

MATH 543 – Homework 5

Parham Khodadi

Problem 12.3 – Trefethen & Bau

The goal of this problem is to explore some properties of random matrices. Your job is to be a laboratory scientist, performing experiments that lead to conjectures and more refined experiments. Do not try to prove anything. Do produce well-designed plots, which are worth a thousand numbers.

Define a random matrix to be an $m \times m$ matrix whose entries are independent samples from the real normal distribution with mean zero and standard deviation \sqrt{m} — In MATLAB, $A = \text{randn}(m,m)/\text{sqrt}(m)$ — the factor \sqrt{m} is introduced to make the limiting behavior clean as $m \rightarrow \infty$.

- A. What do the eigenvalues of a random matrix look like? What happens, say, if you take 100 random matrices and superimpose all their eigenvalues in a single plot? If you do this for $m = 8, 16, 32, 64, \dots$, what pattern is suggested? How does the spectral radius $\rho(A)$ (see Exercise 3.2) behave as $m \rightarrow \infty$?
- B. What about norms? How does the 2-norm of a random matrix behave as $m \rightarrow \infty$? Of course, we must have $\rho(A) < \|A\|$ (see Exercise 3.2). Does this inequality appear to approach an equality as $m \rightarrow \infty$?
- C. What about condition numbers—or more simply, the smallest singular value σ_{min} ? Even for fixed m this question is interesting. What proportions of random matrices in $\mathbb{R}^{m \times m}$ seem to have $\sigma_{min} < 2^{-1}, 2^{-2}, 2^{-3}, \dots$? In other words, what does the tail of the probability distribution of smallest singular values look like? How does the scale of all this change with m ?

Part A.

