Assignment instructions: Completing this assignment requires submitting, for each problem, a written part or a Matlab program, and in some cases, both of these. Any written part should be handed in during class; Matlab programs should be emailed to esam448@u.northwestern.edu. Please comment your code to explain your implementation. Including *all* of the programs for a given assignment in a single email will be appreciated. The collaboration policy is still in effect.

1. For this problem the goal is to generate samples according to the Maxwell distribution,

$$p(x) = \sqrt{2}\pi x^2 e^{-x^2/2}$$

using rejection based upon the Wiebull distribution

$$f(x) = \frac{2x}{\lambda^2} e^{-(x/\lambda)^2} .$$

Choose the scale parameter λ so that $p(x)$ and $f(x)$ have maxima at the same location, and then find a parameter A so that $Af(x) \geq p(x)$ for all x.

(a) What should λ and A be?

(b) Write a Matlab function (using the specific function name lastname_firstname_hw2_prob1)

```
function [samples] = lastname_firstname_hw2_prob1(M)

[...any statements you need...]

end
```

that generates M random samples from the Maxwell distribution.
To test your Matlab function, note that the random variable

$$M = \sqrt{Z_1^2 + Z_2^2 + Z_3^2}$$

will be Maxwellian distributed if Z_1, Z_2 and Z_3 are i.i.d. standard normals. Test the samples generated by your function by comparing it to an equal number of Maxwellian samples generated from standard normals using the two-sample K.-S. test (Matlab: kstest2. (This Matlab function is the one you should email to esam448@u.northwestern.edu.)

(c) Generate 10,000 samples 10,000 times, running the K.-S. test on each set of 10,000 samples compared with 10,000 additional samples generated using Gaussian random samples. How many times out of 10,000 do your generated samples fail the K.-S. test?

(d) Make plots of the histogram of the 10,000 p values using both 20 bins and 50 bins. Do you notice anything unusual?

2. For this problem we would like to evaluate the integral

$$I = \int_0^1 x^{-1/\nu}\, dx = \frac{\nu}{\nu-1} x^{(\nu-1)/\nu}\Big|_0^1 = \frac{\nu}{\nu-1}$$

for $1 < \nu < 2$ using Monte-Carlo integration. Specifically, the goal is to use *uniform* random variables $X_n \sim \mathrm{U}(0,1)$, so that

$$\hat{I} = \frac{1}{M}\sum_{n=1}^{M} X_n^{-1/\nu}\,.$$

An issue with this particular choice, however, is that $Var(X_n) = \infty$, which means that the estimate $\hat{I}$ does not converge to the correct answer at the rate $1/\sqrt{M}$. Write a Matlab function `lastname_firstname_hw2_prob2`

```
function [IQR] = lastname_firstname_hw2_prob2(N,M,nu)

[...any statements you need...]

end
```

that generates N estimates of $\hat{I}$ using M samples X_n and returns the *interquartile range* for the N estimates. Note the interquartile range is $Q3 - Q1$, where $Q2$ is the *median* of the N $\hat{I}$ estimates, $Q1$ is the median of the $\hat{I}$ estimates *smaller* than the median and $Q3$ is the median of the $\hat{I}$ estimates *greater* than the median.

Use your function with $\nu = 1.5$ to generate $N = 100$ estimates with $M = 100$, 1000, 10,000, 100,000 and 1,000,000 and determine the IQR in each case. Use these IQR values to determine the rate at which the estimate $\hat{I}$ converges to the correct value as $M \to \infty$.

