EE511 Project#2

Yushu Liu

09/28/2016

2. Project achievement and results

1. Estimate π by the area method (including confidence intervals). Draw a graph of the successive values of the estimator as the number of samples increases. How man y points do you need to use for your estimate to be within ±1% of the true value of π (with probability 0.95)?

```
%% part1
clear all;close all;clc;
fprintf('\nPart 1: \n\n');
for count = 4:6 % test 10000, 100000, 1000000 points
    num = 10^count;
    fprintf('Generating %d corresponding RVs.\n', num);
    %fprintf('Generating %d random points.\n', num);
    x = rand(num, 1); % generate x axis value
    y = rand(num, 1); % generate y axis value
    p = ones(num, 1); % generate random variable
    for i=1:num
    if x(i)^2+y(i)^2>1 % use the circle with radius=1, here the point
inside circle
        p(i)=0;% here is random variable can represent inside the
circle
    end
    end
    m = mean(p); % for (0,1) distribution, mean=p (p=p(1))
    sv = var(p); % var() function
    v = m * (1 - m) / num; % use mean to calculate variance
    % for (0,1) distribution, mean=p (p=p(1)) and variance definition
    fprintf('Mean: %f. Variance: %f.\n', m, v);
    fprintf('%d points in circle and %d out of circle \n',sum(p),num-
sum(p));
    fprintf('Confidence interval from 0.99 from %d points
estimate,\n', num);
    fprintf('first estimate, %f <= pi <= %f\n',4*(m-2.56*sqrt(v)),
4*(m+2.56*sqrt(v)));
    %add the confidence inside
    fprintf('With second estimate, %f <= pi <= %f\n',4*(m-
2.56*sqrt(sv/num)), 4*(m+2.56*sqrt(sv/num)));
    %when alpha=0.01,let beta=2.56, and use the definition of
confidence
    %intervel
    fprintf('confidence interval of 1st estimate are %f points are
needed ', 16*2.56^2*m*(1-m)/(0.01)^2);
    fprintf('to estimate within 0.99 of the true value of pi\n');
    fprintf('confidence interval of 2nd estimate are %f points are
```

```
needed ', 16*2.56^2*sv/(0.01)^2);
    fprintf('to estimate within 0.99 of the true value of pi\n\n');
end
e=zeros(15,1);
for count=10:24
    num=2^count;
    x = rand(num, 1); % generate a(x) coordinate
    y = rand(num, 1); % generate b(y) coordinate
    p = ones(num, 1); % generate random variable
    for i=1:num
    if x(i)^2+y(i)^2>1% use the circle with radius=1
        p(i)=0;% here is random variable can represent inside the
circle
    end
    end
    e(count-9)=4*mean(p);% generate estimate pi value
end
true_pi=ones(15,1).*pi;% true value of pi
low=ones(15,1).*(true_pi-0.01);% lower bound of confidence interval
high=ones(15,1).*(true_pi+0.01);% higher bound of confidence interval
n=2.^(10:24);
figure
plot(n,true_pi,'g','LineWidth',2);hold on;
plot(n,low,'r','LineWidth',2);
plot(n,high,'r','LineWidth',2);
plot(n,e,'LineWidth',3);hold off;
title('graph of estimates with the increase of the number of
points');
legend('true value of pi','lower bound of 0.01','higher bound of
0.01','estimates');
```

And the result are as follows:

1. In the command window:

Part 1:

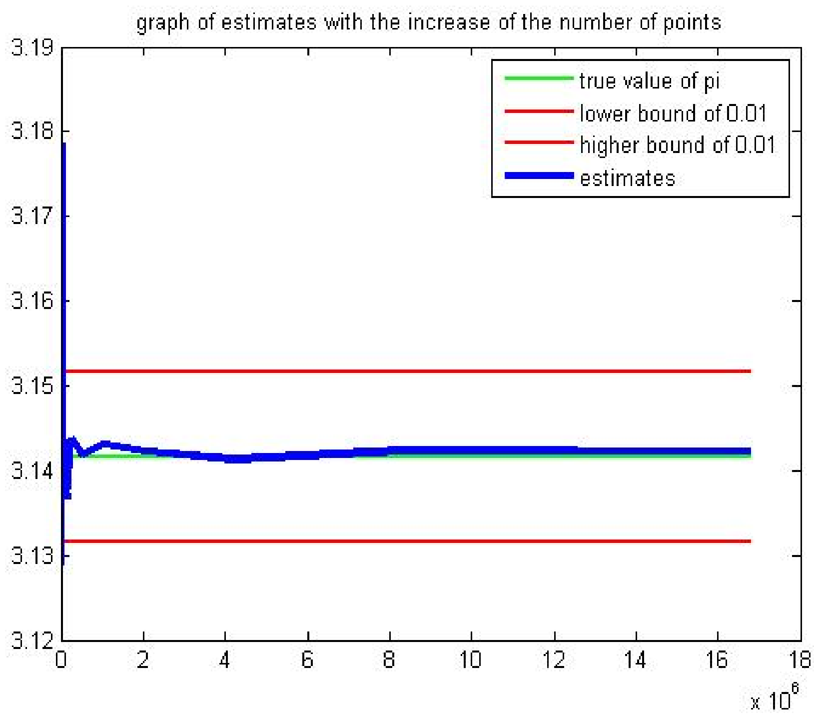
Generating 10000 corresponding RVs.
Mean: 0.792100. Variance: 0.000016.
7921 points in circle and 2079 out of circle
Confidence interval from 0.99 from 10000 points estimate,
first estimate, 3.126846 <= pi <= 3.209954
With second estimate, 3.126844 <= pi <= 3.209956
confidence interval of 1st estimate are 172676.968612 points are needed to estimate within 0.99 of the true value of pi
confidence interval of 2nd estimate are 172694.238036 points are needed to

