



Computational Thinking & AI

Cyber Architects · Engineers of Tomorrow

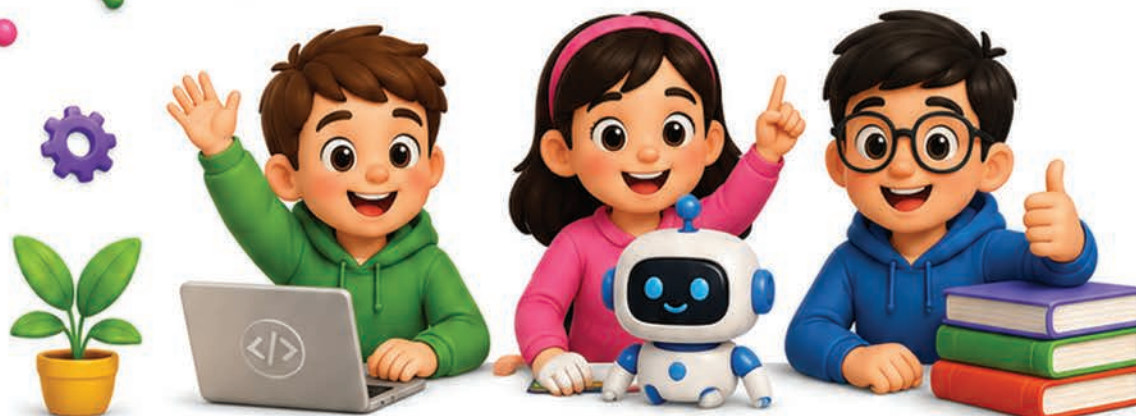


Class 6

> init(Code · Classes · ML · Ethics)



104 Pages · Real Engineering Unlocked



Editone Kids · Editone International Pvt. Ltd.



Computational Thinking & AI

Cyber Architects · Engineers of Tomorrow

For Class 6 · Book 6 of 8

Published by

Editone Kids

A division of Editone International Pvt. Ltd.

Naraina Industrial Area, New Delhi

www.editone.kids

© 2026 Editone International. All rights reserved.



Aligned with CBSE CT & AI Curriculum 2026–27,
NEP 2020, NCF-SE 2023, and DPDP Act 2023



First Edition · 2026





Welcome

A Note for Grown-Ups



Dear Teacher and Parent,

Class 6 is the **foundation year** for real engineering thinking! Your child crosses from "using AI" to **"understanding and building AI"** like a real software engineer. CBSE allocates 100 hours of CT & AI in Class 6 – this book covers the full curriculum and more.



What's NEW in Class 6:

-  **Real Python-style code** – variables, lists, dicts, functions, classes
-  **Object-Oriented Programming** – classes & objects intro
-  **Machine Learning** – supervised vs unsupervised
-  **Statistical patterns** – mean, median, mode
-  **Recursion** – functions calling themselves!
-  **AI ethics & DPDP Act 2023** – India's data privacy law
-  **AI for UN Sustainable Development Goals**
-  **Career paths in AI** – ML engineer, data scientist, ethicist!

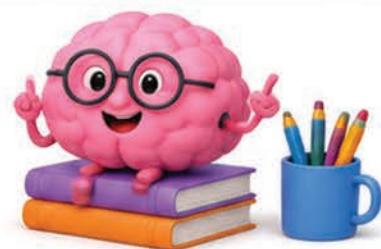


Why this matters:

Class 6 sets the foundation for high school computer science, AI labs, and future careers. Concepts here are real – same vocabulary used at Google, Microsoft, ISRO!



Approx. 100 hours · CBSE CT 2026–27 ready · DPDP Act 2023 aligned · 70% screen-free





Guide

How to Use This Book



Class 6 has 7 sign types — look for them!



Big Idea

Core CT/AI concept (CBSE).



Lab Activity

Hands-on experiments!



India Page

Real Indian context.



Class Diagram

NEW! OOP blueprints!



ML Card

NEW! Real ML concepts!



Code Lab

Real Python code!



100 hours of learning

~3 hours per week + 3 mini-projects. CBSE recommends a notebook + optional computer access for Python.



Class 6 students should keep an **"Engineering Journal"** — write what you build & learn!



Hello!

Meet the Cyber Architects!

Your 8 friends are now engineers — and a **NEW** friend joins!



Diya

ML researcher!



Arjun

Software engineer!



Mithu

Pattern analyst!



Beep

Multi-AI agent!



Tara

Project lead!



Veer

Data scientist!



Riya

Python coder!



Kabir

Data detective!



Ishaan ✨ NEW!

