Members: Wesley Foster

Getting Started

Output:

```
________________________________
Integral Estimates: (Uniform)
Q1 [-10,10] f(x)=1  :  (20.0, 0.0)       Actual integral(-10, 10)(x)dx = 20
Q2 [-1,1] f(x)= eval legendre(l,x)^2 for[1,6]:
         l=0   :   (2.0, 0.0)
         l=1   :   (0.6548017020381981, 0.01874461260040634)
         l=2   :   (0.4021080336652551, 0.013727488323590843)
         l=3   :   (0.2986663305884823, 0.011147180587216404)
         l=4   :   (0.21947783338001695, 0.0080847151429523)
         l=5   :   (0.19108555392616977, 0.007974893931008337)
         l=6   :   (0.14993247113252006, 0.005818925639733308)
Q3 [-10,10] f(x)=standard normal PDF with mean of 0 and variance 1  :  (1.0168529697428152, 0.06808645948155576)
Q4 [-10,10] f(x)=0.7p1(x) + 0.3p2(x)  :  (1.0138582029524243, 0.04971164691275888)


________________________________________
Integral Estimates 1: (Truncated Normal)
Q1 [-10,10] f(x)=1  :  (10.118209236912707, 4.636315322855519)   Actual integral(-10, 10)(x)dx = 20
Q3 [-10,10] f(x)=standard normal PDF with mean of 0 and variance 1  :  (1.0, 5.508765101882708e-18)
Q4 [-10,10] f(x)=0.7p1(x) + 0.3p2(x)  :  (0.8861453938635313, 0.11302927161768314)
```

Q1:
The uniform distribution did a much better job of estimating the value of the integral, this is likely due to the fact that the function f(x)=1 is a uniform distribution. It may have been more accurate to have the truncated normal distribution estimate 1/20th of the function and then multiply it by 20.
Q2:
The l=0 entry had no error within the uniform distribution and thereafter had very little error.
Q3:
While the uniform distribution did a relatively good job at estimating the standard normal PDF integral (or CDF), the truncated normal distribution had a minuscule error which is likely due to the PDF being a normal distribution. In both estimates (because the standard normal PDF is uniformly distributed) I only had the integral estimate half of the function and multiplied the error and estimation by two.
Q4:
The uniform distribution unexpectedly did better on question four, this is likely because of how I split this distribution, instead of splitting the functions into two segments that are smaller than 10 units I split the two functions at their maximum without their coefficients and then multiplied the result of the integral estimate by 2*c where c is the respective coefficient of the function.

Main Project

Rosenbrock Function:

```
Rosenbrock Function:
(0.29534608148904135, 0.012289056940422234)

Using Half Bounds:
(0.3012442968737623, 0.010272733029785971)
```

To evaluate the Rosenbrock function- originally I set up the integral normally and just raised e to the result of the provided ln_f function, this provided the top answer at 10^5 number of points evaluating from [-5,5]^2. This answer is fairly accurate and while not the most accurate out of the two is a decent approximation. The second way I set up the Rosenbrock function was by using half bounds (though the word half is not entirely accurate here), I did everything the same as the previous method but set the bounds from [-5,1]^2 because the minimum and point of reflection of the function is at 1, I then doubled the result and error in order to get the answer. This works well for smaller integrals because the error from the not fully symmetrical evaluation is minimal compared to if the entire function was evaluated on one side. Both functions increase in accuracy in a fairly proportional way, the following picture shows the result of doing 10^7 number of points for both techniques:

```
Rosenbrock Function:
(0.3020281137285295, 0.0012473522680202544)

Using Half Bounds:
(0.30685902544424765, 0.0010488231121489155)
```

It can be seen that the accuracy of the original picture is nearly exactly 10 times the accuracy of the second picture.