Computational Analysis in Stochastic Model of Risk and Profit Problems

Zhijun Yang
Faculty Adivisor: David Aldous

Abstraction

This research project is about analysis of risk and profit from games that can be formulated from mathematical model of stochastic process. We will approach to this kind of problems primarily from computational approach. I will simulate large set of data and by analyzing these data by employing several mathematical and statistical methods to reach our conclusion. This research is primarily concerns with problems arising in analysis of risk and profit from games, we are concern about the optimal strategy interms of maximal return or minimal risk in the those games that can be formulated in terms of Random Walk Model, Markov Chain Model or general Stochastic Model.

Model of Random Walk in Risk and Profit Problem

Let's consider the following problem arising in many real world situation. In a game of finite steps, where person A bet amount of X at each step i, the probability that person A win X amount is p at each of the step i, and probaility person A lose the bet X is $1-p$. Suppose A have amount of M dollars before enter the game, we are concerned about what kind of strategy or mixed strategies that person A can employ to maxmamize his or her profit after a finite step, let's say N and what kind of strategy or mixed strategy that person A can minimize its lose.

In order to formulate our best possible strategy, we first need some mathematical defnition and formulation of this problem. Let's first consider the definition of random walk. Let $X_1, X_2, ..., X_N$ to be identical independent distribution random variables, and let $S_n = X_1 + X_2 + ... + X_n$. Then S_n is called a random walk. In short, a random walk is just a sum of independent random variables we immediately see that our game above can be formulated in terms of the random walk model, where X_i are just independent Bernoulli distribution with parmeter p.

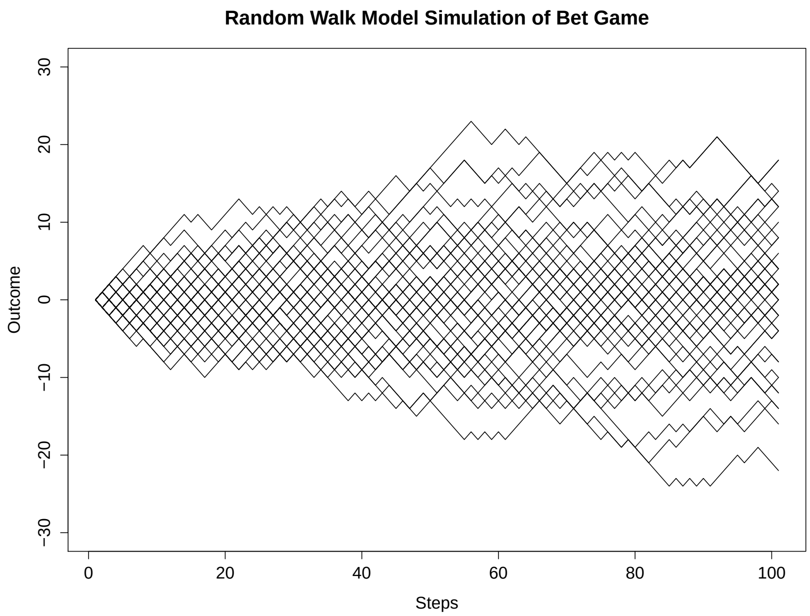
The naive approach would be continue betting money in each step of the game until the game is over or until lose all the money, this strategy would be fine if the probability of wining is high, to see this, we can just take the expectation of the distribution of the random walk model:

$$\begin{aligned} E(X) &= E(M + S_n) = E(M + \sum X_i) \\ &= M + \sum E(X_i) = M + n \times (p \times X + (1-p) \times (-X)) \\ &= M + n \times (2p-1) \times X \end{aligned}$$

It is evidently that if the probability of wining is above $\frac{1}{2}$, then over the long run, we expect the person A win. However, our intuition tells us this is not the optimal strategy in terms of maximaze the profit and minimaze the lose.

We realized that in this random walk game situation, the only strategy type for person A is to determine when to leave this game. However, the future step of random walk X_{i+1} is independent of any of its past step X_j where $0 \leq j \leq i$, thus we are unable to determine when is right to leave the game from the mathematical formulation of the game. This is when our computational approach take place into this problem.