ML Engineer! Loves neural networks & future tech!



9 Cyber Architects • Ready to build!



	Chapter 1 – Decomposition for Real Projects	Page 7
	Chapter 2 – Pattern Mastery & Data Patterns	Page 22
	Chapter 3 – Abstraction with Classes & Objects	Page 38
	Chapter 4 – Algorithms in Real Code (Python!)	Page 54
	Chapter 5 – AI/ML Foundations	Page 70
	Big Brain Review (4 parts!)	Page 87
	Mega Maze (20-column!)	Page 91
	Picture, Logic, Pattern, Code, Data & ML Puzzles	Page 92
	Glossary – Big Words I Learned	Page 98
	Things I Can Do Now	Page 102
	My Engineering Journal	Page 103
	My Certificate	Page 104





Chapter 1 • CT Pillar #1 in Production



Decomposition for Real Projects

How software engineers **REALLY** break problems!





Real Engineering!



Real engineers decompose into **THREE** layers: frontend, backend, data!



NEW Concept: SOFTWARE LAYERS

Every app is decomposed into LAYERS. Each team works on one!

Example: WhatsApp's Architecture



FRONTEND (Layer 1)

What you SEE on phone – buttons, chats, contacts, voice icon. Built by UI engineers!



BACKEND (Layer 2)

The 'brain' that processes – sends messages, encrypts data. Built by backend engineers!



DATABASE (Layer 3)

Where data is STORED – your chats, contacts, photos. Built by database engineers!



CLOUD (Layer 4)

Servers around the world. Built by DevOps engineers!



One app = many engineers, each working on their layer!





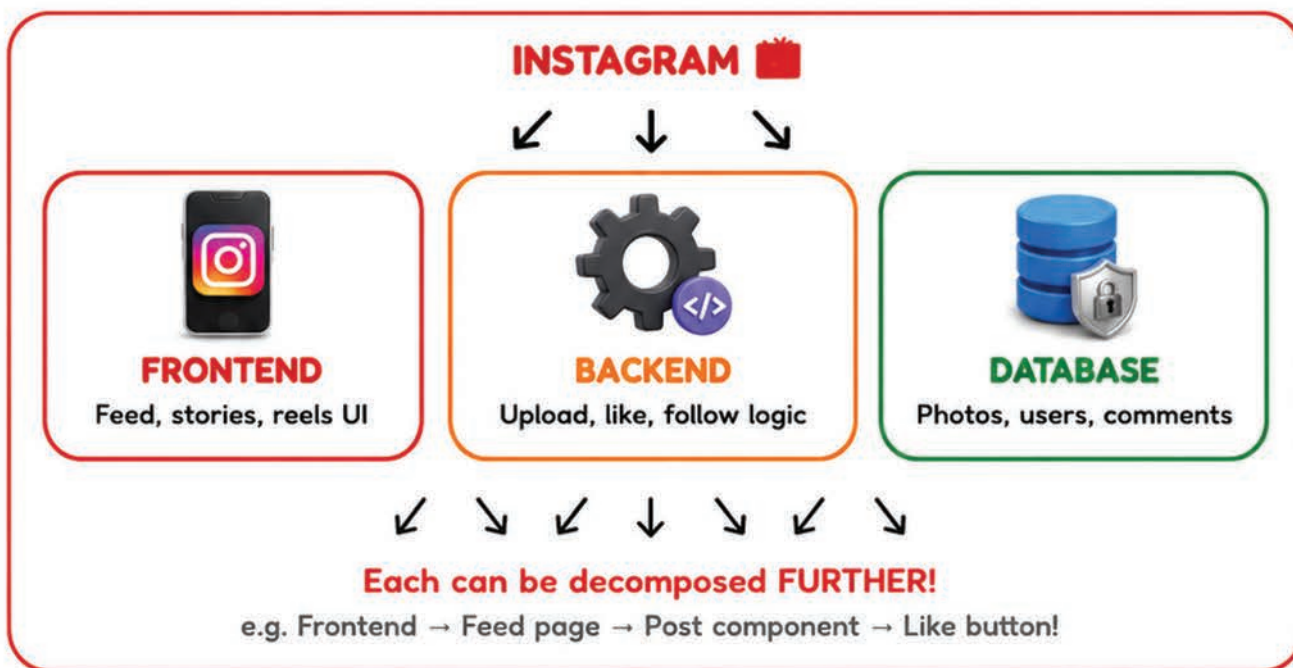
Decomposing Big Software



Lab · Build Your Own Instagram Clone (on paper)



Decompose Instagram into engineering parts!



