# Comparing the performance of Tilted Importance Sampling with Classical Monte Carlo simulation

Fahad Bin Mostafa

Department of Mathematics and Statistics

The Texas Tech University, USA.

Abstract:

Monte Carlo simulation is a powerful statistical tool and widely used in both non-engineering fields and engineering fields. It can deal with large number of random variables, various distribution types as well as highly non-linear engineering models. Monte Carlo simulation perform random sampling and conduct a large number of experiments on computer. On the other hand, importance sampling is a general technique for estimating properties of a particular distribution, while only having samples generated from a different distribution than the distribution of interest. It is related to umbrella sampling in computational physics. In this study, our main aim is to estimate the right tail of an exponential random variable using technique from importance sampling and tilted distributions. In our problem, we want to estimate the probability $\theta = P\,r\{X > a\} = \wedge \int_a^{\infty} e^{-\lambda x} dx$ where we will consider $X$ is an exponential random variable with intensity $\lambda$ (mean $\frac{1}{\lambda}$) with any constant $a > 0$. Since the event $X > a$ occurs with a very small probability, that is $\theta$ is very small, the use of classical Monte Carlo approach will not be efficient. Instead, it is better to use importance sampling with tilted density serving as the density to sample form. In our study, we wish to show computational time as well as accuracy of the estimation between two methods.

Methodology:

Importance Sampling Technique is a well-known tool for sampling. The central idea of importance sampling is to sample the random variables according to an alternative sets of distributions such that more samples will be in the failure region. More Samples will therefore contribute to the probability estimation.

Let $x = (x_1, x_2, \dots x_n)$ then

$$\theta = E[h(X)] = \int h(x) f(x) dx$$

$$\approx \frac{1}{N}\sum_{i=1}^{N} h(X_i)$$

If $X_i \sim F(X)$ and $F(x)$ is the cumulative distribution function for $f(x)$. For many problems, $F$(x) is difficult to sample when $Var(h)$ is large.

If there is a related and recognized pdf $g(x)$ is available, then it is better to use.

$$\theta = E_g\left[\frac{h(x)f(x)}{g(x)}\right] = \int \frac{h(x)f(x)}{g(x)} g(x)\, dx$$

$$\approx \frac{1}{N}\sum_{i=1}^{N} \frac{h(X_i)f(X_i)}{g(X_i)}$$

Where, $X_i \sim G(X)$, which is associated cdf $G(X)$ of pdf $g(x)$. So, now we stuck in a question that what is importance sampling in a sense. The answer is, if the $Var[\frac{h(x)f(x)}{g(x)}]$ is small, $g(x)$ samples concentrated where $h(x)f(x)$ is important. Here $\hat{\theta}_N$ is the importance sampling estimator of $\theta$. If we want to produce $N$ number of values of $X$ from the density $g(x)$ then we have the following output,

$$\hat{\theta}_N = \frac{1}{N}\sum_{i=1}^{N} \frac{h(X_i)f(X_i)}{g(X_i)}$$

In case of tilted density, we can consider any scalar $t$ such a way where the tilted density is

$$f_t(x) = \frac{e^{tx} f(x)}{M(t)};\ \ t > 0$$

In our problem, $f(x) = \lambda e^{\wedge x}$.

Now we show compute $M(t)$

$$M(t) = \wedge \int_0^{\infty} e^{tx} e^{-\lambda x} dx$$

$$= \wedge \int_0^{\infty} e^{-(\lambda - t)x} dx$$

$$= -(\frac{\lambda}{\lambda - t})[0 - 1]$$

$$= \left(\frac{\lambda}{\lambda - t}\right)$$

So now, our tilted density becomes

$$f_t(x) = \frac{e^{tx} f(x)}{\left(\frac{\lambda}{\lambda - t}\right)}; \;\; t > 0$$

$$= (\lambda - t)e^{-(\lambda - t)x}$$

If we compare this tilted density with standard exponential density, then the mean of $f_t(x)$ is $\frac{1}{\lambda - t}$. Using the mean of $f_t(x)$ we can calculate the optimal amount of $t$ to eastimate $\theta$ for the given value $a$. Let's call it $t^*$. Thus, $a = \frac{1}{\lambda - t}$

$$t = \lambda - \frac{1}{a} = t^*(say)$$

Let us denote $\widetilde{\theta_N}$ is the Monte Carlo estimator that uses samples from $X_i \sim f_{t^*}(x)$ where we have the exponential distribution. So, we know it's mean is $\frac{1}{\lambda - t^*}$ which implied $a = \frac{1}{\lambda - t^*}$ and once again

$$\lambda - \frac{1}{a} = t^*$$

$X_i$ has exponential distribution with mean $a$. Using the method we explained above, will generate the $X_i$ from the density of $alog(U)$

The $\widetilde{\theta_N}$ looks like

$$\widetilde{\theta_N} = \{\frac{\lambda}{n(\lambda - t^*)}\} \sum_{i=1}^{N} I_{[a,\infty)}(x_i)M$$

Where, $M = e^{-t^* x_i}$; $i = 1,2, \ldots$

$X_i$ has exponential distribution with mean $a$.

$$\widetilde{\theta_N} = \sum_{i=1}^{N} I_{[a,\infty)}(x_i) e^{-(\lambda - \frac{1}{a})x_i} . \frac{a\lambda}{n}$$

Let's talk about Classical Monte Carlo Simulation. To discuss it, we introduce the following flow chart below.