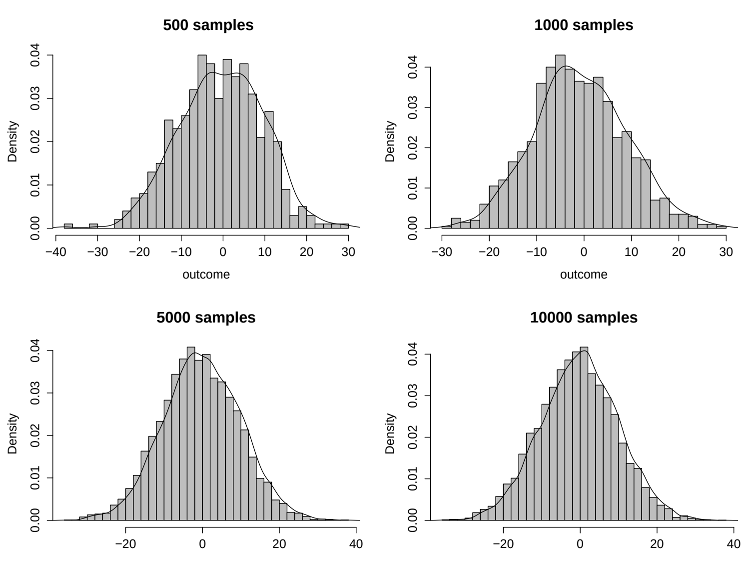
Now we want reach some emprically conclusion using simulation method. For simplicity let's assume this game is fair, that is p = 0.5, and we assume that person A have complete information about this game and let's first consider the case that M is sufficient large comparing to X, such that person A will not encounter running out of money in the game. The following is a simulated graph using 50 samples.



Now we have a certian intuition that as the sample size getting larger, we can approximate our outcome using the normal distribution with mean at our expected value. Actually, this is a generalization of the strong law of large number, let Y_i be a sample of our random walk, then by the ergodic thereom we have

$$lim_{n\to\infty}\frac{1}{n}\sum_{k=1}^{n} Y_k = EY_1$$

In addition, it is evidently that we can use binomial formula to approximate random walk sample, and since as the sample size become sufficiently large, we can use normal distribution to approximate our binomial formula, we expect the distribution of our sample data to have normal distribution. To see this emprically, let's start some simulation with large size of sample data. The graphically display is shown as following:



Thus the emprical data confirm our mathematical formualtion that this random walk game can be estimated use the normal distribution. Since we already able to determine the expectation, the only parameters we need is the standard deviation. Instead of deriving it mathematically, we can take advantage of our simulation data to computing it. Indeed, after we take our sample size large enough, the sample standard deviation approach to the true population parameter. In our case, after computation, we find our standard deviation for this example is about 10.

So now given the game with wining chance p = 0.5 and with unit payoff X = 1, person A is able to predict his or her predicted outcome under the normal curve approximation with a expectation of 0 and standard deviation of 10. A shift of the winning chance p or a scale of payoff X just have the effect of shift the mean and scale the standard deviation of our noral approximation.

General Model of Stochastic Process in Risk and Profit Problem I

Let's consider a general type problem arising in many real world situation. In a game of finite steps, where person A bet amount of X_i at each step i, the probability that person A win X_i amount is p at each of the step i, and probaility person A lose the bet X_i is $1 - p$. Suppose A have amount of M dollars before enter the game, we are concerned about what kind of strategy or mixed strategies that person A can employ to maxmamize his or her profit after a finite step, let's say N.

One of the naive approach would be bet all the money in each step of the game, this strategy would be fine if the number of steps are short and the probability of wining is high, to see this, we can just take the expectation of geometric distribution:

