# Simulation & Modeling

## Course Code: BIT534CO
## Year/ Semester: III/VI

**Compiled by Shishir Ghimire**

Credit Hours: 3hrs

# Unit 4: Random Numbers

Class Load : 10 hrs

# TABLE OF CONTENTS

# Random Numbers and its Properties:

- Random numbers are samples **drawn from a uniformly distributed random variable** between some satisfied intervals, they have **equal probability of occurrence.**
- Random numbers are characterized by the fact that their value **cannot be predicted.**
- **Are the numbers selected by chance?**
- Random numbers are needed in many areas: **Cryptography, Monte Carlo computation and simulation, industrial testing and labelling, hazard games, gambling,** etc.

# Random Numbers and its Properties:

- Random numbers are a **necessary basic ingredient (element) in the simulation of almost all discrete systems.**
- Most computer languages have a **subroutine**, **object**, or **function** that will **generate a random number.**
- To be effective, random numbers **must be**:
  a. **unpredictable**
  b. **statistically independent** (unrelated to any previously generated random numbers)
  c. **uniformly distributed** (equal probability for any number to be generated)
  d. **protected**

# ▶ Random Numbers > Properties :

## General Properties of Random Numbers

- A sequence of random numbers, r1, r2, r3 … .. must have two important properties:
  - **Uniformity:**
    - The random numbers generated should be **uniform**. That means a sequence of random numbers should be equally **probable everywhere**.
    - If we divide all the set of random numbers into several numbers of **class intervals** then numbers of samples in **each class** should be **same**.
    - If 'N' number of random numbers of samples in each class should be equal to $e_i = N/K$.
  - **Independence:**
    - The probability of observing a value in a particular interval is independent of the **previous value drawn**. i.e. the current value of a random variable has **no relation** with the previous values.
    - Each random number should be independent samples drawn from a continuous uniform distribution between 0 and 1.

# Random Numbers and its Properties:

**Types of Random Numbers**

- There are two types of random numbers:
  i. Pseudo-Random Number (PRN) and
  ii. True- Random Numbers (TRN).

- These different types of random numbers have different applications.

# Random Numbers and its Properties:

1. **True Random Numbers**

- A hardware random number generator (HRNG) or **true random number generator (TRNG) is a device that generates random numbers from a physical process, rather than by means of an algorithm.**
  - **Example:** Generated by **dice, roulette wheel, coin toss**
- Such devices are often based on microscopic phenomena that generate low-level, statistically random "**noise**" signals, such as thermal noise, the photoelectric effect, radioactive decay, involving a beam splitter, and other quantum phenomena.
- True random numbers are gained from **physical processes** like radioactive decay or also rolling a dice.

# Random Numbers and its Properties:

## 2. Pseudo Random Numbers

- A set of values or elements that is statistically random, but **it is generated by algorithm and is derived from a known starting point and is typically repeated over and over.**
- It is called "**pseudo**" random, because the algorithm can repeat the sequence, and the numbers are thus **not entirely random**.
- **Every new number is generated from the previous ones by an algorithm.**
- This means that the new value is fully determined by the previous ones. But, depending on the algorithm, they often have properties making them very suitable for simulations.

# ▶ Pseudo Random Numbers > **Problems :**

When generating pseudo-random numbers, **certain problems or errors** can occur. Some examples include the following:

- ❖ The generated numbers **may not be uniformly distributed.**
- ❖ The generated numbers may be discrete-valued **instead continuous valued.** { Numbers are discrete valued and not continuous on [0,1] }
- ❖ The **mean** of the generated numbers may be too high or too low.
- ❖ The **variance** of the generated numbers may be too high or low.
- ❖ There may be **dependence**. The following are the examples:
  - ➢ Autocorrelation between numbers.
  - ➢ Numbers successively higher or lower than adjacent numbers.
  - ➢ Several numbers above the mean followed by several numbers below the mean.