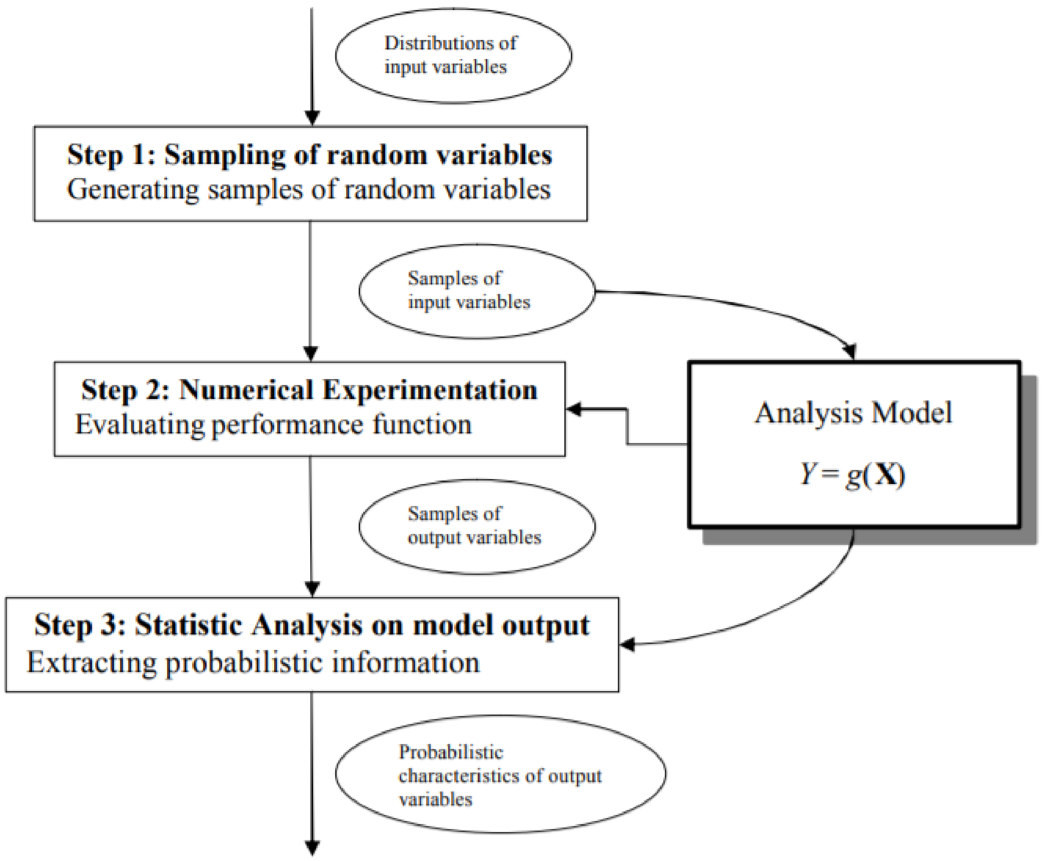


Figure 0: Classical Monte Carlo Simulation

Classical Monte Carlo estimator uses samples from $f(x)$ directly. $\widehat{\theta}_N$ is the Classical Monte Carlo estimator, the $X_i \sim Exp(\frac{1}{\lambda})$

$$\widehat{\theta}_N = \sum_{i=1}^{N} I_{[a,\infty)}(x_i)$$

Now, we are going to compare and $\widetilde{\theta_n}$ with each other, most importantly the Classical Monte Carlo estimator that uses samples from $f(x)$ directly. Particularly we compared the $\widetilde{\theta_n}$ with $\widehat{\theta}_N$.

We will use MATLAB to implement the Classical and the importance-sampling estimators.

We consider the algorithm following:

Step 1: Define a, $\lambda$(lamda), t and the moment generating function ($M(t)$).

Step 2: Considering the sample size, in our case, $n = 10000$.

Step 3: Introduce for loop to generate data from exponential random variable and assign $\hat{\theta}_N$ and $\widetilde{\theta_n}$ for calculating its values. We have calculated the mathematical output of it for our problem.

Step 4: Display it by using MATLAB build in comments {say disp ($\hat{\theta}_N$)}.

Step 5: Plot $\hat{\theta}_N$ and $\widetilde{\theta_n}$ against sample size $n$.

Result Discussion:

After implementing the classical and importance sampling estimator in MATLAB, we can show results in the followings. We considered two things mainly, firstly the computational time and finally the accuracy of the estimation.

