A CONDITIONAL EXPECTATION

LARRY, ZHIRONG LI

Prove that

$$\frac{x+\sqrt{x^2+\frac{8}{\pi}}}{2}<\mathbb{E}\left[X\,|X>x\right]<\frac{x+\sqrt{x^2+4}}{2}$$

where $X\sim N\left(0,1\right)$ and $x>0$.

Proof. Based on the definition of the conditional expectation, $\mathbb{E}\left[X\,|X>x\right]$ can be written as

$$\mathbb{E}\left[X\,|X>x\right]=\int_x^\infty s\cdot d\mathbb{P}\left(X\in ds\,|X>x\right)$$

because X is a continuous random variable, $\mathbb{P}\left(X=s\,|X>x\right)\equiv 0$, the trick is to write

$$\begin{aligned}
d\mathbb{P}\left(X\in ds\,|X>x\right) &= d\frac{\mathbb{P}\left(X\in ds,X>x\right)}{\mathbb{P}\left(X>x\right)}\\
&= \frac{d\mathbb{P}\left(X\in ds,X>x\right)}{\mathbb{P}\left(X>x\right)}\\
&= \frac{1}{1-N\left(x\right)}\frac{1}{\sqrt{2\pi}}\exp\left(-\frac{s^2}{2}\right)\cdot ds
\end{aligned}$$

Therefore

$$\begin{aligned}
\mathbb{E}\left[X\,|X>x\right] &= \frac{1}{1-N\left(x\right)}\frac{1}{\sqrt{2\pi}}\int_x^\infty s\exp\left(-\frac{s^2}{2}\right)ds\\
&= \sqrt{\frac{1}{2\pi}}\frac{1}{1-N\left(x\right)}\exp\left(-\frac{x^2}{2}\right)
\end{aligned}$$

□

We examine the numerical values for $x\in 0:0.01:10$ since the probability for x to be out of $\pm 3\sigma=\pm 3$ around the mean is very small.

Based on the fact that

$$erfc\left(x\right)=\frac{2}{\sqrt{\pi}}\int_x^\infty e^{-t^2}dt$$

and

$$\frac{2}{\sqrt{\pi}}\frac{e^{-x^2}}{x+\sqrt{x^2+2}}<erfc\left(x\right)\le\frac{2}{\sqrt{\pi}}\frac{e^{-x^2}}{x+\sqrt{x^2+\frac{4}{\pi}}}$$

Date: Dec. 13, 2011.

Algorithm 1 Calculate the bound

```
clear all,close all;
x=0:0.01:18;
expectedX=sqrt(1/pi/2)./(1-normcdf(x)).*exp(-x.^2/2);
upperBound=x+sqrt(x.^2+4);
lowerBound=x+sqrt(x.^2+8/pi);
figure(666);
plot(x,expectedX,'-r',x,upperBound,'-g',x,lowerBound,'-b');
grid on;
xlabel('x');
ylabel('numerical value');
legend('conditional expectation for x','upper bound','lower bound');
title('The boundedness of conditional expectation of standard normal');
```

We can get that

$$\mathbb{E}\left[X\left|X>x\right.\right]=\sqrt{\frac{1}{2\pi}}\exp\left(-\frac{x^2}{2}\right)\cdot\frac{1}{\frac{1}{2}erfc\left(\frac{x}{\sqrt{2}}\right)}$$

hence the conclusion can be drawn, i.e.,

$$2\frac{\sqrt{\pi}}{2}e^{\frac{x^2}{2}}\left(\frac{x}{\sqrt{2}}+\sqrt{\frac{x^2}{2}+\frac{4}{\pi}}\right)\le\frac{1}{\frac{1}{2}erfc\left(\frac{x}{\sqrt{2}}\right)}<2\frac{\sqrt{\pi}}{2}e^{\frac{x^2}{2}}\left(\frac{x}{\sqrt{2}}+\sqrt{\frac{x^2}{2}+2}\right)$$
$$\Downarrow$$
$$\frac{x+\sqrt{x^2+\frac{8}{\pi}}}{2}\le\sqrt{\frac{1}{2\pi}}\exp\left(-\frac{x^2}{2}\right)\cdot\frac{1}{\frac{1}{2}erfc\left(\frac{x}{\sqrt{2}}\right)}<\frac{x+\sqrt{x^2+4}}{2}$$

and the diagram is illustrated as below:

Notice that there is an oscillation effect around 8 and the red line disappears after that. Actually the instability problem comes from Matlab's (any software's) finite precision, in Matlab, the eps =2.2204e-016, not a big tolerance value at all and $1-normcdf\,(5.5)\approx 0$.