## PRN vs TRN:

| Aspect | Pseudo-random | True-random |
|---|---|---|
| **Nature** | Algorithm of mathematical formula, later translated into programming code | Extract randomness from physical phenomena and introduce it into a computer |
| **Response Speed** | Fast responses in generating numbers | Slow responses in generating numbers |
| **Reproducibility** | Sequence of numbers can be reproduced | Sequence of numbers cannot be reproduced |
| **Repetition of Sequence** | Sequence of numbers is repeated | Sequence of numbers will or will not get repeated |
| **Deterministic** | Deterministic | Non-deterministic |

# Random Number Tables:

- A random number table is a series of digits (0 to 9) arranged randomly in **rows and columns**.
- Random number tables have been used in statistics for tasks such as **selecting random samples**.
- This was much more **effective than** manually selecting the random samples (with dice, cards, etc.).
- Nowadays, tables of random numbers have been **replaced** by computational random number generators.
- **But how do researchers determine a sample set that will truly be representative of the larger population?**
- The answer to obtaining a **representative sample can be the use of a table of random numbers** to select each member of the sample set.

# ▶ Random Number Tables:

Table of Random Numbers

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 36518 | 36777 | 89116 | 05542 | 29705 | 83775 | 21564 | 81639 | 27973 | 62413 | 85652 | 62817 | 57881 |
| 46132 | 81380 | 75635 | 19428 | 88048 | 08747 | 20092 | 12615 | 35046 | 67753 | 69630 | 10883 | 13683 |
| 31841 | 77367 | 40791 | 97402 | 27569 | 90184 | 02338 | 39318 | 54936 | 34641 | 95525 | 86316 | 87384 |
| 84180 | 93793 | 64953 | 51472 | 65358 | 23701 | 75230 | 47200 | 78176 | 85248 | 90589 | 74567 | 22633 |
| 78435 | 37586 | 07015 | 98729 | 76703 | 16224 | 97661 | 79907 | 06611 | 26501 | 93389 | 92725 | 68158 |
| 41859 | 94198 | 37182 | 61345 | 88857 | 53204 | 86721 | 59613 | 67494 | 17292 | 94457 | 89520 | 77771 |
| 13019 | 07274 | 51068 | 93129 | 40386 | 51731 | 44254 | 66685 | 72835 | 01270 | 42523 | 45323 | 63481 |
| 82448 | 72430 | 29041 | 59208 | 95266 | 33978 | 70958 | 60017 | 39723 | 00606 | 17956 | 19024 | 15819 |
| 25432 | 96593 | 83112 | 96997 | 55340 | 80312 | 78839 | 09815 | 16887 | 22228 | 06206 | 54272 | 83516 |
| 69226 | 38655 | 03811 | 08342 | 47863 | 02743 | 11547 | 38250 | 58140 | 98470 | 24364 | 99797 | 73498 |
| 25837 | 68821 | 66426 | 20496 | 84843 | 18360 | 91252 | 99134 | 48931 | 99538 | 21160 | 09411 | 44659 |
| 38914 | 82707 | 24769 | 72026 | 56813 | 49336 | 71767 | 04474 | 32909 | 74162 | 50404 | 68562 | 14088 |
| 04070 | 60681 | 64290 | 26905 | 65617 | 76039 | 91657 | 71362 | 32246 | 49595 | 50663 | 47459 | 57072 |
| 01674 | 14751 | 28637 | 86980 | 11951 | 10479 | 41454 | 48527 | 53868 | 37846 | 85912 | 15156 | 00865 |
| 70294 | 35450 | 39982 | 79503 | 34382 | 43186 | 69890 | 63222 | 30110 | 56004 | 04879 | 05138 | 57476 |
| 73903 | 98066 | 52136 | 89925 | 50000 | 96334 | 30773 | 80571 | 31178 | 52799 | 41050 | 76298 | 43995 |
| 87789 | 56408 | 77107 | 88452 | 80975 | 03406 | 36114 | 64549 | 79244 | 82044 | 00202 | 45727 | 35709 |
| 92320 | 95929 | 58545 | 70699 | 07679 | 23296 | 03002 | 63885 | 54677 | 55745 | 52540 | 62154 | 33314 |
| 46391 | 60276 | 92061 | 43591 | 42118 | 73094 | 53608 | 58949 | 42927 | 90993 | 46795 | 05947 | 01934 |
| 67090 | 45063 | 84584 | 66022 | 48268 | 74971 | 94861 | 61749 | 61085 | 81758 | 89640 | 39437 | 90044 |
| 11666 | 99916 | 35165 | 29420 | 73213 | 15275 | 62532 | 47319 | 39842 | 62273 | 94980 | 23415 | 64668 |
| 40910 | 59068 | 04594 | 94576 | 51187 | 54796 | 17411 | 56123 | 66545 | 82163 | 61868 | 22752 | 40101 |
| 41169 | 37965 | 47578 | 92180 | 05257 | 19143 | 77486 | 02457 | 00985 | 31960 | 39033 | 44374 | 28352 |
| 76418 | | | | | | | | | | | | |