Your turn: Decompose YouTube:



Frontend parts: _____



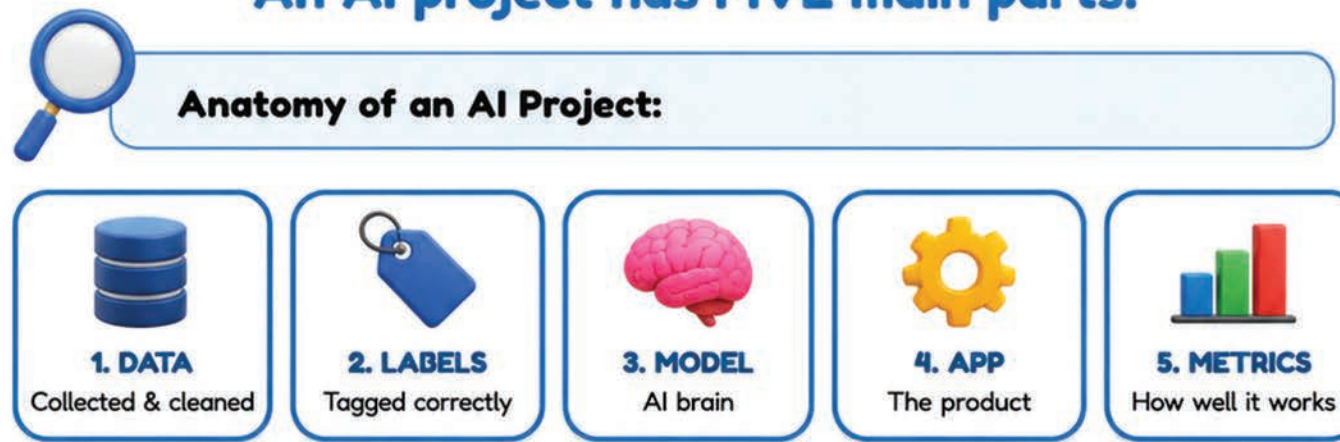
Backend parts: _____



Database parts: _____



An AI project has FIVE main parts!





Example: Spam Email Filter AI

- DATA:** 100,000 emails (spam + normal)
- LABELS:** "spam" or "not spam" tags
- MODEL:** Trained AI that detects spam patterns
- APP:** Your Gmail's spam folder!
- METRICS:** Accuracy = 99.5%!




Now you – design “Homework Grading AI”:



DATA needed: _____



LABELS: _____



MODEL job: _____



APP form: _____





Break it down.
Build it up!



Time Decomposition

Real engineers decompose TIME too – into sprints!



Big Word: SPRINT

A SHORT period (1-2 weeks) where engineers complete one small chunk.
Many sprints = one big project!

Example: 8-week project decomposed into 4 sprints!



Sprint 1

Week 1-2: Plan & design



Sprint 2

Week 3-4: Build frontend



Sprint 3

Week 5-6: Train AI model



Sprint 4

Week 7-8: Test & launch



Why sprints?

- Small wins keep team motivated!
- Can adjust plans every 2 weeks
- Used at Google, Microsoft, Flipkart, ISRO
- Called “Agile” methodology



Decompose YOUR project – “Build a school newsletter” in 4 sprints:



Sprint 1 (Plan): _____



Sprint 2 (Build): _____



Sprint 3 (Review): _____



Sprint 4 (Publish): _____



Break big projects into small sprints. **Win step by step!**



India's UPI handles 10 BILLION transactions per month.
How?

UPI = Unified Payments Interface. Built by NPCI (National Payments Corporation of India).



USER APPS

PhonePe, Google Pay, Paytm, BHIM – they're all FRONTENDS!





UPI SWITCH

NPCI's central system – routes payments between apps!





BANKS

Your bank + receiver's bank – they actually move money!





SECURITY LAYER

AI for fraud detection, encryption, PIN verification!





Speed:

UPI processes a payment in **0.5 seconds!**

Built by decomposition – each layer handles one job perfectly.
1000+ engineers built & maintain it!







UPI is now used in Bhutan, Nepal, France, Singapore, UAE!



Decompose a Game



Activity · Roblox Designer



Big games like Roblox, Minecraft, Free Fire have **HUNDREDS** of parts!

A 3D GAME



GRAPHICS

3D models, textures, lights



CONTROLS

Keyboard, mouse, gamepad



PHYSICS

Gravity, collision, jumps



NETWORK

Multi-player connection



AI (NPCs)

Smart enemies/helpers



SOUND

Music, effects



SAVE

Player progress



SCORING

Levels, XP, rewards

Design **YOUR** game! Pick 5 parts:



Game name: _____



5 parts: _____





Mini-Project • Launch a Startup!



