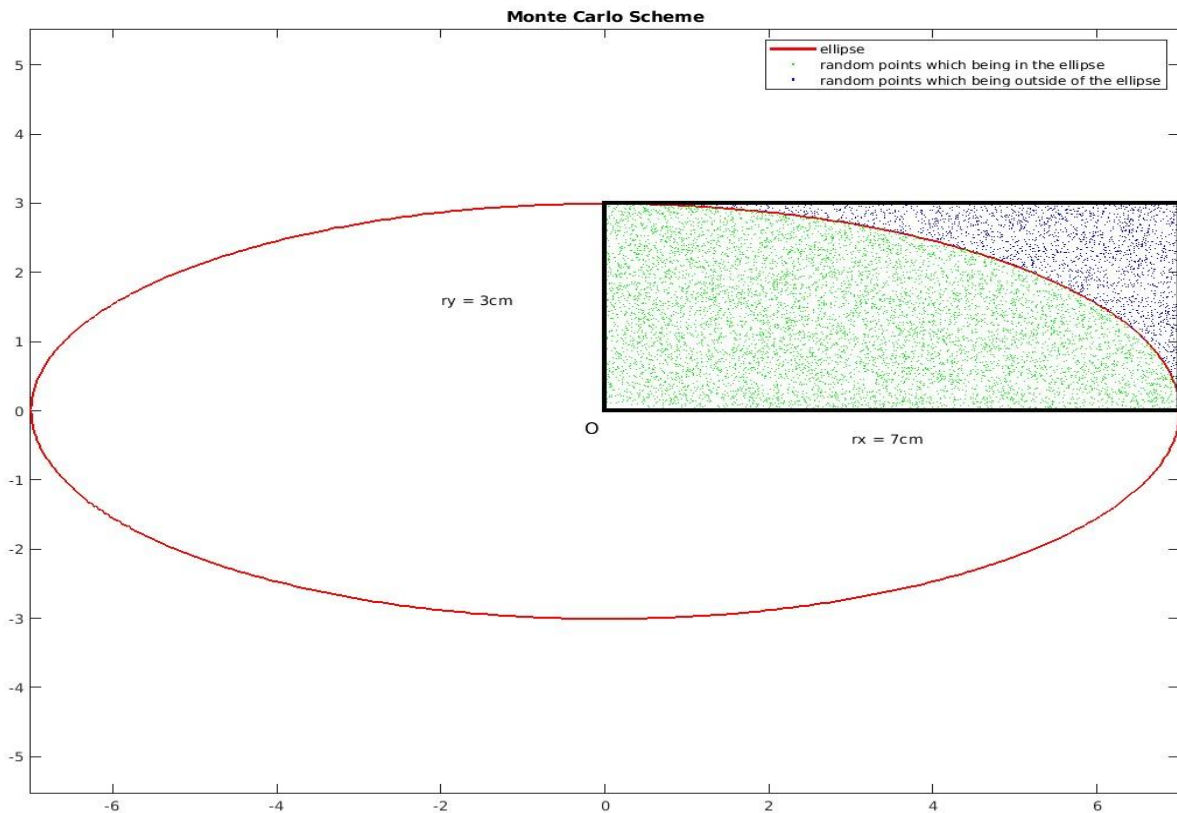


MİDTERM

HASAN MERT GÜNEŞ
090210128

1-a)



Estimating the value of pi using MONTE CARLO METHOD

$$\text{Area of the quarter ellipse} / \text{area of the rectangle} = (A*B*\pi/4) / A*B = \pi/4$$



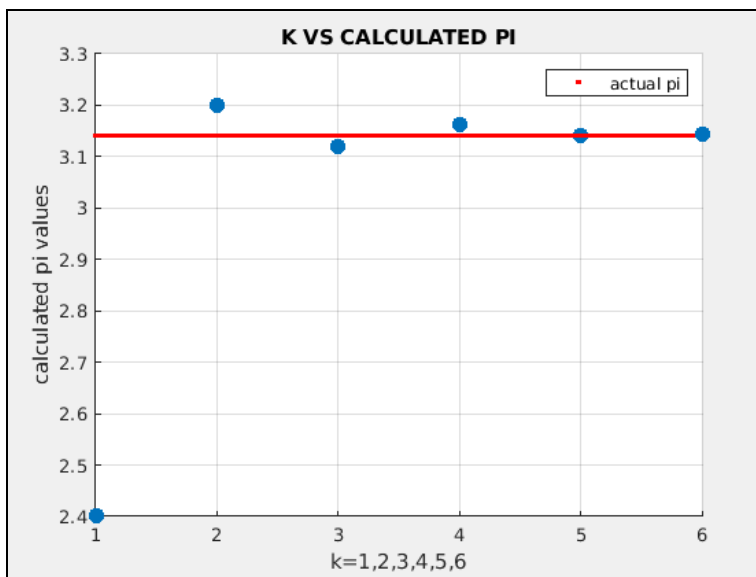
$$(\text{Number of points in the area} / \text{number of points in the area 2}) * 4 = \pi$$

If we generate random points then determine where the point is, we can calculate the equation shown above. Our result approaches the pi value as the number of points increases.