# Generation of Random Numbers:

**Important considerations to generate the Random Numbers:**

1. The routine should be **fast**. Individual computations are inexpensive, but a simulation may require many millions of random numbers.
2. The routine should be **portable** to different computers – ideally to different programming languages. This ensures the program produces same results.
3. The routine should have sufficiently **long cycle.** The cycle length, or period, represents the length of random number sequence **before previous numbers begin to repeat** in an earlier order.
4. The random numbers should be **replicable.** Given the starting point, it should be possible to generate the same set of random numbers, completely independent of the system that is being simulated.
5. The random numbers should closely approximate the **ideal statistical properties** of uniformity and independence.

# Methods of Random Number Generation:

1. Linear Congruential Method
2. Multiplicative Congruential Method
3. Mid Square Method

# Methods of Random Number Generation:

❖ **Uniform - Linear Congruential Method:**

➢ The linear congruential method, initially proposed by Lehrer [1951], produces a sequence of integers, $X_1$, $X_2$,... between zero and m-1 according to the following recursive relationship:

$$X_{i+1} = (a * X_i + c) \bmod m \text{.........................(i)}$$

where,

**i** = 0,1,2..............

$X_0$ - initial value, is called the seed,

**a** is called the constant multiplier,

**c** is the increment

**m** is the modulus.

# Methods of Random Number Generation:

- **Uniform - Linear Congruential Method:**
  - **Case-1:** If $c \neq 0$ in Equation (i), the form is called the **mixed congruential method.**
  - **Case-2:** When $c = 0$ in Equation (i), the form is known as **the multiplicative congruential method.**
  - **Case-3:** When $a = 1$ in Equation (i), the form is known as **the additive congruential method.**
    - The selection of the values for a, c, m and $X_o$ drastically affects the statistical properties and the cycle length.

# Methods of Random Number Generation:

- **Uniform - Linear Congruential Method:**
  - **Example:** For the values selection with $X_{(0)} = 30$, $a = 12$, $c = 21$ and $m = 100$, the sequence of random numbers generated are as follows:
    - $X_{(0)} = 30$
    - $X_{(1)} = (12 * 30 + 21) \bmod 100 = 381 \bmod 100 = 81$
    - $X_{(2)} = (12 * 81 + 21) \bmod 100 = 993 \bmod 100 = 93$
    - $X_{(3)} = (12 * 93 + 21) \bmod 100 = 1137 \bmod 100 = 37$ **and so on…**
  - The random numbers between [0, 1] generated are as follows:
    - R(0) = 0.3
    - R(1) = 0.81
    - R(2) = 0.93
    - R(3) = 0.37 **and so on…….**

**Question: For the values selection with $X_{(0)}$ = 27, a = 17, c = 43 and m = 100, Generate the sequence of random number for i = 5. (Method to follow in exam)**

**Given $X_0$ = 27, a = 17, c = 43 and m = 100, then i = 5**

$X_1 = (17*27 + 43) \bmod 100 = 2$
$R_1 = 2/100 = 0.02$

$X_2 = (17*2 + 43) \bmod 100 = 77$
$R_2 = 77/100 = 0.77$

$X_3 = (17*77 + 43) \bmod 100 = 52$
$R_3 = 52/100 = 0.52$

$X_4 = (17*52 + 43) \bmod 100 = 27$
$R_4 = 27/100 = 0.27$

$X_5 = (17*27 + 43) \bmod 100 = 2$
$R_5 = 2/100 = 0.02$

Therefore, the generated random numbers after 5 iterations are **0.02, 0.77, 0.52, 0.27 and 0.02.**