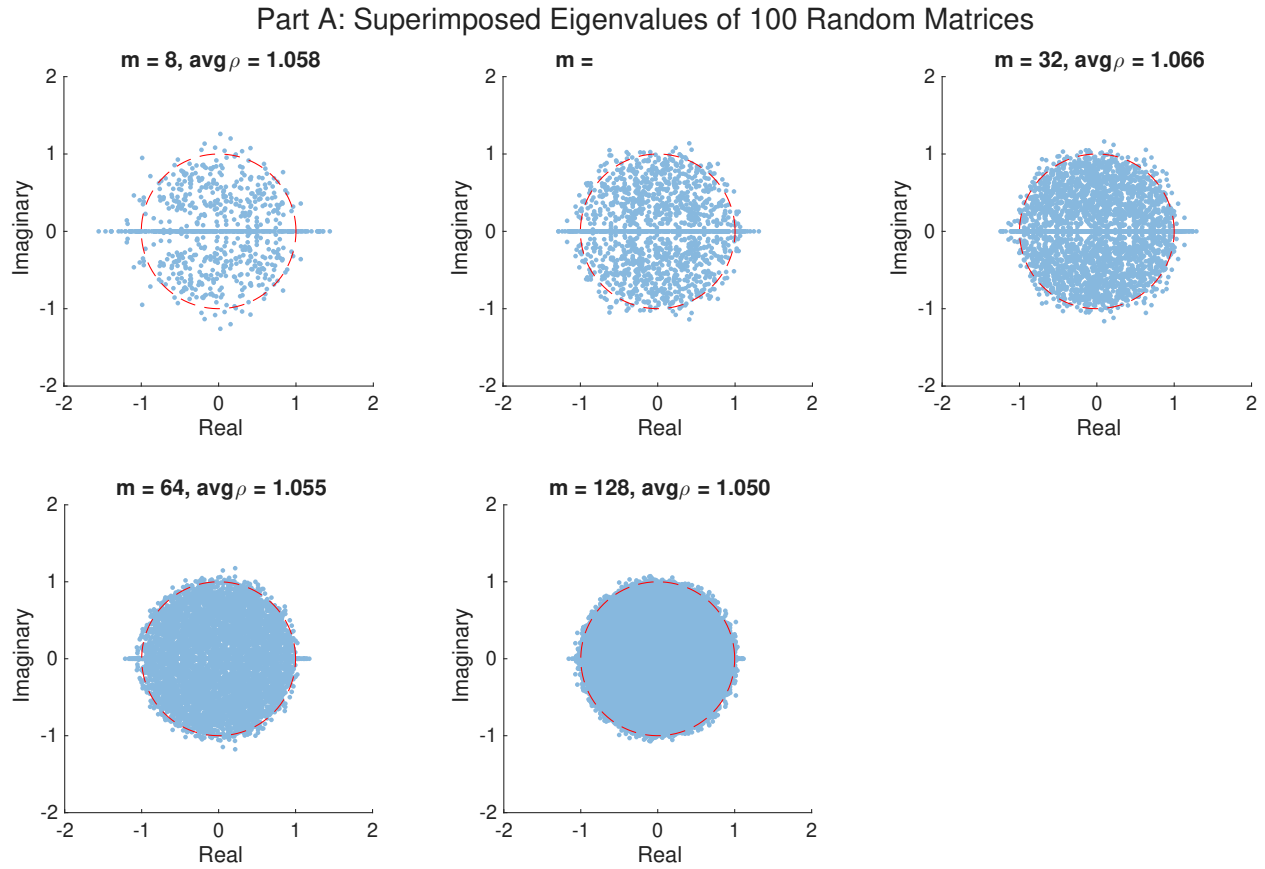


Figure 1: For some reason, LaTeX is having trouble loading the second graph's title. It should read: $m = 16, \rho_{avg} = 1.052$.

My MATLAB code (found in the Appendix) generated Figure 1. This figure shows that the eigenvalues of random matrices superimposed on the complex plane fall into a roughly uniform circular distribution. As the matrix size m increases, this circular "blob" becomes more dense. This increase in density also makes its outer boundary much sharper.

The red dashed unit circle (radius of 1) perfectly bounds the distribution. As $m \rightarrow \infty, \rho \rightarrow 1$. The third graph being a statistical outlier, where the ρ increases rather than decrease towards 1.

Part B.

