Computational Physics
Project 8: Matlab and the Master Matrix

Michael Ryan

May 6, 1999

Abstract

For this project we are temporarily leaving the low-level computer languages like C and Fortran to learn a bit about the capabilities of numerical commercial numerical packages, namely Matlab in this case. Packages like Matlab offer the users specialized function calls that can greatly simplify certain mathematical tasks. Matlab mainly specializes in matrix manipulation. We shall use Matlab to explore an interesting mathematical structure found in randomly generated, symmetric matrices. The interesting thing about these matrices is that their eigenvalues follow a specific repeatable pattern which we shall see in this project.

1 Procedure and Equations

1.1 The Master Matrix

For some $N \times N$ matrix M with elements m_{ij} that are generated randomly following the probability distribution

$$P(m_{ij})\alpha \exp\left(-\frac{1}{2}m_{ij}^2\right) \tag{1}$$

the quantities

$$Q_n = \frac{1}{N^{1+(n/2)}}\mathrm{Tr}[M^n] \tag{2}$$

should be, in the infinite limit, the same for any random matrix generated. So, generate 7 random, symmetric matrices using Matlab for each N of $N =$

10, 100, and 300 and calculate Q_n for $n = 2, 4$, and 6. The expected values for $N \to \infty$ are:

$$Q_2 = 1 \; Q_4 = 2 \; Q_6 = 5. \tag{3}$$

The errors on this calculation are

$$\delta Q_n = \sqrt{\frac{1}{K}\sum_{i=1}^{K}(Q_n^i - \bar{Q}_n)^2} \tag{4}$$

where K = 7, Q_n^i is the value obtained from the i-th matrix, and $\bar{Q}_n$ is the average of the K values.

1.2 The Density of Eigenvalues

The results from the previous section allude to even deeper results. Despite the random elements of the matrix the density of eigenvalues is the same as $N \to \infty$. So for eigenvalues e_i of some randomly generated matrix let

$$\lambda_i = \frac{e_i}{\sqrt{N}} \tag{5}$$

Taking the previous expression for Q_n and using

$$M = \sum_{i=0}^{N} e_i V_i V_i^\dagger \tag{6}$$

where V_i is the normalized eigenvector. I can substitute this into the the equation for Q_i.

$$Q_n = \frac{e_i^n}{N^{1+(n/2)}} \mathrm{Tr}\left[\sum_{i=0}^{N} V_i V_i^\dagger\right] \tag{7}$$

Using the relation between e_i and λ_i and the fact then I can simplify the equation to

$$Q_n = \frac{1}{N}\sum_{j=1} N\lambda_j^n. \tag{8}$$

Given the density of λ as

$$\rho(\lambda) = \frac{1}{N}\frac{\text{no. of } e_i \text{ in the interval}[\lambda, \lambda + d\lambda]}{\Delta\lambda} \tag{9}$$

I can check the universality of the eigenvalue density by graphing it vs. the average eigenvalues. So, I split the eigenvalues into 10 separate bins which are spread evenly over the interval $[-2.5, 2.5]$. I took the average eigenvalue for each bin. I then graphed these values with the number of eigenvalues in each bin divided by the square of the number of eigenvalues.

2 Graphs, Data, and Analysis.

2.1 The Master Matrix.

Here is the matlab program I created to create and print out the values of Q_n for all values of n as well as the error δQ_n.

```
fa = fopen('data.mat', 'w');
fb = fopen('avedata.mat', 'w');
fc = fopen('avedata2.mat', 'w');
N = [10 100 300];
for l = 1:3,
        for k = 1:7,
                randn('state', 100*sum(clock)) ;
                M = randn(N(l)) ;
                m1 = tril(M) ;
                M = M - triu(M);
                m1 = transpose(m1);
                M = M + m1;
                n = 2 ;
                while n < 8
                        Q(n/2, k, l) = (1/(N(l)^(1+n/2)))*trace(M^n);
                        fprintf(fa, '%i        %i         %i       %f\n', n/2, k,
l, Q(n/2,k,l));
                        n = n + 2;
                end
        end
end
B = mean(Q, 2);
for i = 1:3
        for j = 1:7
                dq1(:, j, i) = Q(:, j, i)-B(:,:,i);
```

```
        end
end
dq1 = dq1.^2;
dq1 = dq1./7;
dq1 = cumsum(dq1, 2);
dq = sqrt(dq1);
fprintf(fc, '%f\n\n', B );
fprintf(fb, '%f\n\n', dq(:,7,:));
fclose(fa);
fclose(fb);
fclose(fc);
```

