EE511 PROJECT 2
SAMPLES AND STATISTICS

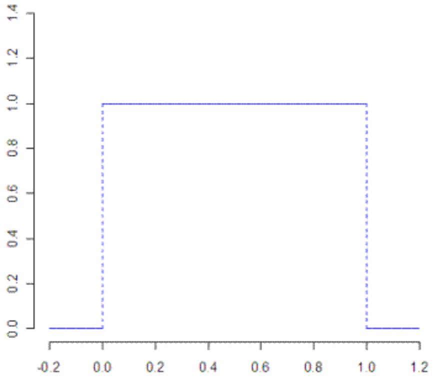
NAME	SONALI B SREEDHAR
EMAIL ID	sbsreedh@usc.edu
USCID	1783668369

PROBLEM 1:

1. Simulate sampling uniformly (how many?) on the interval ⍰⍰3,2⍰.
    1. Generate a histogram of the outcomes.
    2. Compute the sample mean and sample variance for your samples. How do these values compare to the theoretical values? If you repeat the experiment will you compute a different sample mean or sample variance?
    3. Compute the bootstrap confidence interval (what width?) for the sample mean and sample standard deviation.

Uniform Distribution:

A uniform distribution, also called a rectangular distribution, is a probability distribution that has constant probability.



This distribution is defined by **two parameters**, a and b:

- a is the minimum.
- b is the maximum.

The distribution is written as U(a,b).

Like all probability distributions for continuous random variables, the area under the graph of a random variable is **always equal to 1.** In the above graph, the area is: A = l x h = 2 * 0.5 = 1.

Expectation and Variance

If X ~ U(a,b), then:

- E(X) = ½ (a + b)

- Var(X) = (1/12)(b - a)^2

Experiment:

To simulate uniformly distributed samples over the interval [-3,2].
The samples are generated using rand() function which generated a uniform continuous distribution. These samples are plotted using histogram and the uniformity of the distribution is observed.

Code:

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% EE511-FAll 2019                                                   %
% Project #2                                                        %
% Samples and Statistics                                            %
% Author: Sonali B Sreedhar                                         %
% Date:09/10/2019                                                   %
% USC ID:1783668369                                                 %
% Email ID: sbsreedh@usc.edu                                        %
%-----------------------------------------------------------------------%
%Simulate sampling uniformly (how many?) on the interval [-3,2].
% 1.    Generate a histogram of the outcomes.
% 2.    Compute the sample mean and sample variance for your samples.
% .     How do these values compare to the theoretical values?
% .     If you repeat the experiment will you compute a different
%       sample mean or sample variance?
% 3.    Compute the bootstrap confidence interval (what width?)
%       for the sample mean and sample standard deviation.
%-----------------------------------------------------------------------%
% Bootstrapping means random sampling with replacement.
% bootci - Bootstrap Confidence interval
% bootstat = bootstrp(nboot,bootfun,d1,...) draws nboot bootstrap data
samples,
% computes statistics on each sample using bootfun, and returns the results
% in the matrix bootstat. nboot must be a positive integer. bootfun is a
function
% handle specified with @. Each row of bootstat contains the results of
applying
% bootfun to one bootstrap sample. If bootfun returns a matrix or array,
% then this
% output is converted to a row vector for storage in bootstat.
% nboot is a positive integer indicating the no. of bootstrap data sample.
%------------------------------------------------------------------------%
clear;
n = input('Please enter the number of samples: ');
X = []; % X stores the sample outcomes
for i = 1:n
    % Generate random number in [-3,2] and store in X
    X = [X 5*rand() - 3];
end
```

```
%To plot the histogram for the samples
hist(X)
title(['n=',num2str(n)])
% Theoritical Calculations through simulations of set of samples
disp(['The sample mean is: ',num2str(mean(X))]);
disp(['The sample variance is: ',num2str(var(X))]);
M = sort(bootstrp(n,@mean,X));
S = sort(bootstrp(n,@std,X));
% Get index of 2.5% point and 97.5% point
disp(['The bootstrap confidence interval for sample mean is: [',...
    num2str(M(ceil(n*0.025))),',',num2str(M(floor(n*0.975))),']']);
disp(['The bootstrap confidence interval ',...
    'for sample standard deviation is: [',...
    num2str(S(ceil(n*0.025))),',',num2str(S(floor(n*0.975))),']']);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Result:

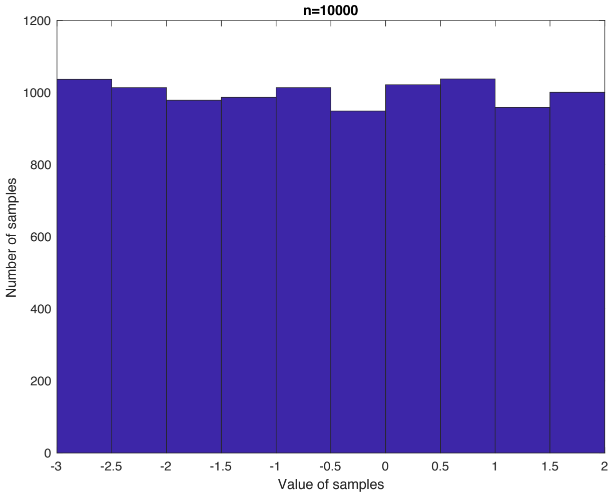


Figure 1: Histogram plot for uniformly distributed samples with a sample space of 10000

```
Please enter the number of samples: 10000
The sample mean is: -0.50841
The sample variance is: 2.101
The bootstrap confidence interval for sample mean is: [-0.5373,-0.47984]
The bootstrap confidence interval for sample standard deviation is: [1.4367,1.462]
```

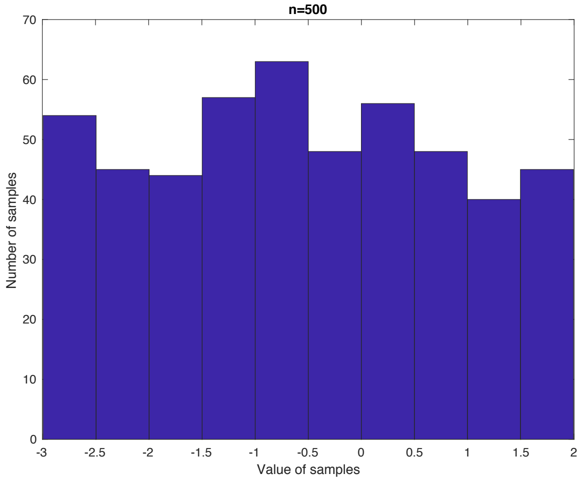


Figure 2: Histogram plot for uniformly distributed samples with a sample space of 500

```
Please enter the number of samples: 500
The sample mean is: -0.55658
The sample variance is: 1.9886
The bootstrap confidence interval for sample mean is: [-0.68076,-0.44311]
The bootstrap confidence interval for sample standard deviation is: [1.35,1.4712]
```

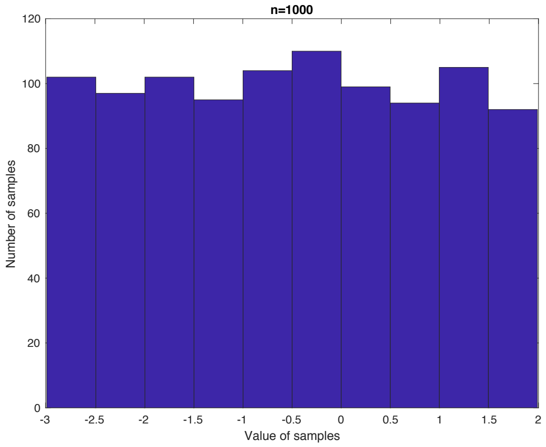


Figure 3:Histogram plot for uniformly distributed samples with a sample space of 1000

```
Please enter the number of samples: 1000
The sample mean is: -0.5173
The sample variance is: 2.0382
```

The bootstrap confidence interval for sample mean is: [-0.60878,-0.42792]
The bootstrap confidence interval for sample standard deviation is: [1.3844,1.4675]

Analysis:

Statistical Parameters	Theoretical Values	Number of Samples =500	Number of Samples =1000	Number of Samples =10000
Mean	-0.5	-0.55658	-0.5173	-0.50841
Variance	2.08	1.9886	2.0382	2.101

We observe uniformity on the graph due to huge numbers of samples considered. From figure 1,2,3 we can observe the samples are uniformly distributed. Considering different size of samples we observe the histograms show uniformity here too. Concept is crystal clear in the sample size 10000.