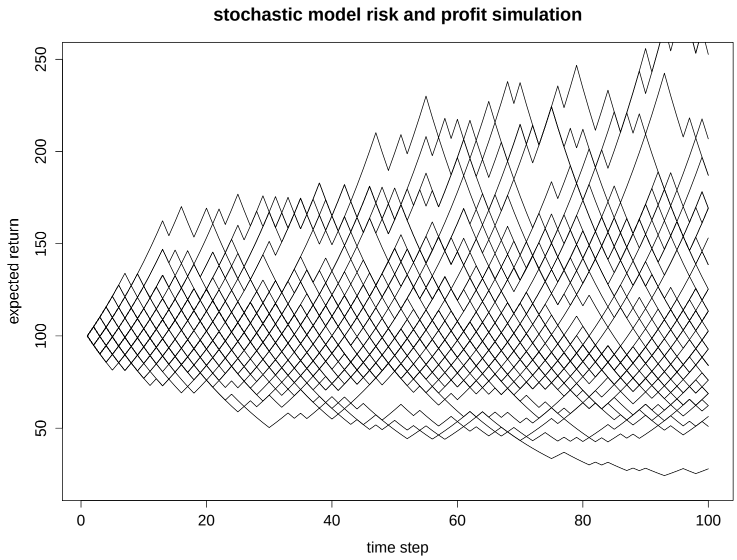
$$E(X) = E(\sum X_i) = 2^N \times M \times p^N$$

Since at step i, the expectation of X_i would be 0 if any of the steps before resulting the probability of losing, then the Person A lose all its bet, such at the outcome of the game is 0; thus the only possible way with nonzero outcome is to win at each of the steps, with a return of $2^N \times M$ and a probability of p^N.
Since at step i, the expectation of X_i would be 0 if any of the steps before resulting the probability of losing, then the Person A lose all its bet, such at the outcome of the game is 0; thus the only possible way with nonzero outcome is to win at each of the steps, with a return of $2^N \times M$ and a probability of p^N.

We immedaitely could see that this formula becomes $E(X) = (\frac{2}{p})^N \times M$, that is the expectation of the game is favor to person A if the probaility of winning is greater than $\frac{1}{2}$, this confirms with our intuition about the game. However, our intuition also tells us that this is not the optimal return outcome for person A. And over a long run, this strategy is not desirable, since most of the time Person A would end up with nothing.

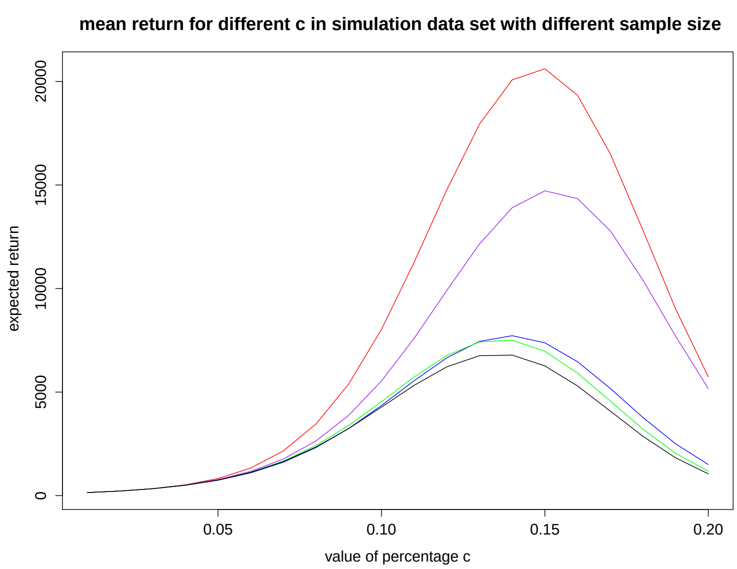
A general strategy to approach this problem is each time bet a percentage c of Person A's total amount M. When c = 1 this problem become the same as our discussion before. Now the question here is that we want to find the percentage c of the total amount to be invested in each step that maximaze Person A's profit or minimize Person A's loss.

Let's first simulated our result silimar to the previous problem. For consistency, let's take our probability of wining p to be 0.52, and let's take the number of steps to be 100 and sample size to be 50 in order to match our first simulation. In the graph, I call the parameter c, percentage amount to be invested, to be 0.05. The simulation graph is shown as below.



