Probability of No Tails in Three Coin Tosses
Z-Transform Approach

Y. Harsha Vardhan Reddy
EE24BTECH11063
IIT HYDERABAD

Problem Statement

Question:

Find the probability that no toss results in a tail when a coin is tossed three times (independent tosses).

Theoretical Solution

- The probability of getting a tail (T) in a single toss is:

$$P(T) = \frac{1}{2}$$

- Since the tosses are independent:

$$P(\text{Three Tails}) = P(T) \times P(T) \times P(T)$$

- Substituting $P(T) = \frac{1}{2}$:

$$P(\text{Three Tails}) = \left(\frac{1}{2}\right)^3 = \frac{1}{8}$$

PMF for a Single Coin Toss

Probability Mass Function (PMF):

$$P_X(k) = \begin{cases} p, & \text{if } x = 1 \text{ (Tails)} \\ 1 - p, & \text{if } x = 0 \text{ (Heads)} \end{cases}$$

For a fair coin: $p = 0.5$.

Conditions for a Valid PMF:

1. **Non-negativity:** $\forall k, P_X(k) \geq 0$.
2. **Normalization:** $\sum_k P_X(k) = 1$.

For a single coin toss:

$$P_X(0) + P_X(1) = 1. \quad \text{(Normalization condition)}$$

$$\text{Substituting } p = 0.5 :$$

$$P_X(0) = 0.5,$$

$$P_X(1) = 0.5.$$

Problem Statement and Z-Transform

Problem Statement: In n coin tosses, let X denote the number of tails, which follows a binomial distribution. The probability mass function (PMF) is:

$$P(X = k) = \binom{n}{k} p^k (1 - p)^{n-k}$$

Z-Transform Definition: The Z-transform of a sequence a_k is defined as:

$$T(z) = \sum_{k=0}^{n} a_k z^k$$

Substituting $a_k = P(X = k)$:

$$T(z) = \sum_{k=0}^{n} \binom{n}{k} p^k (1 - p)^{n-k} z^k$$

Derivation and Final Result

Combine z^k with p^k:

$$T(z) = \sum_{k=0}^{n} \binom{n}{k} (pz)^k (1-p)^{n-k}$$

Using the binomial theorem:

$$(x+y)^n = \sum_{k=0}^{n} \binom{n}{k} x^k y^{n-k}$$

Let $x = pz$ and $y = 1 - p$. Then:

$$T(z) = (pz + (1-p))^n$$

Final Result:

$$T(z) = (p + (1-p)z)^n$$

Z-Transform Definition

Z-Transform for the Number of Tails in n Coin Tosses:

$$T(z) = (p + (1 - p)z)^n \quad \text{where:}$$

- p is the probability of tails ($p = 0.5$ for a fair coin),
- $1 - p$ is the probability of heads ($1 - p = 0.5$ for a fair coin),
- n is the number of tosses.

Expansion of $T(z)$

Expansion of $T(z)$ using the Binomial Theorem:

$$T(z) = \sum_{k=0}^{n} \binom{n}{k} p^{n-k} (1-p)^k z^k$$

where:

$$\binom{n}{k} = \frac{n!}{k!(n-k)!}$$

Interpretation:

- The coefficient of z^k in $T(z)$ gives the probability $P_X(k)$, where X is the number of tails.

Z-Transform for $n = 1$

For $n = 1$:

$$T(z) = p + (1 - p)z$$

For a fair die,

$$P_X(0) = 0.5, P_X(1) = 0.5$$

Probability Mass Function (PMF)

The PMF is computed as:

$$P_X(k) = \binom{n}{k} p^{n-k} (1-p)^k$$

Substitute $p = 0.5$ and $1 - p = 0.5$ for a fair coin:

$$P_X(k) = \binom{n}{k} (0.5)^n$$

Computational Steps

For each $k \in \{0, 1, \ldots, n\}$:

1. Compute the binomial coefficient:

$$\binom{n}{k} = \prod_{i=0}^{k-1} \frac{n-i}{k-i}$$

2. Multiply by $(0.5)^n$ to compute $P_X(k)$:

$$P_X(k) = \binom{n}{k} \cdot (0.5)^n$$

Result for $n = 3$

Substituting $n = 3$ and expanding:

$$T(z) = (0.5 + 0.5z)^3 = 0.125 + 0.375z + 0.375z^2 + 0.125z^3$$

The PMF values are:

$$P_X(0) = 0.125, \quad P_X(1) = 0.375, \quad P_X(2) = 0.375, \quad P_X(3) = 0.125$$

CDF Conditions

The CDF must satisfy:

1. $F_X(k)$ is non-decreasing.
2. $F_X(k) \in [0, 1]$ for all k.
3. $F_X(n) = 1$, where n is the maximum number of tails.

