قســـــــم الامـــــــــــــــن
الــــــــــــــــسيبرانـــــــــــــــــــ
ي

Department of Cyber Security

Subject:

Measure of Randomness

Class:

Second

Lecturer:

Asst. raed alshmary

Lecture: (9)

Introduction

Introduction

In the field of cybersecurity and cryptography, randomness plays a critical role in ensuring the security and robustness of systems. Random sequences are essential for tasks such as generating cryptographic keys, initializing secure connections, and creating unpredictable outcomes in simulations and algorithms. However, not all sequences that appear random are truly random. To evaluate the randomness of a binary sequence, statistical tests are employed.Measure of Randomness refers to a set of techniques and tests used to determine whether a sequence exhibits characteristics typical of a random sequence. These tests help identify patterns or predictability in sequences that could compromise their security. Understanding and applying these measures ensure that the random number generators and cryptographic systems remain secure and reliable.This lecture will introduce key concepts such as runs, gaps, and blocks, and explain five fundamental statistical tests used to assess randomness in binary sequences.

Key Concepts

1. Run: A sequence of identical bits (e.g., 0 or 1).
   Example: In the sequence '01110000111', the runs are '0', '111', and '0000', '111'.

2. Gap: A run of zeros.
   Example: In '1000011', the gap is '0000'.

3. Block: A run of ones.
   Example: In '1111001110', blocks are '1111' and '111'.

Five Basic Tests

These tests help determine whether a binary sequence exhibits characteristics of randomness.

1. Frequency Test (Monobit Test)

Purpose: To check if the number of 0's and 1's in the sequence are approximately equal. Statistic:

$$X_1 = \frac{(n_0 - n_1)^2}{n}$$

Where:

- n_0: Number of 0's in the sequence.
- n_1: Number of 1's in the sequence.
- n: Total length of the sequence.

Distribution: χ1 with 1 degree of freedom if n ≥ 10.

2. Serial Test (Two-Bit Test)

Purpose: To check if occurrences of '00', '01', '10', and '11' are approximately equal. Statistic: Based on a χ^2 test for overlapping bit pairs.

$$X_2 = \frac{4}{n-1}(n_{00}^2 + n_{01}^2 + n_{10}^2 + n_{11}^2) - n$$

Where:

- n_{00},n_{01} ,n_{10} ,n_{11} : Counts of each two-bit pair.
- $n-1$: Total number of two-bit pairs.

Distribution: χ^2 with 3 degrees of freedom.

3. Poker Test

Purpose: Divides the sequence into blocks of size m and checks if each possible block appears equally often.

Statistic: χ3 with 2^m - 1 degrees of freedom.

$$X_3 = \frac{2^m}{k}\sum_{i=1}^{2^m} n_i^2 - k$$

Where:

- m: Block size.
- k: Number of blocks $= n/m$.
- ni: Number of occurrences of each block.

Note: Setting m=1 reduces this to the frequency test.

4. Runs Test

Purpose: To evaluate if the number and lengths of runs (gaps or blocks) match expected values.
Statistic: Based on expected number of gaps or blocks of each length.

$$X_4 = \sum_{i=1}^{k}\frac{(B_i - e_i)^2}{e_i} + \sum_{i=1}^{k}\frac{(G_i - e_i)^2}{e_i}$$

Where:

- *Bi*,*Gi*: Observed number of blocks and gaps of length *i*.
- *ei*: Expected value for gaps/blocks of length *i*.

Distribution: χ4 with *2k* - *2* degrees of freedom.

5. Autocorrelation Test

Purpose: Detects correlations between the sequence and shifted versions of itself.
Statistic: A(d) compares s to its d-shift using XOR.

$$X_5 = \frac{A(d) - (n-d)/2}{\sqrt{(n-d)/4}}$$

Where:

- $A(d)$: Number of bits differing from their d- shifted versions.