The end results for Q_n are

n/2	k	l	Q_n
1	1	1	0.896183
2	1	1	1.465394
3	1	1	2.763742
1	2	1	0.900775
2	2	1	1.448791
3	2	1	2.733351
1	3	1	0.900775
2	3	1	1.448791
3	3	1	2.733351
1	4	1	0.900775
2	4	1	1.448791
3	4	1	2.733351
1	5	1	0.900775
2	5	1	1.448791
3	5	1	2.733351
1	6	1	1.175299
2	6	1	3.421487
3	6	1	12.13000
1	7	1	1.175299
2	7	1	3.421487
3	7	1	12.13000

1	1	2	1.033704
2	1	2	2.155468
3	1	2	5.601833
1	2	2	1.007463
2	2	2	2.020622
3	2	2	5.115207
1	3	2	0.997549
2	3	2	2.000477
3	3	2	5.045538
1	4	2	1.000972
2	4	2	2.000607
3	4	2	5.010525
1	5	2	0.982161
2	5	2	1.908916
3	5	2	4.581576
1	6	2	1.022774
2	6	2	2.074977
3	6	2	5.261363
1	7	2	1.027759
2	7	2	2.107611
3	7	2	5.374225
1	1	3	1.007517
2	1	3	2.021227
3	1	3	5.065481
1	2	3	1.008366
2	2	3	2.024504
3	2	3	5.075926
1	3	3	1.010114
2	3	3	2.038650
3	3	3	5.141705
1	4	3	0.996226
2	4	3	1.998608
3	4	3	5.024686
1	5	3	0.999369
2	5	3	1.988219
3	5	3	4.933880
1	6	3	1.012722
2	6	3	2.062859

3	6	3	5.258246
1	7	3	1.012510
2	7	3	2.048609
3	7	3	5.193549

The averages for each n and their corresponding errors δQn are

N	n	Q_n	δQ_n
10	2	0.978555	0.124442
10	4	2.014790	0.889691
10	6	5.422450	4.242241
100	2	1.010340	0.017145
100	4	2.038383	0.075386
100	6	5.141467	0.297850
300	2	1.000668	0.005959
300	4	2.026096	0.024589
300	6	5.099067	0.100403

Notice how the errors decrease as N increases.

2.2 The Density of Eigenvalues.

Figure 1 is a plot of the eigenvalue density for three randomly generated matrices for $N = 100$.

Figure 2 is a plot of the eigenvalue density for three randomly generated matri ces for $N = 300$.

Figure 3 is a plot of the eigenvalue density for three randomly generated matri ces for $N = 1000$.