Mini-Project • Be the Founder!



Mission: Decompose how to launch a tech startup!

Step 1: What problem solves your startup?

Problem: _____



Step 2: Name + tagline:

Name: _____ Tagline: _____



Step 3: Decompose into key parts:



Product idea



Team needed



Funding



Marketing



Build product



Launch!

Step 4: 90-day plan (3 sprints of 30 days):



Sprint 1 (Days 1–30): _____



Sprint 2 (Days 31–60): _____



Sprint 3 (Days 61–90): _____



Indian startup success stories:

Flipkart, Ola, Zomato, Razorpay – all started small!





Resource Decomposition



Every project needs: PEOPLE · TIME · MONEY!

Engineers decompose RESOURCES, not just tasks!



PEOPLE

Who works on what?



TIME

How many hours/days needed?



MONEY

What does it cost to build?

Example: Building a school website

Task	People · Time	Cost (₹)
 Design	1 person · 5 days	5,000
 Develop	2 people · 10 days	20,000
 Content	1 person · 3 days	3,000
 Hosting	0 people · 1 day	2,000/yr
TOTAL	4 people · 19 days	₹30,000



Smart decomposition lets you BUDGET!

Without it = surprises & delays. With it = clear plan!





Decomposition in Careers!



Different careers – same decomposition skill!



DOCTOR

Patient → symptoms →
tests → diagnosis →
treatment



CHEF

Menu → ingredients →
prep → cook → serve



TEACHER

Lesson → intro →
explain → practice → test



FILMMAKER

Story → script → shoot →
edit → release



ARCHITECT

Site → plans →
blueprint → construction →
finish



CODER

Problem → design →
code → test → deploy



All careers use decomposition!

Whether you become a doctor, chef, or programmer – this skill helps you SUCCEED!



Your dream career: _____

Decompose it into 5 steps:

1. _____ 2. _____ 3. _____
4. _____ 5. _____



The "Atomic" Task

Smallest possible task = can't break down more!



Big Word: **ATOMIC TASK**

A task SO SMALL it can't be split further. Like an atom in chemistry – the smallest unit!

Decomposing "Write an essay":



Level 1: Write essay

↓ break further

Level 2: Intro · Body · Conclusion

↓ break further

Level 3: Each paragraph has 4–5 sentences

↓ break further

Level 4 (ATOMIC!): Each sentence = 1 idea + words



STOP! Can't break further!



Why find atomic tasks?

- You know EXACTLY what to do
- Easy to estimate time
- Easy to assign to one person
- Easy to check if done!



Find atomic tasks for "Plan a birthday":

1 Atomic 1: _____

2 Atomic 2: _____

3 Atomic 3: _____





Decomposing AI Ethics

Ethics problems are **TRICKY**. Decompose to think clearly!



✓ The Ethics Decomposition Framework:

1. **WHO?** Who is affected?
2. **WHAT?** What harm could happen?
3. **WHY?** Why is this being done?
4. **ALTERNATIVES?** What **ELSE** could we do?
5. **DECIDE!** What's the right choice?



Real case: Your school wants AI cameras to watch students all day!



- WHO?** All students & teachers
- WHAT?** Privacy lost • feeling watched • pressure
- WHY?** To prevent bullying & cheating
- ALTERNATIVES?** Trust + counselors + class talks
- DECIDE:** What do **YOU** think?



Your verdict:



- ☐ Allow AI cameras ☐ Block them ☐ With limits – Why? _____



Real engineers **MUST** think about ethics – companies have “**Ethics Committees**” now!



Brain Puzzle · Decomposition Pro!



1 Puzzle 1 · Reverse Engineer

An app shows your photos in a beautiful grid, lets you share & like. People follow each other. What is it?

Decompose its 6 main parts: _____



2 Puzzle 2 · Resource Decomp

Build a small mobile app. Needs: 1 designer (5 days, ₹500/day), 2 developers (15 days, ₹800/day each), hosting ₹3000.

Total cost: ₹ _____



3 Puzzle 3 · 3-Level Tree

Decompose "Olympics 2028":

Level 1: _____ Sports · _____ Countries · _____ Venues

Level 2 (Sports): _____ Track · _____ Swimming · _____ Gymnastics

Level 3 (Swimming): _____



4 Puzzle 4 · Atomic Hunter

Which of these is ATOMIC?