# Methods of Random Number Generation:

## **Multiplicative Congruential Method:**

$$X_{i+1} = (a * X_i) \bmod m$$

Let, $a = 9$, $m = 31$ & $X_0 = 2$

Then, $X_1 = (a * X_0) \bmod m$

$= (9 \times 2) \bmod 31 = 18 \bmod 31 = 18$

$X_2 = (aX_1) \bmod m$

$= (9 \times 18) \bmod 31 = 162 \bmod 31 = 7$

And $X_3=1$, $X_4=9$, $X_5=19$

Hence, random number are 2,18,7,1,9,19,16,20.....

# ▶ Methods of Random Number Generation:

❖ **Classwork-1:**

- ➢ Let m = 100, a = 19, c = 0, and $X_0$ = 63, and generate a sequence random integers. Find first 7 random number generate using any suitable method?
- ➢ Use the linear congruential method to generate a sequence of random numbers with $X_0$ = 27, a= 17, c = 43, and m = 100.
- ➢ Use the Multiplicative congruential method to generate a sequence of four-three digit random integers, with seed = 117, constant multiplier = 43 and modulus = 1000.
- ➢ Use multiplicative congruential method to generate a sequence of three digits random numbers between (0, 1) with X = 27, a=3 and m =1000.
- ➢ Use the linear congruential method to generate a sequence of three two-digit random integers. Let $X_0$ =29, a= 9, c=49, m=100
- ➢ Use the multiplicative congruential method to generate five three digit random integers. $X_0$ =118, a=45 and m = 1000

**Question:** Given the following parameters for a Mixed Congruential Generator (MCG): m = 100, a = 37, b = 10 and Seed $r_0$ = 5. Using the Mixed Congruential Method, generate the first 5 random numbers. Show the steps and the numbers generated. **2025 ( 8 Marks)**

# Methods of Random Number Generation:

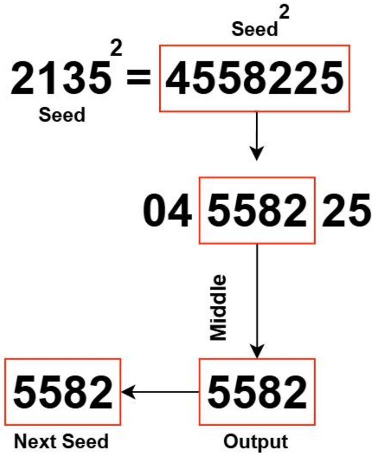
## **Mid-Square Method:**

- This method was proposed by John Von Neumann to generate a sequence of 4-digit random numbers and was described at a conference in 1949.
- This was **one of the first methods** used to generate pseudo-random numbers.
- In this method, **we have a seed and then the seed is squared and its midterm is fetched as the random number.**
- It takes the square of the seed to make it an 8-bit number, if not 8 bits, add zeros to the start to make it an 8-bit number. After making it an 8-bit number, take the middle 4 digits. We have a new random number, and the process goes on.

# ▶ Methods of Random Number Generation:

❖ **Steps of Mid-Square Method (Algorithm):**

1. Start with an $n$-digit number. (**Note:** n is even)
2. Square the seed to produce a number of length up to $2n$ digits.
3. Extract the middle $n$ digits of the squared number based on the following rules:
   - If the result is $2n$ digits long → take exactly the middle $n$ digits.
   - If the result has less than $2n$ digits → add leading zeros, then take middle $n$ digits.
4. Use the extracted $n$-digit number as the next seed.
5. Repeat the process from Step 2.