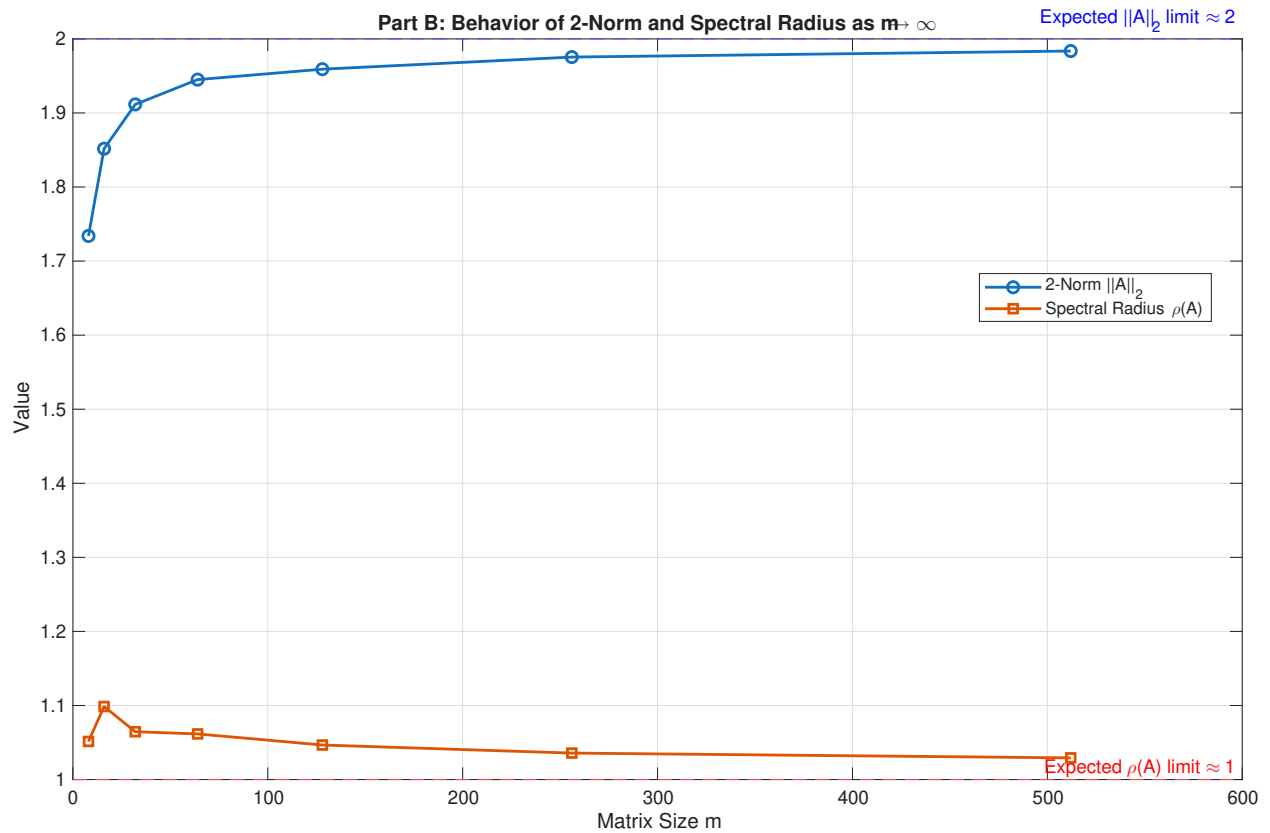


Figure 2: It may be hard to tell, but there exists a blue dashed line at $y = 2$ representing the expected value for the 2-norm limit.

My MATLAB code (found in the Appendix) generated Figure 2. In this figure, you can see that the 2-norm of the random matrix A rapidly parabolically approaches the asymptote of 2. In other words: As $m \rightarrow \infty$, $\|A\|_2 \rightarrow 2$. At the same time, the spectral radius approaches 1. In other words: As $m \rightarrow \infty$, $\rho(A) \rightarrow 1$.

This graph also answers the question of whether the fundamental inequality $\rho(A) \leq \|A\|_2$ approaches equality with increasing m . Clearly, it does not as $\rho(A) \rightarrow 1$, it remains less than $\|A\|_2 \rightarrow 2$.

Part C.