A) "Plan trip" B) "Write story" C) "Lift my pencil" D) "Eat lunch"

Answer: _____





Case Study · ISRO Chandrayaan-3

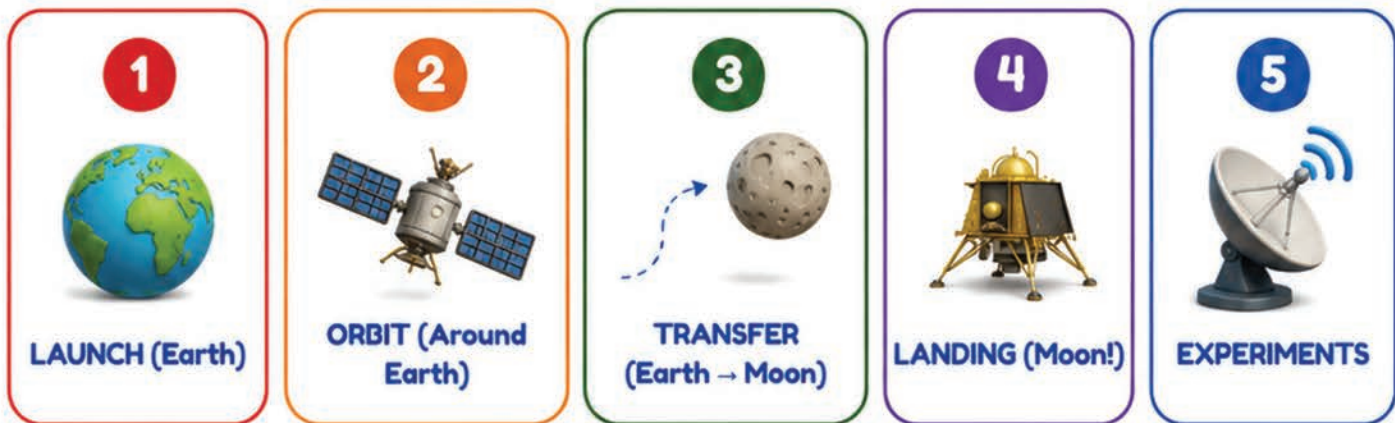


India's moon landing – DECOMPOSED!

August 23, 2023: India became the 4th country to land on the Moon (and FIRST near the south pole!)



The 4-Phase Decomposition:



Each phase was decomposed into 100s of sub-tasks!

- **Engineers:** 1000+ ISRO scientists
- **Time:** 4 years
- **Cost:** ₹615 crores (~\$75 million) – cheaper than many Hollywood movies!
- **Result:** SAFE landing! Pragyan rover walked on Moon!



ISRO uses computational thinking just like **YOU**'re learning!

Try it Yourself!



Real-Engineering Challenge



1. **App Decompose:** Pick any app on your parent's phone. Decompose into Frontend/Backend/Database!



2. **Startup Pitch:** Design a 3-min pitch to start a tech company. Use sprints!



3. **Atomic Tasks:** Decompose "Get ready for school" until you reach atomic tasks!



4. **Ethics Decompose:** Use the framework on "AI grading exams". Discuss with family!



5. **Resource Plan:** Plan a class trip – people, time, cost!



In Chapter 1 you learned:



Software **layers** (Frontend, Backend, Database, Cloud)



5-part **AI project** anatomy



Sprints & Agile thinking



Resource decomposition (People, Time, Money)



Atomic tasks



Ethics framework for AI decisions



ISRO & UPI engineering!



Software Architect!

Next: **Pattern Mastery!** ➔

Chapter 2

CT Pillar #2 in

Production

Pattern Mastery & Data Patterns

Stats • Trends • Anomalies • AI patterns!



Pattern Types in Real Data



Real data has 3 main pattern types!



LINEAR

Steady increase or decrease
e.g. your height every year



EXPONENTIAL

Doubles or triples!
e.g. virus spread, internet users



CYCLICAL

Repeats in cycles!
e.g. seasons, day/night



Real examples:

Data	Pattern Type	Notes
 Your savings ₹/month	Linear 	+₹100/month
 COVID cases 2020	Exponential 	Doubled every 5 days!
 Monsoon rain	Cyclical 	Every year June-Sept
 City population	Linear 	+2% per year
 Smartphone users	Exponential 	Doubled in 5 years!



Identify pattern type **FIRST** → then forecast!



Time Series Patterns

Data over TIME = a time series!



Big Word: TIME SERIES

Numbers measured at **REGULAR** intervals – every second, day, year. Used for forecasting!