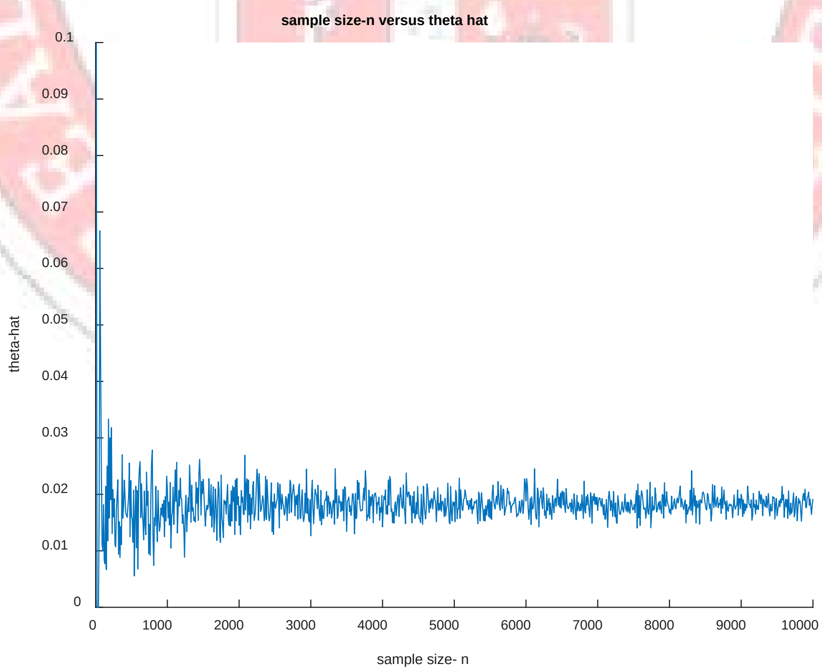


Figure-1: Convergence of Classical Monte Carlo when $a = 4$ & $\lambda = 1$.

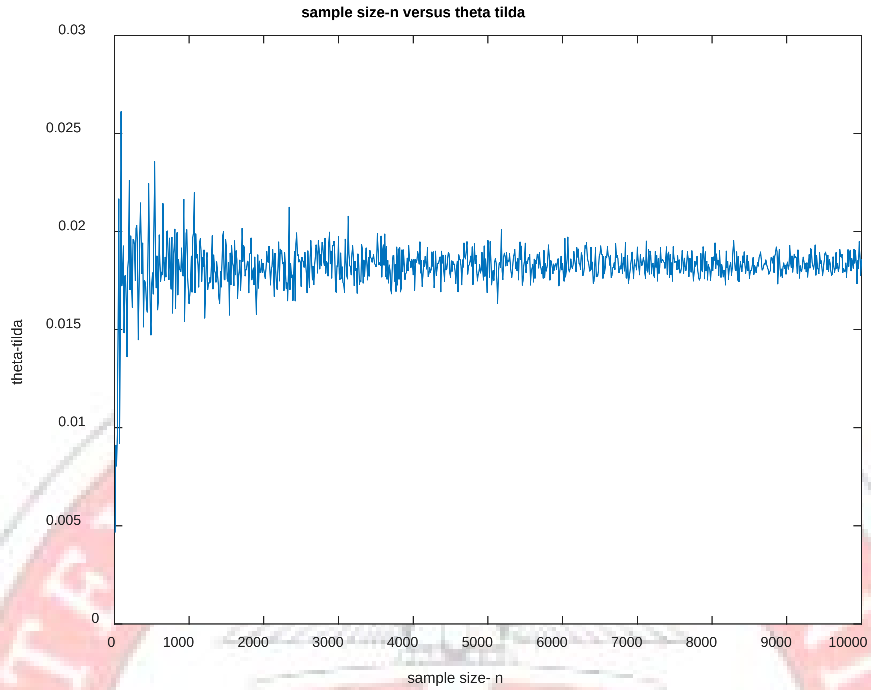


Figure-2: Convergence of Importance Sampling with tilted density when $a = 4$ & $\lambda = 1$.

From Figure-1 and Figure two we can understand the convergence of two technique easily. If we increase the sample size $n$, then both converges to a particular value. However, Classical MC shows less performance than that of Importance Sampling. If we analyze the graph closely then we see that, when the value of $n$ is in between 0 to 2000 them it shows more fluctuation which means that we have larger variance however, it declines slowing for both case. If we compare the graph with the Table-1, then it seems to be perfect.

Table-1: Computation Time and Accuracy of Estimation

| | Original Estimate | Classical Monte Carlo | Importance Sampling |
|---|---|---|---|
| Computational Time (sec.) | NA | 0.820890000000000 | 1.199001000000000 |
| Accuracy of Estimation | 0.0183 | 0.019200000000000 | 0.018599993274431 |

From Table-1 we see that both Classical Monte Carlo Method and Importance Sampling converges approximately to 0.019200000000000 and 0.018599993274431 respectively where Classical Monte Carlo technique take relatively less time than Importance Sampling. We took different values of $a$

To compare our result with different $a$ we have shown many figures.