# Methods of Random Number Generation:

- **Mid-Square Method (4 digit seed):**
  - For example, we are using 4 digit which is called seed number, we are generating 5 random numbers here
  - $(5673)^2 = 32182929 = 1829$ (next seed number)
  - $(1829)^2 = 03325241 = 3452$
  - $(3452)^2 = 11916304 = 9163$
  - $(9163)^2 = 83960569 = 9605$
  - $(9605)^2 = 92256025 = 2560$

# Methods of Random Number Generation:

- **Mid-Square Method (2-digit seed):**
  - Example: Consider the seed to be 14 and we want a two digit random number.
    - **Number --> Square --> Mid-term**

| | | |
|---|---|---|
| 14 | --> 0196 | --> 19 |
| 19 | --> 0361 | --> 36 |
| 36 | --> 1296 | --> 29 |
| 29 | --> 0841 | --> 84 |
| 84 | --> 7056 | --> 05 |
| 05 | --> 0025 | --> 02 |
| 02 | --> 0004 | --> 00 |
| 00 | --> 0000 | --> 00 |

# Methods of Random Number Generation:

## Mid-Square Method > Problems:

- Difficult to assure that the **sequence will not degenerate** over a long period of time.
  - Zeros once they appear are carried in subsequent numbers (try 5197 as a seed).
- If the **middle k digits are all zeroes**, the generator then outputs zero forever.
  - If the first half of a number in the sequence is zeroes, the subsequent numbers will be decreasing to zero. While these runs of zero are easy to detect, they occur too frequently for this method to be of practical use.

## Classwork > Lets Try !!!

1. **Q1.** Using the Mid-Square Method, generate the first five random numbers starting with the seed 37 (2-digit seed). Show each step and express the random numbers in the range [0,1].

2. **Q2.** Using the Mid-Square Method, generate the first five random numbers starting with the seed 5731 (4-digit seed). Show each step and convert the sequence into uniform random numbers in the range [0,1].

# Methods of Random Number Generation:

- **Homework:**

  a. Use Multiplicative congruential method to generate a sequence of 10 **three-digit** random integers and corresponding random variables. Let $X_0$ =4, a=3

     ***Hint : for three-digit random integers m=1000***

  b. Use the mixed congruential method to generate a sequence of three two-digit random numbers with $x_0$=37, a=7, c=29 and m =100

  c. Use the multiplicative congruential method to generate a sequence of four three-digit random numbers. Let $X_0$ =118, a=4 and m =100

## Qualities of an Efficient Random Number Generator:

1. The sequence of random numbers generated must follow the uniform (0, 1) distribution.
2. The sequence of random numbers generated must be statistically independent.
3. The sequence of random numbers generated must be reproducible. This allows replication of the simulation experiment.
4. The sequence must be non-repeating for any desired length. Although not theoretically possible, a long repeatability cycle is adequate for practical purposes.
5. Generation of the random numbers must be fast because in simulation studies a large number of random numbers are required. A slow generator will greatly increase the time and cost of the simulation studies/experiments.
6. The technique used in generating random numbers should require little computer memory.

# Tests for Randomness:

- **Testing for Randomness:**
  - The desirable **properties of random numbers — uniformity and independence to ensure that these desirable properties are achieved**, a number of tests can be performed.
  - The tests can be placed in two categories according to the properties of interest.
    - Testing for uniformity
    - Testing for independence.

# Tests for Randomness > Uniformity and independence

There are different types of test:

- **Frequency test:** Uses the **Kolmogorov-Smirnov (KS) or Chi-square test** to **compare the distribution** of the set of numbers generated to a **uniform distribution**.
- **Runs test:** Tests the **runs up and down** or the runs above or below the mean by comparing the **actual value to expected value.** The statistics for comparison in Chi-square.
- **Autocorrelation test:** Tests the **correlation between numbers** and compares the sample correlation to the expected correlation of zero.
- **Gap test:** Counts the **number of digits that appear between repetition** of a particular digit and then uses KS test to compare with the expected size of gaps.
- **Poker test:** Treats numbers group together **as a poker hand.** Then the hands obtained are compared to what is expected using the Chi-square test.