Example: Delhi temperature this week

Day	Time	Temp °C
 Mon	9 AM 	28
 Mon	3 PM 	38
 Mon	9 PM 	30
 Tue	9 AM 	27
 Tue	3 PM 	39
 Tue	9 PM 	31



Spot patterns:

- Hot in afternoon! (CYCLICAL within a day)
- Slightly hotter than yesterday (LINEAR — heating up!)
- Pattern: peaks at 3 PM, lows at 9 AM



Predict tomorrow 3 PM: _____ °C



Stock markets, weather, traffic – ALL use time series patterns!



Pattern Matching in Code



Computers **SEARCH** for patterns in text using "pattern matching"!



Real Python – **Pattern Matching:**

```
# Find names starting with "A"
names = ["Arjun", "Diya", "Aanya", "Veer", "Ananya"]
for name in names:
    if name.startswith("A"):
        print(name)
```



Output: Arjun, Aanya, Ananya



Common pattern-matching tasks:



startswith("A")

Begins with A?



endswith(".jpg")

Ends with .jpg?



contains("@")

Has @ symbol?



len(text) > 5

Length check!



Email checks, password rules, search bars – all use pattern matching!




REGEX = the **SUPERPOWER** of pattern matching!
 




Big Word: REGEX

"Regular Expression" – a **CODE** for patterns!
 Lets you match any text pattern in milliseconds!

Examples of REGEX:

Pattern (Regex)	Matches	Example
 <code>^A.*</code>	Starts with A	 "Arjun", "Apple"
 <code>\d{10}</code>	10 digits	 Indian phone numbers!
 <code>. * @ . * \ . com</code>	Email-like	 diya@school.com
 <code>[A-Z]{3}</code>	3 capitals	 "BMW", "IIT", "AIM"



- Where regex is used:**
- Email validation (every signup form!)
 - Phone number checking
 - Search & replace in text editors
 - Spam filters!



You try! Which regex matches Indian PIN codes (6 digits)?

A) `\d{5}` B) `\d{6}` C) `[A-Z]{6}` → ____



India Page • Stock Market Patterns!



BSE Sensex & NSE Nifty – patterns matter!



Stock Pattern Types:



BULL MARKET

Prices going UP! Investors happy!



BEAR MARKET

Prices going DOWN! Worry!



SIDeways

No clear direction. Boring!



VOLATILE

Big jumps both ways. Risky!



Indian Stock Markets:

- **BSE Sensex** – 30 top companies (Mumbai)
- **NSE Nifty 50** – 50 top companies
- **AI traders** – bots that buy/sell in milliseconds!
- **Pattern detection** – AI spots trends humans miss!



Important warning:

Stocks are **RISKY!** Patterns help – but not always. Even AI loses sometimes. Never invest without learning & advice!





Pattern Recognition by AI

AI is the **WORLD'S BEST** pattern matcher!

How AI sees patterns vs humans:



HUMANS see:

- About 7 things at once
- 2-3 patterns max
- Slow learning
- Get tired!



AI sees:

- MILLIONS at once!
- 1000s of patterns
- Super fast learning
- Never tires!



ML PATTERN INSIGHT

A typical computer vision AI can recognize 10,000+ different objects.
ChatGPT learned patterns from 175 BILLION text fragments.
Spotify learns your music taste from 1000s of patterns!

What AI patterns YOU use daily:



YouTube recommendation: _____



Google search ranking: _____



Camera face detection: _____





Anomaly Detection!



➤ When the **PATTERN BREAKS** – that's an **ANOMALY**! ➤



Big Word: ANOMALY

An-aw-muh-lee! Something that DOESN'T fit the pattern.
AI loves finding anomalies!

➤ Real-world anomaly detection: ➤



Credit Card Fraud

You always shop in Delhi. Suddenly Mumbai?
AI alerts!



Disease Detection

Heart-rate pattern weird? AI warns doctor!



Cyber Attacks

Unusual login from foreign country!



Factory Defects

AI spots faulty product on conveyor belt!



Anomaly Detection Algorithm:

- 1 Learn the **NORMAL** pattern
- 2 Set boundaries (typical range)
- 3 Flag anything **OUTSIDE** boundaries!



Find the anomaly: **[10, 12, 11, 13, 250, 11, 12, 14]**

Anomaly: _____ Why? _____

Apps **WATCH** how you use them – and learn patterns!



Big Word: **CLICKSTREAM**

The **TRAIL** of clicks & swipes you leave on an app/website.
AI studies this to improve apps!

What apps track:



Where you tap

Which buttons
attract you



Time per page

What's
interesting



Watch patterns

Skip vs
watch full



Search history

What you
look for



Why this is important:

- Helps apps improve experience
- BUT – your privacy can be compromised!
- Companies sell this data!
- India's DPDP Act 2023 protects you!