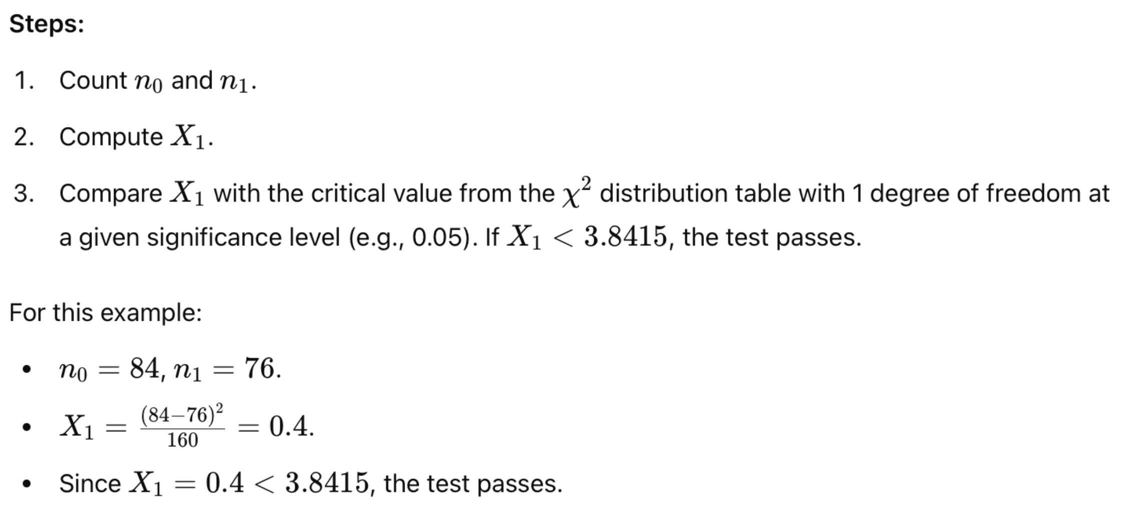
Distribution: N (0,1) for sufficiently large n.

Example

Consider the sequence of length n = 160, obtained by replicating the following sequence four times:

11100 01100 01000 10100 11101 11100 10010 01001.
Results of the tests:
1. Frequency Test:

$$X_1 = \frac{(n_0 - n_1)^2}{n}$$



Steps:

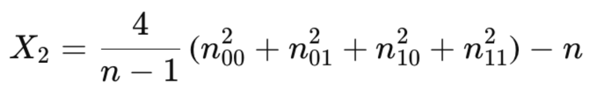
1. Count n_0 and n_1.
2. Compute X_1.
3. Compare X_1 with the critical value from the χ^2 distribution table with 1 degree of freedom at a given significance level (e.g., 0.05). If $X_1 < 3.8415$, the test passes.

For this example:

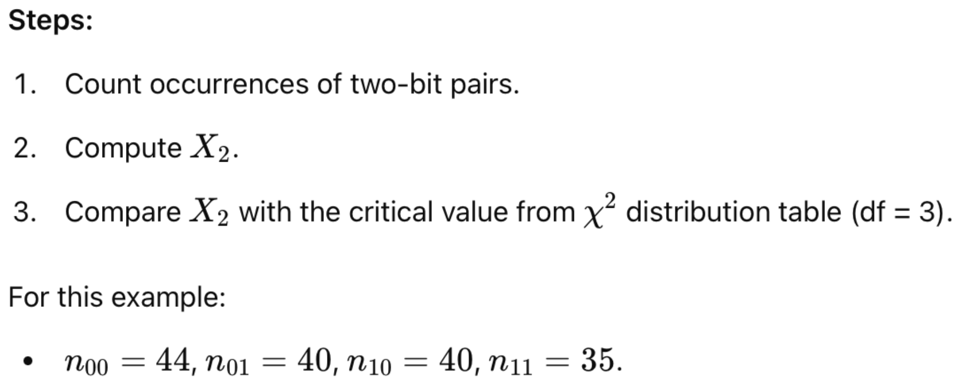
- $n_0 = 84$, $n_1 = 76$.
- $X_1 = \frac{(84-76)^2}{160} = 0.4$.
- Since $X_1 = 0.4 < 3.8415$, the test passes.

Pass (X1 = 0.4)

2. Serial Test:



$$X_2 = \frac{4}{n-1}\left(n_{00}^2 + n_{01}^2 + n_{10}^2 + n_{11}^2\right) - n$$



Steps:

1. Count occurrences of two-bit pairs.
2. Compute X_2.
3. Compare X_2 with the critical value from χ^2 distribution table (df = 3).

For this example:

- $n_{00} = 44$, $n_{01} = 40$, $n_{10} = 40$, $n_{11} = 35$.

X2 = 4/159 (44^2 + 40^2 + 40^2 + 35^2) - 160 = 0.6252

- Since X2<7.815 the test passes.

Pass (X2 = 0.6252)

3. Poker Test:

$$X_3 = \frac{2^m}{k} \sum_{i=1}^{2^m} n_i^2 - k$$

Steps:

1. Divide the sequence into k blocks of size $m = 3$.
2. Count occurrences of each block.
3. Compute X_3.
4. Compare X_3 with the critical value (χ^2, df = $2^m - 1$).