1-b,d,e)

CODE

```
1 % 090210128 Hasan Mert Güneş
2
3 tic
4 % radius of ellipse;
5 rxHMG = 7;
6 ryHMG = 3;
7
8 for k=1:6
9     %random point vectors
10    vecxHMG = [];
11    vecyHMG = [];
12
13    N=10^k;
14    % random point generator, Number of sample:N
15    for i=1:N
16        kxHMG = 0 + 7*rand(1,1); %
17        kyHMG = 0 + 3*rand(1,1); %
18        vecxHMG = [vecxHMG kxHMG(1)];
19        vecyHMG = [vecyHMG kyHMG(1)];
20    end
21
22    % mHMG(2) is number of points which being inside the ellipse
23    % found by leaving alone y value of "x^2a^2+y^2b^2=1" ellipse equation
24    mHMG = size(vecyHMG(vecyHMG< sqrt((rxHMG^2-vecxHMG.^2)*ryHMG^2/rxHMG^2)));
25
26    % results
27    piHMG = 4*mHMG(2)/N;
28    errorHMG = abs(pi - piHMG)/pi*100;
29    disp("k= "+k+ " | calculated pi= "+piHMG)
30
31
32    % plot k vs pi table
33    hold on
34    plot((1:0.01:6),3.1415,"r.")
35    grid on
36    plot(k,piHMG,Marker=".", Color="#0072BD", MarkerSize=30)
37    legend("actual pi")
38    title("K VS CALCULATED PI")
39    xlabel("k=1,2,3,4,5,6")
40    ylabel("calculated pi values")
41
42 end
43 timeHMG = toc;
```




OUTPUTS

```
>> clear
k= 1 | calculated pi= 2.4
k= 2 | calculated pi= 3.2
k= 3 | calculated pi= 3.12
k= 4 | calculated pi= 3.1636
k= 5 | calculated pi= 3.1407
k= 6 | calculated pi= 3.1436
>>
```

1-c)

Calculate the number pi (π) approximately using a Monte Carlo scheme with a quarter ellipse. Code in matlab

 Ask ChatGPT

Here is a MATLAB code that uses a Monte Carlo method to calculate the value of π approximately using a quarter ellipse. ×

```
% Set the number of random points to generate.
num_points = 1000000;

% Generate random points in the unit square.
x = rand(1, num_points);
y = rand(1, num_points);

% Use the equation of the quarter ellipse to determine whether each point is
% inside or outside the quarter ellipse.
inside_ellipse = (x.^2 + y.^2 <= 1) & (x >= y);

% Estimate the value of pi using the ratio of the number of points inside the
% quarter ellipse to the total number of points, multiplied by four.
pi_approx = sum(inside_ellipse) / num_points * 4;

% Display the estimated value of pi.
disp(['Estimated value of pi: ', num2str(pi_approx)]);
```

You can run this code in CoCalc by creating a new MATLAB file and copying and pasting the code into the file. Then, you can run the code by clicking the "Run" button or by typing "run" in the MATLAB console. The estimated value of π will be displayed in the console.

```
>> clear
Estimated value of pi: 1.5713
fx >>
```

- AI solved the question by very efficient code blocks and almost got the right answer. But AI multiplied num_points by four instead of eight. So the estimated value of pi calculated as 1.5713 instead of 3.1426.
- Our approaches are very similar.

1-f)

For $N=10^6$ measured calculation times

Circle calculation time	Ellipse calculation time	AI Ellipse calculation time
740.9726 s	759.3239 s	0.0242 s

- AI code runs very fast and returns very accurate results. But my code is slow because I tried to print, plot and save data at the same time.

%Error = $\text{abs}(\pi - \pi_{\text{HMG}})/\pi * 100$

Circle error	Ellipse error	AI Ellipse error
% 1.5261	% 0.0653	% 49.9567, (%0.035)*

- *The most accurate results coming from AI if the AI would have multiplied by the right number.
- Ellipse returns a more accurate result than circle, despite they both run exactly the same processes.

2-a)

The Chi-Square Test of Independence determines whether there is an association between categorical variables.

1. $P \leq 0.05$ (Hypothesis interpretations are rejected)
2. $P > 0.05$ (Hypothesis interpretations are accepted)

Here P denotes the probability; hence for the calculation of p-values, the Chi-Square test comes into the picture.

