

Binomial Probability Calculation



 Uploaded image

explain me the math part of how did we get value 0.0004?

The value **0.0004** represents the **probability** that more than 10 users are simultaneously active when using **packet switching** with **35 users**, each being **active 10% of the time**.

Let's break down the **math** using **binomial probability**:

Problem Setup:

- **Number of users (N) = 35**
- **Probability of a user being active (p) = 0.1**
- **We need the probability that more than 10 users are active at the same time**

We model this as a Binomial Distribution:

Let **X** be the number of users active at a given moment. Then,
 $X \sim \text{Binomial}(n = 35, p = 0.1)$

We want:

$$P(X > 10)$$

But instead of computing $P(X > 10)$ directly, we compute the **complement**:

$$P(X > 10) = 1 - P(X \leq 10)$$

Using a binomial cumulative distribution function (CDF):

$$P(X \leq 10) = \sum_{k=0}^{10} \binom{35}{k} \cdot (0.1)^k \cdot (0.9)^{35-k}$$

You can compute this using a calculator, Python, or statistical software. The result is:

$$P(X > 10) \approx 1 - 0.9996 = 0.0004$$

Python Example (for reference):

```
python

from scipy.stats import binom

# n = 35 users, p = 0.1 active probability
```

```
P = 1 - binom.cdf(10, 35, 0.1)
print(P)
```

This outputs:

```
scss

0.000405... (approximately 0.0004)
```

✓ Conclusion:

The value **0.0004** comes from computing the **tail probability** in a **binomial distribution** — the probability that more than 10 users (out of 35) are active simultaneously when each is active 10% of the time.

Would you like me to generate a probability distribution chart or table for better visualization?



explain everything about 32 bit IPV4 and how the host and network parts are determined using subnet

Sure! Let's walk through **everything about 32-bit IPv4 addressing** and how **subnetting** determines the **host and network parts**.

What is IPv4?

IPv4 (Internet Protocol version 4) is the **most widely used protocol** for assigning addresses to devices on a network.

IPv4 Address Format:

- IPv4 addresses are **32-bit** binary numbers.
- They are typically written in **dotted decimal notation**:
Example: `192.168.1.10`
→ Binary: `11000000.10101000.00000001.00001010`

Each section is **8 bits** (1 byte), so:

- 4 bytes × 8 bits = **32 bits total**

Structure: Network vs Host Part

The 32 bits of an IPv4 address are divided into:

- **Network Part:** Identifies the **network** a host belongs to.
- **Host Part:** Identifies the **specific device (host)** within that network.

🔧 This division is controlled by the Subnet Mask.

🎭 Subnet Mask

A **subnet mask** is also a 32-bit number that shows how many bits are **network bits** (set to 1) and how many are **host bits** (set to 0).

Example:

- IP: ``192.168.1.10``
- Subnet Mask: ``255.255.255.0`` → Binary: ``11111111.11111111.11111111.00000000``
 - First 24 bits = **Network**
 - Last 8 bits = **Host**
- CIDR notation: ``192.168.1.10/24``

So:

- Network address = ``192.168.1.0``
- Host range = ``192.168.1.1`` to ``192.168.1.254``
- Broadcast address = ``192.168.1.255``

📊 Classes of IP Addresses (Legacy)

IPv4 originally had **5 classes**, but modern networking uses **CIDR** (Classless Inter-Domain Routing). Still, classes help understand ranges:

Class	Leading Bits	Address Range	Default Subnet Mask	# Hosts
A	0	0.0.0.0 – 127.255.255.255	255.0.0.0 (/8)	~16 million
B	10	128.0.0.0 – 191.255.255.255	255.255.0.0 (/16)	~65,000
C	110	192.0.0.0 – 223.255.255.255	255.255.255.0 (/24)	254

🧠 How Subnetting Works

Step-by-step:

1. **IP Address:** ``192.168.10.34``
2. **Subnet Mask:** ``/26`` → 255.255.255.192 → ``11111111.11111111.11111111.11000000``
 - 26 bits → Network
 - 6 bits → Host

Determine:

- **Network Address:** AND the IP and Subnet Mask
``192.168.10.34` & `255.255.255.192` → `192.168.10.0``
- **Broadcast Address:** All host bits set to 1
`→ `192.168.10.63``
- **Host Range:** ``192.168.10.1`` to ``192.168.10.62``
- **Total hosts:** $2^6 - 2 = 62$ (subtract 2 for network & broadcast)

How to Calculate Subnets

If you borrow bits from the host part:

- **Borrow 2 bits from /24 → /26**
- Subnets: $2^2 = 4$
- Hosts per subnet: $2^6 - 2 = 62$

CIDR Notation

CIDR = Classless Inter-Domain Routing

Instead of traditional classes, you use ``/n`` to indicate how many bits are the **network portion**.

Examples:

- ``/24`` → 255.255.255.0 → 256 addresses
- ``/30`` → 255.255.255.252 → 4 addresses (2 usable)
- ``/16`` → 255.255.0.0 → 65,536 addresses

Summary Table

CIDR	Subnet Mask	# of Hosts	Host Bits
/24	255.255.255.0	254	8
/26	255.255.255.192	62	6
/30	255.255.255.252	2	2
/16	255.255.0.0	65,534	16

Would you like a diagram to visually explain the network/host division or a subnetting problem solved step-by-step?