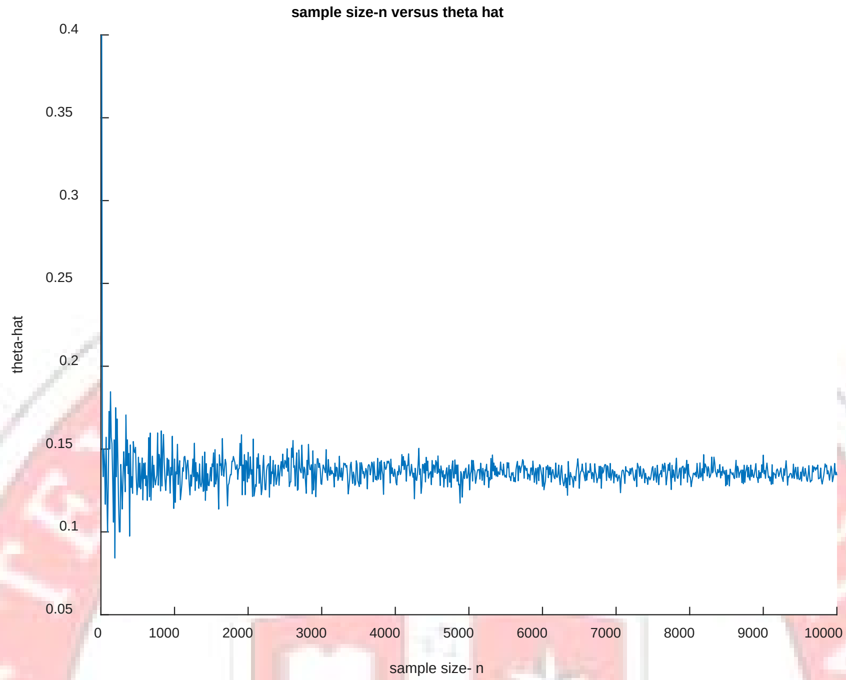


Figure-3: Convergence of Classical Monte Carlo when $a = 2$ & $\lambda = 1$.

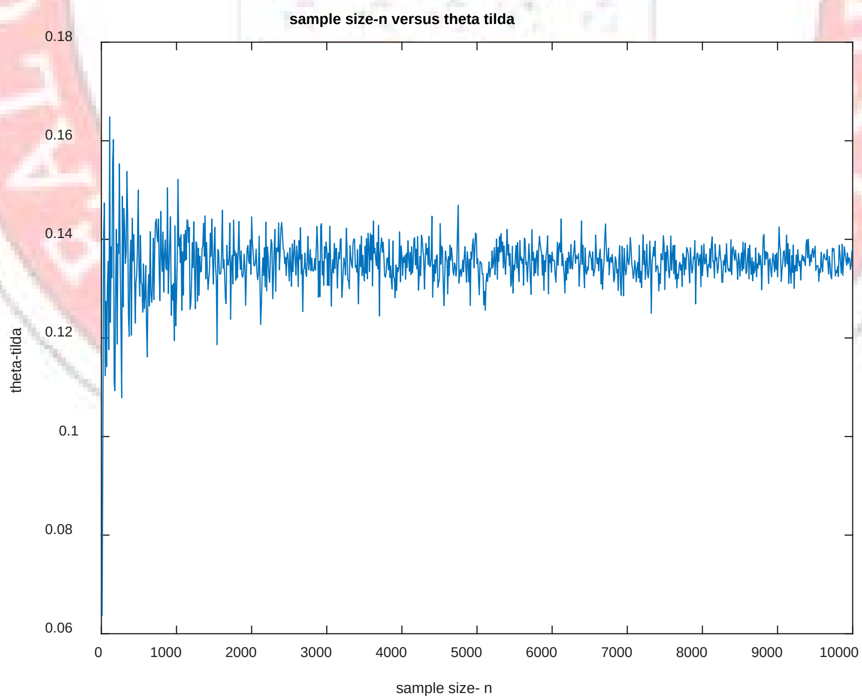


Figure-4: Convergence of Importance Sampling with tilted density when $a = 2$ & $\lambda = 1$.

