(100,1);
percent = zeros(100,1);
for i=10:10:1000
    A = randn(i) / sqrt(i);
    eigen = eig(A); radius = abs(eigen(1));
    norm2(i/10) = norm(A);
    sm_sig(i/10) = svds(A,1,'smallest');
    percent(i/10) = radius / norm2(i/10);
end


subplot(1,2,1);
plot([10:10:1000], norm2, '.'); title("2-norm of random matrices");
subplot(1,2,2);
plot([10:10:1000], percent, '.'); title("Spctral radius / 2-Norm");
```

Increasing the size of matrices , I plot the 2-norm of random matrices and the ratio of the spectral radius and the 2-norm as well. (x-axis : the size of matrices)

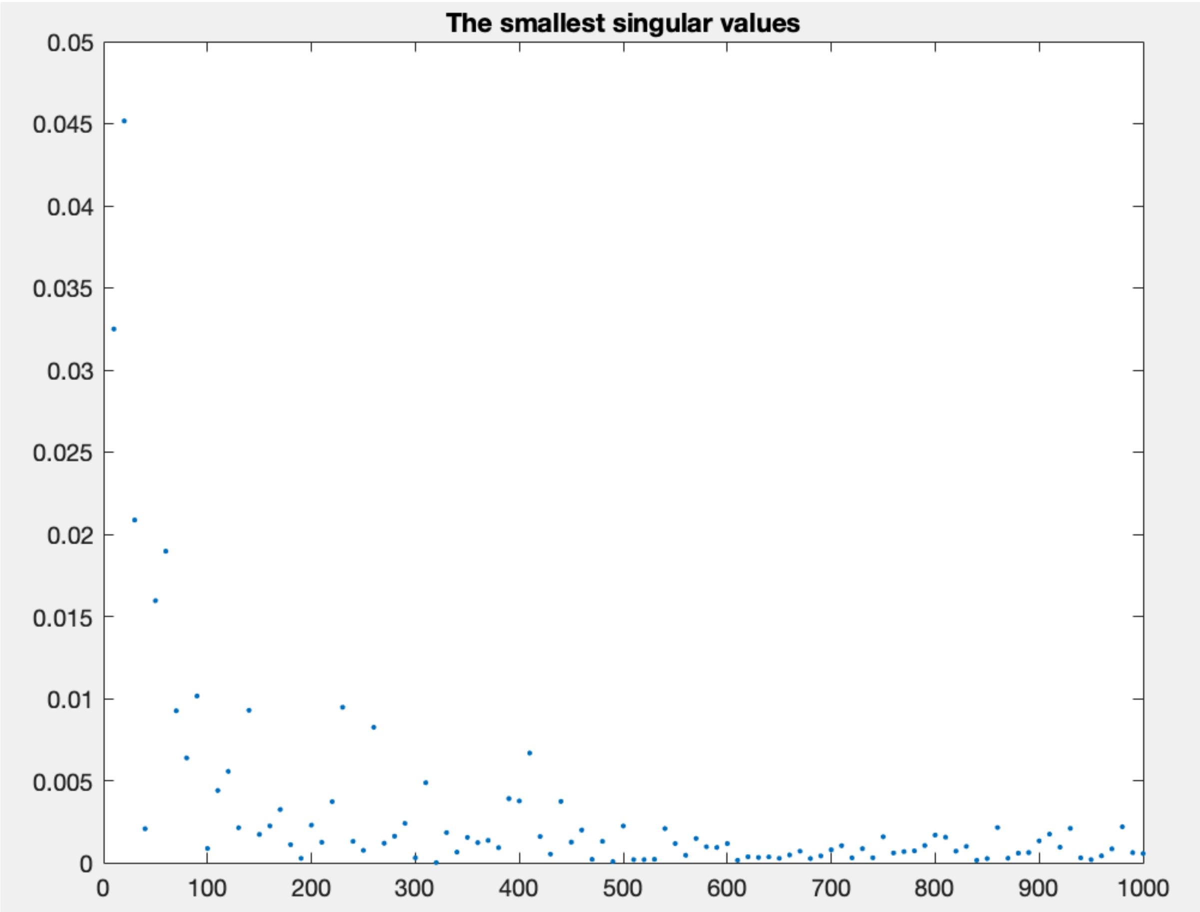



The 2-norm of random matrices approaches 2 as the size of matrices increases. Hence, the ratio of the spectral radius and the 2-norm approaches 0.5 as the size increases. Hence, we can say the inequality between the spectral radius and 2-norm seems not to approach an equality as the size goes to infinity.

**(c)**

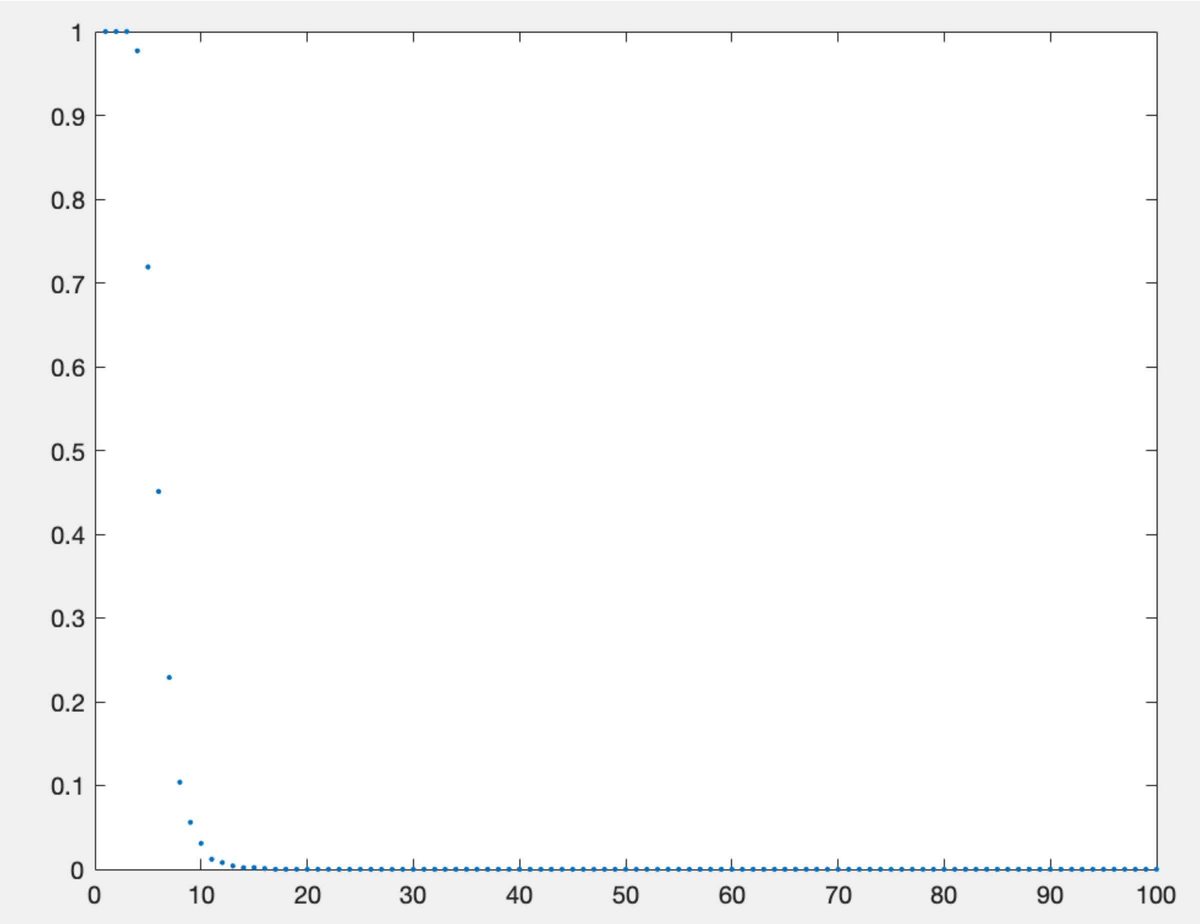
```
% (c)
% Pattern of the smalles singular values as m increases
figure;
plot([10:10:1000], sm_sig, '.'); title ("The smallest singular values")
```