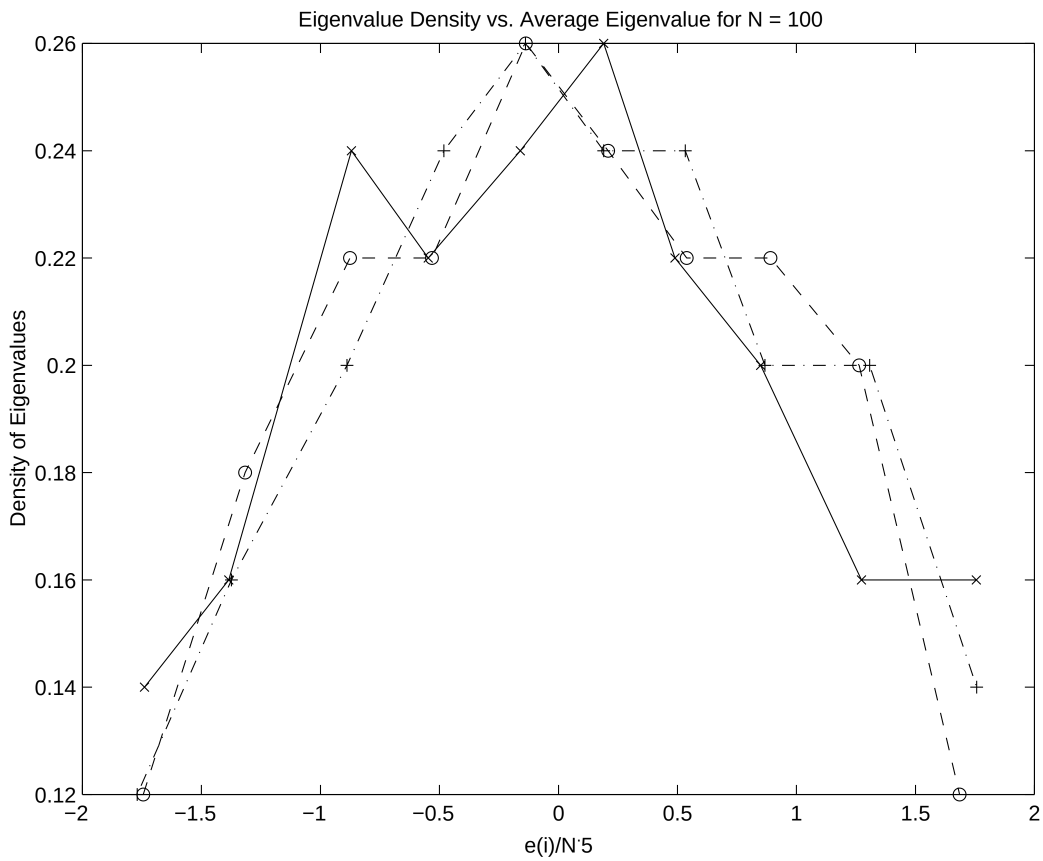

Figure 1: Graph of eigenvalue density vs. average eigenvalue for $N = 100$.

3 Conclusion

I think that this project has definitely shown the relationship of the eigenvalues of randomly generated symmetric matrices. It was put forth that as $N \to \infty$ the eigenvalues of any $N \times N$ randomly generated symmetric matrix would fall into absolutely repeatable pattern. As shown in the previous section there was evidence of a pattern supported by my low errors which decreased in magnitude as N increased.