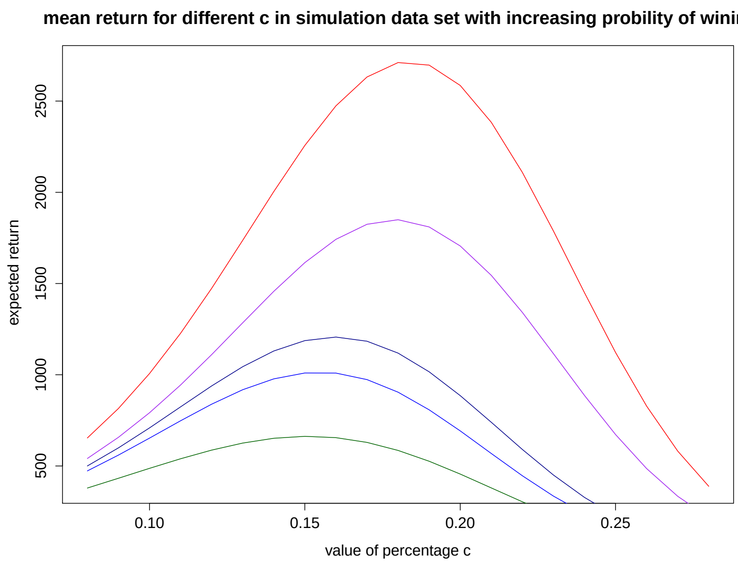
As we can see, this model seems more unstable comparing to the simple random walk model case. There are several data that seems more volatile than others. The mathematical prediction to solve this problem would be complicated, instead we take the computational approach as usual by simulation and making conclusion from our data.

Now let's use out computational approach to solve this problem, consider the case where the probability of winning is about 0.52, and the steps of the game is sufficient large for us to analyze its long time behavior, let's say 1000. Now let's take the simualtion with increase size of sample data and take the mean return for each data set with different parameter c. The following graph gives us a rough idea what value of c will Person A maximize his or her profit after a long term of the game, the different color curves from red to black correponding to a increase number in sample size.



We see as the sample size increases, our expected return after a certian long time become smooth out and eventually become stable. It is evidnetly form those simulation graph that we can conclude that the percentage of total amout money that will maximize the return of person A is about 0.12-0.14 percent according to our stochastic model formulation of the game.

Now we are interested how the shifting of the probability of winning p has effect on the value of c. Our intuition tells us that there is a postive correlation bewteen these two parameters, the larger the chance of winning, person A should be more aggressive in order to maximze his or her expected return. Indeed, after we simulate several sets of data wiht different probability of winning chance p and plot the curve of different c value expectation return curve, we should assume that p have a huge impact on the value of c. In other word, if the probaility of winning of the game increase by a small amount, we should adjust our percentage c to a much larger quantity and the expected return is exponential increased.



The graphical display is above, the different color curves corresponding to a probability of wining chance of 0.518, 0.519, 0.520, 0.521, 0.522, the curve become more rapid which confirms with our assumption that the expected return increases as the probability of wining. In addition, we see that the value of percentage c that maximize return also shift to the right more rapidly comparing to the increase of proability of winning p, suggesting that if there is a modest increase of wining probability in the game, Person A should largely increase the his or her total amount to be in order to maximize profits. And the optimal percantage of investment c can easily obtained by computing the apex of each curves' x-axis positition.

Appendix: Simulation Coding

1. The Random Walk Model Simulation Function

```
abet = function(N,X,p,S){
r = runif(N+1, min = 0, max = 1);
Stop = S; Amount = rep(NA,N+1);
Amount[1] = 0;
if(S > N){
Stop = N;
}
for(i in 2:(S+1)){
if(r[i] < p){
Amount[i] = Amount[i-1] + X;
```

```
}else{
Amount[i] = Amount[i-1] - X;
}}
return(Amount);
}
```

2. The Stochastic Model Simulation Function and Ploting

```
smbet = function(N,A,p,c){
amount = 1:N;
amount[1] = A;
bet = amount[1] * c
r = runif(N, min = 0, max =1)
; for(i in 1:(N-1)){
if(r[i] ¡= p){
amount[i+1] = amount[i] + bet;
bet = amount[i+1] * c;
}else{
amount[i+1] = amount[i] - bet;
bet = amount[i+1] * c;
}}
return(amount);
}

p = 0.52; N = 100; A = 100; size = 50; c = 0.05;
rw = matrix(nrow = size, ncol = N)
for(i in 1:size){
rw[i,] = rwbet(N,A,p,c)
}
plot(rw[1,],type="l",ylim = c(20,250), xlab = "time step", ylab ="expected return", main= "stochastic model
risk and profit simulation")
for(i in 2:size){
lines(rw[i,],type="l")
}
```