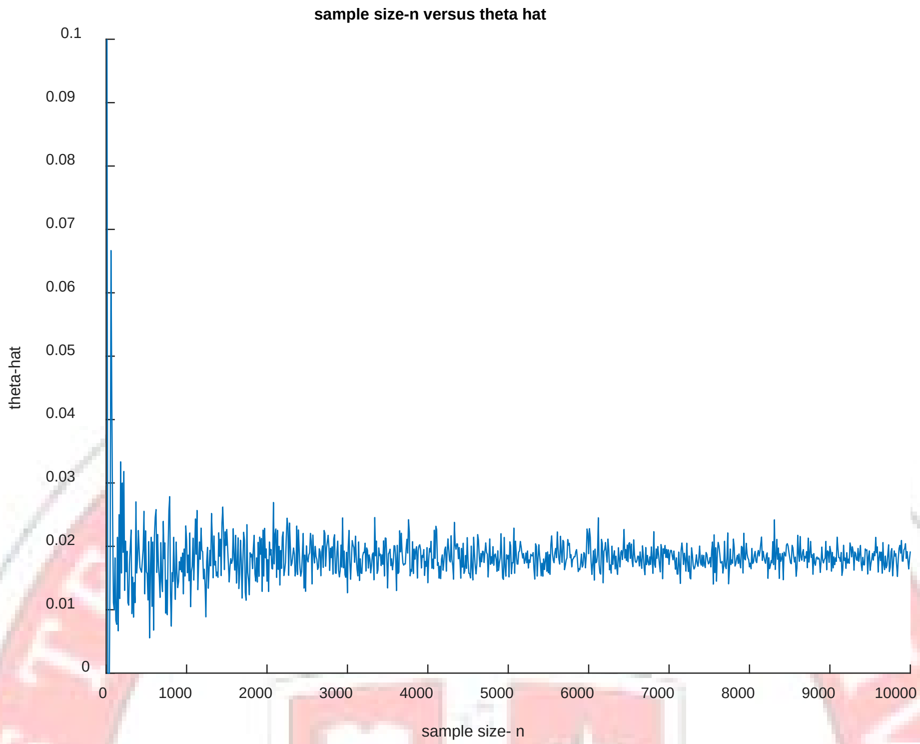


Figure-5: Convergence of Classical Monte Carlo when $a = 4$ & $\lambda = 1$.

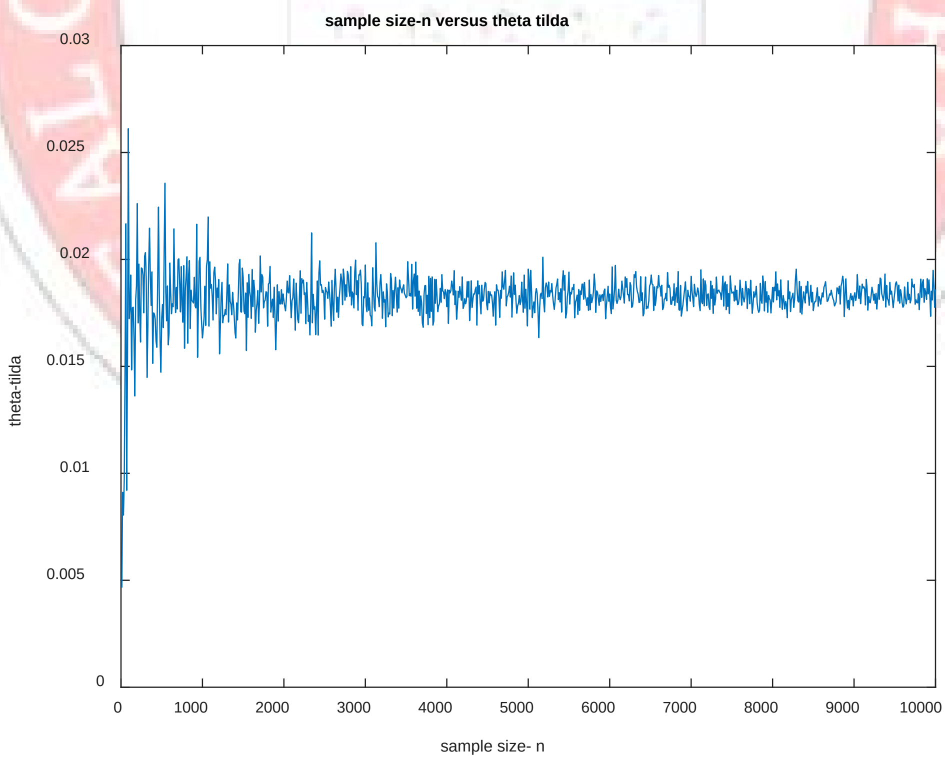


Figure-6: Convergence of Importance Sampling with tilted density when $a = 4$ & $\lambda = 1$.