From the result we observe the accuracy of the statistical functions increase as the number pf samples increases. Therefore, moving forward large number of samples i.e, 10000 samples will be considered. From the above figures we observe that for 10000 samples compared to the 1000 samples, we get normalized distribution. This proves central limit theorem.

Central Limit theorem:
In order to make a statistical statement about the observed and expected results, we use the “CENTRAL LIMIT THEOREM”, which states that ‘The arithmetic mean of a sufficiently large number of iterates of independent random variables, each with a well defined expected value and finite variance will be approximately normally distributed , regardless of the underlying distribution”.

Introduction to Bootstrap:
Bootstrapping is a statistical technique that falls under the broader heading of resampling. This technique involves a relatively simple procedure but repeated so many times that it is heavily dependent upon computer calculations. Bootstrapping provides a method other than confidence

intervals to estimate a population parameter. Bootstrapping very much seems to work like magic. Read on to see how it obtains its interesting name.

Confidential Interval:

In statistical inference, one wishes to estimate population parameters using observed sample data. A ***confidence interval*** gives an estimated range of values which is likely to include an unknown population parameter, the estimated range being calculated from a given set of sample data.
The purpose of taking a random sample from a lot or population and computing a statistic, such as the mean from the data, is to approximate the mean of the population. How well the sample statistic estimates the underlying population value is always an issue. A confidence interval addresses this issue because it provides a range of values which is likely to contain the population parameter of interest.

Confidence intervals are constructed at a confidence level, such as 95 %, selected by the user. What does this mean? It means that if the same population is sampled on numerous occasions and interval estimates are made on each occasion, the resulting intervals would bracket the true population parameter in approximately 95 % of the cases. A confidence stated at a $1-\alpha$ level can be thought of as the inverse of a significance level, α.

In the same way that statistical tests can be one or two-sided, confidence intervals can be one or two-sided. A two-sided confidence interval brackets the population parameter from above and below. A one-sided confidence interval brackets the population parameter either from above or below and furnishes an upper or lower bound to its magnitude.

Matlab provides a bootstrapping function that does essentially the same thing as bootstrap. It calculates the confidence interval using the bias accelerated correction.

Analysis:

Bootstrapping algorithm tells us about reliability of our statistic based on our simple sample. We can use the CI to test the hypothesis that the statistic run on our sample of huge number is significantly different from the population average.

Problem 2:

1. Produce a sequence X by drawing samples from a standard uniform random variable.
   1. Compute $Cov(X_k,X_{k+1})$. Are $X_k$and X_{k+1} uncorrelated? What can you conclude about the independence of X_k and X_{k+1}?
   2. Compute a new sequence Y where: $Y[k] = Xk\text{- } 2\cdot X[k\text{-}1] + 0.5\cdot X[k\text{-}2]– X[k\text{-}3]$.

Code:

```
%-----------------------------------------------------------------------------
%
% Problem 2                                                                   %
%-----------------------------------------------------------------------------
%
n = input('Please enter the number of samples: ');
Xk = rand(1,n);                   % Xk is X_k
Xk1 = circshift(Xk,1,2);          % Xk1 is X_k+1 by shift array circularly
Xk1(1) = 0;                       % Set the first element to 0 to get X_k+1
Cov_a = cov(Xk,Xk1)               % Get covariance matrix

disp(['Cov[X_k, X_k+1] is: ',num2str(Cov_a(2,1))]);

Yk = [];                          % Create Yk

for k =4:n
Yk(k)= Xk(k) - 2*Xk(k - 1) + 0.5*Xk(k - 2) - Xk(k - 3);
end

Cov_b = cov(Xk,Yk)

disp(['Cov[X_k, Y_k] is: ',num2str(Cov_b(2,1))]);
%-----------------------------------------------------------------------------
%
```

Results:

a) Number of Samples=500

$Cov(X_k,X_{k+1})$= 0.0815   0.0040
                  0.0040   0.0820

$Cov(X_k,X_{k+1})$: 0.0039542

$Cov(X_k,Y_k)$= 0.0815   0.0757
               0.0757   0.4693

$Cov(X_k, Y_k)$: 0.075731

b) Number of Samples=1000

$Cov(X_k, X_{k+1})$= 0.0854   0.0006
                    0.0006   0.0856
$Cov(X_k, X_{k+1})$: 0.00063817