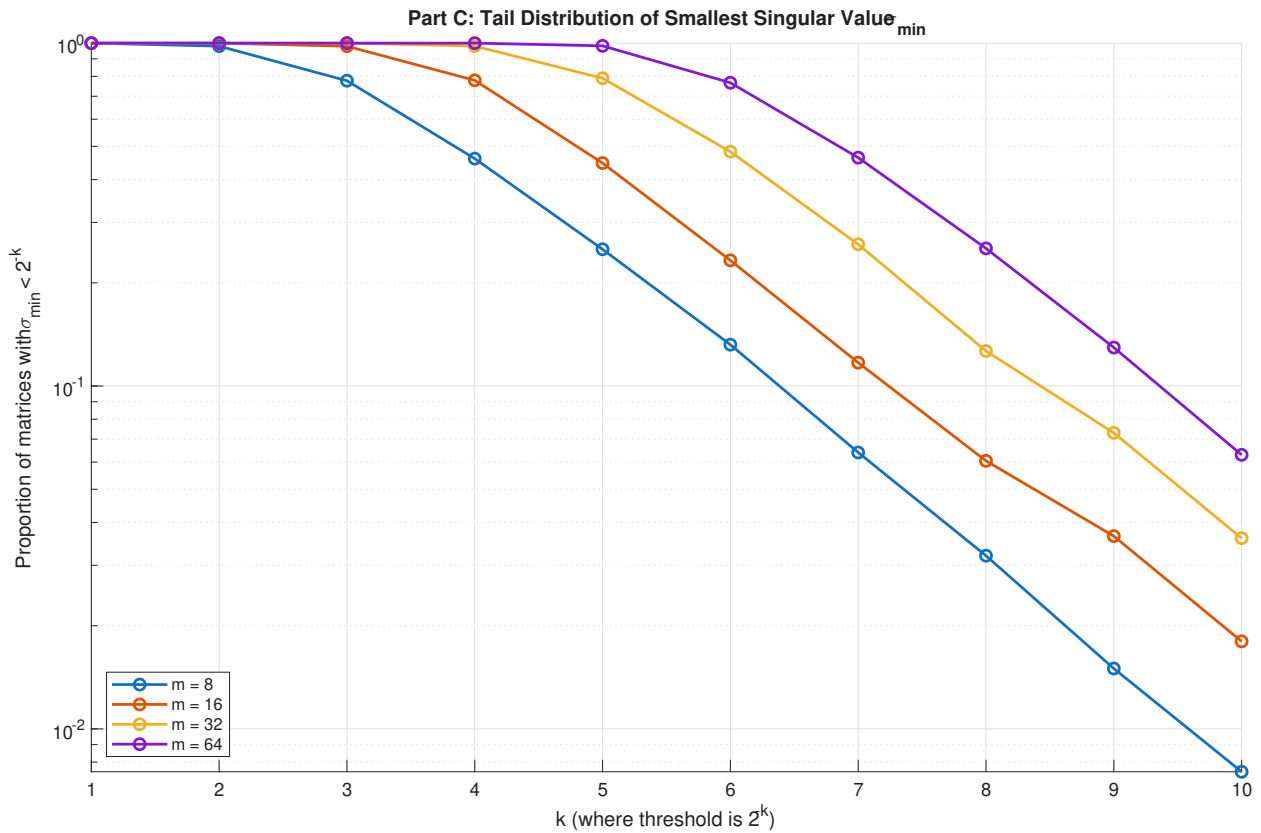


Figure 3: Enter Caption

My MATLAB code (found in the Appendix) generated Figure 3. Because the y-axis is on a logarithmic scale, the straight, parallel lines for higher values of k (roughly $k \geq 3$) indicate a power-law relationship. Since I plotted against k (where the threshold is 2^{-k}), the linear drop-off confirms that the probability of finding a smallest singular value $\sigma_{\min} < \epsilon$ is directly proportional to ϵ .

Look at the vertical spacing between the lines. For any given k , the proportion is higher for larger m . If you compare $m = 8$ to $m = 64$ (an $8\times$ increase in size), the proportion line shifts up by roughly a factor of 8. This suggests that the probability of drawing an ill-conditioned random matrix increases linearly with the matrix dimension m .