# Tests for Randomness:

- **Testing for Randomness:**
  - **Uniformity Test**
    - Kolmogorov Smirnov Test (n <= 20)
    - Chi-square Test (n > 20)
  - **Independent Test**
    - Autocorrelation Test
    - Gap Test
    - Poker Test

# Tests for Randomness > Hypothesis :

To check on the desirable properties of random number i.e. uniformity and independence, a number of tests can be performed.

1. **Test for Uniformity:**

   The hypotheses are as follows:

   $H_0: R_i \sim U[0,1]$

   $H_1: R_i \neq U[0,1]$

   Null hypothesis, $H_0$ reads that the numbers are uniformly distributed on the interval [0,1]. **Failure to reject** null hypothesis means that evidence of non uniformity has not been detected by this test.

2. **Test for Independence:**

   The hypotheses are as follows:

   $H_0: R_i =$ *independently generated*

   $H_1: R_i \neq$ *independently generated*

   Null hypothesis, $H_0$ reads that the numbers are independent. **Failure to reject** the null hypothesis means an evidence of dependence has not been detected by this test.

# K-S Test

## Tests for Uniformity> Kolmogorov-Smirnov (KS) Test:

- Test is based on the largest absolute deviation statistic between $F(x)$ *and* $S_N(x)$ over the range of the random variable: $D = max|\ F(x) - S_N(x)|$
- ***Algorithm for Kolmogorov- Smirnov Test :***
  1. ***Rank the data from smallest to largest***. Let $R_{(i)}$ denote $i^{th}$ smallest observation, so that $R_{(1)} \leq R_{(2)} \leq \ldots \leq R_{(N)}$
  2. Compute
  $$D^{+} = \max_{1 \leq i \leq N} \left\{ \frac{i}{N} - R_{(i)} \right\}; \quad D^{-} = \max_{1 \leq i \leq N} \left\{ R_{(i)} - \frac{i-1}{N} \right\}$$
  3. Compute $D = max(D^{+}, D^{-})$
  4. Locate in Table the *critical value* $D\alpha$, for the specified significance level $\alpha$ and the sample size $N$.
  5. If the sample statistic ***D is greater than*** the critical ***value*** $D\alpha$, the null ***hypothesis is rejected***. If $D \leq D\alpha$, conclude ***there is no difference.***

# Numericals Done in Class

**Q1. Perform Kolmogorov-Smirnov test for the following random numbers with level of significance of $\alpha = 0.05$. The critical value for $\alpha = 0.05$ is 0.410 for sample size N = 10. Random numbers are: 0.24, 0.89, 0.11, 0.61, 0.23, 0.86, 0.41, 0.64, 0.50, 0.65.**

Q2. Suppose that the five numbers 0.44, 0.81, 0.14, 0.05, 0.93 were generated, and it is desired to perform a test for uniformity using the Kolmogorov-Smirnov Test at $\alpha = 0.05$ level of significance.

## Homework > Kolmogorov-Smirnov (KS) Test:

1. The following numbers have been generated 0.39, 0.67, 0.45, 0.78, and 0.55. Use the Kolmogorov- Smirnov Test to check whether given numbers are uniformly distributed or not. [*Use the critical value of D for* $\alpha=0.05$ *and* $N=5$ *is 0.565.*] .

2. The following numbers have been generated 0.44, 0.19, 0.88, 0.27, 0.55, 0.13, 0.63, 0.74, 0.24 and 0.33. Use the Kolmogorov- Smirnov Test with $\alpha=0.05$ to determine, if the hypothesis that the numbers are uniformly distributed on the interval[0,1] can be rejected. [*Use the critical value of D for* $\alpha=0.05$ *and* $N=10$ *is 0.410.*] .

# Chi-Square Test

# Tests for Uniformity > Chi Square test:

The Chi-square test compares the observed frequencies (actual data) to the expected frequencies (what we would expect if there was no relationship).