By plotting the smallest singular values of random matrices with the sizes increased, we can observe the pattern that the smallest singular values decrease and seem to converge to 0 as the size increases. (x-axis : the size of matrices)



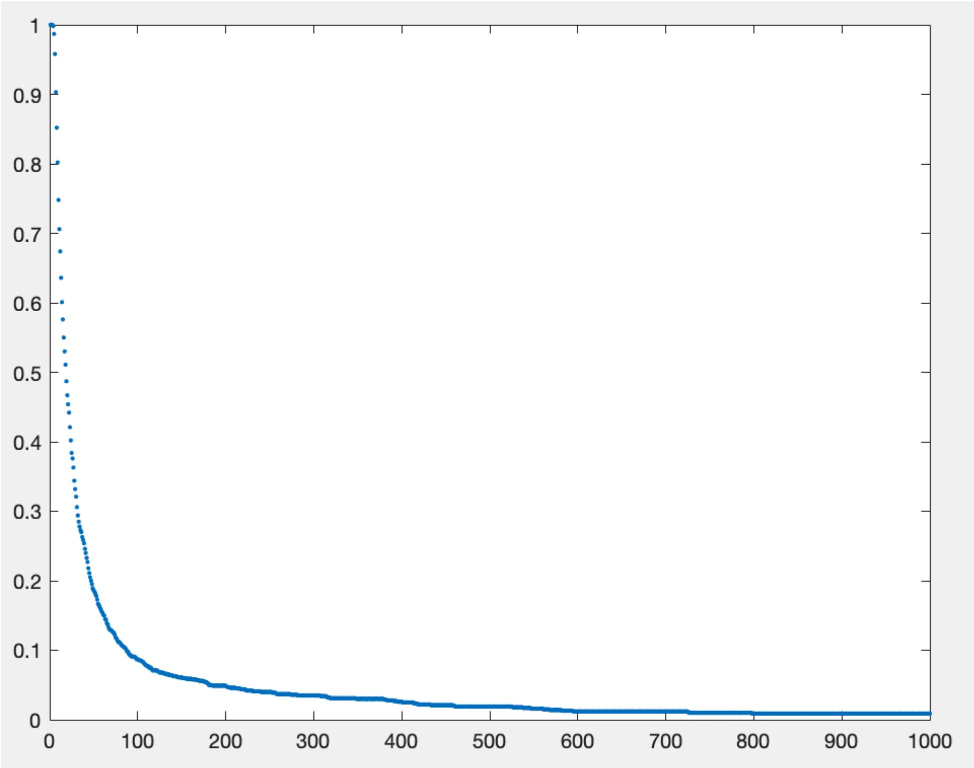

Now, let's investigate the proportion of random matrices whose smallest eigenvalues is smaller than 2^(-i) for fixed size=30. Here, the x-axis means the minus exponent of 2. Namely, the y value of x=10 means the proportion of random matrices whose smallest eigenvalues is smaller than 2^(-10).

