MIT 18.095, IPA 2006

Numerical Methods for Random Matrices

Problem Set

In class we showed how to compute the probability distribution for a random variable by generating a large number of samples and plotting a histogram. Here is a MATLAB example which creates numbers from the normal distribution and plots a histogram which is scaled to make it a probability distribution:

```
trials=10000;
dx=.2;

y=[];
for k=1:trials
  sample=randn(1);
  y=[y; sample];
end

[count,x]=hist(y,-4:dx:4);
bar(x,count/(numel(y)*dx))
```

To check our simulation we can draw the expected distribution $f(x) = \frac{1}{\sqrt{2\pi}} e^{-x^2/2}$ on top of the histogram:

```
xx=-4:0.01:4;
yy=exp(-xx.^2/2)/sqrt(2*pi);
line(xx,yy,'color','m','linewidth',2)
```

1. Modify the first example to histogram the eigenvalues of a 2-by-2 random GUE matrix.

2. Modify the second example to draw the expected distribution on top of the histogram in part 1. The expression for the eigenvalue distribution is (with our scaling):

$$g(x) = \frac{f(x)}{n} \sum_{j=0}^{n-1} \frac{H_j(x/\sqrt{2})^2}{2^j j!}, \tag{1}$$

   where n is the size of the matrix ($n = 2$ in this case) and $H_j(x)$ is the jth Hermite polynomial (search for "Hermite" on `www.mathworld.com`).

3. Make histograms as in part 1 for 5-by-5, 10-by-10, and 20-by-20 matrices. You have to increase the histogram region (and maybe also the box size `dx`), since the largest eigenvalue grows as $2\sqrt{n}$. Which shape do the distributions seem to converge to?

4. **(Optional)** Write MATLAB code to evaluate (1) for arbitrary n, and plot the curves for $n = 5, 10$, and 20 on top of the histograms in part 3. You might want to compute the Hermite polynomials using the recurrence relation, see Mathworld for details.