**Formula for Chi-square test:**

Chi-square statistic is calculated as:

$$\chi_c^2 = \sum_{i=1}^{n} \frac{(O_i - E_i)^2}{E_i}$$

Where,

c is degree of freedom , n is the number of classes

$O_i$ is the observed number in i-th class

$E_i$ is the expected number in the i-th class

## ▶ Tests for Uniformity > Chi Square test:

**Algorithm:**

Step 1: Determine Order Statistics

$$R_{(1)} \leq R_{(2)} \leq \ldots \leq R_{(N)} \quad (7.3)$$

Step 2: Divided Range $R_{(N)} - R_{(1)}$ in $n$ equidistant intervals $[a_i, b_i]$, such that each interval has at least 5 observations.

Step 3: Calculate for $i = 1, \ldots, N$

$$O_i = N \cdot \{S_N(b_i) - S_N(a_i)\},\ E_i = N \cdot \{F(b_i) - F(a_i)\}$$

Step 4: Calculate

$$\chi_0^2 = \sum_{i=1}^{n} \frac{(O_i - E_i)^2}{E_i}$$

Step 5: Determine for significant level $\alpha$, $\chi^2_{\alpha,n-1}$

$$\begin{cases} \chi_0^2 \leq \chi^2_{\alpha,n-1} & \text{Accept: No Difference between } S_N(x) \text{ and } F(x) \\ \chi_0^2 > \chi^2_{\alpha,n-1} & \text{Reject: Difference between } S_N(x) \text{ and } F(x) \end{cases}$$

## Tests for Uniformity > Chi Square test:

Theory + Numericals Done in Class

## Tests for Uniformity > Chi Square test:

- Use Chi-square test for the data shown below with $\alpha=0.05$. The test uses $n=10$ intervals of equal length, namely [0,0.1),[0.1,0.2), ….., [0.9,1.0).

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 0.34 | 0.90 | 0.25 | 0.89 | 0.87 | 0.44 | 0.12 | 0.21 | 0.46 | 0.67 |
| 0.83 | 0.76 | 0.79 | 0.64 | 0.70 | 0.81 | 0.94 | 0.74 | 0.22 | 0.74 |
| 0.96 | 0.99 | 0.77 | 0.67 | 0.56 | 0.41 | 0.52 | 0.73 | 0.99 | 0.02 |
| 0.47 | 0.30 | 0.17 | 0.82 | 0.56 | 0.05 | 0.45 | 0.31 | 0.78 | 0.05 |
| 0.79 | 0.71 | 0.23 | 0.19 | 0.82 | 0.93 | 0.65 | 0.37 | 0.39 | 0.42 |
| 0.99 | 0.17 | 0.99 | 0.46 | 0.05 | 0.66 | 0.10 | 0.42 | 0.18 | 0.49 |
| 0.37 | 0.51 | 0.54 | 0.01 | 0.81 | 0.28 | 0.69 | 0.34 | 0.75 | 0.49 |
| 0.72 | 0.43 | 0.56 | 0.97 | 0.30 | 0.94 | 0.96 | 0.58 | 0.73 | 0.05 |
| 0.06 | 0.39 | 0.84 | 0.24 | 0.40 | 0.64 | 0.40 | 0.19 | 0.79 | 0.62 |
| 0.18 | 0.26 | 0.97 | 0.88 | 0.64 | 0.47 | 0.60 | 0.11 | 0.29 | 0.78 |

## Homework > Chi Square test:

A set of 100 random numbers have been generated as below:

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| 0.23 | 0.30 | 0.07 | 0.39 | 0.43 | 0.47 | 0.66 | 0.19 | 0.49 | 0.37 | 0.53 |
| 0.80 | 0.56 | 0.10 | 0.28 | 1.00 | 0.23 | 0.04 | 0.93 | 0.25 | 0.90 | 0.24 |
| 0.88 | 0.75 | 0.54 | 0.84 | 0.42 | 0.68 | 0.12 | 0.68 | 0.16 | 0.72 | 0.93 |
| 0.14 | 0.71 | 0.23 | 0.76 | 0.96 | 0.89 | 0.82 | 0.99 | 0.42 | 0.22 | 0.77 |
| 0.36 | 0.95 | 0.35 | 0.14 | 0.89 | 0.26 | 0.99 | 0.78 | 0.23 | 0.75 | 0.67 |
| 0.34 | 0.04 | 0.95 | 0.41 | 0.89 | 0.64 | 0.88 | 0.92 | 0.62 | 0.84 | 0.36 |
| 0.88 | 0.32 | 0.86 | 0.24 | 0.42 | 0.54 | 0.35 | 0.60 | 0.88 | 0.84 | 0.15 |
| 0.86 | 0.99 | 0.63 | 0.56 | 0.16 | 0.53 | 0.86 | 0.69 | 0.74 | 0.70 | 0.38 |
| 0.48 | 0.67 | 0.54 | 0.97 | 0.50 | 0.09 | 0.20 | 0.72 | 0.27 | 0.72 | 0.96 |