<https://www.cuemath.com/chi-square-formula/>

<https://libguides.library.kent.edu/spss/chisquare>

<https://www.simplilearn.com/tutorials/statistics-tutorial/chi-square-test>

Chi Square Formula



$$\chi^2 = \sum \frac{(O_i - E_i)^2}{E_i}$$

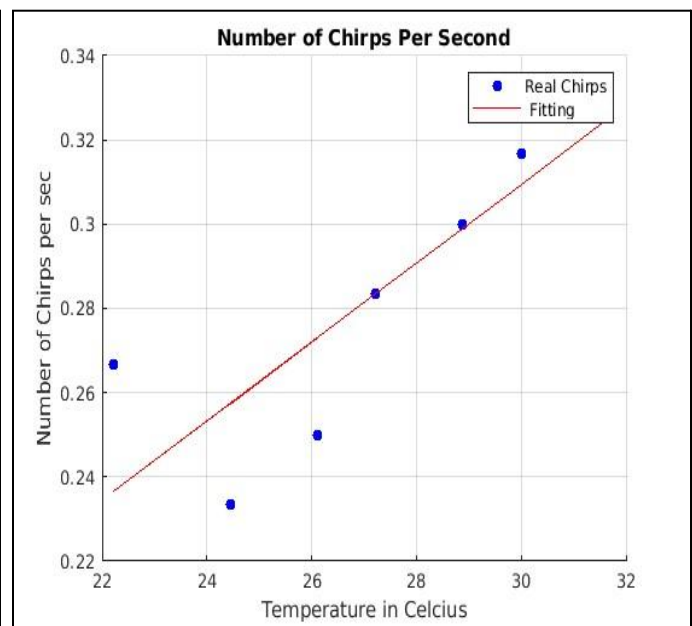
where

O_i = observed value (actual value)

E_i = expected value

2-b)

```
Editor - /home/mert/272/crickets.m *
crickets.m * sleeplessnights.txt +
1 % 090210128 Hasan Mert Güneş
2
3 dataHMG = readmatrix("sleeplessnights.txt");
4
5 chirpsHMG = dataHMG(:,1); % number of per min
6 chirps_per_secHMG = chirpsHMG/60; % number of per sec
7 fahrHMG = dataHMG(:,2); % temperature in fahrenheit
8 celciusHMG = (fahrHMG-32)*5/9; % temperature in celcius
9
10 % plotting temp and chirps to determine a relationship
11 scatter(celciusHMG,chirps_per_secHMG)
12 hold on
13
14 % linear fitting process
15 pHMG = polyfit(celciusHMG,chirps_per_secHMG,1);
16 chirps_fit_HMG = pHMG(1)*celciusHMG+pHMG(2);
17
18 % fitting plot
19 plot(celciusHMG,chirps_fit_HMG,'r-',MarkerSize=20);
20 legend("Real Chirps","Fitting")
21 title("Number of Chirps Per Second")
22 xlabel("Temperature in Celcius")
23 ylabel("Number of Chirps per sec")
24 grid on
25
26 % chi square test by " $\chi^2 = \sum (O_i - E_i)^2 / E_i$ " equation
27 chi_squHMG = 0; % starting sum loop with 0
28 for i = 1:size(chirps_per_secHMG)
29     chi_squHMG = chi_squHMG + (chirps_per_secHMG(i)- ...
30         chirps_fit_HMG(i))^2/chirps_fit_HMG(i);
31 end
32
33 disp("χ² test result= " + chi_squHMG);
```



Command Window

χ² test result= 0.0084036

fx >>

Results of Test

Before you can conclude, you must first determine the critical statistic, which requires determining our degrees of freedom.

$$(r-1)(c-1) = (7-1)(2-1) = 6$$

“r=row number, c=column number”

Our $X^2 = 0.0084$, this means
We have sufficient evidence to say
that there is an association between
number of chirps and temperature.

**Critical values of the Chi-square distribution
with d degrees of freedom**

Probability of exceeding the critical value							
d	0.05	0.01	0.001	d	0.05	0.01	0.001
1	3.841	6.635	10.828	11	19.675	24.725	31.264
2	5.991	9.210	13.816	12	21.026	26.217	32.910
3	7.815	11.345	16.266	13	22.362	27.688	34.528
4	9.488	13.277	18.467	14	23.685	29.141	36.123
5	11.070	15.086	20.515	15	24.996	30.578	37.697
6	12.592	16.812	22.458	16	26.296	32.000	39.252
7	14.067	18.475	24.322	17	27.587	33.409	40.790
8	15.507	20.090	26.125	18	28.869	34.805	42.312
9	16.919	21.666	27.877	19	30.144	36.191	43.820
10	18.307	23.209	29.588	20	31.410	37.566	45.315

INTRODUCTION TO POPULATION GENETICS, Table D.1
© 2013 Sinauer Associates, Inc.

2-c)

```

data = load("data.txt");

chirps = data(:,1);
temp = data(:,2);

fit = polyfit(chirps, temp, 1);

x = linspace(min(chirps), max(chirps), 1000);
y = polyval(fit, x);
hold on;
plot(x, y);

expected = polyval(fit, chirps);
chi2 = sum((temp - expected).^2 ./ expected);
dof = length(chirps) - 2; % degrees of freedom
p = 1 - chi2cdf(chi2, dof); % p-value

```

The p-value will tell you whether or not the linear function provides a good fit for the data. If the p-value is less than a significance level of your choice (e.g. 0.05), you can conclude that the data does not support the linear relationship claimed by the theory.

- AI did the similar processes for the first section but then, AI found the degree of freedom differently and used special commands to find the final p value.
- The interpretation section is the same.