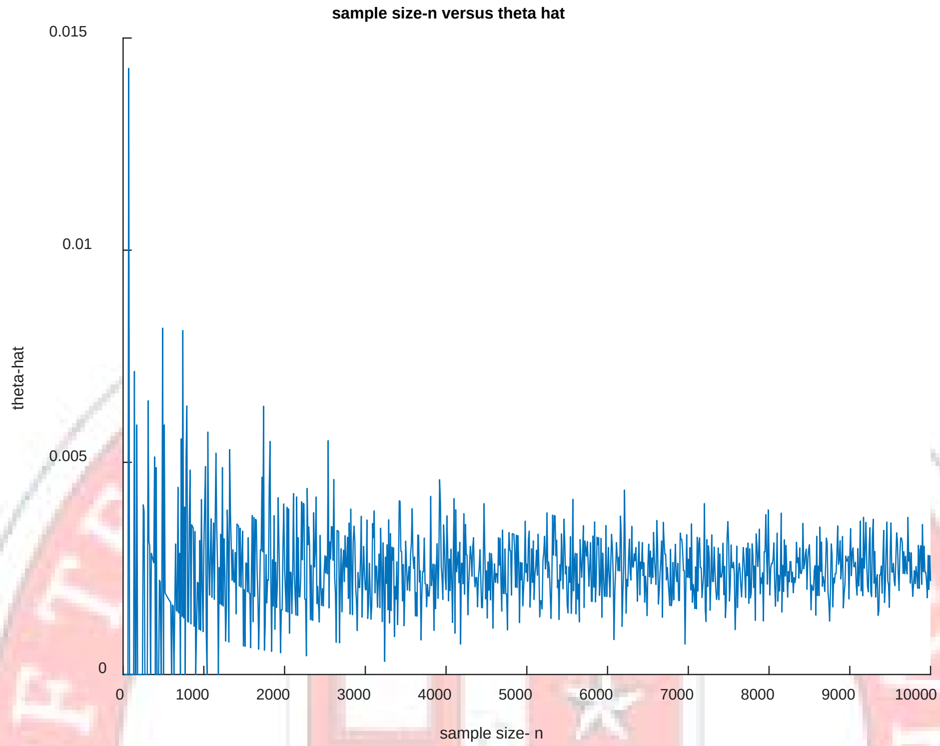


Figure-7: Convergence of Classical Monte Carlo when $a = 6$ & $\lambda = 1$.

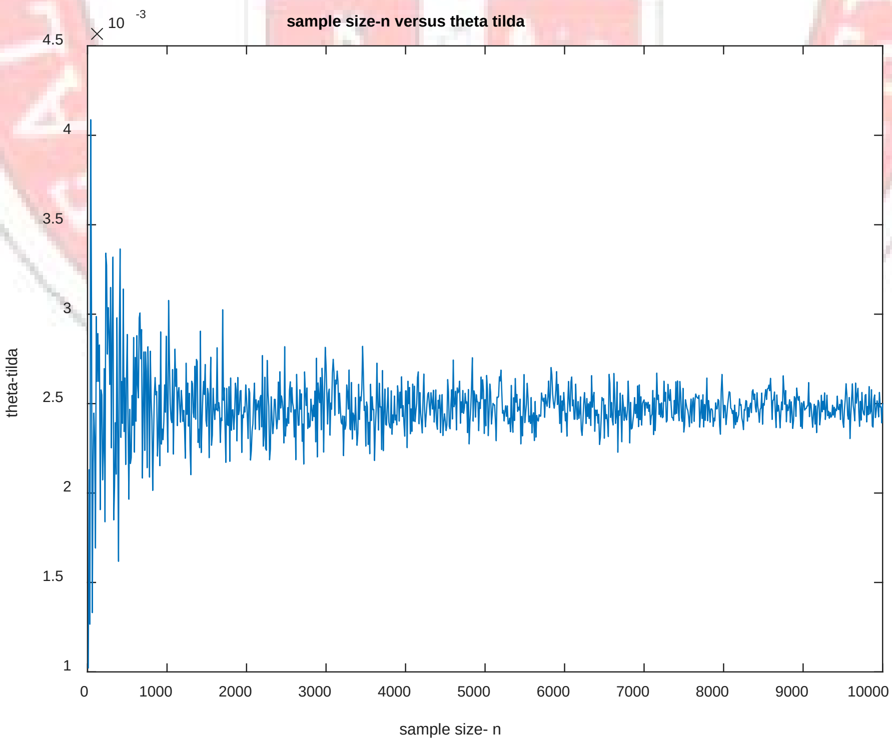

Figure-8: Convergence of Importance Sampling with tilted density when $a = 6$ & $\lambda = 1$.