Test these values for uniform distribution using Chi Square Test (Use $\alpha=0.05$)

# Autocorrelation Test

## Tests for Independence > AutoCorrelation Test :

# Theory + Numericals Done in Class

## Tests for Independence > AutoCorrelation Test :

- Example: Test whether the *3rd*, *8th*, *13th*, and so on are auto-correlated or not.

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 0.12 | 0.01 | 0.23 | 0.28 | 0.89 | 0.31 | 0.64 | 0.28 | 0.83 | 0.93 |
| 0.99 | 0.15 | 0.33 | 0.35 | 0.91 | 0.41 | 0.60 | 0.27 | 0.75 | 0.88 |
| 0.68 | 0.49 | 0.05 | 0.43 | 0.95 | 0.58 | 0.19 | 0.36 | 0.69 | 0.87 |

# Poker Test ( 3 & 4 Digit )

# Tests for Independence > Poker Test :

## Theory + Numericals Done in Class

# Poker Test > 3-Digit :

- ***Example:*** A sequence of ***1000 three-digit numbers*** has been generated and an analysis indicates that ***680 have three different digits***, ***289 contain exactly one pair of like digits***, and ***31 contain three like digits***. Based on the poker test, are these numbers independent? Let $a = 0.05$.

# Poker Test ➤ 4-digit :

- A sequences of 1,000 four-digit numbers has been generated & an analysis indicates the following combinations and frequencies. Based on poker test check whether the numbers are independent. Use $\alpha$ =0.05 and $X^2$ = 9.49.

| Combination | Observed Frequencies |
|---|---|
| Four Different Digits | 540 |
| One Pair | 380 |
| Two Pairs | 34 |
| Three Like digits | 25 |
| Four Like digits | 1 |
| Total | 1,000 |

# PYQs:

1. Explain the independence and uniformity property of random number. For the following sample of random numbers, perform test for independence using K-S test. (D0.05,10 = 0.41) 0.35, 0.77, 0.12, 0.33, 0.88, 0.45, 0.19, 0.25, 0.91, 0.54 **2024 PU (8)**
2. What do you mean by Poker Test? Write significance of testing number for randomness in statistical testing.**2024 PU (8)**
3. What are the properties of random number ? The sequence of numbers 0.54, 0.73, 0.98, 0.11 and 0.68 has been generated. Use the Kolmogorov Smirnov test with level of significance (α) 0.05 to determine if the hypothesis that the numbers are uniformly distributed on the interval 0 to 1 can be rejected. (Note that the value of D for α = 0.05 and N = 5 is 0.565). **2025 (8), 2024 PU (12)**
4. Given the following parameters for a Mixed Congruential Generator (MCG): m = 100, a = 37, b = 10 and Seed $r_0$ = 5. Using the Mixed Congruential Method, generate the first 5 random numbers. Show the steps and the numbers generated. **2025 PU (8)**

# References:


1. Jerry Banks, John S. Carson II, Barry L. Nelson, David M. Nicol – *Discrete-Event System Simulation* – Pearson (2013)
2. Averill M. Law – *Simulation Modeling and Analysis* – McGraw-Hill Education (2014)
3. Geoffrey Gordon – *System Simulation*

# THANKS!

**Do you have any questions?**


**theciceerguy@p m.me**

**9841893035**

**ghimires.com.np**