Try this: Track **YOUR** YouTube usage today:



Time spent: _____ min



How many videos? _____



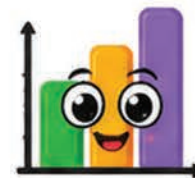
What topic mostly? _____



You can **RESET** your "ad profile" in Google settings!



Statistical Patterns!



Statistics = math of patterns in data!



The 3 Magic Numbers:



MEAN

Average!
Add all ÷ count



MEDIAN

Middle value when sorted



MODE

MOST FREQUENT value

Example: Class test scores: [85, 90, 75, 90, 80, 95, 90]

Statistic	Calculation	Result
MEAN	$605 \div 7$	86.4
MEDIAN	Sort: 75, 80, 85, 90 , 90, 90, 95	90
MODE	90 appears 3 times	90



When to use each:

- **MEAN** – overall average
- **MEDIAN** – typical, ignores outliers
- **MODE** – most common (popular!)



You try!

Heights of friends (cm): [140, 145, 142, 150, 145, 138]

Mean = ____ Median = ____ Mode = ____



Patterns in Epidemics!

⇒ Diseases spread in PATTERNS – AI helps predict! ⇐



Big Word: **EPIDEMIC**

When a disease spreads FAST to many people.
Math & AI help predict it!



The famous “R-number”:

R

R (reproduction number) = average people 1 sick person infects

- **R = 1** → stable
- **R < 1** → disease dying out
- **R > 1** → SPREADING fast
- **R = 2** → each person spreads to 2 → exponential!



Spread example with R = 2:

Day	New Cases	Total
0	1	1
1	2	3
2	4	7
3	8	15
4	16	31
5	32	63



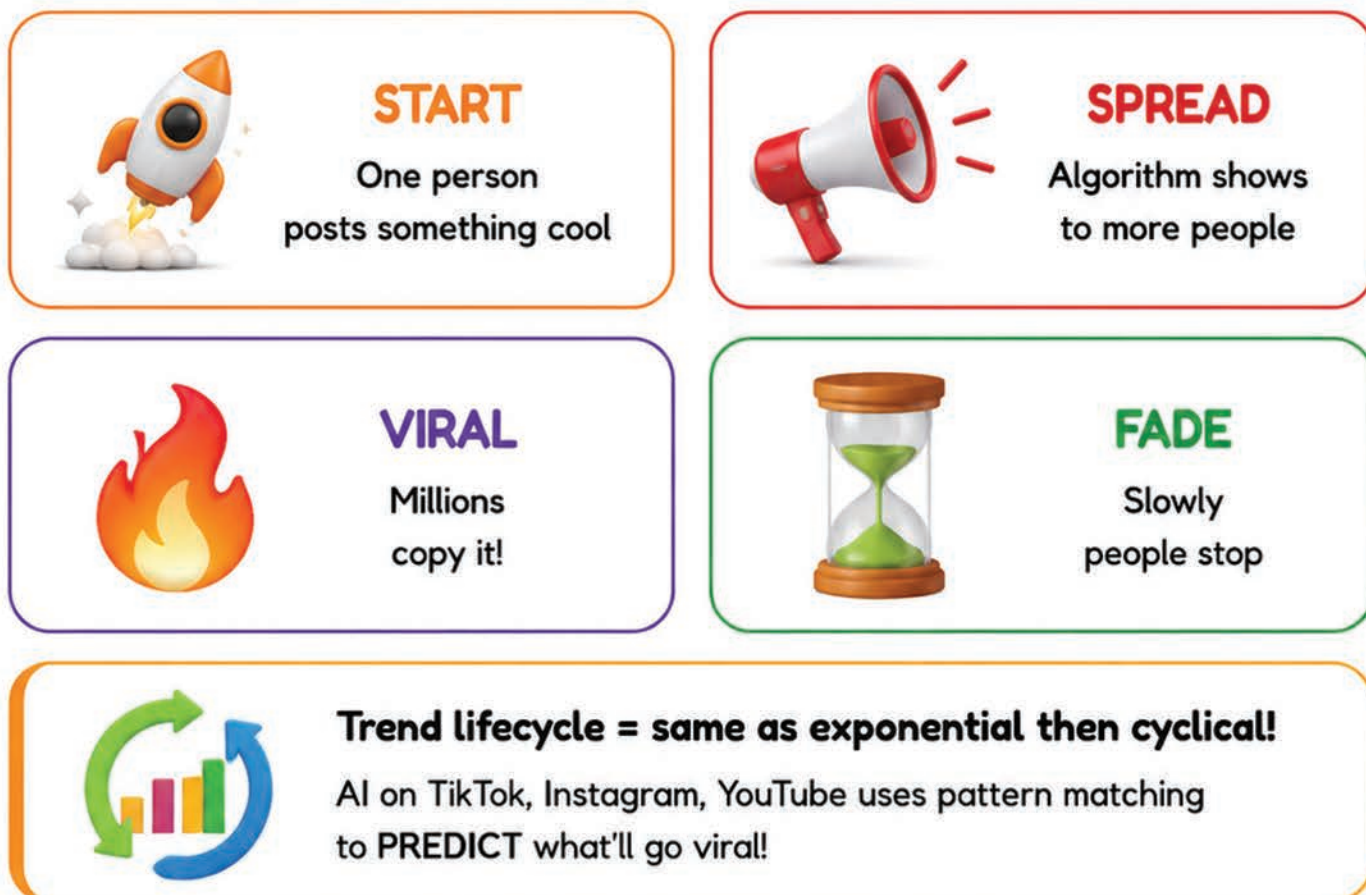
In 5 days, 1 person became **63!** That's exponential growth!