estimate within 0.99 of the true value of pi

Generating 100000 corresponding RVs.
Mean: 0.784270. Variance: 0.000002.
78427 points in circle and 21573 out of circle
Confidence interval from 0.99 from 100000 points estimate,
first estimate, 3.123760 <= pi <= 3.150400
With second estimate, 3.123760 <= pi <= 3.150400
confidence interval of 1st estimate are 177409.168087 points are needed to estimate within 0.99 of the true value of pi
confidence interval of 2nd estimate are 177410.942197 points are needed to estimate within 0.99 of the true value of pi

Generating 1000000 corresponding RVs.
Mean: 0.785099. Variance: 0.000000.
785099 points in circle and 214901 out of circle
Confidence interval from 0.99 from 1000000 points estimate,
first estimate, 3.136190 <= pi <= 3.144602
With second estimate, 3.136190 <= pi <= 3.144602
confidence interval of 1st estimate are 176914.232979 points are needed to estimate within 0.99 of the true value of pi
confidence interval of 2nd estimate are 176914.409895 points are needed to estimate within 0.99 of the true value of pi

2. The Line which can show the changing under the sample number changed:



3. Part 2 code and results

```
%% part 2
clear all;close all;clc;
n=1000000;
i=0:1/n:1;
s=0;
for k=1:length(i)-1
    s=s+(1/(1+((i(k)+i(k+1))/2)^2))*1/n;
end
pi=vpa(4*s,20)
```

And the result is as follows:

pi =

3.1415926535899751926

4. Part 3 code and result

```
%% part 3
clear all;close all;clc;
N = 52;
sample=rand(1,N);% produce 52 samples
randnum=randperm(N);% produce 52 random variables
selected=sample(randnum(1:13));% select 13 samples

num_simu = input('simple used to test: ');
% input the number of simulations

 % total number of cards
new_deck = linspace(1, N, N); % generate a deck of new cards
% map Card 1 to the Ace of Clubs, Card 2 to the 2 of Clubs,...
% Card 14 to the Ace of Diamonds, Card 27 to the Ace of Hearts
% Card 40 to the Ace of Spades, Card 52 to the King of Spades

X_num_points = zeros(1, num_simu); % record the points of each
simulation

for k = 1:1:num_simu

    sample=rand(1,52);
    randnum=randperm(52);% produce 52 random variables
selected=sample(randnum(1:13));% select 13 samples

    for c = 1:1:52 % randomly shuffle the new deck of cards
        randnum(c) = randperm(ceil((N-c+1)*randnum(c)));
```

```
        selected(ceil((N-c+1)*randnum(c)):1:end-1) = ...
            selected(ceil((N-c+1)*randnum(c)+1):1:end);
        selected(N-c+1:1:end) = 0;
    end

    bridge_hand = zeros(1, N/4);
    bridge_hand(1:1:N/4) = selected(1:4:end);


    num_points = 0;

    for c1 = 1:1:N/4 % get the number of points in the bridge hand
        switch bridge_hand(c1)
            case {1, 14, 27, 40}
                num_points = num_points+4;
            case{13, 26, 39, 52}
                num_points = num_points+3;
            case{12, 25, 38, 51}
                num_points = num_points+2;
            case{11, 24, 37, 50}
                num_points = num_points+1;
            otherwise
        end
    end

    X_num_points(k) = num_points;
end

figure(1)
histogram(X_num_points, edges, 'Normalization', 'probability');
% plot the histogram of the number of points
grid on
title('The Histogram of the Number of Points');
xlabel('Number of points');
ylabel('Probability');
```