The Monte Carlo simulation for computing the conditional expectation is given by the following Matlab algorithm.

and the simulated results are consistent for $x\in(0,4)$ and Monte Carlo method has large deviation from 4 and gets cut off after 5.5.

As you can see from the Figure 0.2, (in green for monte carlo), for large values (5.5 or above) Monte Carlo method can not find sufficiently large number of condition-satisfying points to do the sample mean, since the probability for X to be greater than 5.5 is almost 0, i.e., in Matlab , type in normcdf(5.5), you will get ans =1.0000.

References

[1] Erfc: http://mathworld.wolfram.com/Erfc.html

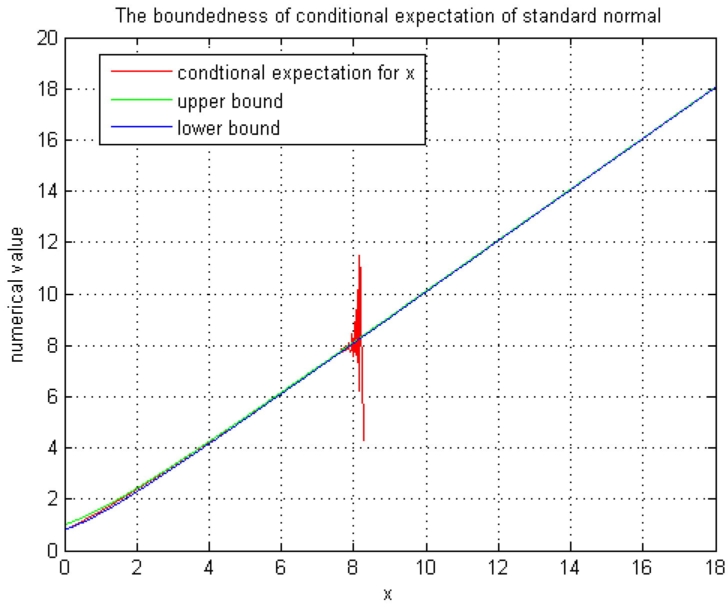

FIGURE 0.1. The theoretical values and the bounds

Algorithm 2 Monte Carlo Simulation for Conditional Expectation

```
% Monte Carlo verification of the conditional expectation
t=0:0.01:10;
calculatedExpectation=sqrt(1/pi/2)./(1-normcdf(t)).*exp(-t.^2/2);
monteCarloExpectation=zeros(size(t));
%number of points generated in simulation, notice that if t is taking large
%value, the probability for the random point in (x,\infty) is decreasing
num_points=1e7;
% Prefer to use vectors since Matlab excels in this
generatedRandomPoints=randn(1,num_points);
for i=1:length(t)
    isLargerThanX=generatedRandomPoints>t(i);
    isLargerThanXIndex=find(isLargerThanX==1);
    monteCarloExpectation(1,i)=sum(generatedRandomPoints(isLargerThanXIndex))
    /length(isLargerThanXIndex);
end
figure(888);
plot(t,calculatedExpectation,'-r',t,monteCarloExpectation,'-g');
grid on; xlabel('x'); ylabel('numerical value');
legend('calculated expectation for x','monte carlo simulation for x');
title('Monte Carlo Verification of Conditional Expectation');
```

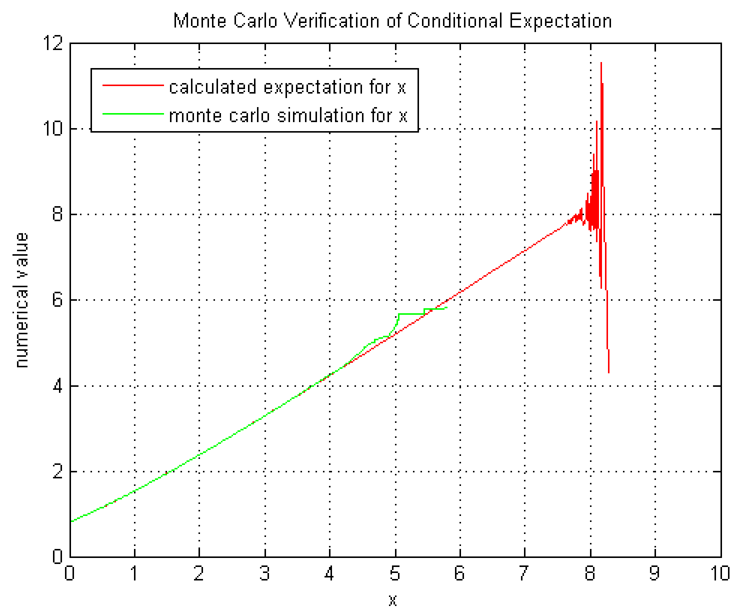

Figure 0.2. Monte Carlo vs Theoretical Value