During COVID-19, Aarogya Setu AI tracked R-numbers across India!

➤ Trends are **PATTERNS** people follow on social media! ➤

Anatomy of a trend



Famous Indian viral trends:





Trends come & go. Don't chase them — be **ORIGINAL!**



Mini-Project • Be a Pattern Detective!



Mission: Find a real pattern in your school using data!

 **Step 1:** Pick ONE thing to track for 7 days:



Library visits/day



Lunch likes/day



Sports field users

 **Step 2:** Collect data!

Day	Number	Notes
Mon	_____	_____
Tue	_____	_____
Wed	_____	_____
Thu	_____	_____
Fri	_____	_____

 **Step 3:** Calculate & predict!



Mean: _____ Median: _____ Mode: _____



Pattern type: ☐ / ☐ / ☐ / ☐ _____



Predict next Monday: _____



Bonus • Music Patterns!

Music is full of math & patterns!



BPM = Beats Per Minute

How FAST is a song's beat? Each BPM has its own feel!

BPM Range	Mood	Examples
 60-70	Calm 	Lullabies, Bhajans
 80-100	Chill 	Slow Bollywood, Pop
 120-130	Dance 	Pop, Bollywood remix
 140-180	Workout 	EDM, Punjabi beats!



Spotify uses BPM & mood patterns to make playlists like "Workout Hits", "Calm Sleep"!



Find BPM at home:

1	Play your favourite song		
2	Tap your foot in rhythm		
3	Count taps in 15 seconds		
4	Multiply $\times 4 = \text{BPM!}$		$\times 4$

Song: _____ BPM: _____ Mood: _____



Brain Teasers • Pattern Pros!



1 Teaser 1 • Spot the Anomaly

Daily steps: [8500, 8200, 8400, 8600, 50, 8300, 8500]

Anomaly day: _____ Why probably? _____



2 Teaser 2 • Pattern Math

Sequence: 2, 6, 12, 20, 30, 42, _____

Hint: differences are 4, 6, 8, 10, 12, ?

Next: _____



3 Teaser 3 • R-number

If $R=3$ and starts with 1 person, after 4 days how many TOTAL?

Day 1: 3 new • Day 2: 9 • Day 3: 27 • Day 4: 81

Total: _____



4 Teaser 4 • Statistics

Marks: 70, 75, 80, 75, 90, 75, 85

Mode: _____ Median: _____ Mean: _____



5 Teaser 5 • Time Series

App users monthly: Jan 100, Feb 200, Mar 400, Apr 800. May?

Pattern: _____

May: _____





Try it Yourself!



Pattern Master Challenge

1. **Stock Watcher:** Pick Reliance or TCS. Watch Sensex prices for 5 days. Note pattern!
2. **Anomaly Hunt:** Track your daily routine for 7 days. Find an anomaly day!
3. **Stats Master:** Get heights of 10 family/friends. Calc mean/median/mode!
4. **BPM Detective:** Find BPM of 5 different songs. Pattern matches mood?
5. **Regex Sleuth:** Write regex for "all valid email IDs"!



In Chapter 2 you learned:

- 3 pattern types: **linear, exponential, cyclical**
- **Time series** data
- Pattern matching with Python (startswith, contains)
- **REGEX** – the superpower!
- **Anomaly detection** in fraud/disease
- Statistics: **mean, median, mode**
- R-number for epidemics
- Music BPM patterns!



Pattern Analyst!



Next: Classes & Objects!