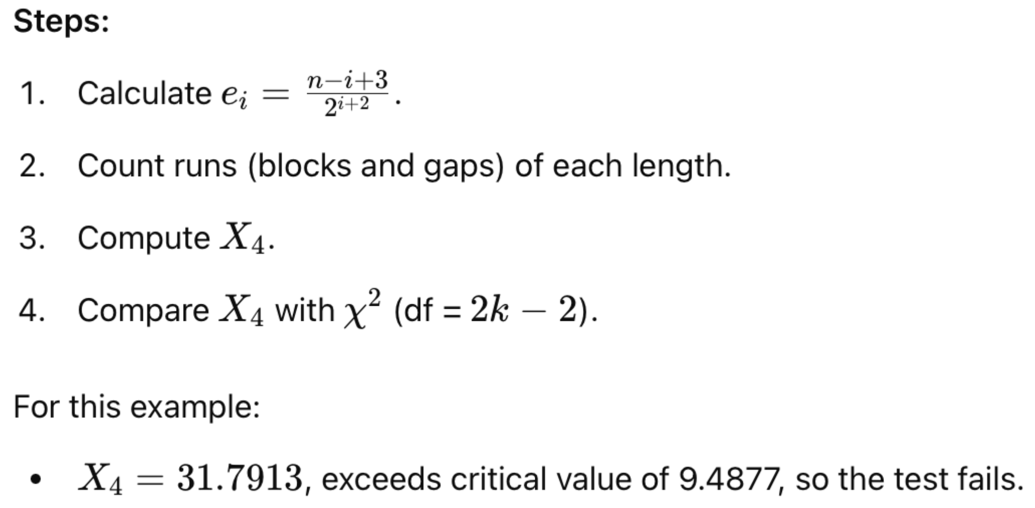
For this example:

- Blocks: `000`, `001`, `010`, `011`, `100`, `101`, `110`, `111`.
- Frequencies: 5, 10, 6, 4, 12, 3, 6, 7.
- $X_3 = \frac{8}{53} \times (5^2 + 10^2 + 6^2 + 4^2 + 12^2 + 3^2 + 6^2 + 7^2) - 53 = 9.6415.$
- Since $X_3 < 14.0671$, the test passes.

Pass (X3 = 9.6415)

4. Runs Test:

$$X_4 = \sum_{i=1}^{k} \frac{(B_i - e_i)^2}{e_i} + \sum_{i=1}^{k} \frac{(G_i - e_i)^2}{e_i}$$



Steps:

1. Calculate $e_i = \frac{n-i+3}{2^{i+2}}$.
2. Count runs (blocks and gaps) of each length.
3. Compute X_4.
4. Compare X_4 with χ^2 (df = $2k - 2$).

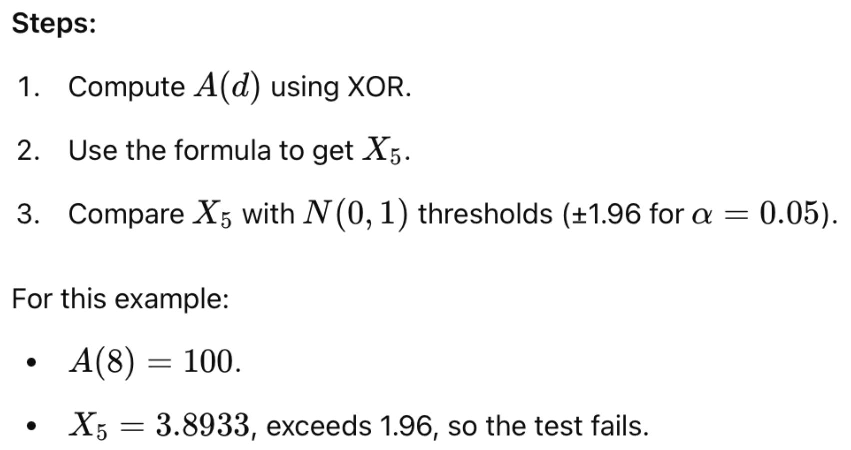
For this example:

- $X_4 = 31.7913$, exceeds critical value of 9.4877, so the test fails.

Fail (X4 = 31.7913)

5. Autocorrelation Test:

$$X_5 = \frac{A(d) - (n-d)/2}{\sqrt{(n-d)/4}}$$



Steps:

1. Compute $A(d)$ using XOR.
2. Use the formula to get X_5.
3. Compare X_5 with $N(0,1)$ thresholds (±1.96 for $\alpha = 0.05$).

For this example:

- $A(8) = 100$.
- $X_5 = 3.8933$, exceeds 1.96, so the test fails.

Fail (X5 = 3.8933)