3. The Stochastic Model Simulation Function

```
bbet = function(N,A,p,c){
amount = A;
bet = amount * c
r = runif(N, min = 0, max =1);
for(i in 1:N){
if(r[i] <= p){
amount = amount + bet;
bet = amount * c;
}else{
amount = amount - bet;
bet = amount * c;
}}
return(amount);
}
```

4. Find the mean return value of percentage c with different sample size

```
p = 0.52; N = 1000; A = 100; size = 5000
c=c(0.01,0.02,0.03,0.04,0.05,0.06,0.07,0.08,0.09,0.1,0.11,0.12,0.13,0.14,0.15,0.16,0.17,0.18,0.19,0.2)
```

this examine diiferent value of c

```
M = length(c)
outcome = matrix(nrow = size, ncol = length(c))
for(i in 1:size){
outcome[i,] = betit(N,A,p,c)
}
mean = 1:M
for(i in 1:M){
mean[i] = mean(outcome[,i])
}
```

Reference

Probaility Theory and Examples, Rick Durrett, Fourth Edition, Cambridge Press, 2010 .

An introduction to Stochastic Modeling, Howard M.Taylor and Samuel Karlin, Third Edition, Academic Press, 1998.

Random walk in random and non-random environments (Third Edition), Pal Rvsz (2013), World Scientific Pub Co. ISBN 978-981-4447-50-8

Computational Analysis in Risk and Profit Problem II

Zhijun Yang
Faculty Adivisor: David Aldous

Abstraction

This research project is about analysis of risk and profit from games that can be formulated from mathematical model of stochastic process. We will approach to this kind of problems primarily from computational approach. I will simulate large set of data and by analyzing these data by employing several mathematical and statistical methods to reach our conclusion. This research is primarily concerns with problems arising in analysis of risk and profit from games, we are concern about the optimal strategy interms of maximal return or minimal risk in the those games that can be formulated in terms of Random Walk Model, Markov Chain Model or general Stochastic Model.

Model in Risk and Profit Problem

Let's consider a general type problem arising in many real world situation. In a game of finite steps, where person A bet amount of X_i at each step i, the probability that person A win X_i amount is p at each of the step i, and probaility person A lose the bet X_i is $1-p$. Suppose A have amount of M dollars before enter the game, we are concerned about what kind of strategy or mixed strategies that person A can employ to maxmamize his or her profit after a finite step, let's say N.

A general strategy to approach this problem is each time bet a percentage c of Person A's total amount M. Now the question here is that we want to find the percentage c of the total amount to be invested in each step that maximaze Person A's profit or minimize Person A's loss.

We will use computational analysis to approach this problem. In the folowing discussion, for simplicity , we will always consider our original amount of money to be 100 and we always set the time steps to be suffcient large, let's say 100. In our first case, considering taking the percentage of c to be 0.02, 0.04, 0.08 respectively, and for each percentage c we simulate a set of data, where each data corresponding to a person bet in the game with corresponding c. The following tables are obtained from our simulation of 10 players who play this game with respective percentage rate c as described above.

A sample of results from 10 players who bet with percentage of current amount c = 0.04

108.33102	92.30572	92.30572	67.01626	61.86116
127.13849	127.13849	85.20528	44.91275	127.13849

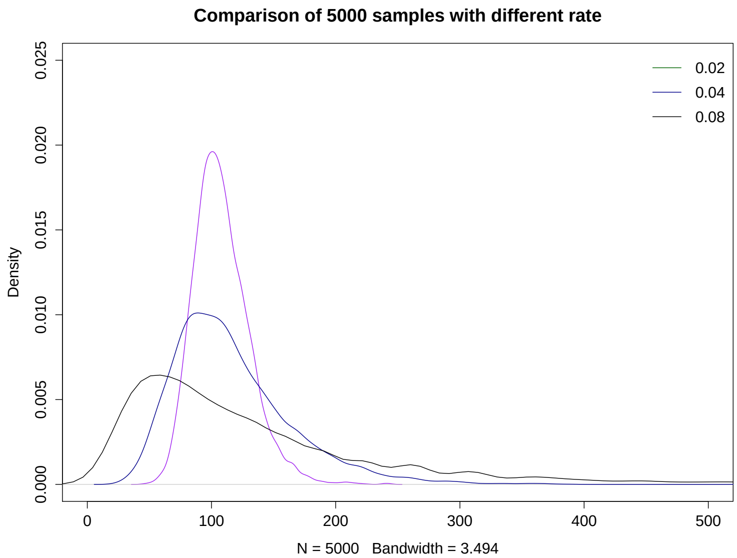
A sample of results from 10 players who bet with percentage of current amount c = 0.02

