

# Recommender Systems - Theory Behind It



## The Big Idea

We are building a system that can **suggest products** to customers:

- If they are **new customers** (no history).
- If they are **returning customers** (with past purchases).
- If the **website is brand new** (no ratings yet).

So the system has 3 parts:

---

### ◆ Part I: Popularity-Based Recommendations (for new users)

- Imagine you walk into a new shop for the first time. You don't know what's good.
- The easiest thing the shopkeeper can do is:
  - 👉 Show you the **most popular products** (the ones most people buy).
- In our code:
  1. We count how many times each product has been rated/purchased.
  2. We sort products by this count.
  3. The top 10 or 20 items are shown to you.

This is called a **cold-start strategy** because it works even if we know nothing about you.

---

## ◆ Part II: Collaborative Filtering (for returning users)

- Once you buy something, we know a bit about your taste.
- Collaborative filtering means:  
👉 *“People who bought similar things to you also liked these other items.”*

How it works:

1. We create a big table (called a **utility matrix**) where:
  - Rows = users.
  - Columns = products.
  - Values = ratings (or 0 if no rating).
2. This table is huge and mostly empty (because no one buys *everything*).
3. We use a technique called **SVD (Singular Value Decomposition)** to simplify this table into hidden “patterns”:
  - Example pattern: “people who like lipsticks also like perfumes.”
  - Example pattern: “people who buy baby products also buy toys.”
4. Using these patterns, we find products that are **similar** to what you already bought.
5. Then we recommend those similar products to you.

So, if you bought a face cream, the system may suggest related items like cleansers or moisturizers.

---

## ◆ Part III: Cold-Start for New Products (content-based)

- What if a product is **new** and has no ratings yet?
- Then we look at its **description or category** (e.g., “red lipstick, beauty product”).
- We compare product descriptions using **TF-IDF** (a text similarity measure).
- This lets us recommend new items even before anyone has rated them.

Example: A new “organic shampoo” arrives → the system looks at its description and suggests it to people who bought similar “hair care” items.

---

## ⚡ Putting It All Together

- If you're a **brand new user** → show **popular items**.
  - If you're a **returning user** with purchase history → use **collaborative filtering**.
  - If it's a **new product** (no ratings yet) → use **content similarity**.
  - If nothing else works → always fall back to **popularity**.
- 

## 🧠 In Short

- **Popularity-based** = “Best sellers.”
- **Collaborative filtering (SVD)** = “People like you also liked these.”
- **Content-based** = “This product is similar to the one you looked at.”

Together, this makes a **hybrid recommender system** that can handle many real-world situations.