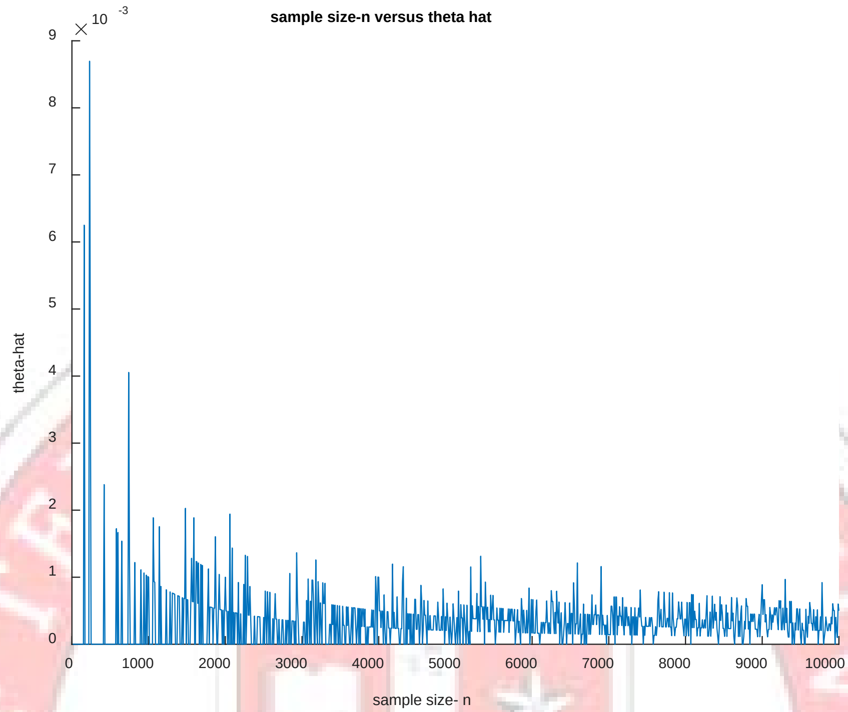


Figure-9: Convergence of Classical Monte Carlo when $a = 8$ & $\lambda = 1$.

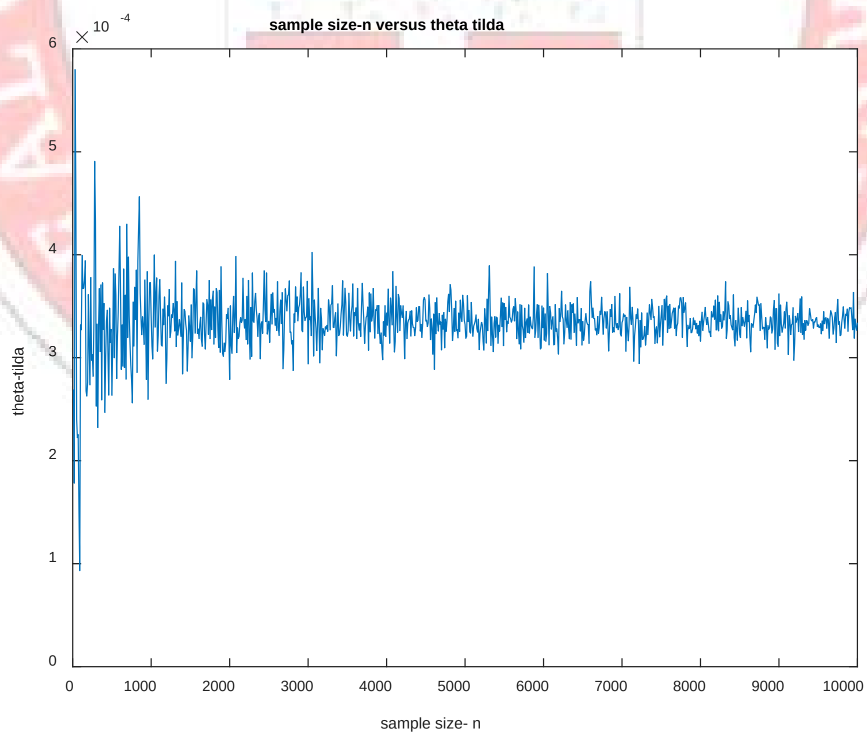


Figure-10: Convergence of Importance Sampling with tilted density when $a = 8$ & $\lambda = 1$.