CDF Calculation

The CDF is given by:

$$F_X(k) = \sum_{i=0}^{k} P_X(i)$$

CDF Computation

Compute the values:

$$F_X(0) = P_X(0) = 0.125$$

$$F_X(1) = P_X(0) + P_X(1) = 0.125 + 0.375 = 0.5$$

$$F_X(2) = F_X(1) + P_X(2) = 0.5 + 0.375 = 0.875$$

$$F_X(3) = F_X(2) + P_X(3) = 0.875 + 0.125 = 1$$

For $n = 3$, the CDF values are:

$$F_X(0) = 0.125, \quad F_X(1) = 0.5, \quad F_X(2) = 0.875, \quad F_X(3) = 1$$

Probability of $X = 3$

Using the CDF, the probability of $X = 3$ is:

$$P_X(3) = F_X(3) - F_X(2)$$

Substituting the values:

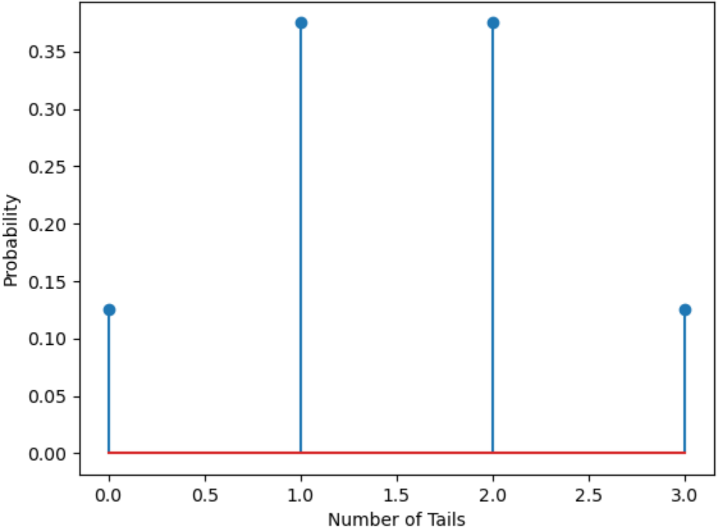
$$P_X(3) = 1 - 0.875 = 0.125$$

Conclusion

The probability of getting all three tails is:

$$P(X = 3) = \mathbf{0.125}$$

Plot of Probabilities



C Code: PMF Calculation

```c
// Function to calculate PMF using Z-transform
void calculate_pmf_z_transform(double *pmf) {
    int n = 3; // Number of tosses
    double p = 0.5; // Probability of tails for a fair coin

    for (int k = 0; k <= n; k++) {
        double binomial_coefficient = 1;
        for (int i = 0; i < k; i++) {
            binomial_coefficient *= (n - i) / (double)(i +
                1);
        }
        pmf[k] = binomial_coefficient * pow(p, n - k) * pow
            (1 - p, k);
    }
}
```

C Code: Expose to Python

```c
// Expose to Python
__attribute__((visibility("default"))) __attribute__((used))
void get_probabilities(double *pmf) {
    calculate_pmf_z_transform(pmf);
}
```

Python Code: Visualization

```
import ctypes
import matplotlib.pyplot as plt
# Load the shared library
lib = ctypes.CDLL('./c1.so')
# Create an array to hold PMF values (4 elements for 0 to 3
    tails)
pmf = (ctypes.c_double * 4)()
# Call the C function to calculate PMF
lib.get_probabilities(pmf)
# Convert the PMF to a Python list
pmf_values = [pmf[i] for i in range(4)]
# Plotting the PMF
x = [0, 1, 2, 3]  # Number of tails
plt.stem(x, pmf_values, use_line_collection=True)
plt.xlabel('Number of Tails')
plt.ylabel('Probability')
plt.show()
```