102.02027	110.51842	115.02937	102.02027	102.02027
98.01948	134.99110	129.69733	124.61116	90.48241

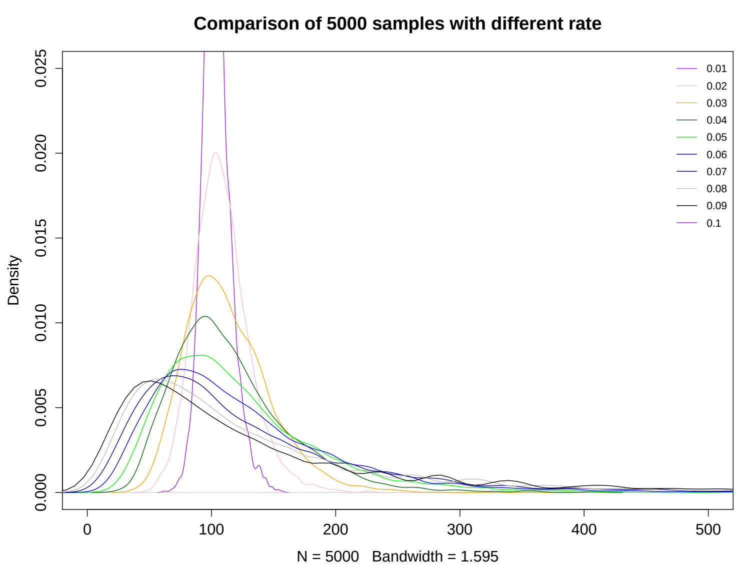
A sample of results from 10 players who bet with percentage of current amount c = 0.08

85.15596	44.84059	61.79356	222.85924	189.84305
32.53865	222.85924	222.85924	137.75991	85.15596

After a simulation of a large set of data, we take the sample size to be 5000, we plot the distribution graph with different percentage c in the same graph as following. The different colors curve in the graph corresponding to different values of c as stated in the legend.

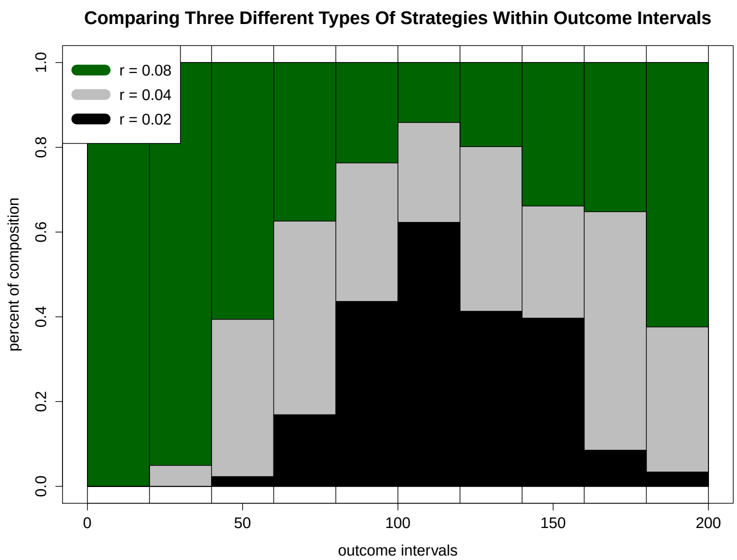


From the graph above, we immediately notice that the distribution corresponding to the larger value of c is smooth out and have mean more shifted to the right, that is most likely to lose money than small percentage of c. Now we comparing more different values of c. We ploting the percentage of mount being invested in each game from 0.01 to 0.1 with increment of 0.1 in the following graph. obversed that the distribution corresponding to smaller value of c is more concentrated near 100, while the distribution corresponding to larger c is smooth out through the graph.