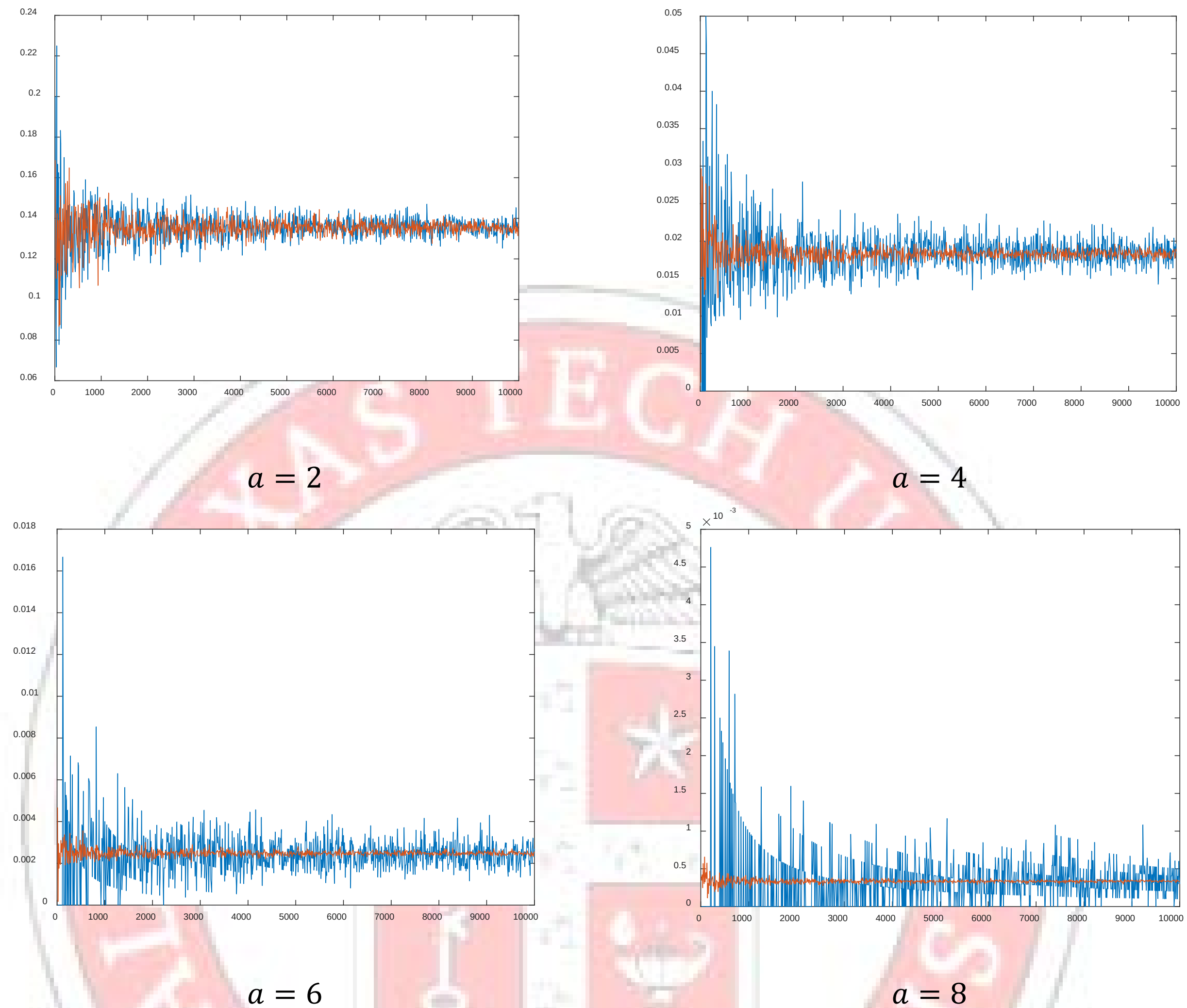


Figure 11: Comparison between Classical Monte Carlo and Impotence Sampling for different values of $a$.

In the Figure 11, we show the comparison between Classical Monte Carlo and Impotence Sampling for different values of $a$ and it is quite clear that the blue plots have more variance than the red plots. It is important to note that dark black is indicting Classical Monte Carlo and light black is indicating Importance Sampling.

Table-2: Accuracy of Estimation

| $a$ | Original Estimate | Classical Monte Carlo | Importance Sampling |
|---|---|---|---|
| 2 | 0.1377 | 0.134200000000000 | 0.138084468527881 |
| 4 | 0.0183 | 0.019200000000000 | 0.018599993274431 |
| 6 | 0.0024 | 0.002200000000000 | 0.002502389167019 |
| 8 | $4.00e^{-4}$ | 5.000000000000e-04 | 3.38154030683e-04 |

Table-3: Time of Computation in seconds

| $a$ | Classical Monte Carlo | Importance Sampling |
|---|---|---|
| 2 | 0.771849 | 1.223210 |
| 4 | 0.820890 | 1.199001 |
| 6 | 0.796049 | 1.179852 |
| 8 | 0.815366 | 1.166194 |

Acknowledgement:

I want to say special thanks to our wonderful course instructor who clarify all the materials in his class.

Conclusion:

Importance sampling is more efficient approach to simulation. In essence, we can take an alternative distribution for estimating our desired distribution. Although Classical Monte Carlo approach takes less computational time to calculate than that of Importance Sampling but Importance Sampling shows improved computational efficiency and our study bolsters this concept. In one sentence, Classical Monte Carlo technique is computationally inefficient relative to Importance Sampling with tilted density when samples fall into the failure region which can contribute to the probability estimation.

Appendix

Relevant MATLAB codes

```
clear all;
clc;
a=8;
lamda = 1;
t=lamda -(1/a);
mt= lamda/(lamda-t);
tic  % time of convergence
m=1000;
for j=1:m
    n=10*j;
    u=rand(1,n);
    expo=-log(u); % generating data from exponential
```

```
%classical monte carlo estimate
for i=1:length(expo)
if expo(i)>=a
    gx1(i)=1;
else
    gx1(i)= 0;
end
end
theta_hat(j)=(1/n)*sum(gx1);   % monte carlo
end
toc
figure(1), clf;
hold on;
plot(10*[1:m],theta_hat)
xlabel('sample size- n')
ylabel('theta-hat')
title('sample size-n versus theta hat')
theta_hat(j)
format long

%Importance Sampling estimate

for j=1:m
    n=10*j;
    u=rand(1,n);
    x = -a*log(u); % generating data from exponential

%tilted sampling
for i=1:length(x)
if x(i)>=a
gx(i)=exp(-t*x(i));
else
gx(i)= 0;
end
end
   theta_tilda(j)=(mt/n)*sum(gx);
end
toc;
figure(2)
plot(10*[1:m],theta_tilda)
%plot(theta_tilda)
xlabel('sample size- n');
ylabel('theta-tilda')
title(' sample size-n versus theta tilda')

theta_tilda(j)

format long
```

End