3. **Rayleigh flights.** Lord Rayleigh proposed the problem of a three-dimensional random walk, where each step has a fixed length but is made in an arbitrary direction. We will consider 20 steps of length 1. The goal is to determine the probability distribution associated with the total distance from the origin after these 20 steps. (Clearly, this distance is ≤ 20.) In particular, the goal will be to use importance sampling to determine the probability distribution associated with large distances from the origin, when the probability is very small. Here is a procedure to do this:

   (a) Take the first step in an arbitrary direction. You can pick an arbitrary direction by using spherical coordinates: chose the polar angle (i.e., the angle between the pole of the sphere and the direction) randomly so that $\cos\varphi$ is uniformly distributed on $(-1, 1)$, while the azimuthal or equatorial angle ϑ is uniformly distributed on $(0, 2\pi)$. Show either analytically or numerically that this generates a random step with no preferred direction.

   (b) For each subsequent step, use the vector sum of the previous steps to define the direction of the polar axis of a coordinate system, and choose the polar angle of the next step that will be taken using $\cos\varphi_n = 2x^{1/\alpha} - 1$, where x is a uniform random variable in $[0,1]$ and $\alpha \geq 1$ is a biasing parameter. Note that $\alpha = 1$ is the unbiased case. Again take the equatorial angle ϑ to be uniformly distributed on $(0, 2\pi)$.

(c) Use an importance-sampled Monte-Carlo simulation[1] to estimate the probability distribution for the total distance. Do this by estimating the probability that the distance lies within a set of bins $r_i < r < r_{i+1}$, where $r_i = i\,\Delta r$, i.e.,

$$p_{i+\frac{1}{2}} = \frac{1}{\Delta r}\int_{r_i}^{r_{i+1}} p_r(r)dr = \frac{1}{\Delta r}\int I_i(r)\,p_{\vec{u}}(\vec{u})\,d\vec{u}\,,$$

where $I_i(r)$ is an indicator function for the distance being in the ith interval and $\vec{u}$ denotes the uniform random variables used to generate the samples. Plot the distribution obtained in this way using 100 bins. Note that $p_{i+\frac{1}{2}}$ should be centered at $r_{i+\frac{1}{2}}$.

(d) To obtain a good distribution you should use a sufficient number of samples, multiple biasing parameters and the balance heuristic, as discussed in class and in the online notes.

(e) To help check things, plot along with your results the asymptotic result for large values of r,

$$p_N(r) \sim \frac{\rho\, L^{-1}(\rho)}{r_{\max}\sqrt{(\pi/2N^3)(1-\rho^2-2\rho/L^{-1}(\rho))}}\;e^{-N\rho L^{-1}(\rho)}\left(\frac{\sinh[L^{-1}(\rho)]}{L^{-1}(\rho)}\right)^N,$$

where $\rho = r/r_{\max}$ and $L(x) = 1/\tanh x - 1/x = \rho$ (so $L^{-1}(\rho) = x$; note, therefore, that you can plot $p_N(r)$ vs. ρ *parametrically* since both can be expressed explicitly in terms of x). There is also an exact solution to this problem, but it is an infinite series that can be difficult to compute at small probability.

(f) To help check the results another way, also plot the estimated *coefficient of variation* in each bin (the standard error of the estimate — or, the standard error of the mean — divided by the estimated probability). This gives an idea of the relative accuracy of the estimated probability in each bin.

(g) Write and submit (to `esam448@u.northwestern.edu`) a Matlab function (it will need no input arguments) that returns a vector of the estimated probabilities and the estimated coefficient of variation in each of the 100 bins. Use the name `lastname_firstname_hw2_prob3.m`

Note: Using the law of conditional probabilities, i.e., $p(AB) = p(A|B)p(B)$, we can express a joint probability distribution as

$$p(u_1,\ldots,u_N) = p(u_N\,|\,u_{N-1},\ldots,u_1)\,p(u_{N-1}\,|\,u_{N-2},\ldots,u_1)\cdots p(u_2\,|\,u_1)\,p(u_1)\,.$$

Thus, even if the random variables aren't independent, we can still write the joint pdf as a product *if* each random variable is selected using a conditional probability distribution. Importance sampling can be done with products of the ratios of such conditional distributions.

[1] See the note at the end of this problem