$Cov(X_k, Y_k)$= 0.0854   0.0915
                 0.0915   0.5476
$Cov(X_k, Y_k)$: 0.091498

b) Number of Samples=10000

$Cov(X_k, X_{k+1})$= 0.0823   -0.0000
                    -0.0000   0.0824
$Cov(X_k, X_{k+1})$: -2.0592e-05

$Cov(X_k, Y_k)$= 0.0823   0.0829
                 0.0829   0.5201
$Cov(X_k, Y_k)$: 0.082945

Observations:
From the above results we can see that though the results are close to zero ,they are not zero. Hence we can conclude that sequence Xk,Xk+1 and Xk,Yk are corelated.

Problem 3 :
Let M = 10. Simulate (uniform) sampling with replacement from the outcomes 0, 1, 2, 3, ..., M-1.

1. Generate a histogram of the outcomes.
2. Perform a statistical goodness-of-fit test to conclude at the 95% confidence level if your data fits samples from a discrete uniform distribution 0, 1, 2, ..., 9.
3. Repeat (b) to see if your data (the same data from b) instead fit an alternate uniform distribution 1, 2, 3, ..., 10

The goodness of fit of a statistical model describes how well it fits a set of observations. Measures of goodness of fit typically summarize the discrepancy between observed values and the values expected under the model in question . Such measures can be used in statistical hypothesis testing.

A **chi-squared test**, also written as χ2 test, is any statistical hypothesis test where the sampling distribution of the test statistic is a chi-squared distribution when the null hypothesis is true. Without other qualification, 'chi-squared test' often is used as short for Pearson's chi-squared test. The chi-squared test is used to determine whether there is a significant difference between the expected frequencies and the observed frequencies in one or more categories.

In the standard applications of this test, the observations are classified into mutually exclusive classes, and there is some theory, or say null hypothesis, which gives the probability that any observation falls into the corresponding class. The purpose of the test is to evaluate how likely the observations that are made would be, assuming the null hypothesis is true.

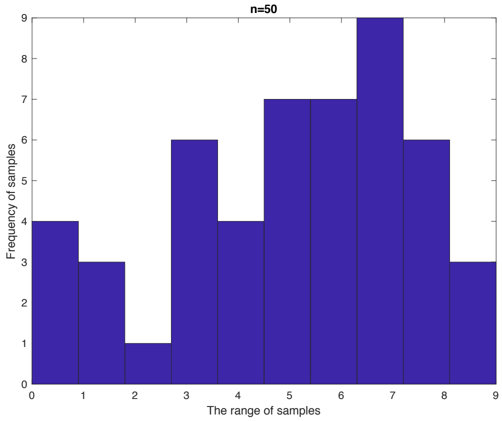
Chi-squared tests are often constructed from a sum of squared errors, or through the sample variance. Test statistics that follow a chi-squared distribution arise from an assumption of independent normally distributed data, which is valid in many cases due to the central limit theorem. A chi-squared test can be used to attempt rejection of the null hypothesis that the data are independent.

Also considered a chi-squared test is a test in which this is asymptotically true, meaning that the sampling distribution (if the null hypothesis is true) can be made to approximate a chi-squared distribution as closely as desired by making the sample size large enough.

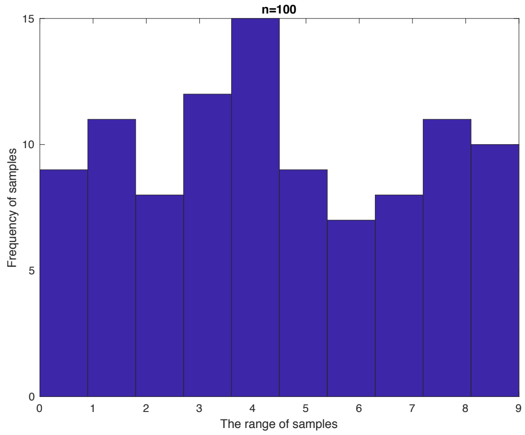
Code:

```
%---------------------------------------------------------------------------
%
% Problem 3                                                                 %
%---------------------------------------------------------------------------
%
n = input('Please enter the number of samples: ');
M = 10;
A = randi([0 M-1],1,n);      % A is the randaom number from 0,1,...,9

hist(A)
title(['n=',num2str(n)])

X = hist(A);
X_theo = repmat(n/M,1,M);    % Expected number of samples
ChisquaredTest = sum((X-X_theo).^2./X_theo);
ChisquaredThreshold_95 = chi2inv(0.95,M-1);

disp(['ChisquaredTest = ',num2str(ChisquaredTest), ...
    ',  ChisquaredThreshold_95 = ',num2str(ChisquaredThreshold_95)]);

X2 = [X(2:10) 0];            % The data in the range 1,2,...,10
ChisquaredTest = sum((X2-X_theo).^2./X_theo);
ChisquaredThreshold_95 = chi2inv(0.95,M-1);

disp(['ChisquaredTest = ',num2str(ChisquaredTest), ...
    ',  ChisquaredThreshold_95 = ',num2str(ChisquaredThreshold_95)]);
%--------------------------------------------------------------------------%
```