```
% Fix m=10. Examine the tail of the probability distribution of smallest
% singular values.
sm_sigs = zeros(1000,1);
for i=1:1000
    A = randn(30) / sqrt(30);
    sm_sigs(i) = svds(A,1,'smallest');
end

prop = zeros(100,1);
for i=1:100
   prop(i) = sum(sm_sigs < 1 / 2^i)/1000;
end
figure;
plot(prop,'.');
```



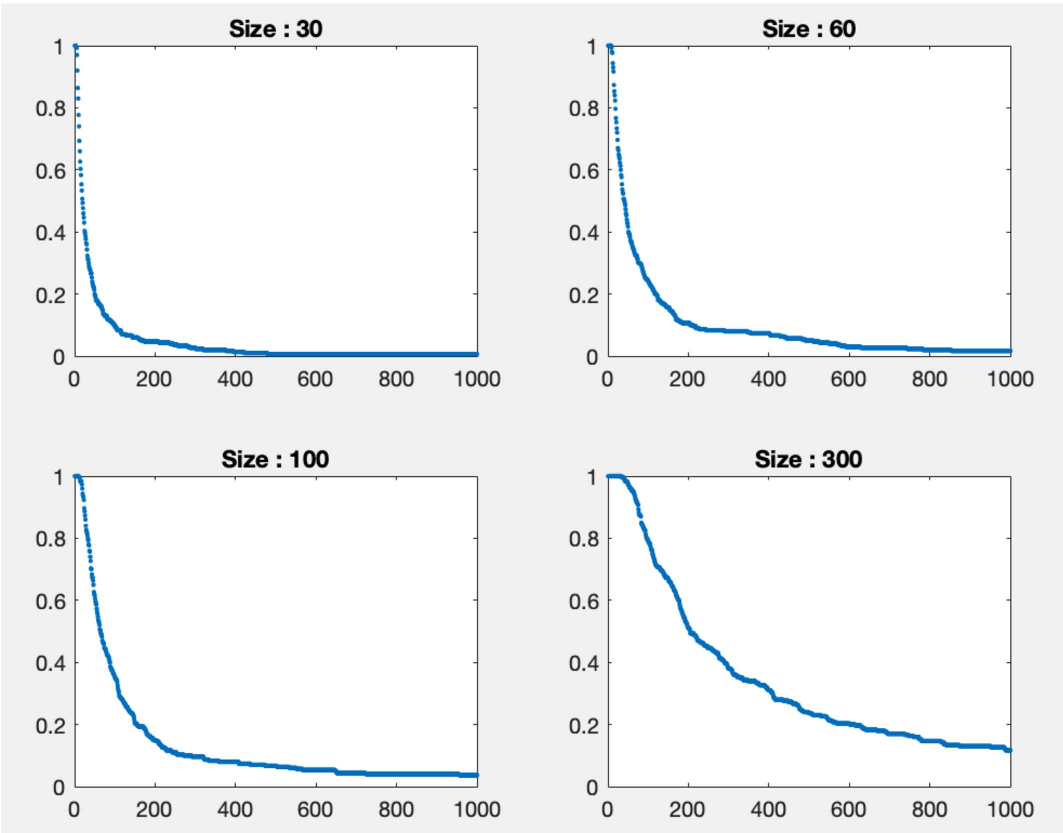

It seems that the proportion significantly drops at some points. Hence, I proceed more refined experiment (checking the proportion smaller than 1 / (3 * i) from i=0 to 1000.

```
% Refined experiments
prop = zeros(1000,1);
for i=1:1000
    prop(i) = sum(sm_sigs < 1 / (3*i))/1000;
end
figure;
plot(prop,'.');
```




From this plot, the proportion stays as 1 initially. However, after the proportion starts to drop, the behavior of proportion resembles long-tail distribution. Finally, I compare this pattern among various sizes of matrices.

```matlab
% Trial for various sizes
figure;
sm_sigs = zeros(300,4);
for i=1:300
    A = randn(30) / sqrt(30);
    B = randn(60) / sqrt(60);
    C = randn(100) / sqrt(100);
    D = randn(300) / sqrt(300);
    sm_sigs(i,1) = svds(A,1,'smallest');
    sm_sigs(i,2) = svds(B,1,'smallest');
    sm_sigs(i,3) = svds(C,1,'smallest');
    sm_sigs(i,4) = svds(D,1,'smallest');
end

prop = zeros(1000,4);
for i=1:1000
   prop(i,1) = sum(sm_sigs(:,1) < 1 / (3*i))/300;
   prop(i,2) = sum(sm_sigs(:,2) < 1 / (3*i))/300;
   prop(i,3) = sum(sm_sigs(:,3) < 1 / (3*i))/300;
   prop(i,4) = sum(sm_sigs(:,4) < 1 / (3*i))/300;
end

size = [30,60, 100, 300];
for i=1:4
   subplot(2,2,i);
   plot(prop(:,i), '.'); title(sprintf("Size : %d", size(i)));
end
```




As the size grows, the rate of decrease in the proportion is slower and the proportion initially stays 1 much longer. In other words, the proportion graphs get smooth.