Now we want to focus on how three different strategies, let's say r = 0.02, r = 0.04, r =0.08, (as in the first simulation graph shown), to have outcomes in some arbitrary intervals, let's say taking the width of interval to be 20, and range from 0 to 200. Keeping all the other factors the same as in the simulation of our first graph, we generate the plot with a sample size of 1000.

The following two graphs just a brief illustration of how many people from each groups using three different strategies end up in different outcome intervals with adjustment to the total number in the interval. It gives us a demonstration of how different strategies could help us reach our desired outcome. The sample size taking is 5000.



Appendix: Example Of My Simulation Coding

1. Simulation of density graph

```
p = 0.52
N = 100
A = 100

betit = function(N,A,p,c){
amount = A;
bet = amount × c
r = runif(N, min = 0, max =1);
for(i in 1:N){
if(r[i] ¡= p){
amount = amount + bet;
bet = amount × c;
}else{
amount = amount - bet;
bet = amount × c;
}}
return(amount);
}
```

```
c = (1:10)/100
samplesize = 5000;
result = matrix(nrow = 10, ncol = samplesize);

for(i in 1:10){
for(j in 1:samplesize){
result[i,j] = betit(N,A,p,c[i]);
}}

colors = c(”purple”,”red”,”pink”,”orange”,”dark green”,”green”,”blue”,”dark blue”,”grey”,”black”);
plot(density(result[1,]),main = ”Comparison of 1000 sample with different strategies”,xlim = c(0,500),ylim
= c(0,0.02),col= colors[1])
for(i in 2:9){
lines(density(result[i,]), col = colors[i])
}

legend(’topright’,c(”0.01”,”0.02”,”0.03”,”0.04”,”0.05”,”0.06”,”0.07”,”8”,”9”,”10”), lty=1, col=colors,
bty=’n’, cex=1.2)
```

2. Simulation of Percentage Distribution

```
samplesize = 10000;
result = matrix(nrow = 3, ncol = samplesize);
for(i in 1:3){
for(j in 1:samplesize){
result[i,j] = betit(N,A,p,c[i]);
}}

count = matrix(data = rep(0,30), nrow = 3, ncol = 10)
for(i in 1:3){
for(j in 1:samplesize){
for(k in 1:10){
if((result[i,j] ¡ 20*k) & (result[i,j] ¿= 20*(k-1))){
count[i,k] = count[i,k] + 1
}}}}

count[3,8] = 64
total = apply(count,2,sum)
for(j in 1:10){
count[,j] = count[,j]/total[j];
}

count = apply(count,2,cumsum);
ostepx = c(0,20,40,60,80,100,120,140,160,180,200);
stepx = rep(ostepx, each = 2);

stepy1 = count[1,];
stepy1 = rep(stepy1,each = 2);
stepy1 = c(0,stepy1,0)
stepy2 = count[2,];
stepy2 = rep(stepy2,each = 2);
stepy2 = c(0,stepy2,0)
```

stepy3 = count[3,]; I change an entry in this matrix here because of the random number genertor in R recur-

sive avoid any outcome from outcome value from 140 to 160 for the 8 percentage case, that is it will appear 0 for all the time, so I use matlab to do this special case simulation and insert the result into the entry of the count matrix

```
stepy3 = rep(stepy3,each = 2);
stepy3 = c(0,stepy3,0)

plot(c(0,200),c(1,1),xlim= c(0,200), ylim = c(0,1),type="l", xlab="outcome intervals", ylab = "percent of composition", main = "Comparing Three Different Types Of Strategies Within Outcome Intervals");
polygon(stepx, stepy3, col="coral1")
polygon(stepx, stepy2, col="aquamarine2")
polygon(stepx, stepy1, col="cornflowerblue")
abline(v=ostepx)
legend("topleft",c("r = 0.08","r = 0.04", "r = 0.02"),lty=c(1,1,1),lwd=c(15,15,15),col=c("coral1","aquamarine2",""))
```

Reference

Probaility Theory and Examples, Rick Durrett, Fourth Edition, Cambridge Press, 2010 .

An introduction to Stochastic Modeling, Howard M.Taylor and Samuel Karlin, Third Edition, Academic Press, 1998.

Random walk in random and non-random environments (Third Edition), Pal Rvsz (2013), World Scientific Pub Co. ISBN 978-981-4447-50-8