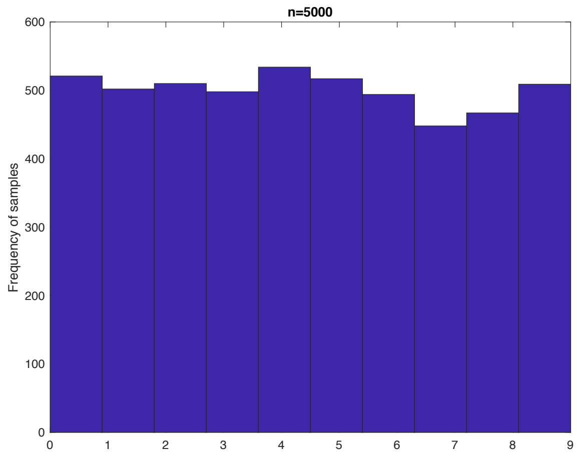
Results:



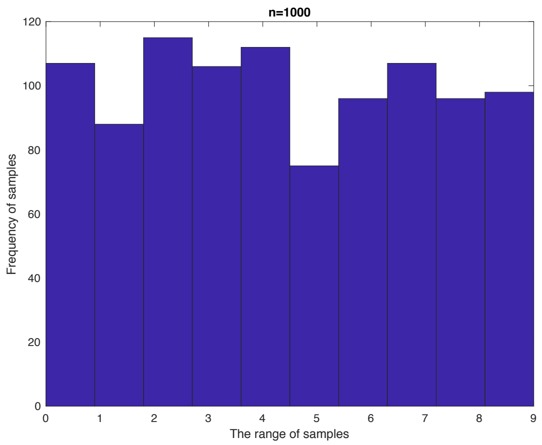
(a)



(b)



(c)



(d)

Here figures a,b,c,d denote the histogram of experiments with sample number 50,100,1000 and 5000.

The results of statistical goodness of fit test for question 3 b are as shown below:

Repeat Time	1	2	3	4	5
Chi-Square test statistic	7.94	6.46	9.58	6.34	13.8
Repeat Time	6	7	8	9	10
Chi-Square test statistic	6.36	4.28	12	9.66	13.88
Repeat Time	11	12	13	14	15
Chi-Square test statistic	12.2	13.48	16.32	15.86	17.2
Repeat Time	16	17	18	19	20
Chi-Square test statistic	10.98	8.6	12.64	10.56	4.96

Observation:

Here we observe that the threshold of chi square test at 95% confidence level is 16.9198 and in the 20 time repeat shown above, only 15th time repeat yielded larger value than the threshold (17.2). Hence it can be concluded that the raw data fits the samples from discrete uniform distributions (1-9).

The results of statistical goodness of fit test for question 3 c are as shown below:

Repeat Time	1	2	3	4	5
Chi-Square test statistic	107.45	106.1	108.37	105.85	113.64
Repeat Time	6	7	8	9	10
Chi-Square test statistic	106.2	103.92	112	106.77	112.67
Repeat Time	11	12	13	14	15
Chi-Square test statistic	112.16	113.39	115.83	110.76	112.29
Repeat Time	16	17	18	19	20
Chi-Square test statistic	109.98	108.56	109.75	110.47	104.47

Observations:

Here we observe that the threshold of chi-square test at the 95% confidence level is also 16.9189 and in the 20 times repeat shown above, all the results are larger than the threshold (16.9189). Hence, it can be concluded that the raw data does not fit the samples from a discrete uniform distribution (1-10).