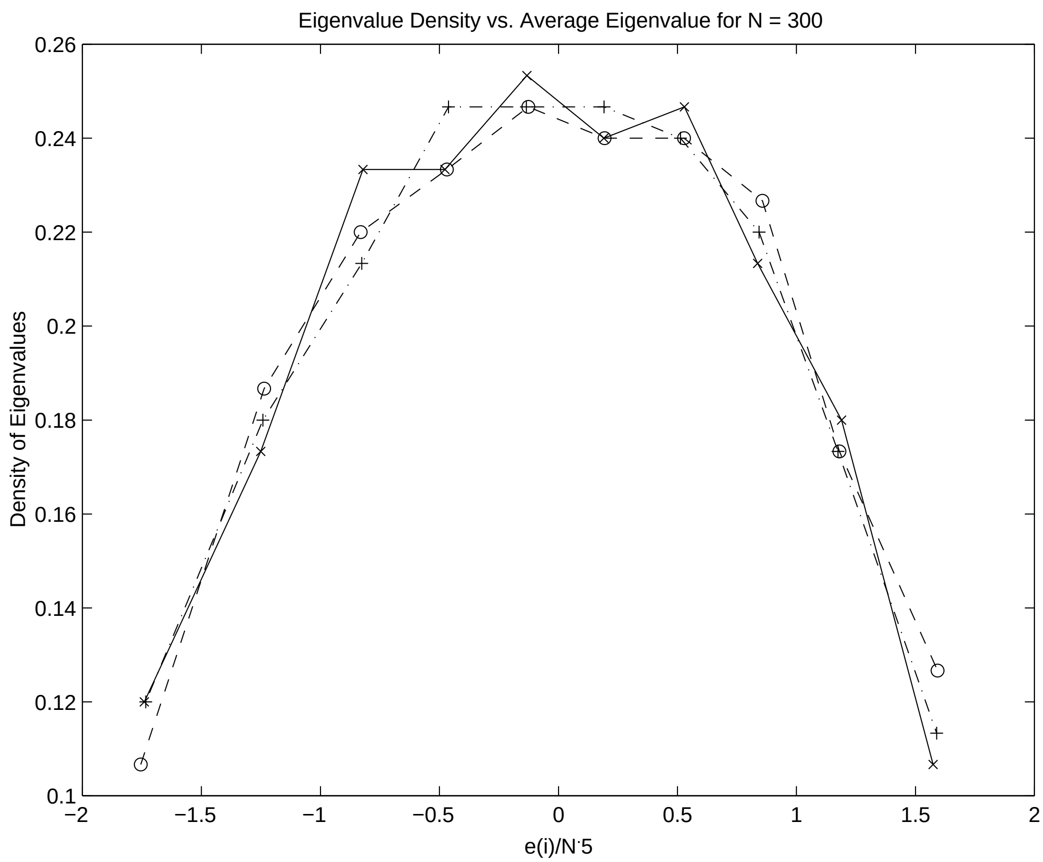

Figure 2: Graph of eigenvalue density vs. average eigenvalue for $N = 300$.

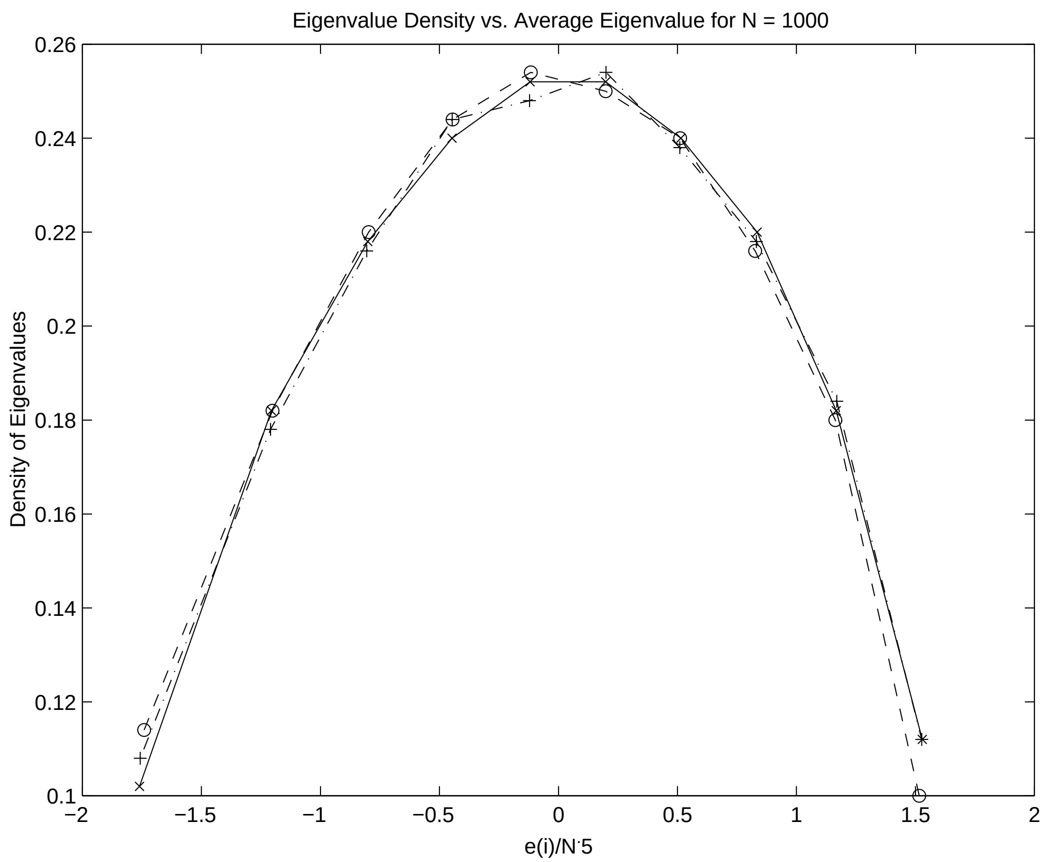

Figure 3: Graph of eigenvalue density vs. average eigenvalue for $N = 1000$.