Appendix: MATLAB Code

```
1 %% MATH 543 - Homework 4
2 % By Parham Khodadi
3 clear; clc; close all;
4
5 %% Part A: Eigenvalues and Spectral Radius
6 m_vals = [8, 16, 32, 64, 128];
7 num_matrices = 100;
8 figure('Position', [100, 100, 1000, 600]);
9
10 for i = 1:length(m_vals)
11     m = m_vals(i);
12
13     % 1. Preallocate a column vector of the exact required size
14     all_eigenvalues = zeros(m * num_matrices, 1);
15     avg_rho = 0;
16
17     for k = 1:num_matrices
18         A = randn(m,m)/sqrt(m);
19         eigs_A = eig(A);
20
21         % 2. Calculate the start and end indices for the current batch
22         start_idx = (k - 1) * m + 1;
23         end_idx = k * m;
24
25         % 3. Insert the eigenvalues into the preallocated array
26         all_eigenvalues(start_idx:end_idx) = eigs_A;
27
28         avg_rho = avg_rho + max(abs(eigs_A)); % Compute spectral radius
29     end
30     avg_rho = avg_rho / num_matrices;
31
32     % Plot eigenvalues in the complex plane
33     subplot(2, 3, i);
34     scatter(real(all_eigenvalues), imag(all_eigenvalues), 5, 'filled', '
35         MarkerFaceAlpha', 0.5);
36     title(sprintf('m=%d, avg_rho=%.3f', m, avg_rho));
37     axis equal;
38     xlim([-2 2]); ylim([-2 2]);
39     xlabel('Real'); ylabel('Imaginary');
40
41     % Draw unit circle for reference
42     hold on;
43     theta = linspace(0, 2*pi, 100);
44     plot(cos(theta), sin(theta), 'r--');
45     hold off;
46 end
47 sgtitle('Part A: Superimposed Eigenvalues of 100 Random Matrices');
48
49 % Save Figure for Part A
50 exportgraphics(gcf, 'Figures\PartA_Eigenvalues.eps', 'ContentType', '
51     vector');
```

```

50
51
52 %% Part B: Norms vs Spectral Radius
53 m_vals_large = 2.^(3:9); % Test m values from 8 up to 512
54 trials = 50;
55 avg_norms = zeros(size(m_vals_large));
56 avg_rhos = zeros(size(m_vals_large));
57
58 for i = 1:length(m_vals_large)
59     m = m_vals_large(i);
60     temp_norm = 0;
61     temp_rho = 0;
62     for k = 1:trials
63         A = randn(m,m)/sqrt(m);
64         temp_norm = temp_norm + norm(A, 2);
65         temp_rho = temp_rho + max(abs(eig(A)));
66     end
67     avg_norms(i) = temp_norm / trials;
68     avg_rhos(i) = temp_rho / trials;
69 end
70
71 figure;
72 plot(m_vals_large, avg_norms, '-o', 'LineWidth', 1.5, 'DisplayName', '2-
73     Norm_||A||_2');
74 hold on;
75 plot(m_vals_large, avg_rhos, '-s', 'LineWidth', 1.5, 'DisplayName', '
76     Spectral_Radius_\rho(A)');
77 yline(2, 'b--', 'Expected_||A||_2_limit_\approx_2', 'HandleVisibility', '
78     off');
79 yline(1, 'r--', 'Expected_\rho(A)_limit_\approx_1', 'HandleVisibility', '
80     off');
81 xlabel('Matrix_Size_m');
82 ylabel('Value');
83 title('Part B: Behavior of 2-Norm and Spectral Radius as m \rightarrow
84     inf');
85 legend('Location', 'best');
86 grid on;
87
88 % Save Figure for Part B
89 exportgraphics(gcf, 'Figures\PartB_Norm_Rho.eps', 'ContentType', 'vector')
90 ;
91
92 %% Part C: Condition Numbers and Smallest Singular Value (\sigma_min)
93 m_vals_C = [8, 16, 32, 64];
94 num_samples = 2000; % Large sample size for distinct tail probabilities
95 k_vals = 1:10;
96 thresholds = 2.^(-k_vals);
97
98 figure;
99 hold on;
100 for i = 1:length(m_vals_C)
101     m = m_vals_C(i);
102     sigma_min_vals = zeros(num_samples, 1);

```

```

98
99     for j = 1:num_samples
100         A = randn(m,m)/sqrt(m);
101         s = svd(A); % Returns singular values in descending order
102         sigma_min_vals(j) = s(end); % Extract smallest singular value
103     end
104
105     proportions = zeros(length(thresholds), 1);
106     for t = 1:length(thresholds)
107         proportions(t) = sum(sigma_min_vals < thresholds(t)) / num_samples
108         ;
109     end
110     plot(k_vals, proportions, '-o', 'LineWidth', 1.5, 'DisplayName',
111         sprintf('m=%d', m));
112 end
113 set(gca, 'YScale', 'log'); % Log scale to see the tail behavior clearly
114 xlabel('k (where threshold is 2^{-k})');
115 ylabel('Proportion of matrices with \sigma_{min} < 2^{-k}');
116 title('Part C: Tail Distribution of Smallest Singular Value \sigma_{min}')
117 ;
118 legend('Location', 'southwest');
119 grid on;
120 % Save Figure for Part C
121 exportgraphics(gcf, 'Figures\PartC_SigmaMin.eps', 'ContentType', 'vector')
122 ;

```